Addressing Missing Labels in Large-Scale Sound Event Recognition Using a Teacher-Student Framework With Loss Masking

Eduardo Fonseca , *Student Member, IEEE*, Shawn Hershey, Manoj Plakal , Daniel P. W. Ellis , *Fellow, IEEE*, Aren Jansen, *Member, IEEE*, and R. Channing Moore

Abstract—The study of label noise in sound event recognition has recently gained attention with the advent of larger and noisier datasets. This work addresses the problem of *missing labels*, one of the big weaknesses of large audio datasets, and one of the most conspicuous issues for AudioSet. We propose a simple and model-agnostic method based on a teacher-student framework with loss masking to first identify the most critical missing label candidates, and then ignore their contribution during the learning process. We find that a simple optimisation of the training label set improves recognition performance without additional computation. We discover that most of the improvement comes from ignoring a critical tiny portion of the missing labels. We also show that the damage done by missing labels is larger as the training set gets smaller, yet it can still be observed even when training with massive amounts of audio. We believe these insights can generalize to other large-scale datasets.

Index Terms—Sound event recognition, label noise, missing labels, teacher-student, loss masking.

I. INTRODUCTION

AS SOUND Event Recognition (SER) has gained attention in recent years [1], research in this field has moved from small datasets encompassing few hours of audio [2]–[5], to larger datasets with much greater coverage and duration [6], [7]. A milestone was the release of AudioSet—a dataset of 527 everyday sound classes organized with a hierarchical ontology, that includes around 2 million segments of $\approx$10 s in its released version [6].

However, large-scale audio datasets inevitably bring in label noise issues, since it is intractable to exhaustively annotate such massive amounts of audio. The resulting issues of less-precise labels can cause various problems including performance decreases and longer training times [8], and can become a critical

impediment to the success of SER. Consequently, label noise in SER has lately become a focus of interest. Previous work analyses the impact of label noise in these tasks [9], [10], as well as proposes ways to mitigate its negative effect [11]–[14]. A DCASE 2019 Challenge Task was launched to foster research in this topic [7].

AudioSet presents a number of label noise problems. Some are due to shortcomings in the annotation process, e.g., missing or incorrect labels. Others are related to the hierarchical structure of the AudioSet Ontology, e.g., a segment may be annotated with a leaf class label but not with its parent one, or annotated with a label that is not the most specific within its hierarchical path. Still other problems arise from the temporally-weak labels (i.e., clip-level labels), where the class label may be active only during a small (and unknown) portion of the audio segment. Finally, some semantic inconsistencies may exist as the ontology allows for several sound attributes to be associated to one type of sound event (while not all of them may have been annotated). Despite these label noise problems, they have been directly addressed in only a few of the previous works using AudioSet (e.g., [13], [14]), while the majority of efforts focus on deriving more sophisticated network architectures that ignore or downplay the idiosyncrasies of the labeled audio data (e.g., [15], [16]).

In this work, we focus on one of the most frequent label noise problems in AudioSet: its missing labels. The study of missing labels in SER has received very little attention. To our knowledge, this specific topic has been covered only by [9], where robustness to missing labels is studied by simulating them in a synthetic dataset of 20 classes. Our contribution is two-fold. First, we propose a simple and model architecture-agnostic method based on a teacher-student framework to first identify the most critical potentially missing labels in AudioSet, and then ignore their contribution in the learning process through a loss masking approach. We then analyse the effect of the proposed method via a set of experiments using two model architectures of different capacity and two train sets of different size. We find that a simple optimisation of the training label set can lead to a non-negligible improvement in recognition performance without additional compute. We also discover that most of the improvement comes from ignoring a tiny portion of the missing labels. The ultimate goal is to demonstrate how prior knowledge of a dataset can be leveraged to build simple, efficient, and model-agnostic solutions to improve recognition performance, which can complement other approaches focused on improving network architectures.
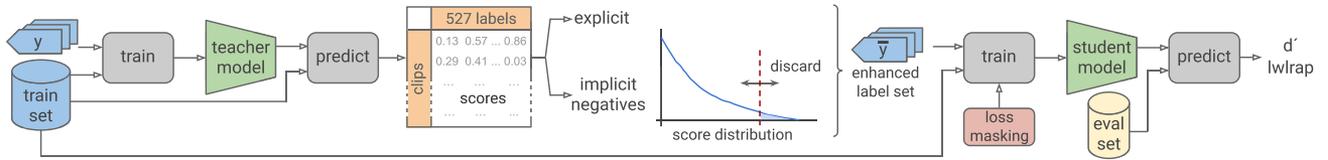
Fig. 1. Proposed method. First: Identification of potential missing labels per class using teacher's predictions and creation of enhanced label set. Second: Training a student model while ignoring missing labels through loss masking.

II. MISSING LABELS IN AUDIOSET

We refer to missing labels as those labels that would be included in an ideal, exhaustive annotation but which are missing from the current set. The existence of missing labels in AudioSet is due to the dataset curation process. This process consisted of two steps: the compilation of a list of candidate labels per clip, and the human validation of the labels nominated in that list. The list of candidate labels was compiled by means of a series of automatic methods, including the processing of the available metadata (e.g., video title and/or description) as well as a query-by-example method. These methods can be sub-optimal due to the high inter- and intra-class variation of sound events in the AudioSet Ontology [6]. In addition, the list of candidate labels was limited to a maximum of ten labels per clip. There are therefore several ways by which some existing sound events fail to be nominated by the system, or are nominated but ranked below the top ten, thus leading to missing labels. We call the nominated labels that have received human validation *explicit* labels (that can in turn be positive or negative, depending on the human rating being "Present" or "Not Present"). The remaining labels which are not proposed by the nomination system (the vast majority) are referred to as *implicit* negative labels, and have received no human validation. Hence, it is likely that some of the implicit negative labels are indeed missing (positive) labels.

AudioSet poses a multi-label audio tagging problem, which is usually addressed by a deep network with an output layer composed by C binary classifiers, with C being the number of classes. Binary classification loss functions are typically adopted, composed by two terms, one accounting for the positive examples, and the other for the negative ones. The default option is binary cross-entropy, expressed by: (1)

$$\mathcal{L} = -\sum_{c=1}^{C} y_c \log(p_c) + (1 - y_c) \log(1 - p_c), \qquad (1)$$

where p_c represents the network output prediction and y_c the ground truth label for class c. The implicit negative labels are considered negative examples, hence they are covered by the second term of (1). If a sound event is actually present, we want the model to emit a high score even if the "Present" label is missing. However, this behaviour will be penalized (with the penalty increasing for higher output predictions) due to the backpropagation of an artificially high loss contribution, which causes a misleading gradient update. We hypothesize this hinders the learning process to some extent.

III. METHOD

We propose a two-step strategy based on a teacher-student framework [17] depicted in Fig. 1. First, a teacher model is trained using the original AudioSet labels, y, and the teachers

predictions are used to build a new enhanced label set, $\bar{y}$, where the suspected missing labels are flagged. Second, $\bar{y}$ is used to train a student model where the flagged labels are ignored through a loss masking approach. Next we explain the proposed method in detail. The first step consists of identifying the potential missing labels per class. To do so, a *teacher* model is trained using the original AudioSet labels, y. We use the trained teacher model to predict scores for the train set, leading to a set of $\mathbb{R}^{C \times 1}$ scores per audio clip. The teacher's predicted scores are used to take decisions on labels' veracity. We focus on the predictions associated with the implicit negative labels. Our hypothesis is that the top-scored implicit negative labels (henceforth, *top-scored negatives*) are likely to correspond mostly to missing "Present" labels, i.e., false negatives. Under this hypothesis, we rank implicit negative labels based on the teacher's predictions and we create a new label set, $\bar{y}$, by flagging a given percentage of the top-scored negatives per class, with the intention of ignoring them in the students learning. Note that, unlike other teacher-student pipelines where teachers predictions are used as ground truth to train a student (e.g., via soft labels [17], [18]), our case features a *skeptical* teacher whose supervision is used to highlight flaws in the current ground truth, estimating potentially missing labels and flagging them in a new label set. The outcome of this step is an enhanced training label set, $\bar{y}$, where the label information is encoded as multi-hot target vectors with three states (positive, negative, and to-be-ignored labels).

The second step consists of training a *student* model using the label set optimised through the teachers predictions. The goal here is to ignore the loss contributions of the previously flagged labels in the loss function computation. This is done through a loss masking approach, where we modify the student's learning pipeline so as to create a binary mask of size $C \times 1$ per input example, using the information encoded in $\bar{y}$. Each element of the binary mask, M_c, is defined as (2)

$$M_c = \begin{cases} 0, & \text{if label is implicit negative and score } > t_c \\ 1, & \text{otherwise,} \end{cases} \qquad (2)$$

where t_c is a per-class threshold computed as a given percentile of the per-class scores distribution. In practice, we compute the loss function $\mathcal{L}$ following (1), and M_c is applied to the negative term of $\mathcal{L}$ in order to discard the loss contributions of potentially missing labels. A similar masking approach to ignore false negatives in SER was recently used in [19].

IV. EXPERIMENTAL SETUP

We evaluate the proposed method using an internal version of AudioSet [6], including information about which labels are explicit/implicit. To study the impact of missing labels as a function of training data size, we use two train sets with similar class distributions but one roughly five times larger than the

TABLE I
TRAIN SETS AND ARCHITECTURES USED IN OUR EXPERIMENTS

Train set	clips	hours
tr_small	506,721	1407
tr_large	2,467,357	6853

Architecture	parameters	Mult-Adds
ResNet-50	30M	1860M
MobileNetV1	3.7M	69.2M

other (see Table I). For evaluation, we use an internal *eval set* of 47,132 audio clips. Incoming clips are transformed to log-mel spectrograms using a 25 ms Hann window with 10 ms hop, and 64 mel log-energy bands. The network is presented with time-frequency patches of 96 frames (corresponding to 0.96 s) with 50% overlap. To assess the impact of missing labels on models of different capacity, we employ two CNNs as students: ResNet-50 [20] and MobileNetV1 [21]. Both are taken from the computer vision literature and have proven successful in audio recognition (see [22] and YAMNet[1]). Table I shows model size and Mult-Adds for both architectures. We use Adam optimizer [23] with learning rate 1e-5, and random weight initialization with a standard deviation of 0.001.

A. Evaluation

We pass each 0.96 s evaluation patch through the model to compute output scores, which are then averaged per-class across all patches in a clip to obtain clip-level predictions, as in [6]. As evaluation metrics we use d' and *lωlrap*. d' (d-prime) is a within-class metric, i.e., it ranks all test samples according to the classifier score for a given class. d' can be computed as a monotonic transform of ROC AUC, and describes the separation between unit-variance normal distributions that would achieve the same AUC [22], [24]. In order to avoid the impact of potential missing positive labels in the eval set, only samples with explicit labels for a given class (both positive and negative) are used in the calculation of d'. (Because this excludes many "easy" samples, the resulting d' values are substantially lower than including all non-positive samples as negatives.) Label-weighted label-ranking average precision (*lωlrap*) is a between-class metric, i.e., it evaluates the overall ranking across all classifier outputs for every test sample. Specifically, *lωlrap* measures, for every ground truth test label c, what fraction of the predicted top-ranked labels down to c are among the ground truth [7]. Both metrics are computed on a per-class basis, then averaged with equal weight across all classes to yield the overall performance shown in Table II and Fig. 2. Non-pathological $d' \in [0, \infty)$ while *lωlrap* $\in [0, 1]$.

V. EXPERIMENTS

As explained in Section III, we first train a teacher model with the unmodified labels and use it to predict scores in the train set. We used an internal ResNet-50 model for the teacher which had been trained using several tweaks to improve performance, similar to those used in the publicly-released YAMNet model.[1] Based on the teachers predicted scores, we generate 18 new label
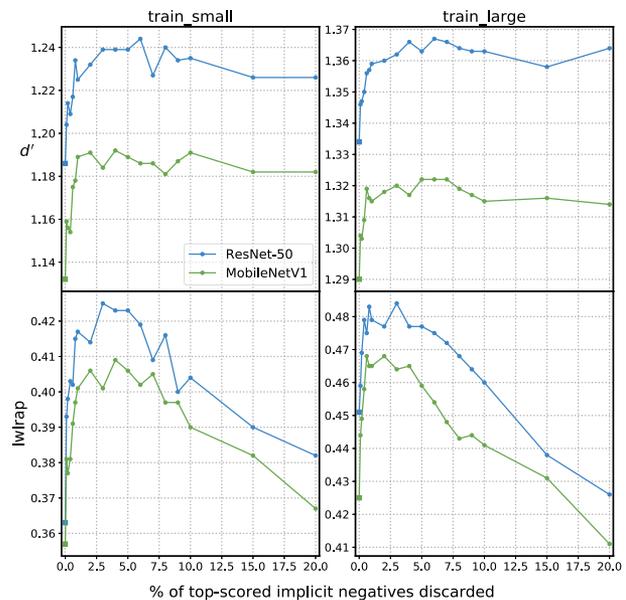
Fig. 2. Classification performance as a function of the proportion of top-scored negative labels that are discarded. Each point in the lines corresponds to one operating point. The leftmost point in each curve, marked with a square, corresponds to using all negative labels.

sets, each of them using a different threshold t_c, i.e., discarding a different proportion of top-scored negatives in the train set. Finally, for every enhanced label set, we train a student model on the train set, and predict on the evaluation set, reporting the best performance obtained.[2]

Fig. 2 illustrates the impact of missing labels by plotting performance as a function of the amount of top-scored negatives discarded, similar to the treatment of noisy ImageNet labels in [25]. We experiment with progressively discarding $t_c \in \{0, 0.1, 0.2, 0.4, 0.6, 0.8, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 15, 20\}$ % of top-scored negatives for the two train sets and architectures mentioned. Each point in the lines is the result of one experiment trial using one label set with a given amount of discarded negatives. The leftmost point (at $x = 0.0\%$, marked with a square) corresponds to *normal training* (no labels ignored and all false negative labels included). We use it as our baseline.

Common to all the curves of Fig. 2, we observe three regions from left to right: a steep increase at the beginning of the curve, followed by a sweet-spot, and a final decay that is more severe in *lωlrap* than in d'. A possible interpretation of this behaviour is as follows. We conjecture that the top-scored negatives correspond either to missing "Present" labels (i.e., false negatives (FNs)), *or* they are "decoys," difficult (and thus informative) true negatives (TNs), perhaps from similar classes, and especially useful in learning. First, we remove some critical FNs that damage the learning process, hence the sudden performance increase at the left of the curves. As we continue discarding more top-scored negatives, we keep removing FNs, but we also start to remove some TNs. Therefore, performance increases more slowly, until a sweet-spot is reached where both effects cancel out. Finally,

TABLE II
CLASSIFICATION PERFORMANCE FOR BASELINES AND BEST OPERATING
POINTS FOR ARCHITECTURES AND TRAIN SETS CONSIDERED

Model	Train set	*d'*		*lωlrap*	
		baseline	best	baseline	best
ResNet-50	*tr_small*	1.186	1.244	0.363	0.425
	tr_large	1.334	1.367	0.451	0.484
MobileNetV1	*tr_small*	1.132	1.192	0.357	0.409
	tr_large	1.290	1.322	0.425	0.468

if we keep ignoring more top-scored negatives, performance is degraded. As to why the decay in d' is much less pronounced than in $l\omega lrap$, a possible explanation lies in the way d' works. d' characterizes the separation of the positive and negative score distributions as the distance between their *means*. It may be that removing the high scoring tail changes the mean of the negative distribution (hence d' increases suddenly), but as we remove more labels with much more frequent scores the mean of the negative distribution barely changes (and consequently so d'). By contrast, $l\omega lrap$ does not suffer from this issue, its curves showing a decay as expected. Based on this intuition, we consider the right end of the d' curves less reliable. Table II lists the performances for baselines and best operating points for all the train sets and architectures considered. Based on Fig. 2 and Table II, next we make a number of observations.

Effect of ignoring the highest ranked top-scored negatives. The proposed method yields performance improvements in all cases considered. The best operating points are usually between 3 and 6% discarded for d', and between 1 and 4% for $l\omega lrap$. We believe this result is relevant as AudioSet training examples are often treated as if they had complete labels. However, the most important pattern we observe in all cases is the consistent steep increase at the beginning of the curves. In all cases, most of the improvement comes from removing just $\approx 1\%$ of the top-scored negatives. Further, in most cases, just by removing a tiny percentage ($<=0.2\%$) of (potentially) missing labels, approximately half of the total boost is already attained. Two observations can be made from these findings. First, this indicates that a tiny portion of labels is troublesome and it is moderately affecting classifier performance, a concept which is basis for disciplines like instance selection, where it is assumed that not all training examples are equally informative, some of them being redundant and some being harmful [26]. Second, these findings become interesting as they contrast with the common trend of acquiring more and more training data to improve recognition performance, even if noisily labeled [27] (something we also find useful in our experiments in general).

Effect of train set size. Table II shows improvements with respect to the baseline of ≈ 0.060 for d' when training with *tr_small* for both architectures, whereas when using *tr_large*, improvements are almost half of that (≈ 0.033). This relationship also holds for $l\omega lrap$ when using ResNet-50,[3] whereas when using MobileNetV1, the performance difference between training with *tr_small* and *tr_large* is smaller. These results seem to indicate that the damage done by missing labels, and consequently the performance boost obtained by discarding them, can be higher when the dataset is smaller. A possible explanation is that larger amounts of data help to mitigate the effect of these errors in

the label space, which accords with [27]. However, even when training with massive amounts of audio (almost 7000 h, see Table I), the impact of these labelling errors can still be observed. The d' sweet spot occurs roughly in the same region for both train sets. The $l\omega lrap$ sweet spot seems to move slightly to minimal discards when training with larger amounts of data.

Effect of model architectures. The proposed method is effective for both model architectures despite having different underlying principles and significantly different numbers of parameters. The overall trend of the curves in Fig. 2 is similar for both architectures. As can be seen in Table II, in terms of d', both architectures show very similar improvements with respect to their corresponding baselines. In terms of $l\omega lrap$, however, results are inconsistent, with ResNet-50 providing a greater improvement than MobileNetV1 when training on *tr_small*, and vice versa when training on *tr_large*. We do not observe consistently larger improvements using ResNet-50, even though its much larger number of parameters might lead one to expect it to overfit labeling errors more readily. Regardless of missing labels, when comparing baselines, ResNet-50 outperforms MobileNetV1, but not by a particularly large margin considering the huge difference in parameters.

Effect on evaluation metrics. By looking at Table II, it can be seen that d' improvements reach up to relative 5.3% (MobileNetV1) and $l\omega lrap$ improvements reach up to relative 17.1% (ResNet-50), both cases occurring when using *tr_small* ($\approx$half a million clips), where improvements are more evident.

Finally, we carried out a small informal listening test in which we inspected some of the clips associated with the discarded top-scored negatives for a few classes. As expected, most clips were missing "Present" labels, some of them being flagrant labelling errors, but difficult to detect considering the train set size. These findings indicate that the proposed method, while simple, is effective in identifying missing labels in a human annotated dataset like AudioSet, and it is able to improve training over unnoticed missing labels. Additionally, it can be useful for dataset cleaning or labeling refinement. Re-labelling a small amount of flagged top-scored negatives may lead to even better results than the proposed method. While the presented results are specific to AudioSet, the insight found can also apply to other large-scale audio datasets, especially those annotated via human validation of sub-optimally nominated candidates, e.g., the FSD dataset [28].

VI. CONCLUSION

We have identified missing labels as a pathology in the labelling of AudioSet. We have proposed a simple method based on a teacher-student framework with loss masking to first identify the most critical potentially missing labels, and then ignore them during the learning process. Our main findings are: i) most of the improvement comes from filtering out a tiny portion ($<1\%$) of the most critical estimated missing labels, showing a moderate impact on performance; ii) the damage done by missing labels (and the performance boost obtained by discarding them) becomes higher as the train set gets smaller—however, even when training with massive amounts of audio, the impact of these labelling errors can still be observed; iii) when applied to two CNN architectures of different nature and size the proposed method behaves similarly in both cases. We believe these insights will apply equally to large-scale audio datasets beyond AudioSet, since the problem of missing labels is endemic.

---

[3]By chance, absolute improvements for both metrics are numerically similar in this case, despite the metrics and their numeric range are different.

References

[1] T. Virtanen, M. D. Plumbley, and D. Ellis, *Computational Analysis of Sound Scenes and Events*. Berlin, Germany: Springer, 2018.

[2] P. Foster, S. Sigtia, S. Krstulovic, J. Barker, and M. D. Plumbley, "CHiME-home: A dataset for sound source recognition in a domestic environment," in *Proc. Workshop Appl. Signal Process. Audio Acoust.*, 2015.

[3] K. J. Piczak, "ESC: Dataset for environmental sound classification," in *Proc. ACM Int. Conf. Multimedia*, 2015, pp. 1015–1018.

[4] J. Salamon, C. Jacoby, and J. P. Bello, "A dataset and taxonomy for urban sound research," in *Proc. ACM Int. Conf. Multimedia*, 2014, pp. 1041–1044.

[5] E. Fonseca *et al.*, "General-purpose tagging of Freesound audio with AudioSet labels: Task description, dataset, and baseline," in *Proc. Detect. Classification Acoust. Scenes Events Workshop*, 2018, pp. 69–73.

[6] J. F. Gemmeke *et al.*, "Audio Set: An ontology and human-labeled dataset for audio events," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2017, pp. 776–780.

[7] E. Fonseca, M. Plakal, F. Font, D. P. W. Ellis, and X. Serra, "Audio tagging with noisy labels and minimal supervision," in *Proc. Detect. Classification Acoustic Scenes Events Workshop*, 2019, pp. 69–73.

[8] B. Frénay and M. Verleysen, "Classification in the presence of label noise: A survey," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 5, pp. 845–869, May 2014.

[9] M. Meire, P. Karsmakers, and L. Vuegen, "The impact of missing labels and overlapping sound events on multi-label multi-instance learning for sound event classification," in *Proc. Detect. Classification Acoust. Scenes Events Workshop*, 2019, pp. 159–163.

[10] A. Shah, A. Kumar, A. G. Hauptmann, and B. Raj, "A closer look at weak label learning for audio events," 2018, *arXiv:1804.09288*.

[11] E. Fonseca, M. Plakal, D. P. W. Ellis, F. Font, X. Favory, and X. Serra, "Learning sound event classifiers from web audio with noisy labels," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2019, pp. 21–25.

[12] E. Fonseca, F. Font, and X. Serra, "Model-agnostic approaches to handling noisy labels when training sound event classifiers," in *Proc. IEEE Workshop Appl. Signal Process. Audio Acoust.*, 2019, pp. 16–20.

[13] A. Kumar, A. Shah, A. Hauptmann, and B. Raj, "Learning sound events from webly labeled data," 2018, *arXiv:1811.09967*.

[14] A. Kumar and V. K. Ithapu, "Secost: Sequential co-supervision for weakly labeled audio event detection," 2019, *arXiv:1910.11789*.

[15] Q. Kong, Y. Cao, T. Iqbal, Y. Wang, W. Wang, and M. D. Plumbley, "PANNs: Large-scale pretrained audio neural networks for audio pattern recognition," 2019, *arXiv:1912.10211*.

[16] L. Ford, H. Tang, F. Grondin, and J. Glass, "A deep residual network for large-scale acoustic scene analysis," *Proc. Interspeech*, 2019, pp. 2568–2572.

[17] J. Ba and R. Caruana, "Do deep nets really need to be deep?" in *Proc. Advances Neural Inf. Process. Syst.*, 2014, pp. 2654–2662.

[18] J. Li, R. Zhao, J.-T. Huang, and Y. Gong, "Learning small-size DNN with output-distribution-based criteria," in *Proc. 15th Annu. Conf. Int. Speech Commun. Assoc.*, 2014.

[19] B. Kim and B. Pardo, "Sound event detection using point-labeled data," in *Proc. IEEE Workshop Appl. Signal Process. Audio Acoust.*, 2019, pp. 1–5.

[20] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2016, pp. 770–778.

[21] A. G. Howard *et al.*, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*.

[22] S. Hershey *et al.*, "CNN architectures for large-scale audio classification," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2017, pp. 131–135.

[23] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Representations*, 2015, *arXiv:1412.6980*.

[24] D. M. Green *et al.*, *Signal Detect. Theory and Psychophysics*, vol. 1. New York, NY, USA: Wiley, 1966.

[25] C. G. Northcutt, L. Jiang, and I. L. Chuang, "Confident learning: Estimating uncertainty in dataset labels," 2019, *arXiv:1911.00068*.

[26] H. Liu and H. Motoda, "On issues of instance selection," *Data Mining Knowl. Discovery*, vol. 6, no. 2, pp. 115–130, 2002.

[27] C. Sun, A. Shrivastava, S. Singh, and A. Gupta, "Revisiting unreasonable effectiveness of data in deep learning era," in *Proc. IEEE Int. Conf. Comput. Vision*, 2017, pp. 843–852.

[28] E. Fonseca, *et al.* "Freesound Datasets: A platform for the creation of open audio datasets," in *Proc. 18th Int. Soc. Music Inf. Retrieval Conf.*, 2017, pp. 486–493.

[29] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," in *Proc. NIPS Deep Learn. Representation Learn. Workshop*, 2015.