ANGULAR DISTANCE DISTRIBUTION LOSS FOR AUDIO CLASSIFICATION

Antonio Almudévar[1]*, *Romain Serizel*[2], *Alfonso Ortega*[1]

[1] ViVoLab, Aragón Institute for Engineering Research (I3A), University of Zaragoza, Spain
[2] Université de Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France
almudevar@unizar.es

ABSTRACT

Classification is a pivotal task in deep learning not only because of its intrinsic importance, but also for providing embeddings with desirable properties in other tasks. To optimize these properties, a wide variety of loss functions have been proposed that attempt to minimize the intra-class distance and maximize the inter-class distance in the embeddings space. In this paper we argue that, in addition to these two, eliminating hierarchies within and among classes are two other desirable properties for classification embeddings. Furthermore, we propose the Angular Distance Distribution (ADD) Loss, which aims to enhance the four previous properties jointly. For this purpose, it imposes conditions on the first and second order statistical moments of the angular distance between embeddings. Finally, we perform experiments showing that our loss function improves all four properties and, consequently, performs better than other loss functions in audio classification tasks.

Index Terms— angular distance, audio classification, loss

1. INTRODUCTION

Classification is one of the main tasks to be solved with machine learning. In this task, there are typically high-dimensional elements and the goal is to decide to which class of a finite set each of these elements belongs. For this purpose, most of the solutions, particularly those based on deep learning, involve obtaining intermediate representations of reduced dimension of the elements to be classified. These representations are called embeddings and they can be considered as a summary of these elements containing the information that is relevant for classification. This problem is very popular not only because of its intrinsic importance, but also because it provides a simple way to obtain embeddings compared to other methods. Embeddings are useful for a multitude of tasks such as anomaly detection [1, 2], biometric recognition [3, 4], etc. The standard loss function to solve the classification task is the cross-entropy. As a secondary result of using this loss function, the embeddings of the different classes usually end up being somewhat separated. However, it is common to impose certain conditions directly on them due to two reasons: (i) this tends to improve the performance in the classification problem by guiding more the optimization [5, 6]; and (ii) it may be desirable for embeddings to have certain properties when used for a specific task other than classification [7, 8]. These conditions on embeddings are usually imposed through the loss function. Typically, a term is added to the cross-entropy or a modification is made to it.

In this paper we propose a loss function that is added to cross-entropy and we call it Angular Distance Distribution Loss because it imposes conditions on the first and second order statistical moments of the angular distances between embeddings in order to organize the embeddings in the space. Specifically, this organization consists of: (i) bringing embeddings of the same class closer, (ii) moving embeddings of different class away, (iii) minimizing the variation of the distances of the embeddings of the same class, and (iv) making the embeddings of a class equal in distance to the embeddings of any class. Traditionally, only the first two have been considered in the literature. However, in section 3 we formalize all four, arguing why they are all important. In addition, we reason how they relate to the statistical moments of the distances between embeddings. Furthermore, we propose an experimental framework with different Audio Classification datasets. In these experiments, on the one hand, we verify that our embeddings satisfy the properties described in the previous paragraph, so we verify that our loss function encourages the properties to be satisfied. On the other hand, we obtain a better accuracy than other loss functions that aim to establish conditions on the embeddings. Thus, we verify that the described properties translate into better classification performance. The details of these experiments are presented in the section 4 and can be replicated using the code in `https://github.com/antonioalmudevar/distance_distribution_loss`

2. RELATED WORK

Audio Classification consists of identifying to which class an audio belongs [9, 10]. In recent years it has received a lot of interest from the community [11, 12]. Solutions to this problem typically involve an embedding extractor followed by a small classifier net which are trained by minimizing cross-entropy. In many SOTA solutions the embedding extractor has a large number of parameters, so it is common to pre-train it with a large dataset and perform finetuning for the desired dataset. Although convolutional architectures has been widespread used [13–15], the most popular systems nowadays are transformer-based. These include Audio Spectrogram Transformer (AST) [16] and BEATs [17].

Loss Functions. It has been observed in different works that separating the embeddings of different classes often results in better performance in the classification task [5, 18–20]. Two loss functions that stand out are Focal Loss [7] and Orthogonal Projection Loss (OPL) [5], with which we compare our proposal.

*This work was supported by MCIN/AEI/10.13039/501100011033 under Grants PDC2021-120846-C41 & PID2021-126061OB-C44, and in part by the Government of Aragón (Grant Group T36 23R). This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 101007666.

3. PROPOSED METHOD

The problem we seek to solve in this paper is that of canonical classification, which has two characteristics: (i) all errors are considered equally critical; and (ii) all elements are considered equally similar to each other within a class. This means that intra-class and inter-class hierarchies are not desirable. In fact, the standard evaluation metric is accuracy, which considers all errors and correct predictions equally relevant. The presence of these hierarchies is desirable in some scenarios, but our goal is not to solve the latter.

3.1. Classification Solution Formulation

Let $\mathcal{D} = \{x^{(i)}, y^{(i)}\}_{i=1}^{N}$ be the dataset, where $x^{(i)}$ is each input and $y^{(i)}$ the label of $x^{(i)}$ and is a vector containing at each position j the probability that $x^{(i)}$ belongs to class j. The objective is to design a system that allows us to obtain a prediction of $y^{(i)}$ which we denote $\tilde{y}^{(i)}$. Typical deep learning classifier solutions consists of (i) an embeddings extractor f_θ, which provides the embedding as $z^{(i)} = f_\theta(x^{(i)}) \in \mathbb{R}^k$; and (ii) a classifier net g_ϕ, which gives the predictions as $\tilde{y}^{(i)} = g_\phi(z^{(i)}) \in \mathbb{R}^c$. Cross-entropy between y_i and $\tilde{y}_i$ is used as loss function, which we call $\mathcal{L}_{CE}$.

3.2. Desirable Properties of Embeddings

We explain below some desirable properties of embeddings for classification. In figs. 1 to 4 the dots correspond to low dimensional representations of the embeddings and different colors are used to indicate different classes.

- **Intra-class clustering**: The embeddings of the same class are close to each other in space. This has been shown to improve performance in classification and additional tasks.

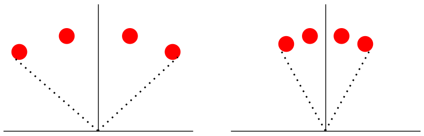


Figure 1: Low (left) and high (right) Intra-class clustering

- **Intra-class equidistance**: All the embeddings from the same class have approximately the same distance from each other. From a conceptual perspective, all the elements in a given class should be equally similar.

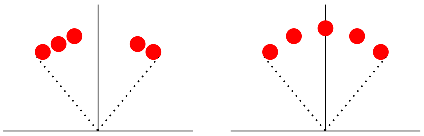


Figure 2: Low (left) and high (right) Intra-class equidistance

- **Inter-class separation**: Embeddings of different classes are far away from each other. This allows to take better advantage of all the space and, as a consequence, improves the performance in different tasks, especially when coupled with intra-class clustering.

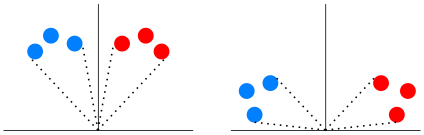


Figure 3: Low (left) and high (right) Inter-class separation

- **Inter-class equidistance**: All embeddings from different classes are approximately equally spaced from each other. This allows removing hierarchies between classes, which is conceptually desirable since all errors have the same penalty in the classical classification problem.

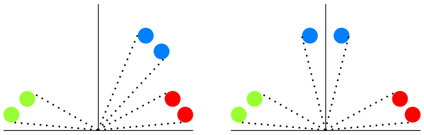


Figure 4: Low (left) and high (right) Inter-class equidistance

Traditionally, only intra-class clustering and inter-class separation have been considered as desirable properties. However, we also consider it convenient to have intra-class and inter-class equidistance, since these allow to avoid intra-class and inter-class hierarchies, respectively, which is desirable in the canonical classification problem, since all errors and correct predictions are equally critical. As a result, as we will see in section 4, maximizing these two improves the accuracy.

3.3. Angular Distance Distribution Loss

Having described the above properties and argued why they are desirable, we now present Angular Distance Distribution Loss, which encourages these properties. It imposes conditions on the first and second order statistical moments of the angular distances between embeddings. For now, we assume that the labels are hard, i.e. $y_k^{(i)} = 1$ for one k and 0 for the rest. With this idea, we can define the sets:

$$D_p = \left\{ d_c\left(z^{(i)}, z^{(j)}\right)^2 \,\middle|\, y^{(i)} = y^{(j)};\, i \neq j \right\} \tag{1}$$

$$D_n = \left\{ \left(1 - d_c\left(z^{(i)}, z^{(j)}\right)\right)^2 \,\middle|\, y^{(i)} \neq y^{(j)} \right\} \tag{2}$$

where $d_c(x, y) = 1 - x^T \cdot y$, which takes values in the interval $[0, 2]$, being 0 when the two vectors are proportional, 1 when they are orthogonal and 2 when they are opposites. Next, we define the following terms from the previous ones:

$$\mu_p = \frac{1}{|D_p|} \sum_{k \in D_p} k \tag{3}$$

$$\sigma_p = \sqrt{\frac{\sum_{k \in D_p} (k - \mu_p)^2}{|D_p| - 1}} \tag{4}$$

and μ_n and σ_n analogously for D_n. Each term can be related to one of the properties in 3.2 as follows:

- Minimizing μ_p implies boosting intra-class clustering, since it implies minimizing the average distance between embeddings of the same class.
- Minimizing σ_p implies promoting the intra-class equidistance, since we are reducing the variation of all the distances between embeddings of the same class.
- Minimizing μ_n implies boosting the inter-class separation, since we are promoting the embeddings of different classes to be orthogonal
- Minimizing σ_n implies favoring the inter-class equidistance, since we are reducing the variation between embedding distances of different classes.

With all this, we define our loss function to minimize ADD as:

$$L_{ADD} = \lambda_\mu^p \mu_p + \lambda_\sigma^p \sigma_p + \lambda_\mu^n \mu_n + \lambda_\sigma^n \sigma_n \tag{5}$$

where $\boldsymbol{\lambda} = \{\lambda_\mu^p, \lambda_\sigma^p, \lambda_\mu^n, \lambda_\mu^n\}$ are hyperparameters. In section 4 we explore how each of these terms separately affects the accuracy and distribution of embeddings in space.

Finally, the loss function of our system is as follows:

$$\mathcal{L} = \mathcal{L}_{CE} + \mathcal{L}_{ADD} \tag{6}$$

3.4. Soft Labels Adaptation

In some scenarios, the labels used to optimize our model are soft, i.e., they represent the probability that an element belongs to each class instead of considering that an element belongs to a single class [21]. One of the main causes of having soft labels is the use of data augmentation techniques such as mixup [22]. As mixup is widely used in audio classification [23], we propose a modification of our loss function to deal with soft labels.

When we have soft labels, it is still important to maximize intra-class clustering and intra-class equidistance, since we are interested that elements belonging to the same class should be close and at a similar distance from each other. However, inter-class separation must be reinterpreted, so that it would be desirable that if $y^{(i)}$ is more similar to $y^{(j)}$ than to $y^{(k)}$, then $z^{(i)}$ should be closer to $z^{(j)}$ than to $z^{(k)}$, and vice versa. For this, we must strive that $d_c\left(z^{(i)}, z^{(j)}\right) = d_c\left(y^{(i)}, y^{(j)}\right)$ for each pair i, j. To modify the loss function, we first define:

$$\mathcal{L}_\mu = \frac{1}{N_B} \sum_{i \in B} \sum_{j \neq i} \left(d_c\left(y^{(i)}, y^{(j)}\right) - d_c\left(z^{(i)}, z^{(j)}\right) \right)^2 \tag{7}$$

where $N_B = |B|(|B| - 1)$ is the number of pairs in a batch. Optimizing $\mathcal{L}_\mu$ we manage to jointly maximize intra-class clustering and inter-class separation. In fact, we note that we do not lose generality with respect to the hard scenario, since $d_c\left(y^{(i)}, y^{(j)}\right)$ holds 0 if $y^{(i)} = y^{(j)}$ and 1 otherwise and, therefore, optimizing $\mathcal{L}\mu$ is equivalent to optimizing $\lambda_\mu^p \mu_p + \lambda_\mu^n \mu_n$ with $\lambda_\mu^n = \frac{|D_n|}{|D_p|} \lambda_\mu^p$. In addition, since elements do not belong to a single class, it does not make sense to maximize the inter-class equidistance. Thus, when we have soft labels, we define the ADD as:

$$L_{ADD}^{soft} = \lambda_\mu \mathcal{L}_\mu + \lambda_\sigma^p \sigma_p \tag{8}$$

4. EXPERIMENTS

4.1. Datasets

Environmental Sound Classification (ESC-50) [24] contains 2,000 5-second ambient sound recordings annotated with 5 classes, so that each audio belongs to a single class. In our experiments we follow the standard 5-fold cross-validation to evaluate our systems.
Speech Commands V2 (KS2) [25] is composed of 105,829 clips of 1-second spoken keywords annotated with 35 word classes. It is officially divided into 84,843, 9,981 and 11,005 clips for training, test and validation, respectively.
IEMOCAP (ER) [26] contains about 12 hours of speech with four different emotions. We use the standard 5-fold cross-validation proposed in [27] for evaluation.

Table 1: Hyperparam. per embeddings extractor and dataset

	AST			BEATs		
	ESC	KS2	ER	ESC	KS2	ER
Window type	Hanning			Povey		
Freq. Mask	24	48	24	0		
Time Mask	96	48	96	0		
Mixup λ	0	0.5	0	0	0.5	0
Epochs	25			30		
Batch Size	32			16		
Optimizer	AdamW			Adam		
Learning rate	7e-4	6e-5	7e-4	8e-6	1e-4	8e-6
Momentum	$\boldsymbol{\beta} = \{0.9, 0.98\}$			$\boldsymbol{\beta} = \{0.95, 0.999\}$		
Weight Decay	1e-2			5e-6		

4.2. Embeddings Extractors Architectures

Audio Spectrogram Transformer (AST) [16] is the first to use Transformer type architectures for audio. The original AST model is pre-trained on Imagenet [28] and Audioset [9]. We fine-tune it for each scenario.
Bidirectional Encoder representation from Audio Transformers (BEATs) [17] is a pre-training framework for learning representations from Audio Transformers, in which an acoustic tokenizer and a self-supervised audio model are optimized. We use the original pre-trained model with Audioset and finetune for each scenario.

4.3. Hyperparameters

For all our systems we use the audio signals at 16kHz. The input to the systems are 128 mel-spectrograms coefficients computed on 25 ms windows every 10 ms. We normalize the mean and standard deviation to 0 and 0.5, respectively. Some hyperparameters vary between scenarios. These details can be found in table 1 and are inspired by the experiments in the original papers, with slight modifications due to computational limitations.

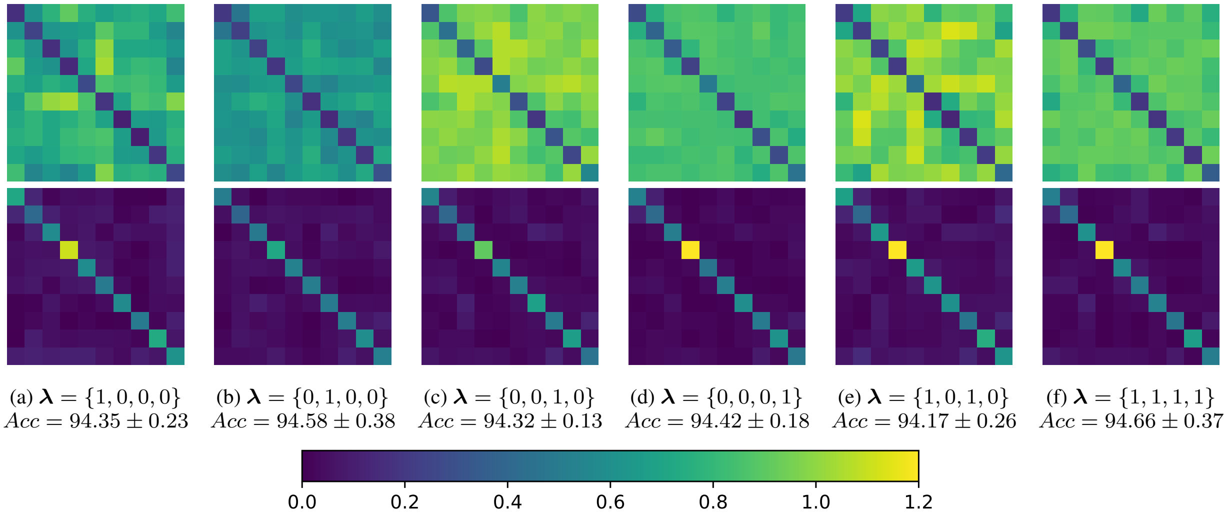
4.4. Ablation Study on each term of the ADD

We are going to analyze the influence of each of the terms of $\mathcal{L}_{ADD}$. First, we want to see if the hypotheses outlined in section 3.3 about the relationship between each ADD term and the properties of 3.2 hold. Second, we want to analyse the impact of each particular term in the ADD and their combination on the accuracy. For this, we train four classifiers, each with one of the elements of $\boldsymbol{\lambda}$ equal to 1 and the rest equal to zero. Third, we want to test whether optimizing intra-class and inter-class equidistance provides an advantage despite already optimizing intra-class clustering and inter-class separation. To do so, we train two classifiers: one with $\boldsymbol{\lambda} = \{1, 0, 1, 0\}$ and another with $\boldsymbol{\lambda} = \{1, 1, 1, 1\}$ and compare them. The dataset to be classified is ESC-50 and the embedding extractor used an AST in all cases. In figure 5 we present the mean and coefficient of variation of the d_c between the embeddings of 10 pairs of classes and the accuracy calculated for all the classes.

- In figure 5a we see that the distance between embeddings of the same class is in general the minimum in mean, i.e. the intra-class clustering is the maximum.
- In figure 5b we observe that the distances between the embeddings of the same class is the least spread, which means that the intra-class equidistance is the highest.

Table 2: Accuracy for the different Datasets, Embeddings Extractors and Loss Functions

	ESC-50		KS2		ER	
	AST	BEATs	AST	BEATs	AST	BEATs
Cross-entropy	93.97 ± 0.21	91.05 ± 0.41	92.05 ± 0.04	88.94 ± 0.13	59.91 ± 0.60	61.66 ± 0.31
Focal Loss [7]	94.40 ± 0.36	91.10 ± 0.49	-	-	60.79 ± 0.16	62.17 ± 0.05
OPL [5]	94.11 ± 0.37	91.50 ± 0.20	-	-	60.53 ± 0.42	$\mathbf{63.06 \pm 0.32}$
ADD ($\boldsymbol{\lambda} = \{1, 1, 1, 1\}$)	$\mathbf{94.68 \pm 0.09}$	$\mathbf{92.22 \pm 0.06}$	$\mathbf{97.54 \pm 0.06}$	$\mathbf{90.49 \pm 0.16}$	$\mathbf{61.30 \pm 0.38}$	62.73 ± 0.17



(a) $\boldsymbol{\lambda} = \{1, 0, 0, 0\}$ $Acc = 94.35 \pm 0.23$

(b) $\boldsymbol{\lambda} = \{0, 1, 0, 0\}$ $Acc = 94.58 \pm 0.38$

(c) $\boldsymbol{\lambda} = \{0, 0, 1, 0\}$ $Acc = 94.32 \pm 0.13$

(d) $\boldsymbol{\lambda} = \{0, 0, 0, 1\}$ $Acc = 94.42 \pm 0.18$

(e) $\boldsymbol{\lambda} = \{1, 0, 1, 0\}$ $Acc = 94.17 \pm 0.26$

(f) $\boldsymbol{\lambda} = \{1, 1, 1, 1\}$ $Acc = 94.66 \pm 0.37$

Figure 5: Mean (top row) and coefficient of variation (bottom row) of the d_c values between embeddings of 10 classes of ESC-50. The accuracy given is calculated for the 50 classes. Coefficient of variation is defined as $\frac{\sigma}{\mu}$ and is used here instead of σ because it normalizes the variation by normalizing by the mean, which changes depending on $\boldsymbol{\lambda}$, so it represents better intra-class equidistance.

- In figure 5c we contemplate that the inter-class distance or separation is the maximum in mean.
- In figure 5d we find that the distances between embeddings of different classes are similar regardless of the class pairs, thus achieving a higher inter-class equidistance.
- In figure 5e we see that if we only optimize intra-class clustering and inter-class separation, the distances between different pairs of embeddings of different classes are very far from each other. In addition, there are classes for which embeddings are closer to each other than for other classes.
- In figure 5f we obtain embeddings that do not satisfy each property as well as when we try to optimize them separately, but with a good balance between all of them.

Finally, the best accuracy obtained is for $\boldsymbol{\lambda} = \{1, 1, 1, 1\}$, that is, when we optimize all four properties together. This leads us to believe that all these properties have an influence in achieving a higher accuracy. In addition, we see that the properties that separately have most positively influence accuracy are inter-class and intra-class equidistance.

4.5. Quantitative Results

We have found in the previous section that our loss function allows us to meet the properties that we consider desirable. Moreover, we have verified that for the analyzed scenario, the accuracy when the four properties are optimized jointly is better than when they are optimized separately. Here we perform a more extensive study in which we compare in terms of accuracy the ADD with other loss functions with good performance. We do not use Focal Loss and OPL for KS2, as these do not support soft labels and we use mixup for this dataset. We have performed all the experiments three times and we provide the mean and standard deviation of the accuracy. In table 2 we can see that the results of our loss function is superior to the rest except in one case. This suggests that, in general, the described properties are desirable to improve accuracy and that the ADD function performs superiorly in different scenarios.

5. CONCLUSIONS AND FUTURE WORK

In this paper we have presented four properties for embeddings of a classifier arguing why we consider these properties to be desirable. In addition, we have designed Angular Distance Distribution Loss, a loss function that is intended to allow us to obtain each of these properties. First, we have verified that, indeed, our loss function allows us to obtain emebddings that satisfy these properties separately and jointly. Subsequently, we have observed, for a given scenario, that the performance in terms of accuracy is better when all four properties are satisfied jointly than separately. Finally, we have found for different datasets and architectures that the fact that our embeddings satisfy these properties translates into better accuracy than other relevant loss functions in the literature. This validates the hypothesis about the importance of these properties to improve accuracy. We believe that the properties described in this work may be desirable also for other applications, such as anomaly detection or biometric recognition. Thus, experiments testing the ADD in these fields can be developed in the future

6. REFERENCES

[1] V. Chandola, A. Banerjee, and V. Kumar, “Anomaly detection: A survey,” *ACM computing surveys (CSUR)*, vol. 41, no. 3, pp. 1–58, 2009.

[2] Y. Kawaguchi, K. Imoto, Y. Koizumi, N. Harada, D. Niizumi, K. Dohi, R. Tanabe, H. Purohit, and T. Endo, “Description and discussion on dcase 2021 challenge task 2: Unsupervised anomalous sound detection for machine condition monitoring under domain shifted conditions,” *arXiv preprint arXiv:2106.04492*, 2021.

[3] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, “Arcface: Additive angular margin loss for deep face recognition,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 4690–4699.

[4] H. Wang, Y. Wang, Z. Zhou, X. Ji, D. Gong, J. Zhou, Z. Li, and W. Liu, “Cosface: Large margin cosine loss for deep face recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 5265–5274.

[5] K. Ranasinghe, M. Naseer, M. Hayat, S. Khan, and F. S. Khan, “Orthogonal projection loss,” in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 12 333–12 343.

[6] A. Almudévar, A. Ortega, L. Vicente, A. Miguel, and E. Lleida, “Variational Classifier for Unsupervised Anomalous Sound Detection under Domain Generalization,” in *Proc. INTERSPEECH 2023*, 2023, pp. 2823–2827.

[7] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, “Focal loss for dense object detection,” in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.

[8] Y. Zeng, H. Liu, L. Xu, Y. Zhou, and L. Gan, “Robust anomaly sound detection framework for machine condition monitoring,” DCASE2022 Challenge, Tech. Rep., July 2022.

[9] J. F. Gemmeke, D. P. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, “Audio set: An ontology and human-labeled dataset for audio events,” in *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2017, pp. 776–780.

[10] A. Mesaros, T. Heittola, A. Diment, B. Elizalde, A. Shah, E. Vincent, B. Raj, and T. Virtanen, “Dcase 2017 challenge setup: Tasks, datasets and baseline system,” in *DCASE 2017-Workshop on Detection and Classification of Acoustic Scenes and Events*, 2017.

[11] N. Turpault, R. Serizel, A. P. Shah, and J. Salamon, “Sound event detection in domestic environments with weakly labeled data and soundscape synthesis,” in *Workshop on Detection and Classification of Acoustic Scenes and Events*, 2019.

[12] F. Ronchini, R. Serizel, N. Turpault, and S. Cornell, “The impact of non-target events in synthetic soundscapes for sound event detection,” *arXiv preprint arXiv:2109.14061*, 2021.

[13] S. Hershey, S. Chaudhuri, D. P. Ellis, J. F. Gemmeke, A. Jansen, R. C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold, *et al.*, “Cnn architectures for large-scale audio classification,” in *2017 ieee international conference on acoustics, speech and signal processing (icassp)*. IEEE, 2017, pp. 131–135.

[14] E. Cakır, G. Parascandolo, T. Heittola, H. Huttunen, and T. Virtanen, “Convolutional recurrent neural networks for polyphonic sound event detection,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 6, pp. 1291–1303, 2017.

[15] R. Serizel, N. Turpault, H. Eghbal-Zadeh, and A. P. Shah, “Large-scale weakly labeled semi-supervised sound event detection in domestic environments,” *arXiv preprint arXiv:1807.10501*, 2018.

[16] Y. Gong, Y.-A. Chung, and J. Glass, “Ast: Audio spectrogram transformer,” *arXiv preprint arXiv:2104.01778*, 2021.

[17] S. Chen, Y. Wu, C. Wang, S. Liu, D. Tompkins, Z. Chen, and F. Wei, “Beats: Audio pre-training with acoustic tokenizers,” *arXiv preprint arXiv:2212.09058*, 2022.

[18] G. Sun, S. Khan, W. Li, H. Cholakkal, F. S. Khan, and L. Van Gool, “Fixing localization errors to improve image classification,” in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXV 16*. Springer, 2020, pp. 271–287.

[19] X. Zhang, R. Zhao, Y. Qiao, and H. Li, “Rbf-softmax: Learning deep representative prototypes with radial basis function softmax,” in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVI 16*. Springer, 2020, pp. 296–311.

[20] I. Sheth and S. Ebrahimi Kahou, “Auxiliary losses for learning generalizable concept-based models,” *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[21] I. Martín-Morató, M. Harju, P. Ahokas, and A. Mesaros, “Strong labeling of sound events using crowdsourced weak labels and annotator competence estimation,” in *Proc. IEEE Int. Conf. Acoustic., Speech and Signal Process. (ICASSP)*, 2023.

[22] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, “mixup: Beyond empirical risk minimization,” *arXiv preprint arXiv:1710.09412*, 2017.

[23] Z. Zhang, S. Xu, S. Cao, and S. Zhang, “Deep convolutional neural network with mixup for environmental sound classification,” in *Chinese conference on pattern recognition and computer vision (prcv)*. Springer, 2018, pp. 356–367.

[24] K. J. Piczak, “Esc: Dataset for environmental sound classification,” in *Proceedings of the 23rd ACM international conference on Multimedia*, 2015, pp. 1015–1018.

[25] P. Warden, “Speech commands: A dataset for limited-vocabulary speech recognition,” *arXiv preprint arXiv:1804.03209*, 2018.

[26] C. Busso, M. Bulut, C.-C. Lee, A. Kazemzadeh, E. Mower, S. Kim, J. N. Chang, S. Lee, and S. S. Narayanan, “Iemocap: Interactive emotional dyadic motion capture database,” *Language resources and evaluation*, vol. 42, pp. 335–359, 2008.

[27] S.-w. Yang, P.-H. Chi, Y.-S. Chuang, C.-I. J. Lai, K. Lakhotia, Y. Y. Lin, A. T. Liu, J. Shi, X. Chang, G.-T. Lin, *et al.*, “Superb: Speech processing universal performance benchmark,” *arXiv preprint arXiv:2105.01051*, 2021.

[28] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “Imagenet: A large-scale hierarchical image database,” in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.