PAT: Parameter-Free Audio-Text Aligner to Boost Zero-Shot Audio Classification

Ashish Seth, Ramaneswaran Selvakumar, Sonal Kumar, Sreyan Ghosh, Dinesh Manocha
University of Maryland, College Park
{aseth125, ramans, sonalkum, sreyang, dmanocha}@umd.edu

Abstract

Audio-Language Models (ALMs) have demonstrated remarkable performance in zero-shot audio classification. In this paper, we introduce PAT (**P**arameter free **A**udio-**T**ext aligner), a simple and *training-free* method aimed at boosting zero-shot audio classification performance of CLAP-like ALMs. To achieve this, we propose to improve the cross-modal interaction between audio and language modalities by enhancing the representations for both modalities using mutual feedback. Precisely, to enhance textual representations, we propose a prompt ensemble algorithm that automatically selects and combines the most relevant prompts from a datastore with a large pool of handcrafted prompts and weighs them according to their relevance to the audio. On the other hand, to enhance audio representations, we reweigh the frame-level audio features based on the enhanced textual information. Our proposed method does not require any additional modules or parameters and can be used with any existing CLAP-like ALM to improve zero-shot audio classification performance. We experiment across 18 diverse benchmark datasets and 6 ALMs and show that the PAT outperforms vanilla zero-shot evaluation with significant margins of 0.42%-27.0%. Additionally, we demonstrate that PAT maintains robust performance even when input audio is degraded by varying levels of noise. We make our code publicly available [1].

1 Introduction

Advancements in multimodal language models (MLMs) have significantly improved performance across various modalities and applications (Turian et al., 2022; Yang et al., 2021). Audio-Language Models (Ghosh et al., 2024b), specifically Audio-Language Encoders (ALEs) like CLAP (Guzhov et al., 2022; Wu* et al., 2023), are a distinct type of

[1]Code: https://github.com/cs20s030/PAT.git

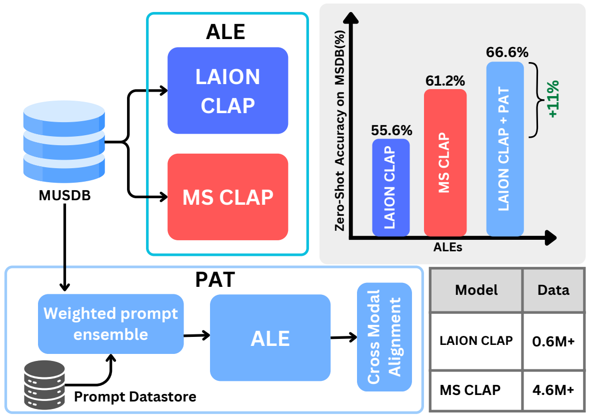

Figure 1: Comparison of zero-shot audio classification performance of ALEs (LAION CLAP (Wu* et al., 2023) and MS CLAP (Elizalde et al., 2023a)) with and without PAT on MUSDB (Bertin-Mahieux et al., 2011) (music genre classification). Our proposed training-free method significantly enhances zero-shot performance, even in low-resource domains where the ALEs have limited training data.

MLMs that learn a shared representation space between the audio and language modalities. Trained on large-scale audio-caption pairs, these models acquire diverse audio concepts during pre-training, enabling them to generalize to new, unseen audio categories. This capability allows ALEs to excel in zero-shot audio classification, accurately categorizing any set of classes described with natural language during inference. Such flexibility is essential for ALEs to effectively adapt to dynamic environments with diverse and unknown sounds.

Significant progress has been made to enhance the zero-shot performance of ALEs across a wide range of audio classification tasks. For instance, Contrastive Language Audio Pretraining (CLAP) (Wu* et al., 2023) was one of the first ALE models to achieve notable zero-shot improvements by utilizing descriptive audio captions, as opposed to the class labels used in earlier models like Wav2CLIP(Wu et al., 2022) and AudioCLIP (Guzhov et al., 2022). Additionally, ALEs such as LAION-CLAP (Wu* et al., 2023) and MS-

Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 12376–12394
April 29 - May 4, 2025 ©2025 Association for Computational Linguistics

CLAP (Elizalde et al., 2023a) employ vast collections of audio-text pairs, while models like CompA-CLAP (Ghosh et al., 2023) leverage complex learning objectives during pretraining to show zero-shot performance gains across various audio classification tasks. While these models show zero-shot improvement, it comes at the additional cost of pre-training them with either more refined learning objectives or by improving both the quality and quantity of audio-text pairs. To overcome these challenges, researchers have attempted to improve ALEs' audio classification capabilities through parameter-efficient transfer learning. Inspired by prompt learning (Brown, 2020) and adapter fine-tuning approaches (Houlsby et al., 2019), Treff Adapter (Liang et al., 2023) and Audio Prompt Learner (Liang et al., 2023) incorporate learnable prompts or lightweight adapters to adapt both textual and audio features. Although these methods result in performance improvements, they introduce learnable parameters and require additional training phases with few-shot labelled data. On the other hand, training-free methods can utilize the zero-shot capabilities of ALEs, making them more scalable and efficient for real-world applications which can contain diverse and unknown sounds. To the best of our knowledge, there exists no *training-free* method to improve the zero-shot classification capabilities of such models.

Our Contribution. To this end, we propose **PAT**: **P**arameter-free **A**udio **T**ext Aligner, a simple, parameter-free, and training-free approach for boosting zero-shot audio classification performance in ALEs. We first identify two major issues with current ALEs: 1) Current ALEs rely on simplistic prompts like "The sound of <label>" during zero-shot evaluation, leading to suboptimal performance. 2) During zero-shot transfer, when we align audio and text representations through cosine-similarity, audio representations are typically average pooled, which results in information loss and reduced discriminative power (Ruderman et al., 2018; Springenberg et al., 2014). Inspired by our findings, we propose two novel components: *1) Weighted Prompt Ensemble:* To enhance zero-shot performance for ALEs, we enrich the textual embeddings by creating a task-agnostic prompt datastore comprising 400 unique prompts. These prompts are specifically designed to reduce the distribution shift between the zero-shot setting and the training data, where audio captions typically contain more than a single-word description, unlike previous approaches. Each prompt is generated from handcrafted templates, capable of accommodating unique sound labels while ensuring semantic coherence. Moreover, we show that naively using all the prompts for various downstream applications may not consistently yield the desired performance improvements. To address this, we introduce *weighted prompt ensemble*, a method which computes a score for each prompt using the ALEs' response as feedback without the need for additional labeled data or training. These scores are then used as weights to perform a weighted ensemble of the text embedding associated with a prompt and a class label. *2) Cross Modal Aligner:* We introduce a *cross-modal aligner* that enhances audio representations with text-guided information. Specifically, we first compute an attention map using parameter-free attention mechanisms between the frame-level audio representations and the enriched textual representations associated with each label. Next, we use this attention map to perform weighted pooling on the frame-level audio representations. Finally, we compute the cosine similarity between the enriched audio and text representations to perform zero-shot classification. To summarize, our main contributions are:

- We propose PAT, a novel approach to improve zero-shot audio classification performance in a *training-free* fashion. PAT introduces a cross-modal interaction approach aimed at improving audio-text alignment by enhancing both audio and textual representations in a zero-shot setting.
- We evaluate PAT across multiple ALEs on 18 audio classification datasets and show that PAT achieves 0.42%-27.0% improvement over our baselines.
- We further investigate PAT's robustness to noisy audio to show that PAT consistently outperforms our baselines under varied noise augmentation settings.

2 Related Work

Audio Language Encoders (ALEs). Previous explorations on developing multimodal encoders to learn shared representations across different modalities have shown significant promise. For example, building on contrastive pre-training techniques from vision-language models like CLIP (Radford et al., 2021), audio-language encoders (ALEs) have

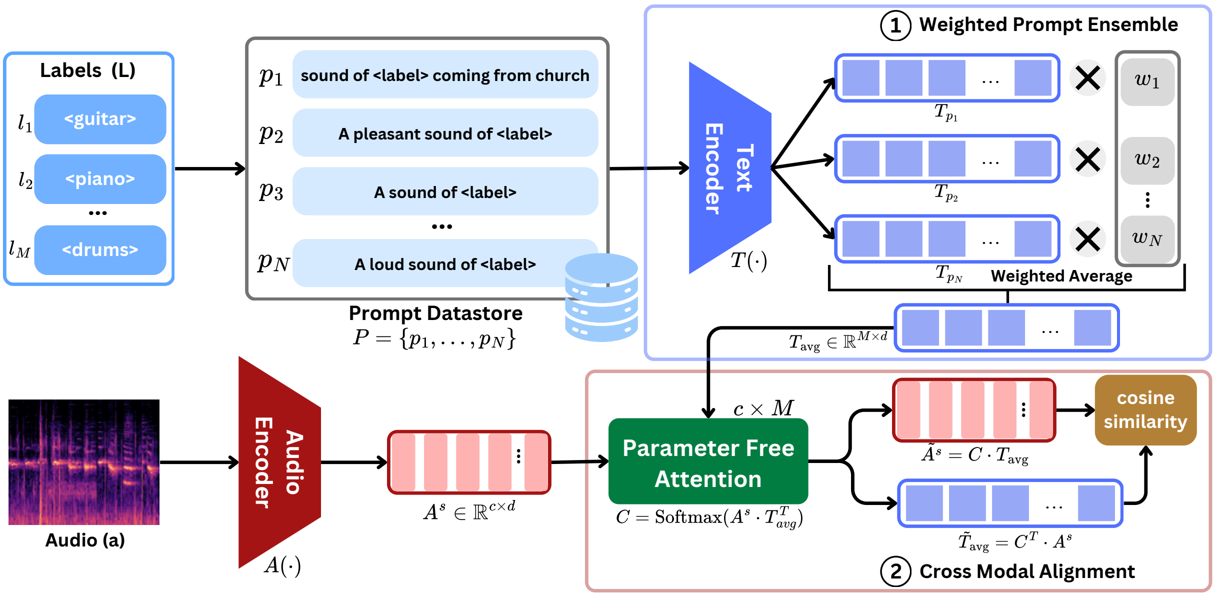

Figure 2: Illustration of PAT. PAT improves the zero-shot capabilities of ALEs by enriching audio-text representations in a parameter and training-free fashion. PAT consists of two major components: ① *Weighted prompt ensemble* that first utilizes an in-house generic prompt datastore to transform class labels into diverse textual descriptions, which are then encoded by a text encoder. Further, each prompt is assigned a unique score based on the level of uncertainty it introduces during zero-shot prediction (less uncertainty results in a higher score). A weighted average is then performed to generate task-specific, semantically rich textual representations. ② Next, the enriched textual representations are used to guide the enhancement of audio representations using a novel zero-shot *cross model alignment*. Precisely, frame-level audio representations are paired with enhanced textual representations to compute a parameter-free attention map, which is used in performing audio and text-guided transformations. Finally, the transformed frame-level audio representations are pooled, and the audio-text-guided information is added to the original logit space, boosting the ALE's zero-shot prediction capabilities.

made advances in audio-language understanding, achieving state-of-the-art zero-shot performance across various audio classification tasks. Early efforts, such as Wav2Clip (Wu et al., 2022) and AudioCLIP (Guzhov et al., 2022), focused primarily on aligning audio representations with labels. More recent approaches, like CLAP (Wu* et al., 2023; Elizalde et al., 2023a) have shifted towards explicitly mapping audio representations to textual descriptions, leading to significant improvements in zero-shot performance across diverse audio classification tasks. However, most prior ALEs have relied on compute-intensive strategies, such as refining existing multimodal alignment objectives during pre-training (Ghosh et al., 2023; Kim et al., 2024; Ghosh et al., 2024a) or increasing the number of parameters and training data (Elizalde et al., 2023a). In contrast, training-free approaches to improve zero-shot performance for ALEs remain largely underexplored.

Downstream Adaptation of ALEs for Audio Classification. Recent methods have employed various parameter-efficient transfer learning approaches, primarily inspired by prompt learning (Brown, 2020) and adapter fine-tuning (Houlsby et al., 2019). For instance, Treff Adapters (Liang et al., 2023) introduce an additional learnable cross-attention module to enhance cross-modal interaction in few-shot settings. Other approaches, such as Audio Prompt Learners (Li et al., 2024) and PALM (Hanif et al., 2024), use soft prompting to append learnable prompt tokens to textual representations. Although these methods show improvements in audio classification tasks, they fail to leverage the core advantage of ALEs, which is zero-shot transfer.

3 Preliminaries

3.1 Zero-shot transfer in ALEs

Let D be a downstream audio classification dataset with M class labels, $L \leftarrow \{l_1, \dots, l_M\}$. First, we insert these labels into a fixed prompt template p, such as "The sound of <label>," where the "<label>" token is replaced with each unique class label. This creates M textual inputs: $p(L) \leftarrow p(l_1), \dots, p(l_M)$. Next, we use text encoders $T(\cdot)$ to transform these textual inputs into latent representations: $T_p \leftarrow T_p(l_1), \dots, T_p(l_M)$. Similarly, for each audio input a_i, we use audio encoders $A(\cdot)$ to obtain frame-level audio representations $A_i^s \in \mathcal{R}^{c \times d}$, which we then pool to produce compact representations $A_i \in \mathcal{R}^{1 \times d}$. To ensure consis-

(1) Prompt Template: A <*attribute*> sound of <label>
Attribute: “loud”, “pleasant”, etc.
Example: “A loud sound of a car”, “a pleasant sound of a guitar.”
Paraphrased: “A car produces a loud sound.”, “A guitar emits a pleasant sound.”

(2) Prompt Template: A sound of <label> coming from <source>
source: “church”, “garden”, etc.
Example: “A sound of bells coming from a church”, “a sound of a cat coming from a garden”.
Paraphrased: “Bells ringing can be heard from the church.”, “The sound of a cat is coming from the garden.”

Figure 3: Examples of prompt templates and their paraphrased versions used in the prompt datastore.

tent dimensionality between audio and text representations, we pass both through fully connected layers of the same size. Finally, we compare the features using cosine similarity to generate the classification logits: $\mathcal{L}_{\text{pred}}(a_i) \leftarrow A_i \cdot T_p^T$

3.2 Prompt Ensemble

Let $P \leftarrow \{p_1, ..., p_N\}$ be a collection of unique prompts. We first encode each prompt p_i with class labels l_j using a text encoder and then average the textual representations of similar labels: $T_{\text{avg}} \leftarrow \{\sum_{i=i}^{N} T_{p_i}(l_1)\}, ..., \sum_{i=i}^{N} T_{p_i}(l_m)\}$. We recognize that one key challenge in applying prompt ensembles to Audio-Language Encoders (ALEs) is the lack of a collection of diverse prompts specifically tailored for zero-shot audio classification.

4 Methodology

Fig.2 demonstrates our proposed method PAT. We propose two simple training-free extensions for existing zero-shot transfer learning in ALEs. To enhance textual representation, we introduce a weighted prompt ensemble that selectively identifies and scores prompts which are more relevant to an unseen downstream task. To enhance audio representation, we introduce a cross-modal alignment that utilizes a parameter-free attention mechanism to align frame-level audio representations with textual descriptions in a zero-shot setting. In upcoming subsections, we explain PAT in detail.

4.1 Prompt Datastore

We adopt prompt ensemble for zero-shot audio classification tasks by developing a robust prompt datastore for ALEs. Specifically, our prompt datastore consists of 400 semantically and linguistically diverse prompts. These prompts are designed to minimize the distribution shift between the zero-shot setting and training data, where audio captions typically contain more than a single-word description. We first use GPT-4o (OpenAI et al., 2024) to generate 600 prompts by providing diverse templates and examples to guide generation. To ensure semantic diversity in our prompts, we design a variety of prompt-generation templates to avoid vague or overly open-ended prompts, such as "How does the sound <label> occur?". Furthermore, to ensure linguistic diversity, we ask GPT-4o to generate a paraphrased version of the initial set of generated prompts. Finally, we conduct a manual filtering process, selecting 400 prompts by discarding those that are inaccurate or repetitive. Fig. 3 shows a few examples of prompt templates (See Appendix 4 for more examples of prompts generated).

4.2 Weighted Prompt Ensemble

Motivation. Through our experiments, we discover that despite the vanilla prompt ensemble method consistently outperforming single prompts like “The sound of <label>”, not all prompts in a large prompt datastore P are equally relevant for unseen downstream tasks. To address this in VLEs, prior work has utilized prompt engineering to create task-specific prompts by analyzing output labels for unseen tasks (Zhou et al., 2022). However, we argue that such an approach requires additional manual effort and is not scalable across diverse downstream tasks.

In response to this, we propose a Weighted Prompt Ensemble (WPE) algorithm in PAT to adapt prompt ensembles across unseen downstream tasks by scoring individual prompts automatically without requiring additional parameters or training. Our approach is guided by a simple intuition: *Given a set of audio data and class labels, prompts that introduce greater uncertainty in label predictions should receive lower scores, and vice versa.* In line with the prior works (Zhang and Xiang, 2023), we quantify uncertainty in model predictions using *max prediction logits*. As shown in Fig. 2, for a given prompt p in prompt datastore P, audio representation A, and textual representation

Algorithm 1 Weighted Prompt Ensemble

Require: Downstream task D; Audio Representations A; Textual Representation, T for N prompts with M class labels; Prompt scores $w \leftarrow \{w_1, \ldots, w_N\}$; N no of prompts: $p \in \{p_1, p_2, ..., p_\text{N}\}$

for $p \leq N$ **do**

  Compute prediction logits $\mathcal{L}_p \leftarrow A \cdot T_p^T$

  Compute max prediction logits: $\mathcal{L}_p^{max} \leftarrow max_j \, \mathcal{L}_p[i, j]$.

  Compute prompt score $w_p \leftarrow \sum_{i=1}^{|D|} \mathcal{L}_{p,d}^{max}$

  Update prompt score $w \leftarrow w \cup \{w_p\}$

end for

Normalize prompt scores $w \leftarrow \text{Softmax}(w)$

Perform weighted prompt ensemble: $T_\text{avg} \leftarrow \sum_{p=1}^{N} w_p \times T_p$

$T_p \in \mathbb{R}^{M \times d}$, where M, d are the number of class labels and embedding dimensions respectively, we first compute the prediction logits $\mathcal{L}_p \leftarrow A \cdot T_p^T$. The prompt score w_p is then determined by calculating a cumulative sum of maximum prediction logits. We formally define this as:

$$w_p = \sum_{i=1}^{|D|} \text{max}_j \mathcal{L}_\text{pred}[i, j], \forall j \in \{1, \ldots, M\} \quad (1)$$

where $|D|$ is the number of samples in an unseen downstream task D. Finally, we apply softmax normalization to the prompt scores and perform a weighted averaging of the prompt-specific textual representations to obtain enriched textual representation T_avg, as shown below:

$$T_\text{avg} = \sum_{p=1}^{N} w_p * T_p \quad (2)$$

where N is the number of unique prompts. We summarize the complete process of our weighted prompt ensemble in Algorithm 1

4.3 Cross-Modal Alignment

Motivation. We observe that ALEs encode audio and text modalities independently, preventing them from utilizing cross-modal interactions before making final predictions. Additionally, audio representations are typically average-pooled, leading to a loss of fine-grained information in the audio.

Inspired by our findings, we propose parameter-free cross-modal alignment in PAT to improve the audio-text alignment in ALEs in a zero-shot fashion. As shown in Fig. 2, we utilize the intermediate frame-level audio representations $A^s \in \mathbb{R}^{c \times d}$ and enriched textual representations $T_{avg} \in \mathbb{R}^{M \times d}$ to compute parameter free attention weights $c_{i,j} \in C$ that denotes the correlation between individual audio frames and class labels. We formulate this below:

$$C = A^s \cdot T_{avg}^T \in \mathbb{R}^{c \times M} \quad (3)$$

Next, we utilize our parameter-free attention weights C to enhance cross-modal alignment between the audio and textual representations. To achieve this, we project the original audio representation A^s and the average textual representation T_avg using C as follows:

$$\tilde{A}^s = \text{Softmax}(C) \cdot T_\text{avg} \in \mathbb{R}^{c \times d} \quad (4)$$

$$\tilde{T}_\text{avg} = \text{Softmax}(C^T) \cdot A^s \in \mathbb{R}^{M \times d} \quad (5)$$

where $\tilde{A}^s$ represents the audio-guided representations that amplify the information of labels strongly associated with the audio, and $\tilde{T}_\text{avg}$ represents the text-guided representations that enhance the information of the audio frames strongly correlated with specific labels. We then pool the transformed frame-level audio representation $\tilde{A}^s$ as follows: $\tilde{A} \leftarrow \text{AvgPool}(\tilde{A}^s)$. The transformed representations $\tilde{A}$ and $\tilde{T}_\text{avg}$ are then used to create audio-guided and text guide logits, $\mathcal{L}_\text{audio} \leftarrow \tilde{A} \cdot T_\text{avg}^T$ and $\mathcal{L}_\text{text} \leftarrow A \cdot \tilde{T}_\text{avg}^T$ respectively. Finally, we combine the audio and text-guided logits with the original logits $\mathcal{L}_\text{pred} \leftarrow A \cdot T_\text{avg}^T$. We formulate this below:

$$\mathcal{L}_\text{combine} = \mathcal{L}_\text{pred} + \beta_1 \mathcal{L}_\text{audio} + \beta_2 \mathcal{L}_\text{text} \quad (6)$$

where β_1 and β_2 are the hyper-parameter weights for audio and text-guided logits, respectively. Additionally, we provide details on hyper-parameter tuning in Appendix E

5 Experimental Setup

5.1 Evaluation Datasets and Metric

For zero-shot evaluation, we utilized 18 open-source audio classification datasets covering a broad range of musical and non-verbal audio types. Specifically, for music-related downstream tasks such as instrument and genre classification, we present results on eight widely-used musical datasets, including NSynth (NSynth Inst/Src) (Engel et al., 2017b), Beijing Opera (Beijing Op) (Tian et al., 2014), MedleyDB (Bittner et al., 2014),

Model → **Dataset ↓**	**L-CLAP**		**LM-CLAP**		**MSCLAP-22**		**MSCLAP-23**		**Wav2CLIP**		**CompA**	
	ZS	PAT	ZS	PAT	ZS	PAT	ZS	PAT	ZS	PAT	ZS	PAT
Sound												
ESC-50	89.00	**93.00** +4.00%	85.60	**92.65** +7.05%	76.95	**78.35** +1.40%	91.80	**94.80** +3.00%	24.85	**31.60** +6.08%	91.35	**93.20** +1.85%
USD-8K	76.00	**80.00** +4.00%	28.09	**39.93** +11.84%	72.54	**74.80** +2.26%	77.70	**82.50** +4.80%	20.97	**22.69** +1.72%	73.53	**78.32** +4.79%
TUT	36.00	**39.00** +3.00%	28.09	**39.93** +11.84%	24.44	**25.61** +1.17%	45.00	**47.00** +2.00%	11.54	**15.18** +3.64%	40.12	**46.28** +6.16%
VS	78.20	**80.00** +1.80%	74.46	78.91 +4.45%	43.78	**54.94** +11.16%	79.00	**79.60** +0.60%	22.72	**24.06** +1.31%	65.22	**71.26** +6.04%
DCASE	44.88	**50.81** +5.93%	**56.76**	55.94 -0.82%	13.93	**23.77** +9.84%	45.90	**45.96** +0.06%	09.63	**17.21** +7.58%	33.20	**34.29** +1.09%
Gunshot Tri	10.23	**22.72** +12.49%	13.64	**29.52** +15.88%	17.05	**23.86** +6.81%	25.00	**25.00** +0.00%	**25.00**	25.00 +0.00%	25.00	**26.15** +1.15%
SESA	67.72	**74.28** +6.56%	72.38	**79.04** +6.66%	66.67	**68.47** +1.80%	70.48	**71.61** +1.13%	29.52	**56.10** +26.58%	64.76	**69.42** +4.66%
AudioSet	31.88	**36.98** +5.10%	33.12	38.21 +5.09%	16.10	**17.81** +1.71%	25.33	**28.73** +3.40%	18.03	**20.12** +2.09%	33.24	**35.12** +1.88%
FSD50K	46.45	**48.76** +2.31%	47.12	**49.10** +2.08%	32.50	**33.80** +1.30%	44.49	**45.52** +1.02%	42.31	**44.14** +2.07%	42.18	**43.22** +1.04%
Cochlscene	38.56	**48.66** +10.10%	50.66	55.35 +4.69%	25.94	**33.51** +7.57%	85.00	**85.22** +0.22%	13.09	**16.11** +3.02%	31.95	**38.21** +6.26%
Music												
Beijing Op.	45.34	**68.64** +23.30%	75.00	**75.42** +0.42%	54.24	**73.72** +19.48%	71.19	**71.61** +0.42%	26.69	**34.32** +7.63%	61.86	**63.21** +1.35%
GTZAN	43.40	**54.20** +10.80%	63.92	**63.93** +0.01%	19.19	**20.75** +1.56%	56.24	**58.56** +2.32%	**30.00**	27.76 -2.24%	50.22	**52.17** +1.95%
MUSDB	55.60	**66.00** +10.40%	73.20	**73.20** +0.00%	47.20	**47.75** +0.55%	61.20	**62.40** +1.20%	51.60	**52.20** +0.60%	56.80	**59.55** +2.75%
Medley	82.50	**92.00** +9.50%	87.88	94.30 +6.42%	84.41	**86.20** +1.79%	45.00	**47.00** +2.00%	42.20	**47.08** +4.88%	56.27	**57.24** +0.97%
Mri. St	10.81	**37.35** +26.54%	47.40	**47.80** +0.40%	14.50	**14.80** +0.30%	44.09	**47.12** +3.03%	06.09	**19.49** +13.40%	06.25	**07.42** +1.17%
Mri. To	25.10	**34.38** +9.28%	27.59	**31.62** +4.03%	16.50	**16.63** +0.13%	22.02	**26.18** +4.16%	15.57	**24.95** +9.38%	17.43	**18.79** +1.36%
NSynth Inst	37.20	**38.00** +0.80%	31.67	**36.49** +4.82%	26.26	**29.63** +3.37%	63.30	**66.30** +3.00%	**24.39**	21.72 -2.67%	27.86	**29.24** +1.38%
NSynth Src	37.00	**41.00** +4.00%	43.92	**46.38** +2.46%	37.06	**41.45** +4.39%	49.70	**61.45** +11.75%	38.28	**42.01** +3.73%	53.66	**55.97** +2.31%

Table 1: Performance comparison between PAT and vanilla zero-shot classification (ZS) across 6 ALEs and 18 diverse audio classification tasks, including 10 sound and 8 musical datasets. The best scores for each ALE are in **bold**. Overall, PAT outperforms vanilla ZS with improvements ranging from 0.42% to 27%.

MUSDB (Rafii et al., 2017a), GTZAN (Tzanetakis et al., 2001), and Mridangam Stroke/Tonic (Mri. St/ Mri. To) (Turian et al., 2022). Additionally, we report results on non-verbal sound classification tasks using datasets like ESC-50 (Piczak), Urban-Sound8K (USD8K)(Salamon et al., 2014), TUT-Urban (TUT)(Mesaros et al., 2018), VocalSound (VS)(Gong et al., 2022), SESA(Spadini, 2019), CochlScene (Jeong and Park, 2022), DCASE Task 4 (Mesaros et al., 2017a), and Gunshot Triangulation (GT) (Turian et al., 2022). We also conduct an evaluation on multi-label datasets like FSD50K (Fonseca et al., 2021) and AudioSet (Gemmeke et al., 2017). For zero-shot evaluation, we report the mean average precision (mAP) for AudioSet and FSD50K and accuracy for the other datasets, which averaged over 5 runs. Appendix C provides additional dataset details.

5.2 Baselines

We demonstrate the scalability and robustness of the PAT across five open-source audio-language encoders (ALEs). Specifically, we integrate PAT with Wav2CLIP (Wu et al., 2022), trained with 200k audio-label pairs and various CLAP models, including LAION CLAP (L-CLAP) (Wu* et al., 2023), trained on over 633K audio-text pairs; CompA CLAP, which is further trained on 60K compositionally aware audio-text pairs; LAION CLAP MUSIC (LM-CLAP), which is further trained on music datasets; MS CLAP-22 (Elizalde et al., 2023b), trained on 128K+ audio-text pairs; and MS CLAP-23 (Elizalde et al., 2023a), which is trained on 4.6M pairs covering diverse audio types such as speech, music, and non-verbal sounds. We re-evaluate the zero-shot accuracy of publicly available ALEs under similar compute settings using A6000 GPUs.

5.3 Audio Augmentations

We further evaluate PAT's robustness in noisy settings by augmenting audio with various kinds of audio augmentations. Specifically, we apply *gaussian noise* to simulate real-world background noise, *pitch shift* to test frequency variation, and *polarity inversion* to check the model's sensitivity to phase changes. Additionally, we use *delay* to introduce echo effects, *gain* to assess performance under varying volume levels, and both *low pass* and *high pass filters* to evaluate the PAT ability to handle reduced frequency ranges. Lastly, we use *reverb* to simulate different reverberant acoustic environments. Appendix D provides additional details on various audio augmentations.

6 Results and Result Analysis

6.1 Main Results

We summarize the results of PAT applied to 6 different ALEs across 18 audio classification datasets, including 10 non-verbal speech and non-speech

Method	Average Accuracy
MSCLAP-23	58.28
MSCLAP-23+PE	58.91
MSCLAP-23+WPE	59.23
MSCLAP-23+PE+CMA	59.32
MSCLAP-23+ PAT (WPE+CMA)	**60.76**

Table 2: Performance comparison of MSCLAP-23 using various components of PAT including Prompt Ensemble (PE), Weighted Prompt Ensemble (WPE), Cross-Modal Alignment (CMA) and their combinations. Baseline ZS scores are highlighted in grey. Overall, WPE consistently outperforms vanilla PE, both with and without CMA.

Logits	Audio Guided	Text Guided	Average Accuracy
✓	×	×	58.28
✓	×	✓	59.24
✓	✓	×	59.55
×	✓	✓	58.29
✓	✓	✓	**60.76**

Table 3: Performance comparison of MS-CLAP with PAT across different combinations of original, audio-guided, and text-guided logits with Cross-Modal Alignment (CMA). Baseline ZS scores are highlighted in grey. Overall, incorporating all three logits in the final prediction yields the best zero-shot performance.

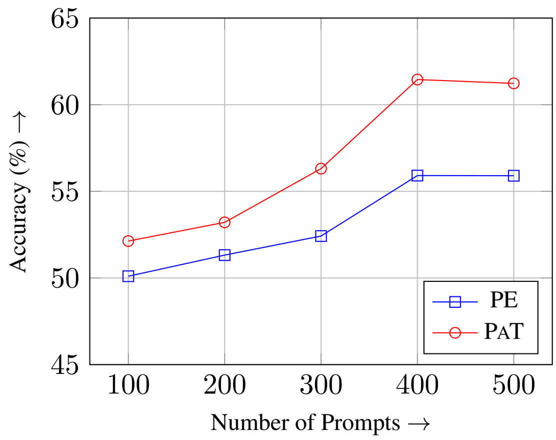

Figure 4: Measuring MSCLAP-23 zero-shot performance with the prompt ensemble (PE) vs. PAT on NSynth-Src by varying the prompt count. Generally, PAT outperforms PE across different prompt counts.

Dataset	Prompt	Score
ESC-50	The sound of <label> coming from a cliff edge.	0.0035
	A sound of a <label> coming from a parking lot	0.0033
NSynth Inst	A major sound of a <label>	0.0038
	A minimal sound of a <label>	0.0037

Table 4: Top two highest scoring prompt by PAT for MSCLAP-23 on ESC-50 and Nsynth-Inst

sound datasets and 8 musical datasets, in Table 1. Our key finding can be summarized as follows: 1) PAT consistently outperforms the vanilla zero-shot (ZS) approach across all baselines and datasets, achieving an absolute improvement of 0.42%-27%. This underlines PAT strength to generalize across diverse audio classification tasks and scale with different ALEs in a training-free fashion. 2) PAT shows a remarkable performance boost even for less-pretrained ALEs. For instance, LAION-CLAP (Wu* et al., 2023) pre-trained on 0.6M audio-text pairs, gains an 11% boost on MUSDB (Rafii et al., 2017b) when combined with PAT, even surpassing MS CLAP (Elizalde et al., 2023a) by 5%, which has been pre-trained on 4.6M pairs. We attribute such gains to our cross-modal aligner that utilizes enriched textual representations to reweigh audio representations. 3) PAT shows a slight performance degradation on ALEs like Wav2CLIP (Wu et al., 2022). As mentioned by (Elizalde et al., 2023a), these models are pre-trained solely on audio labels rather than textual descriptions, thus limiting them to interpret prompts.

6.2 Ablation on various PAT Components

We conduct extensive ablations on individual PAT components, as shown in Table 2 and Table 3. For all ablation studies, we use the best-performing zero-shot model, MS-CLAP, unless stated otherwise. In Table 2, we compare the performance gains from using Prompt Ensemble (PE) versus Weighted Prompt Ensemble (WPE), which is employed in PAT, averaging the zero-shot accuracy across all datasets. Overall, both PE and WPE outperform the baseline, with WPE consistently outperforming PE even with or without Cross-Modal Alignment (CMA). With this, we show that selectively scoring prompts that reduce uncertainty in model predictions, as opposed to using uniform scoring, leads to improved zero-shot performance. Table 3 further explores CMA by ablating different combinations of original, audio-, and text-guided logits. We show that combining all the logits gives the biggest improvement in zero-shot performance.

6.3 Ablation on Number of Prompts

Fig. 4 shows the effect of increasing the number of prompts when applying the weighted prompt ensemble in PAT. The key findings are: 1) While adding more prompts to the prompt datastore improves zero-shot performance in PAT, the performance of MSCLAP-23 plateaus after reaching 400 prompts. This suggests that naively increasing the number of prompts may not lead to further gains

Dataset	Gaussian Noise		Pitch Shift		Polarity Inversion		Gain		High Pass	
	ZS	PAT	ZS	PAT	ZS	PAT	ZS	PAT	ZS	PAT
Sound										
ESC-50	91.80	**94.20**$_{+2.40\%}$	78.05	**80.10**$_{+2.05\%}$	91.85	**94.40**$_{+2.55\%}$	92.05	**94.85**$_{+2.80\%}$	82.35	**86.15**$_{+3.80\%}$
USD8K	77.26	**82.70**$_{+5.44\%}$	63.61	**70.31**$_{+6.70\%}$	77.43	**82.69**$_{+5.26\%}$	77.08	**82.67**$_{+5.59\%}$	71.12	**76.77**$_{+5.65\%}$
TUT	44.94	**45.74**$_{+0.80\%}$	**26.05**	26.04$_{-0.01\%}$	45.68	**47.34**$_{+1.66\%}$	38.95	**41.97**$_{+3.02\%}$	**35.80**	35.00$_{-0.80\%}$
VS	**81.31**	77.86$_{-3.45\%}$	**76.61**	69.64$_{-6.97\%}$	**78.98**	78.00$_{-0.98\%}$	79.00	**79.44**$_{+0.44\%}$	74.07	**76.16**$_{+2.09\%}$
DCASE	38.32	**42.21**$_{+3.89\%}$	31.76	**34.01**$_{+2.25\%}$	38.93	**45.69**$_{+6.76\%}$	43.24	**45.28**$_{+2.04\%}$	33.40	**37.70**$_{+4.30\%}$
Gunshot Tri.	**25.00**	**25.00**$_{+0.00\%}$	**25.00**	**25.00**$_{+0.00\%}$	**25.00**	**25.00**$_{+0.00\%}$	**25.00**	**25.00**$_{+0.00\%}$	19.32	**22.72**$_{+3.40\%}$
SESA	67.62	**69.52**$_{+1.90\%}$	62.86	**64.76**$_{+1.90\%}$	67.62	**69.52**$_{+1.90\%}$	68.57	**69.52**$_{+0.95\%}$	48.57	**58.10**$_{+9.53\%}$
AudioSet	30.40	**31.15**$_{+0.75\%}$	22.37	**23.06**$_{+0.69\%}$	**30.40**	29.22$_{-1.18\%}$	28.78	**30.23**$_{+1.45\%}$	23.89	**24.38**$_{+0.49\%}$
FSD50K	44.54	**45.74**$_{+1.20\%}$	37.16	**43.87**$_{+6.71\%}$	44.39	**44.96**$_{+0.57\%}$	**44.56**	43.79$_{-0.77\%}$	37.94	**43.73**$_{+5.79\%}$
Cochlscene	**85.07**	84.36$_{-0.71\%}$	60.18	**61.42**$_{+1.24\%}$	85.07	**85.17**$_{+0.10\%}$	81.97	**82.09**$_{+0.12\%}$	73.34	**75.15**$_{+1.81\%}$
Music										
Beijing Op.	70.34	**70.62**$_{+0.28\%}$	61.02	**62.74**$_{+1.72\%}$	71.19	**71.61**$_{+0.42\%}$	69.49	**69.61**$_{+0.12\%}$	**65.68**	64.86$_{-0.82\%}$
GTZAN	55.77	**58.53**$_{+2.76\%}$	47.56	**50.38**$_{+2.82\%}$	56.43	**58.26**$_{+1.83\%}$	55.20	**57.79**$_{+2.59\%}$	47.36	**50.54**$_{+3.18\%}$
MUSDB	**63.60**	53.60$_{-10.00\%}$	58.00	**61.60**$_{+3.60\%}$	**68.00**	56.80$_{-11.20\%}$	**68.00**	58.40$_{-9.60\%}$	46.80	**55.20**$_{+8.40\%}$
Medley	**96.61**	95.96$_{-0.65\%}$	92.09	**92.46**$_{+0.37\%}$	95.98	**96.42**$_{+0.44\%}$	95.97	**96.53**$_{+0.56\%}$	**93.37**	90.94$_{-2.43\%}$
Mri. St.	42.63	**48.93**$_{+6.30\%}$	33.15	**44.94**$_{+11.79\%}$	44.09	**47.12**$_{+3.03\%}$	41.31	**44.67**$_{+3.36\%}$	34.73	**37.30**$_{+2.57\%}$
Mri. To.	24.97	**26.61**$_{+1.64\%}$	13.54	**17.12**$_{+3.58\%}$	22.02	**26.18**$_{+4.16\%}$	19.08	**26.78**$_{+7.70\%}$	**17.24**	16.59$_{-0.65\%}$
NSynth Inst	52.86	**53.85**$_{+0.99\%}$	60.11	**64.77**$_{+4.66\%}$	63.89	**66.33**$_{+2.44\%}$	61.89	**64.62**$_{+2.73\%}$	46.22	**47.87**$_{+1.65\%}$
NSynth Src	39.75	**47.85**$_{+8.10\%}$	49.44	**59.37**$_{+9.93\%}$	49.76	**61.45**$_{+11.69\%}$	49.49	**60.64**$_{+11.15\%}$	47.39	**56.46**$_{+9.07\%}$

Table 5: Zero-shot performance measure of MSCLAP-23 using PAT across 18 audio classification tasks under noisy setting. Each audio sample is subjected to 5 different varieties of audio augmentations. PAT outperforms vanilla zero-shot (ZS) classification scores by showing an absolute improvement of 0.10–11.15%.

in zero-shot performance for audio classification tasks. 2) The Weighted Prompt Ensemble (WPE) used in PAT consistently surpasses the standard Prompt Ensemble (PE), even with fewer prompts. Table 4 lists the top 2 highest-scoring prompts generated by MSCLAP-23 using PAT on both sound datasets (e.g., ESC-50) and music datasets (e.g., NSynth Inst). We observe that, for sound datasets, prompts associating sounds to random sources like "parking lot" or "cliff edge" receive higher scores. In contrast, for music datasets, prompts describing random audio attributes like "minimal" or "loud" tend to score higher. We extend this analysis in Appendix G, where we show the top 10 best-scoring prompts for each downstream task.

6.4 Zero-Shot Evaluation under Noisy Setting

Table 5 shows the performance improvement of MSCLAP-23 using PAT across 16 audio classification tasks under noisy conditions, compared to the baseline zero-shot scores. In particular, individual audio samples in each task are subjected to five different types of augmentation, which modify key audio features such as pitch, frequency, amplitude, and phase. Under noisy settings, PAT is able to show promising results by achieving an absolute improvement of 0.10-11.15% when compared with simple zero-shot evaluation. This highlights the robustness of PAT to classify out-of-distribution audio samples without the need for additional few-shot data or training. Additionally, we provide results on 3 other audio augmentations: delay, low pass and reverb in Appendix D

7 Conclusion

We introduce PAT, a parameter- and training-free approach to improve zero-shot audio classification for audio-language encoders (ALEs). Our studies indicate that current methods that guarantee audio classification improvements are not capable of leveraging the core strength of ALEs, which is zero-shot transfer and often require extensive training or additional parameters to adapt audio and textual representations for an unseen task. To overcome this limitation, PAT offers two key contributions: 1) a novel prompt scoring method, called weight prompt ensemble, which adapts prompt ensembling for unseen audio classification tasks in a training-free fashion, and 2) a cross-modal alignment framework, which uses parameter-free attention to better align audio and textual representation. PAT significantly boosts zero-shot performance in ALEs across various audio classification datasets.

8 Acknowledgements

This project is supported in part by NSF#1910940.

9 Limitation and Future Work

In this section, we highlight a few limitations and potential future direction of our proposed method, PAT

- Due to compute constraints, we did not evaluate PAT across speech classification tasks. In future, we plan to release PAT evaluation scores against various speech-related downstream tasks such as Keyword spotting, Emotion Recognition, etc.
- We acknowledge that PAT increases the space complexity of existing zero-shot evaluation algorithms by introducing a prompt datastore.
- While PAT shows significant improvements, it is limited by ALE's existing knowledge space to classify unknown sounds. In future, we plan to integrate parameter-efficient methods such as soft prompting to make PAT robust towards the new evolving sounds.

References

Akshay Anantapadmanabhan, Ashwin Bellur, and Hema A. Murthy. 2014. Mridangam stroke dataset (1.0). 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, Vancouver, BC, Canada. Data set.

Thierry Bertin-Mahieux, Daniel P.W. Ellis, Brian Whitman, and Paul Lamere. 2011. The million song dataset. In *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR 2011)*.

Rachel M Bittner, Justin Salamon, Mike Tierney, Matthias Mauch, Chris Cannam, and Juan Pablo Bello. 2014. Medleydb: A multitrack dataset for annotation-intensive mir research. In *ISMIR*, volume 14, pages 155–160.

Tom B Brown. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Benjamin Elizalde, Soham Deshmukh, Mahmoud Al Ismail, and Huaming Wang. 2023a. Clap learning audio concepts from natural language supervision. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

Benjamin Elizalde, Soham Deshmukh, Mahmoud Al Ismail, and Huaming Wang. 2023b. Clap learning audio concepts from natural language supervision. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

Jesse Engel, Cinjon Resnick, Adam Roberts, Sander Dieleman, Douglas Eck, Karen Simonyan, and Mohammad Norouzi. 2017a. Neural audio synthesis of musical notes with wavenet autoencoders.

Jesse Engel, Cinjon Resnick, Adam Roberts, Sander Dieleman, Mohammad Norouzi, Douglas Eck, and Karen Simonyan. 2017b. Neural audio synthesis of musical notes with wavenet autoencoders. In *International Conference on Machine Learning*, pages 1068–1077. PMLR.

Eduardo Fonseca, Xavier Favory, Jordi Pons, Frederic Font, and Xavier Serra. 2021. Fsd50k: an open dataset of human-labeled sound events. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:829–852.

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

Sreyan Ghosh, Sonal Kumar, Chandra Kiran Reddy Evuru, Oriol Nieto, Ramani Duraiswami, and Dinesh Manocha. 2024a. Reclap: Improving zero shot audio classification by describing sounds. *arXiv preprint arXiv:2409.09213*.

Sreyan Ghosh, Sonal Kumar, Ashish Seth, Chandra Kiran Reddy Evuru, Utkarsh Tyagi, S Sakshi, Oriol Nieto, Ramani Duraiswami, and Dinesh Manocha. 2024b. Gama: A large audio-language model with advanced audio understanding and complex reasoning abilities. *arXiv preprint arXiv:2406.11768*.

Sreyan Ghosh, Ashish Seth, Sonal Kumar, Utkarsh Tyagi, Chandra Kiran Evuru, S Ramaneswaran, S Sakshi, Oriol Nieto, Ramani Duraiswami, and Dinesh Manocha. 2023. Compa: Addressing the gap in compositional reasoning in audio-language models. *arXiv preprint arXiv:2310.08753*.

Yuan Gong, Jin Yu, and James Glass. 2022. Vocalsound: A dataset for improving human vocal sounds recognition. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 151–155.

Andrey Guzhov, Federico Raue, Jörn Hees, and Andreas Dengel. 2022. Audioclip: Extending clip to image, text and audio. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 976–980. IEEE.

Asif Hanif, Maha Tufail Agro, Mohammad Areeb Qazi, and Hanan Aldarmaki. 2024. Palm: Few-shot prompt learning for audio language models. *arXiv preprint arXiv:2409.19806*.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for nlp. In *International conference on machine learning*, pages 2790–2799. PMLR.

Il-Young Jeong and Jeongsoo Park. 2022. Cochlscene: Acquisition of acoustic scene data using crowdsourcing. In *2022 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pages 17–21. IEEE.

Jaeyeon Kim, Jaeyoon Jung, Jinjoo Lee, and Sang Hoon Woo. 2024. Enclap: Combining neural audio codec and audio-text joint embedding for automated audio captioning. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6735–6739. IEEE.

Yiming Li, Xiangdong Wang, and Hong Liu. 2024. Audio-free prompt tuning for language-audio models. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 491–495.

Jinhua Liang, Xubo Liu, Haohe Liu, Huy Phan, Emmanouil Benetos, Mark D Plumbley, and Wenwu Wang. 2023. Adapting language-audio models as few-shot audio learners. *arXiv preprint arXiv:2305.17719*.

Annamaria Mesaros, Toni Heittola, Aleksandr Diment, Benjamin Elizalde, Ankit Shah, Emmanuel Vincent, Bhiksha Raj, and Tuomas Virtanen. 2017a. Dcase 2017 challenge setup: Tasks, datasets and baseline system. In *DCASE 2017-Workshop on Detection and Classification of Acoustic Scenes and Events*.

Annamaria Mesaros, Toni Heittola, and Tuomas Virtanen. 2017b. Tut acoustic scenes 2017, development dataset. Data set.

Annamaria Mesaros, Toni Heittola, and Tuomas Virtanen. 2018. A multi-device dataset for urban acoustic scene classification. In *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018)*, pages 9–13.

OpenAI, Josh Achiam, and Others. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Karol J. Piczak. ESC: Dataset for Environmental Sound Classification. In *Proceedings of the 23rd Annual ACM Conference on Multimedia*, pages 1015–1018. ACM Press.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR.

Zafar Rafii, Antoine Liutkus, Fabian-Robert Stöter, Stylianos Ioannis Mimilakis, and Rachel Bittner. 2017a. The MUSDB18 corpus for music separation.

Zafar Rafii, Antoine Liutkus, Fabian-Robert Stöter, Stylianos Ioannis Mimilakis, and Rachel Bittner. 2017b. The musdb18 corpus for music separation.

Avraham Ruderman, Neil C Rabinowitz, Ari S Morcos, and Daniel Zoran. 2018. Pooling is neither necessary nor sufficient for appropriate deformation stability in cnns. *arXiv preprint arXiv:1804.04438*.

J. Salamon, C. Jacoby, and J. P. Bello. 2014. A dataset and taxonomy for urban sound research. In *22nd ACM International Conference on Multimedia (ACM-MM'14)*, pages 1041–1044, Orlando, FL, USA.

Tito Spadini. 2019. Sound events for surveillance applications (1.0.0). Data set.

Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. 2014. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*.

Mi Tian, Ajay Srinivasamurthy, Mark Sandler, and Xavier Serra. 2014. A study of instrument-wise onset detection in beijing opera percussion ensembles. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2159–2163.

Joseph Turian, Jordie Shier, Humair Raj Khan, Bhiksha Raj, Björn W Schuller, Christian J Steinmetz, Colin Malloy, George Tzanetakis, Gissel Velarde, Kirk McNally, et al. 2022. Hear: Holistic evaluation of audio representations. In *NeurIPS 2021 Competitions and Demonstrations Track*, pages 125–145. PMLR.

George Tzanetakis, Georg Essl, and Perry Cook. 2001. Automatic musical genre classification of audio signals.

Ho-Hsiang Wu, Prem Seetharaman, Kundan Kumar, and Juan Pablo Bello. 2022. Wav2clip: Learning robust audio representations from clip. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4563–4567. IEEE.

Yusong Wu*, Ke Chen*, Tianyu Zhang*, Yuchen Hui*, Taylor Berg-Kirkpatrick, and Shlomo Dubnov. 2023. Large-scale contrastive language-audio pretraining with feature fusion and keyword-to-caption augmentation. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*.

Shu-wen Yang, Po-Han Chi, Yung-Sung Chuang, Cheng-I Jeff Lai, Kushal Lakhotia, Yist Y Lin, Andy T Liu, Jiatong Shi, Xuankai Chang, Guan-Ting Lin, et al. 2021. Superb: Speech processing universal performance benchmark. *arXiv preprint arXiv:2105.01051*.

Zihan Zhang and Xiang Xiang. 2023. Decoupling maxlogit for out-of-distribution detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3388–3397.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. 2022. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348.

A Appendix

In the appendix, we provide:

1. Section B: Baseline Details
2. Section C: Dataset Details
3. Section D: Audio Augmentations
4. Section E: Hyper-Parameter tuning
5. Section F: Additional Details
6. Section G: Filtering Criteria For Prompt Selection

B Baseline Details

LAION-CLAP (Wu* et al., 2023). This is a contrastive language-audio pretraining (CLAP) model from LAION-AI trained on LAION-Audio-630K (Wu* et al., 2023), a large collection of 633,526 audio-text pairs from different data sources. To improve the model's ability to handle audio inputs of variable lengths and boost overall performance, it integrates a feature fusion mechanism and keyword-to-caption augmentation. This enables the model to effectively align and process both audio and text data for enhanced learning.
LAION-CLAP Music (Wu* et al., 2023). This is a music-specific version of the LAION-CLAP model. This version is trained both on audio and music, with the LAION-Audio-630K dataset contributing a major portion of its training data. The details of the music-text data being used for training are not specified.
MS-CLAP 22 (Elizalde et al., 2023b). This is a contrastive language-audio pretraining (CLAP) model from Microsoft. This version is trained on 128k audio and text pairs.
MS-CLAP 23 (Elizalde et al., 2023a). This is a follow-up to the MS-CLAP 22, from Microsoft. This version of CLAP uses two innovative encoders and is trained on massive 4.6M audio-text pairs. To learn audio representations, the authors trained an audio encoder on 22 audio tasks instead of the standard training of sound event classification. To learn language representations, they trained an autoregressive decoder-only model instead of the standard encoder-only models.
Wav2CLIP (Wu et al., 2022). This is a robust audio representation learning method by distilling from Contrastive Language-Image Pre-training (CLIP) (Radford et al., 2021). Wav2CLIP is a model that maps audio into a shared embedding space alongside images and text, enabling multi-modal tasks like zero-shot classification and cross-modal retrieval. It achieves competitive performance on downstream tasks with only about 10% of the data needed by fully supervised models. Additionally, Wav2CLIP is more efficient in pre-training, as it focuses solely on the audio modality and does not require joint training of visual and auditory models, unlike some competing methods.
CompA (Ghosh et al., 2023). This is a CLAP model that is trained specifically to enhance its compositional reasoning abilities. The authors introduce improvements to contrastive training by incorporating composition-aware hard negatives, allowing for more precise and focused training. Additionally, they propose a modular contrastive loss designed to help the model learn fine-grained compositional understanding.

C Dataset Details

ESC-50:[2] (Piczak) The ESC-50 dataset is a labelled collection of 2,000 environmental audio recordings, every 5 seconds in length. The dataset is designed for sound classification tasks and contains recordings organized into 50 semantically distinct classes, with 40 examples per class. These classes are further grouped into 5 major categories, which include Animals, Natural soundscapes & water sounds, Humans, non-speech sounds, Interior/domestic sounds, and Exterior/urban noises.
USD-8K:[3] (Salamon et al., 2014) The UrbanSound8K dataset is an audio collection that contains 8,732 labelled sound excerpts, each with a duration of up to 4 seconds. The dataset is designed to represent various urban sound environments, with recordings organized into 10 distinct classes: air_conditioner, car_horn, children_playing, dog_bark, drilling, engine_idling,

[2] https://github.com/karolpiczak/ESC-50
[3] https://urbansounddataset.weebly.com/urbansound8k.html

gun_shot, jackhammer, siren, and street_music. These classes are derived from the urban sound taxonomy

TUT Urban Acoustic Scenes (TUT):[4] (Mesaros et al., 2017b) The TUT Acoustic Scenes 2019 is a large-scale collection of environmental recordings from various urban sound environments. It consists of 10 classes, each representing a different acoustic scene, such as Airport, Metro station, Park, and Residential area. Each recording is 10 seconds long.

Vocal Sound (VS):[5] (Gong et al., 2022) The VocalSound dataset consists of 21,024 crowdsourced recordings representing 6 classes of vocal sounds: laughter, sighs, coughs, throat clearing, sneezes, and sniffs. These recordings were collected from 3,365 unique subjects, providing a diverse set of vocal sound samples. This dataset is primarily used for the study and classification of non-verbal vocal sounds.

DCASE Task 4:[6] (Mesaros et al., 2017a) The DCASE Task 4 dataset is a sound event detection dataset with heterogeneous data and missing labels. It comprises two primary datasets: DESED and MAESTRO. DESED consists of 10-second-long audio clips that are either recorded in domestic environments or synthesized to simulate such environments. These recordings contain annotated sound events from 10 different classes. MAESTRO provides audio recordings with multiple temporally strong annotated events featuring soft labels across 17 classes. However, for the purposes of DCASE Task 4, only 11 classes from MAESTRO are considered.

Gunshot Triangulation:[7] (Turian et al., 2022) The Gunshot Triangulation dataset is designed for a novel multiclass classification task involving gunshot recordings. The dataset consists of 88 audio clips representing 22 gunshots from 7 different firearms, recorded in an open field using iPod Touch devices (Cooper and Shaw, 2020). The objective of the task is to classify the audio based on the specific iPod Touch that was recorded, effectively identifying the location of the microphone during each gunshot event.

SESA:[8] (Spadini, 2019) The Sound Events for Surveillance Applications (SESA) dataset consists of audio recordings sourced from Freesound. The audio files have durations of up to 33 seconds and are categorized into 4 classes: Casual (not a threat), Gunshot, Explosion, and Siren. This dataset is intended for tasks related to sound event detection and classification in surveillance contexts.

Cochlscene:[9] (Jeong and Park, 2022)The CochlScene dataset is a crowdsourced collection consisting of 76,000 audio samples from 831 participants, covering 13 different acoustic scenes. Each audio file is 10 seconds in length. This dataset is used for the classification of diverse acoustic environments, providing a wide range of real-world audio scenes.

AudioSet[10] (Gemmeke et al., 2017) AudioSet is a large-scale multilabel audio event dataset comprising over 2 million 10-second video clips annotated by humans. It is labeled using a hierarchical ontology of 632 event classes, allowing the same sound to be tagged with multiple labels.

FSD50K[11] (Fonseca et al., 2021) Freesound Dataset 50k (FSD50K) is an open dataset of human-annotated sound events, featuring 51,197 audio clips from Freesound. The clips are unevenly distributed across 200 classes derived from the AudioSet ontology.

Beijing Opera:[12] (Tian et al., 2014) The Beijing Opera Percussion Instrument dataset is a collection of audio examples featuring individual strokes from the four main percussion instruments used in Beijing Opera. These instrument classes include Bangu, Naobo, Daluo, and Xiaoluo. The dataset is designed for the study and classification of traditional Chinese percussion instruments within the context of Beijing Opera.

GTZAN:[13] (Tzanetakis et al., 2001) The GTZAN Genre dataset is widely used for music genre classification tasks. It contains 1,000 audio tracks, each with a duration of 30 seconds, categorized into 10 distinct genres: Blues, Classical, Country, Disco, Hip-hop, Jazz, Metal, Pop, Reggae, and Rock. Each genre is represented by 100 tracks.

MUSDB:[14] (Rafii et al., 2017a) The MUSDB18 dataset consists of 150 full-length music tracks, totalling approximately 10 hours of audio, covering various genres. It is primarily used for mu-

[4] https://zenodo.org/records/2589280
[5] https://github.com/YuanGongND/vocalsound
[6] https://dcase.community/challenge2017/
[7] https://hearbenchmark.com/hear-tasks.html
[8] https://zenodo.org/records/3519845
[9] https://github.com/cochlearai/cochlscene
[10] https://research.google.com/audioset/
[11] https://zenodo.org/records/4060432
[12] https://compmusic.upf.edu/bo-perc-dataset
[13] https://paperswithcode.com/dataset/gtzan
[14] https://sigsep.github.io/datasets/musdb.html

sic source separation tasks. The dataset provides four stem labels: drums, bass, vocals, and others (which include all other non-specific instruments and sounds). Each track is broken down into these stems, allowing for detailed analysis and separation of musical components.

Medley:[15] (Bittner et al., 2014) MedleyDB, is a dataset of annotated, royalty-free multitrack recordings. It was curated primarily to support research on melody extraction. For each song melody, f_0 annotations are provided, as well as instrument activations for evaluating automatic instrument recognition. The original dataset consists of 122 multitrack songs, out of which 108 include melody annotations.

Mridangam Stroke:[16] (Anantapadmanabhan et al., 2014) The Mridangam Stroke dataset is a collection of 7,162 audio examples featuring individual strokes of the Mridangam, a pitched percussion instrument used in Carnatic music, a sub-genre of Indian classical music. The dataset captures strokes in various tonics and includes 10 different stroke types played on Mridangams.

Mridangam Tonic: (Anantapadmanabhan et al., 2014) This dataset is a subset of the Mridangam Stroke dataset. It includes 6 different tonic values associated with the 10 different stroke types played on the Mridangam.

NSynth Instrument:[17] (Engel et al., 2017a) NSynth Instrument is a part of the NSynth audio dataset, which consists of 305,979 musical notes, each characterized by unique combinations of pitch, timbre, and envelope. The NSynth Instrument dataset focuses on the task of identifying the high-level instrument family to which each note belongs. These instrument families include categories like keyboard, string, brass, reed, and mallet.

NSynth Source: (Engel et al., 2017a) NSynth Source is a subset of the NSynth dataset, where the task is to identify the method of sound production for each instrument's note. There are three categories of sound production: acoustic, electronic, and synthetic. Acoustic and electronic labels correspond to instruments recorded from physical sources, while the synthetic label applies to notes generated using digital synthesis.

[15] https://medleydb.weebly.com/
[16] https://compmusic.upf.edu/mridangam-stroke-dataset
[17] https://magenta.tensorflow.org/datasets/nsynth

D Audio Augmentations

We use 8 types of audio augmentations in our experiments. We employ torchaudio-augmentations' [18] implementation to augment the audios.

Gaussian Noise. Gaussian noise augmentation is a technique where random noise, following a Gaussian (normal) distribution, is added to audio signals. This simulates real-world background noise and helps improve the robustness of audio models by exposing them to a variety of noisy conditions during training. In our experiment, we set the Minimum Signal-to-Noise Ratio to 0.0001 and the Maximum Signal-to-Noise Ratio to 0.01.

Pitch Shift. Pitch shift augmentation is an augmentation technique which involves changing the pitch of an audio signal without affecting its tempo. In our experiment, we set the minimum pitch shift to -7.0 semitones (downward shift) and the maximum pitch shift to 7.0 semitones (upward shift).

Polarity Inversion. Polarity inversion is an audio augmentation technique where the polarity of the audio waveform is inverted by multiplying the signal by -1. This flips the waveform vertically, turning positive amplitude values into negative ones and vice versa. Although it doesn't affect the audible characteristics of the sound to human listeners, it can be helpful in testing how models perform phase changes in audio signals.

Gain. Gain augmentation involves adjusting the amplitude of an audio signal by applying a gain factor, effectively changing the volume of the audio. It helps in checking how the models perceive the same audio with different volume levels. For our experiment, we set the minimum gain to -20 dB and the maximum gain to -1 dB.

High Pass Filter. High-pass filter augmentation involves applying a filter that allows frequencies above a certain cutoff frequency to pass through while attenuating frequencies below that cutoff. In our experiment, we set the minimum cutoff frequency to 200 Hz and the maximum cutoff frequency to 1200 Hz.

Low Pass Filter. Low-pass filter augmentation involves applying a filter that allows frequencies below a certain cutoff frequency to pass through while attenuating frequencies above that cutoff. In our experiment, we set the minimum cutoff frequency to 2200 Hz and the maximum cutoff frequency to 4000 Hz.

[18] https://github.com/Spijkervet/torchaudio-augmentations/

Delay. Delay augmentation involves adding a delayed version of the audio signal back onto itself, creating an echo effect. In our experiment, we set the volume factor to 0.5, minimum delay to 200 ms, maximum delay to 500 ms and delay interval to 50 ms.

Reverb. Reverb augmentation involves adding reverberation effects to audio signals, simulating the natural reflections of sound in an acoustic environment like a room or hall. In our experiment, we set the minimum reverberance to 0, maximum reverberance to 100, damping factor to 75 and room size to 100.

E Hyper-parameter tuning

Table 6 shows the hyper-parameter tuning for audio-guided and text-guided logit weights, β_1 and β_2 in PAT. Table 7 shows the list of best-performing hyper-parameters for PAT when applied to MSCLAP-23 across all the evaluation datasets

(β_1, β_2)	(0.01, 0.1)	(0.05, 0.5)	(0.1, 0.02)	(0.01, 0.01)	(0.5, 0.5)
Nsynth Inst.	64.34	62.12	65.10	**66.34**	61.23
ESC-50	**94.80**	92.13	93.21	91.89	93.08

Table 6: The hyper-parameter tuning for PAT applied on MSCLAP-23.

Dataset	β_1 (Audio Guided)	β_2 (Text Guided)
ESC-50	0.01	0.1
USD8K	0.02	1.2
TUT	0.18	2.2
VS	0.04	0.5
DCASE	0.11	3.1
Gunshot Tri	0.01	0.13
SESA	0.01	0.11
AudioSet	0.01	0.01
FSD50K	0.02	0.01
Cochlscene	0.3	2.5
Beijing Op.	0.13	1.8
GTZAN	0.4	3.2
MUSDB	0.02	2.5
Medley	0.04	1.3
Mri. St.	0.23	1.4
Mri. To.	0.1	0.1
NSynth Inst	0.01	0.01
NSynth Src	0.42	2.13

Table 7: Best hyper-parameter for PAT when used with MSCLAP-23 across 18 datasets

Dataset	Delay		Low Pass		Reverb	
	MS CLAP	PAT	MS CLAP	PAT	MS CLAP	PAT
Sound						
ESC-50	90.10	**93.45**$_{+3.35\%}$	86.30	**89.80**$_{+3.50\%}$	90.00	**93.25**$_{+3.25\%}$
USD8K	**76.52**	75.95$_{-0.57\%}$	72.79	**78.45**$_{+5.66\%}$	76.54	**81.88**$_{+5.34\%}$
TUT	**42.72**	42.22$_{-0.50\%}$	39.07	**43.64**$_{+4.57\%}$	42.90	**43.64**$_{+0.74\%}$
VS	78.39	**78.58**$_{+0.19\%}$	**77.47**	76.10$_{-1.37\%}$	**76.47**	74.82$_{-1.65\%}$
DCASE	38.11	**43.44**$_{+5.33\%}$	41.19	**41.80**$_{+0.61\%}$	38.32	**44.67**$_{+6.35\%}$
Gunshot Tri.	**25.00**	**25.00**$_{0.00\%}$	**25.00**	**25.00**$_{0.00\%}$	**30.68**	27.27$_{-3.41\%}$
SESA	68.57	**69.52**$_{+0.95\%}$	68.57	**69.52**$_{+0.95\%}$	68.57	70.48$_{+1.91\%}$
AudioSet	28.76	**29.19**$_{+0.43\%}$	27.05	**28.35**$_{+1.30\%}$	28.92	**29.43**$_{+0.51\%}$
FSD50K	41.84	**43.76**$_{+1.92\%}$	41.10	**43.71**$_{+2.61\%}$	43.21	**43.88**$_{+0.67\%}$
Cochlscene	83.28	**84.43**$_{+1.15\%}$	59.82	**60.32**$_{+0.50\%}$	80.72	**81.44**$_{+0.72\%}$
Music						
Beijing Op.	68.54	**70.21**$_{+1.67\%}$	64.83	**65.44**$_{+0.61\%}$	71.61	**72.45**$_{+0.84\%}$
GTZAN	54.41	**57.49**$_{+3.08\%}$	**56.40**	56.36$_{-0.04\%}$	55.34	**57.33**$_{+1.99\%}$
MUSDB	**71.20**	60.00$_{-11.20\%}$	**62.40**	**62.60**$_{+0.20\%}$	**69.20**	58.80$_{-10.40\%}$
Medley	**96.09**	95.98$_{-0.11\%}$	**93.40**	92.57$_{-0.83\%}$	**96.27**	95.86$_{-0.41\%}$
Mri. St.	45.59	**46.36**$_{+0.77\%}$	37.54	**44.16**$_{+6.62\%}$	**33.51**	28.50$_{-5.01\%}$
Mri. To.	22.36	**31.72**$_{+9.36\%}$	19.56	**22.83**$_{+3.27\%}$	13.49	**19.32**$_{+5.83\%}$
NSynth Inst	59.94	**61.66**$_{+1.72\%}$	53.91	**57.00**$_{+3.09\%}$	56.76	**59.27**$_{+2.51\%}$
NSynth Src	48.12	**58.10**$_{+9.98\%}$	39.06	**49.75**$_{+10.69\%}$	47.24	**55.73**$_{+8.49\%}$

Table 8: Zero-shot performance measure of MSCLAP-23 using PAT across 16 audio classification tasks under noisy settings. Each audio sample is subjected to 3 different varieties of audio augmentations—delay, low pass filtering, and reverb. PAT outperforms vanilla zero-shot (ZS) classification scores by showing an absolute improvement of 0.10–10.69%.

F Additional Details

Model Parameters: Among the ALEs that we use, LAION-CLAP and LAION-CLAP Music have ≈ 158M parameters. MSCLAP-22 has ≈ 196M parameters and MSCLAP-23 has ≈ 159M parameters. Wav2CLIP has ≈ 140M parameters. CompA has ≈ 300M parameters.

Compute Infrastructure: All our experiments are conducted on one NVIDIA A6000 GPUs. No training is required, and depending on the downstream task, a single inference run on a benchmark requires anywhere between 1 and 5 minutes.

Implementation Software and Packages: For our baselines, we use the original GitHub repository provided by the authors: LAION-CLAP [19], CompA-CLAP [20], MS-CLAP [21], Wav2CLIP [22].

Potential Risks: To create the prompt datastore, we use GPT4o, which might encode the biases inherent to an LLM. To avoid this, we manually filter the prompt templates generated by LLM.

G Filtering Criteria For Prompt Selection

Table 9 shows scores for the top 11 prompts selected during zero-shot evaluation of PAT for each dataset. We refine our prompt datastore by filtering out incorrect and redundant prompts. First, we

[19] https://github.com/LAION-AI/CLAP/tree/main
[20] https://github.com/Sreyan88/CompA
[21] https://github.com/microsoft/CLAP/tree/main
[22] https://github.com/descriptinc/lyrebird-wav2clip

remove prompts that create inconsistent source-attribute associations, such as “The sound of a beach coming from a church” or “A feeble sound of thunder,” as they are semantically illogical. This manual filtering ensures meaningful audio-text alignment for zero-shot evaluation. Next, we eliminate repeated prompts, often caused by GPT hallucinations.

Dataset	Prompt	Score
	A sound of a <sound>coming from a playground.	0.00372
	A sound of a <sound>coming from a parade.	0.00365
	A sound of a <sound>coming from a swimming pool.	0.00364
	A sound of a <sound>coming from a park.	0.00363
	A restrained sound of a <sound>.	0.00362
Beijing Op.	A soft sound of a <sound>.	0.00362
	The sound of <sound>can be heard near a playground.	0.00361
	A sound of a <sound>coming from a zoo exhibit.	0.0036
	A sound of a <sound>coming from a gym.	0.0036
	A subtle sound of a <sound>.	0.0036
	A gentle sound of a <sound>.	0.00359
	An even sound of a <sound>.	0.00357
	The sound of <sound>can be heard near a rooftop garden.	0.00357
	A low-key sound of a <sound>.	0.00355
	A sporadic sound of a <sound>.	0.00355
	An irregular sound of a <sound>.	0.00354
Cochlsene	A subdued sound of a <sound>.	0.00354
	A delicate sound of a <sound>.	0.00354
	The sound of <sound>can be heard near a lighthouse.	0.00353
	A moderate sound of a <sound>.	0.00353
	A quiet sound of a <sound>.	0.00353
	A gentle sound of a <sound>.	0.00353
	A reverberating sound of a <sound>.	0.00352
	An extensive sound of a <sound>.	0.00352
	A tremendous sound of a <sound>.	0.00352
	An overwhelming sound of a <sound>.	0.00351
	A sound of a <sound>coming from a church.	0.0035
DCASE	The sound of <sound>can be heard near a hedge maze.	0.00348
	The sound of <sound>can be heard near a cliff edge.	0.00347
	An enormous sound of a <sound>.	0.00347
	A massive sound of a <sound>.	0.00347
	The sound of <sound>can be heard near a fishing pier.	0.00346
	A vibrant sound of a <sound>.	0.00346
	An extensive sound of a <sound>.	0.00363
	An unwavering sound of a <sound>.	0.0036
	A sporadic sound of a <sound>.	0.0036
	A persistent sound of a <sound>.	0.0036
	An all-encompassing sound of a <sound>.	0.00359
ESC50	A sound of a <sound>coming from a garden.	0.00359
	A reverberating sound of a <sound>.	0.00358
	A sound of a <sound>coming from a barber shop.	0.00358
	An even sound of a <sound>.	0.00358
	A continuous sound of a <sound>.	0.00356
	An extreme sound of a <sound>.	0.00356
	A sound of a <sound>coming from a church.	0.00379
	The sound of <sound>can be heard near a church.	0.00374
GTZAN	A sound of a <sound>coming from a park.	0.00374
	The sound of <sound>can be heard near a concert hall.	0.00373

	A sound of a <sound>coming from a theater.	0.0037
	A sound of a <sound>coming from a zoo.	0.0037
	The sound of <sound>can be heard near a fruit orchard.	0.00369
	The sound of <sound>can be heard near a theater.	0.00369
	A sound of a <sound>coming from a concert hall.	0.00368
	An extensive sound of a <sound>.	0.00367
	The sound of <sound>can be heard near a garden.	0.00367
	The sound of <sound>can be heard near a forest trail.	0.00418
	The sound of <sound>can be heard near a suburban neighborhood.	0.00417
	The sound of <sound>can be heard near a wildlife reserve.	0.00416
	A sound of a <sound>coming from a sports field.	0.00409
	A sound of a <sound>coming from a park.	0.00407
Gunshot Tri.	The sound of <sound>can be heard near a golf course.	0.00407
	The sound of <sound>can be heard near a lake.	0.00406
	A sound of a <sound>coming from a parking lot.	0.00405
	A sound of a <sound>coming from a forest.	0.00401
	The sound of <sound>can be heard near a greenfield.	0.00398
	The sound of <sound>can be heard near a bridge.	0.00396
	A gentle sound of a <sound>.	0.00393
	A minor sound of a <sound>.	0.00389
	A mild sound of a <sound>.	0.00387
	A soft sound of a <sound>.	0.00385
	A restrained sound of a <sound>.	0.00382
Medley	A feeble sound of a <sound>.	0.0038
	A subdued sound of a <sound>.	0.00379
	An even sound of a <sound>.	0.00378
	A delicate sound of a <sound>.	0.00378
	A major sound of a <sound>.	0.00376
	A faint sound of a <sound>.	0.00375
	A minimal sound of a <sound>.	0.00413
	A firm sound of a <sound>.	0.00408
	A resounding sound of a <sound>.	0.00407
	A muted sound of a <sound>.	0.00404
	A robust sound of a <sound>.	0.00401
Mridangam St.	An even sound of a <sound>.	0.00399
	A soft sound of a <sound>.	0.00395
	A moderate sound of a <sound>.	0.00393
	A feeble sound of a <sound>.	0.00392
	A major sound of a <sound>.	0.00389
	A gentle sound of a <sound>.	0.00384
	A minimal sound of a <sound>.	0.004
	A sound of a <sound>coming from a music box.	0.00387
	A sound of a <sound>coming from a car wash.	0.00384
	A sound of a <sound>coming from a dock.	0.00383
	A sound of a <sound>coming from a river.	0.0038
Mridangam Tonic	A sound of a <sound>coming from a microwave.	0.00378
	The sound of <sound>can be heard near a vineyard.	0.00376
	A sound of a <sound>coming from a clock.	0.00375
	A sound of a <sound>coming from a washing machine.	0.00375
	A sound of a <sound>coming from a car engine.	0.00371

	A sound of a <sound>coming from a toy.	0.00369
MUSDB	A vibrant sound of a <sound>.	0.00397
	An even sound of a <sound>.	0.0039
	A stentorian sound of a <sound>.	0.00389
	A minor sound of a <sound>.	0.00383
	A serene sound of a <sound>.	0.00379
	A gentle sound of a <sound>.	0.00379
	A sound of a <sound>coming from a theater.	0.00377
	A restrained sound of a <sound>.	0.00374
	A quiet sound of a <sound>.	0.00374
	An all-encompassing sound of a <sound>.	0.00374
	A major sound of a <sound>.	0.00373
Nsynth Inst	A major sound of a <sound>.	0.00389
	A minimal sound of a <sound>.	0.00387
	A resonant sound of a <sound>.	0.00385
	A mild sound of a <sound>.	0.00384
	A gentle sound of a <sound>.	0.0038
	A minor sound of a <sound>.	0.00376
	A moderate sound of a <sound>.	0.00375
	A sharp sound of a <sound>.	0.00373
	A slight sound of a <sound>.	0.00372
	A soft sound of a <sound>.	0.00372
	A resounding sound of a <sound>.	0.00371
Nsynth Source	A resonant sound of a <sound>.	0.00387
	A robust sound of a <sound>.	0.00387
	A minor sound of a <sound>.	0.00382
	A moderate sound of a <sound>.	0.00382
	A sound of a <sound>coming from a piano.	0.00382
	A resounding sound of a <sound>.	0.0038
	A firm sound of a <sound>.	0.0038
	A mild sound of a <sound>.	0.00374
	A slight sound of a <sound>.	0.00373
	A sound of a <sound>coming from a guitar.	0.00372
	An even sound of a <sound>.	0.00371
SESA	The sound of <sound>can be heard near a garden.	0.00364
	A sound of a <sound>coming from a parking lot.	0.00361
	The sound of <sound>can be heard near a fishing pier.	0.00361
	The sound of <sound>can be heard near a hedge maze.	0.0036
	The sound of <sound>can be heard near a rooftop garden.	0.00358
	A sound of a <sound>coming from a garden.	0.00358
	The sound of <sound>can be heard near a golf course.	0.00358
	The sound of <sound>can be heard near a playground.	0.00357
	The sound of <sound>can be heard near a university.	0.00357
	A sound of a <sound>coming from a park.	0.00357
	An echoing sound of a <sound>.	0.00357
TUT	A quiet sound of a <sound>.	0.00372
	A subdued sound of a <sound>.	0.00367
	A low-key sound of a <sound>.	0.00366
	A hushed sound of a <sound>.	0.00366

	A faint sound of a <sound>.	0.00365
	The sound of <sound>can be heard near a beach.	0.00363
	An even sound of a <sound>.	0.0036
	A calm sound of a <sound>.	0.0036
	A sound of a <sound>coming from a hallway.	0.0036
	A soft sound of a <sound>.	0.00359
	A sound of a <sound>coming from a hospital room.	0.00359
USD8K	A sound of a <sound>coming from a park.	0.0036
	A subtle sound of a <sound>.	0.0036
	A soft sound of a <sound>.	0.0036
	A mild sound of a <sound>.	0.0036
	A slight sound of a <sound>.	0.00359
	A feeble sound of a <sound>.	0.00358
	A faint sound of a <sound>.	0.00357
	A muted sound of a <sound>.	0.00357
	The sound of <sound>can be heard near a university.	0.00357
	A minimal sound of a <sound>.	0.00357
	A persistent sound of a <sound>.	0.00357
Vocal Sound	A sudden sound of a <sound>.	0.00377
	A sound of a <sound>coming from a barber shop.	0.00377
	A quiet sound of a <sound>.	0.00377
	An even sound of a <sound>.	0.00376
	An abrupt sound of a <sound>.	0.00375
	A gentle sound of a <sound>.	0.00372
	A low-key sound of a <sound>.	0.00372
	A faint sound of a <sound>.	0.00371
	A sporadic sound of a <sound>.	0.00371
	A major sound of a <sound>.	0.0037
	A subtle sound of a <sound>.	0.0037

Table 9: Score of Top 11 prompts across various audio classification datasets