SELF-TRAINED AUDIO TAGGING AND SOUND EVENT DETECTION IN DOMESTIC ENVIRONMENTS

Janek Ebbers, Reinhold Haeb-Umbach

Paderborn University, Germany
{ebbers, haeb}@nt.upb.de

ABSTRACT

In this paper we present our system for the *Detection and Classification of Acoustic Scenes and Events (DCASE) 2021 Challenge Task 4: Sound Event Detection and Separation in Domestic Environments*, where it scored the fourth rank. Our presented solution is an advancement of our system used in the previous edition of the task.We use a forward-backward convolutional recurrent neural network (FBCRNN) for tagging and pseudo labeling followed by tag-conditioned sound event detection (SED) models which are trained using strong pseudo labels provided by the FBCRNN. Our advancement over our earlier model is threefold. First, we introduce a strong label loss in the objective of the FBCRNN to take advantage of the strongly labeled synthetic data during training. Second, we perform multiple iterations of self-training for both the FBCRNN and tag-conditioned SED models. Third, while we used only tag-conditioned CNNs as our SED model in the previous edition we here explore sophisticated tag-conditioned SED model architectures, namely, bidirectional CRNNs and bidirectional convolutional transformer neural networks (CTNNs), and combine them. With metric and class specific tuning of median filter lengths for post-processing, our final SED model, consisting of 6 submodels (2 of each architecture), achieves on the public evaluation set poly-phonic sound event detection scores (PSDS) of 0.455 for scenario 1 and 0.684 for scenario 2 as well as a collar-based F_1-score of 0.596 outperforming the baselines and our model from the previous edition by far. Source code is publicly available at `https://github.com/fgnt/pb_sed`.

Index Terms— sound event detection, audio tagging, weak labels, self-training

1. INTRODUCTION

Automatic Detection and Classification of Acoustic Scenes and Events (DCASE) has huge potential for various applications such as smart homes, multimedia search and environmental monitoring, to name a few. Due to the high diversity and variability of sounds, however, it is a challenging problem. Driven by the increasing interest from academia and industry and the success of data-driven approaches, the state-of-the-art in DCASE has recently progressed rapidly. The annual DCASE Challenges [1] further push and evaluate the current state-of-the-art in multiple sub-disciplines.

In this contribution we are concerned with the recognition of individual sound events. Here, sound event detection (SED) is the task of recognizing and temporally localizing sound events in an audio clip, whereas audio tagging aims to only recognize their presence within an audio clip without its temporal localization [2].

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) - 282835863. Computational resources were provided by the Paderborn Center for Parallel Computing.

One particular challenge in SED is that large-scale sound databases, such as Google's Audio Set [3], usually only provide tags a.k.a. weak labels, which only indicate the presence or absence of sound events within an audio clip without the information about the temporal location. Several approaches have been proposed for learning to localize sound events from weakly labeled data [4, 5, 6, 7], most of which use some sort of multiple instance learning (MIL) pooling function [8]. Another topic of interest, not only for sound recognition, is semi-supervised learning, which aims to exploit unlabeled data in addition to labeled data to improve performance. Here, approaches are usually based on representation learning [9], pre-training [10], teacher-student learning [11, 12] or self-training [13]. Self-training initially trains models on the available labeled data followed by iterative pseudo labeling [14] of the unlabeled data and retraining on labeled and pseudo labeled data.

For several years now, the Task 4 of the DCASE Challenge [15, 16, 17] tackles both of above challenges. Recently, it also explores the benefit of strongly labeled synthetic data in addition to weakly labeled and unlabeled real data. For this, the Domestic Environment Sound Event Detection (DESED) data set [16] with 10 different target sound events from a domestic environment has been designed. It is composed of 10-sec audio clips and comprises 1578 weakly labeled and 14412 unlabeled real training clips as well as isolated sound events and backgrounds for synthetic soundscape generation. Further, 1168 and 692 strongly labeled real audio clips are provided for validation and public evaluation, respectively.

In this paper we present our solution for the most recent *DCASE 2021 Challenge Task 4: Sound Event Detection and Separation in Domestic Environments*. Here, we built on our previously proposed forward-backward convolutional recurrent neural network (FBCRNN) and tag-conditioned SED [18], and propose three measures to improve performance. First, we introduce an explicit strong label loss in the FBCRNN training to exploit the strong labels from the synthetic data. Second, we perform more extensive self-training. Third, we explore more sophisticated CRNNs and convolutional transformer neural networks (CTNNs) for tag-conditioned SED in addition to the previously used CNN architecture. We show that all three measures improve performance, allowing us to significantly outperform the baseline and, to the best of our knowledge, set a new state-of-the-art in terms of collar-based F_1-score on the public evaluation set.

The rest of the paper is structured as follows. In Sec. 2 we recap the FBCRNN, introduce the strong label loss and outline the proposed FBCRNN self-training. In Sec. 3 we discuss architectures and self-training for the tag-conditioned SED. Sec. 4 presents implementation details w.r.t. data preparation, training and post processing. Finally, results are presented in Sec. 5 after which we draw conclusions in Sec. 6.

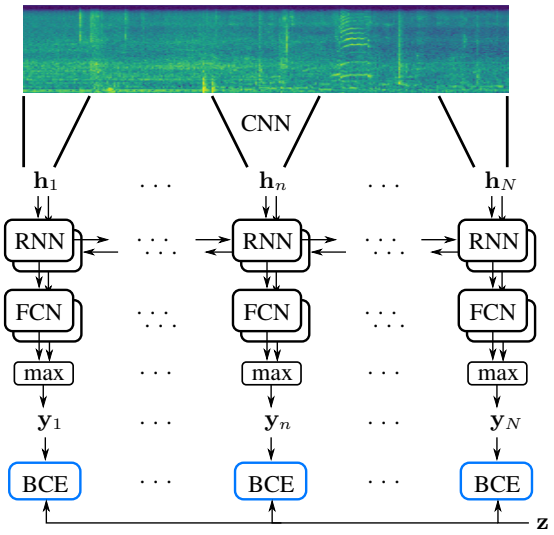

Figure 1: FBCRNN

2. FORWARD-BACKWARD CRNN

The FBCRNN [18] is illustrated in Fig. 1. It consists of a shared CNN front-end and two separate recurrent classifier networks (RNN+fully connected neural network (FCN)) with one processing the audio in forward direction and the other in backward direction. Note that unlike a bidirectional RNN the two classifiers do not exchange hidden representations and, therefore, at each frame one classifier has only seen previous frames and the other only subsequent frames.

To encourage the model to output tag predictions as soon as it has seen the event in the input when training with clip-level (weak) labels, we compute, at each frame, the binary cross entropy (BCE) loss between the point-wise maximum of the predictions of the two classifiers and the weak label. Fig. 2 shows an example, where the weak target and prediction signals are shown purple in the first and fourth subplots, respectively, assuming some decent forward and backward predictions shown in the third subplot. Note, that the FBCRNN training scheme can be seen as MIL with two instances. One instance comprises the current plus all previous frames, which has been processed by the forward classifier, and the other instance comprises the current plus all subsequent frames, which has been processed by the backward classifier. Hence, if an event is labeled positive in the clip at least one of the classifiers has to be able to classify the event as positive, given that the event is either present in previous or in subsequent frames or both.

At test-time a clip-level prediction is obtained by averaging the final forward and the final backward predictions of all models in an ensemble. As the proposed training scheme forces the forward and backward classifiers to output predictions without having processed the whole audio, the FBCRNN generalizes to much shorter segments at test-time. This enables FBCRNN-based SED, where FBCRNNs are applied to small contexts of, say, a couple of $100\,\mathrm{ms}$ around each frame to obtain frame-wise SED scores.

We use the same architecture as in [18], where, however, we removed the last pooling layer between the Conv2d and Conv1d blocks.

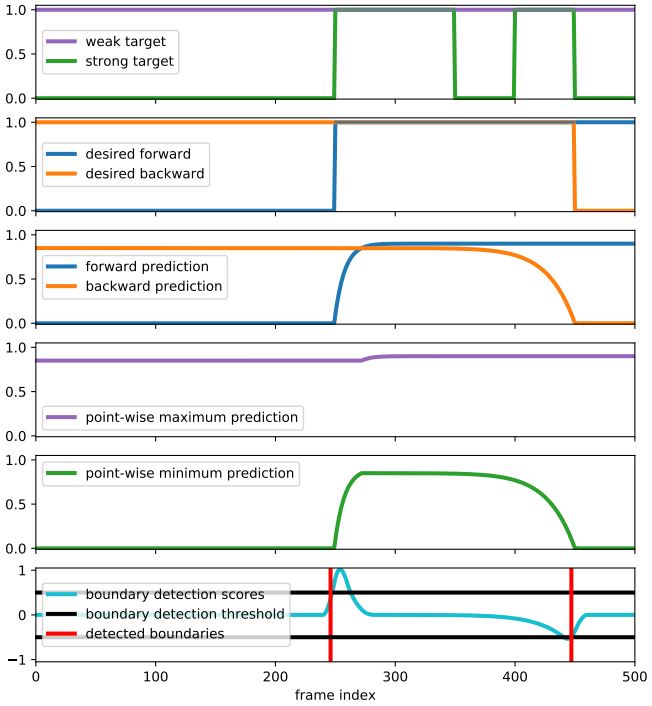

Figure 2: FBCRNN signals

2.1. Strong Label Loss

As the training data of the challenge contains synthetic data which comes with strong labels, it is desirable to make use of the strong labels in the FBCRNN training, which we previously did not do. If strong labels are given, we now, instead of the weak label loss, compute a strong label BCE for both classifiers with respect to the desired outputs, which are illustrated exemplarily in the second subplot of Fig. 2, and average the forward and backward loss terms.

2.2. Self-Training

As a large fraction of the provided data is unlabeled, we now perform more extensive self-training with training 8 initial FBCRNNs on only weakly labeled real and strongly labeled synthetic data followed by three iterations of pseudo labeling and retraining 4 FBCRNN models in each iteration.

In each iteration we generate weak pseudo labels for the complete unlabeled data, where tagging thresholds are tuned on the validation set to maximize the F_1-score. Additionally, we perform a boundary detection for weakly labeled and unlabeled data by filtering the point-wise minimum of the two classifier signals with $\begin{bmatrix}-2/N & \dots & -2/N & 2/N & \dots & 2/N\end{bmatrix}$ where N is the filter size. Exemplary point-wise minimum and subsequent boundary detection are depicted in the two last subplots of Fig. 2. The class-specific filter sizes and thresholds that the output or negative output has to exceed to detect an onset or offset boundary, respectively, are tuned on the validation data such that a minimum collar-based precision of $75\,\%$ is achieved, when using collars of $500\,\mathrm{ms}$. For those events where onset and offset can be detected, the strong label loss from Sec. 2.1 is used in the following FBCRNN retraining.

Finally, we use both the FBCRNN ensemble after the second and third iteration to separately perform strong pseudo labeling of the real data (weakly labeled and unlabeled) giving us a set of strong

pseudo labels for each of the ensembles, i.e. two in total. For the FBCRNN-based SED, class specific context lengths, median filter lengths and detection thresholds are tuned on the validation set to maximize the frame-based F_1-score. The obtained strong pseudo labels allow us to train SED systems in a strongly supervised manner as described in the following.

3. TAG-CONDITIONED SED

As in the previous edition, our SED model uses tag-conditioning [18], which means we also input the predicted tags from a FBCRNN (ensemble) in addition to the audio input features. While in the previous edition we only used a tag-conditioned CNN, we now also train a tag-conditioned bidirectional CRNN and tag-conditioned bidirectional CTNN.

Here, we use similar architectures as in the FBCRNN with, however, only one classifier back-end. For the pure CNN the CNN1d and RNN Blocks are removed. In the bidirectional CRNN, a bidirectional RNN is employed instead of unidirectional RNNs as in the FBCRNN. For the CTNN a Transformer Encoder [19] is used instead of an RNN, where we use 3 Transformer blocks each with 10 heads and 32-dimensional embeddings in each head. Also a positional encoding is added at the Transformer input.

Tag-conditioning is performed by concatenating a 10-dimensional multi-hot encoding of the tags with the inputs of the CNN2d, CNN1d, RNN/Tranformer, and FCN Blocks. For the CNN1d, RNN and FCN the encoding is concatenated along feature dimension at each frame. For the CNN2d the encoding is concatenated along channel dimension at each time-frequency bin.

The models are trained with standard strong label BCE loss. For each set of the 2 strong pseudo label sets we train each of the model architectures giving us 3 models for each of the 2 strong pseudo label sets. For each of the strong pseudo label sets, we perform one iteration of self-training, i.e., generating new strong pseudo labels using the 3 models of that particular set followed by retraining the 3 architectures. Finally, we combine all the models from the two sets of pseudo labels into our final ensemble, i.e., 6 models in total.

4. IMPLEMENTATION DETAILS

4.1. Data Preparation/Augmentation

Initially, waveforms are resampled to $16\,\mathrm{kHz}$ and normalized $x(t) = s(t)/\max(|s(t)|)$ to be within the range of -1 and 1. As our system's input we then extract a M=128-dimensional log-mel spectrogram using a short-time Fourier transform (STFT) with frame-length of $60\,\mathrm{ms}$ and hop-size of $20\,\mathrm{ms}$. Each mel-bin is globally normalized to zero mean and unit variance.

At training time we perform various on-the-fly data augmentations, which is similar to what we already used previously [20, 18] and is described in the following.

Scaling: We randomly scale the waveform with a scale weight sampled out of a Log Truncated Standard Normal distribution with truncation at $\log(3)$.

Shifted superposition: We randomly superpose two audios as $x_i'(t) = x_i(t) + x_j(t-\tau)$ with a random shift τ sampled uniformly such that the superposed signal is not longer than $15\,\mathrm{s}$, i.e., if we, e.g., superpose 2 signals each having a length of $10\,\mathrm{s}$, the shift is uniformly sampled between -5 s and $5\,\mathrm{s}$. Labels are superposed accordingly and clipped at 1 to retain binary labels. We apply superposition with a probability of $2/3$. Due to the similarity to mixup [21], we previously referred to this augmentation also as mixup. However, as we do not interpolate the signals, calling it superposition is more accurate.

Frequency warping: We randomly warp the center frequencies of the mel filter bank similar to vocal tract length perturbation (VTLP) [22]. The boundary frequency is sampled from a Truncated Exponential distribution with $\sigma = M/2$ and truncation at $5 \cdot M$. The warping factor is sampled from a Log Truncated Normal distribution with $\mu = 0$, $\sigma = 0.8$ and truncation at $\log(1.3) \approx 0.26$. Note that the boundary frequency can fall above M, in which case the whole spectrogram is stretched or squeezed and filled with zeros.

Frequency-/Time-Masking: As in SpecAugment [23], we apply one time- and one frequency mask for each input with random locations and widths. The locations are uniformly sampled along the time- and frequency axes, respectively. Widths are uniformly sampled between 0 and $\min(1.4\,\mathrm{s}, 0.2T)$ for the time mask, where T is the length of the audio, or between 0 and 20 bins for the frequency mask.

Gaussian Noise: We add Gaussian noise to the final feature map with its standard deviation being uniformly sampled between 0 and 0.2.

Note, that in contrast to [18], we here neither perform blurring nor reverberation of events in the synthetic data, since it has proven to be not effective.

4.2. Training

Training is performed for $40\,\mathrm{k}$ update steps with a batch size of 16. To balance the different data sets we repeat certain data sets in one epoch multiple times. Here, one epoch consists of 20 times the weakly labeled data, two times pseudo labeled unlabeled data (if used), one time the provided synthetic data from this edition (synthetic21) and two times the provided synthetic data from previous edition (synthetic20). This sums up to $\approx 31\,\mathrm{k}+28\,\mathrm{k}+10\,\mathrm{k}+5\,\mathrm{k}$ audio clips in one epoch. We further ensure that each batch includes at least 6 clips from the weakly labeled data, 2 clips from synthetic21 and 1 clip from synthetic20 as well as at least 1 example of each event class. We employ Adam [24] for optimization with a learning rate of $5 \cdot 10^{-4}$, with a ramp up during the first $1\,\mathrm{k}$ update steps and a reduction to 10^{-4} after $20\,\mathrm{k}$ update steps. We perform validation every $1\,\mathrm{k}$ update steps and choose the checkpoint with best validation performance in terms of (frame-based) F_1-score as the final model.

4.3. Post-Processing

At test-time we use median filtering and a non-linear score transformation for post-processing.

Median filter sizes are tuned for each event class and for each evaluation metric separately to give best performance on the validation set.

The class-specific non-linear score transformation serves the purpose of getting a smooth poly-phonic sound event detection scores (PSDS)-ROC [25] with linearly spaced detection thresholds. It transforms the prediction scores such that in the validation set prediction scores from positively labeled frames are uniformly distributed between 0 and 1. Note that the non-linear score transformation followed by linearly spaced detection thresholds is equivalent to non-linearly spaced detection thresholds.

Table 1: Single model FBCRNN performance on eval-public in %. Bold values indicate best performance in a column. Underlines indicate significant improvements within a block.

Iteration	PSDS1	PSDS2	$F_1^{(\text{collar})}$	$F_1^{(\text{tag})}$
0	31.6±0.6	67.3±1.7	44.1±1.1	83.8±0.8
w/o sll	29.0±2.1	67.2±3.0	41.2±1.9	83.3±0.6
1	36.4±0.5	68.0±1.0	49.1±1.4	84.6±0.3
w/o psll	33.2±0.7	68.9±1.3	47.4±0.6	85.1±0.7
2	**38.2**±0.9	68.9±1.3	**50.9**±1.0	85.1±0.4
3	37.9±1.4	**70.2**±1.2	50.7±1.2	**85.6**±0.6

Table 2: Single model tag-conditioned SED performance on eval-public in %. Bold values indicate best performance in the iteration.

Iteration	Model	PSDS1	PSDS2	$F_1^{(\text{collar})}$
0	CNN	38.2±2.7	64.4±0.4	54.4±0.1
	CRNN	39.7±0.8	**66.7**±0.8	54.5±0.1
	CTNN	**40.9**±1.5	66.2±0.6	**55.7**±0.5
1	CNN	39.6±1.2	64.3±0.6	54.4±0.3
	CRNN	39.8±0.6	**67.0**±1.0	**56.6**±0.1
	CTNN	**40.8**±1.6	66.3±0.4	56.5±0.6

5. RESULTS

We report results on the public evaluation set (eval-public), also referred to as Youtube evaluation [16, 26], as well as official challenge results[1] (eval-2021). Performance is measured in terms of

- PSDS1 / PSDS2: PSDSs [25] with two different sets of parameters as used in the challenge[2]
- $F_1^{(\text{collar})}$: macro-average collar-based F_1-score [27] with a 200 ms collar on onsets and a 200 ms / 20% of the event length collar on offsets
- $F_1^{(\text{tag})}$: macro-average audio tagging F_1-score

For F_1-scores, which evaluate a single operating point, class-specific detection thresholds are tuned to give best performance on the validation set. Note that PSDS1 and $F_1^{(\text{collar})}$ have a focus on accurate temporal localization of sound events, while PSDS2 and $F_1^{(\text{tag})}$ focus on the recognition of active classes.

For FBCRNN-based SED evaluation, context lengths are tuned along with the post-processing hyper parameters for each event class and for each SED evaluation metric separately to give best performance on the validation set. Table 1 presents the single model FBCRNN performance on eval-public over the iterations of the proposed self-training. For reference, we further report in iteration 0 the performance without the strong label loss (sll), as described in Sec. 2.1, and in iteration 1 the performance when not pseudo labeling boundaries in the real data, as described in Sec. 2.2, i.e., without a pseudo strong label loss (psll) on some real data. In each line we report the means and standard deviations over 4 independently trained models.

It can be observed that all metrics improve with the first two iterations of self-training. In the third iteration only PSDS2 and $F_1^{(\text{tag})}$ improve further, whereas PSDS1 and $F_1^{(\text{collar})}$ decrease insignificantly. Further, the proposed strong label loss (sll) and the pseudo strong label loss (psll), in iterations 0 and 1, respectively, allow to significantly improve PSDS1 and $F_1^{(\text{collar})}$ demonstrating their benefit for the temporal localization of sounds.

Table 3: Ensemble results on eval-public and eval-2021 in %. Bold values indicate best performance.

	eval-public			eval-2021		
Model	PSDS1	PSDS2	$F_1^{(\text{collar})}$	PSDS1	PSDS2	$F_1^{(\text{collar})}$
Baseline [12]	35.9	59.6	40.8	31.5	54.7	37.3
Winner [28]	**51.7**	**77.8**	57.4	**45.2**	**74.6**	52.3
FBCRNN	40.6	70.7	52.4	-	-	-
TCSED	45.5	68.4	**59.6**	41.6	63.7	**56.7**

Next, we evaluate the tag-conditioned SED (TCSED) models. Recap from Sec. 3 that we train each of the tag-conditioned architectures (CNN,CRNN,CTNN) on each of the strong pseudo label sets obtained from the FBCRNNs from iterations 2 and 3, followed by one iteration of self-training within each label set separately. In Table 2 we report the means and standard deviations of the results on eval-public over the two label sets.

When comparing performances between iterations 0 and 1, one can see that only for $F_1^{(\text{collar})}$ a significant improvement can be achieved in iteration 1. When comparing results for the different model architectures, it can be observed that tag-conditioned CRNNs and CTNNs perform more or less similar and outperform the tag-conditioned CNNs.

Finally, we report ensemble results in Table 3. Our final FBCRNN ensemble consists of 8 FBCRNNs from after the second and third iterations of the FBCRNN self-training (Table 1). Our TCSED ensemble, which was submitted to the challenge, consists of the 6 models after the single TCSED self-training iteration (Table 2). Both ensembles significantly outperform the challenge baseline w.r.t. all metrics. While the TCSED ensemble significantly outperforms the FBCRNN in PSDS1 and $F_1^{(\text{collar})}$, which both measure temporal localization of sound events, the FBCRNN achieves better results for PSDS2 which primarily measures recognition performance. Here, the FBCRNN-based SED benefits from the tuning of the context lengths, where large contexts are beneficial for PSDS2 evaluation. Compared to the winning system, our system, is outperformed in terms of PSDS1 and PSDS2. However, our TCSED ensemble achieves the highest $F_1^{(\text{collar})}$ of the challenge[2] and, to the best of our knowledge, the highest so far published $F_1^{(\text{collar})}$ on the eval-public set.

6. CONCLUSIONS

In this paper we presented our system for the *DCASE 2021 Challenge Task 4: Sound Event Detection and Separation in Domestic Environments*, where it scored the fourth rank. Starting from FBCRNNs followed by tag-conditioned SEDs, which we proposed in the previous challenge edition, we here presented three measures which significantly improve SED performance. First, we introduced a strong label loss in the FBCRNN training to leverage strong annotations, which is shown to improve temporal sound localization. Then, we performed extensive self-training in both FBCRNN training and tag-conditioned SED training, which particularly improves FBCRNN-based audio tagging and SED performance. Finally, we explored CRNN and CTNN architectures for tag-conditioned SEDs, in addition to CNNs used previously, which gives another performance gain. The proposed measures allow us to set a new, to the best of our knowledge, state-of-the-art in terms of collar-based F_1-score on the public evaluation set of the DESED data set.

[1] http://dcase.community/challenge2021/task-sound-event-detection-and-separation-in-domestic-environments-results

[2] http://dcase.community/challenge2021/task-sound-event-detection-and-separation-in-domestic-environments

7. REFERENCES

[1] "DCASE Challenges," http://dcase.community/events#challenges, accessed: 2021-07-20.

[2] T. Virtanen, M. D. Plumbley, and D. Ellis, *Computational analysis of sound scenes and events*. Springer, 2018.

[3] 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.

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

[5] B. McFee, J. Salamon, and J. P. Bello, "Adaptive pooling operators for weakly labeled sound event detection," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 11, pp. 2180–2193, 2018.

[6] Y. Xu, Q. Kong, W. Wang, and M. D. Plumbley, "Large-scale weakly supervised audio classification using gated convolutional neural network," in *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2018, pp. 121–125.

[7] K. Miyazaki, T. Komatsu, T. Hayashi, S. Watanabe, T. Toda, and K. Takeda, "Weakly-supervised sound event detection with self-attention," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 66–70.

[8] Y. Wang, J. Li, and F. Metze, "A comparison of five multiple instance learning pooling functions for sound event detection with weak labeling," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 31–35.

[9] A. Jansen, M. Plakal, R. Pandya, D. P. Ellis, S. Hershey, J. Liu, R. C. Moore, and R. A. Saurous, "Unsupervised learning of semantic audio representations," in *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2018, pp. 126–130.

[10] E. Fonseca, D. Ortego, K. McGuinness, N. E. OConnor, and X. Serra, "Unsupervised contrastive learning of sound event representations," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 371–375.

[11] L. JiaKai, "Mean teacher convolution system for dcase 2018 task 4," DCASE2018 Challenge, Tech. Rep., September 2018.

[12] N. Turpault and R. Serizel, "Training sound event detection on a heterogeneous dataset," in *DCASE Workshop*, 2020.

[13] B. Elizalde, A. Shah, S. Dalmia, M. H. Lee, R. Badlani, A. Kumar, B. Raj, and I. Lane, "An approach for self-training audio event detectors using web data," in *2017 25th European Signal Processing Conference (EUSIPCO)*. IEEE, 2017, pp. 1863–1867.

[14] D.-H. Lee, "Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks," in *Workshop on challenges in representation learning, ICML*, vol. 3, no. 2, 2013.

[15] R. Serizel, N. Turpault, H. Eghbal-Zadeh, and A. P. Shah, "Large-scale weakly labeled semi-supervised sound event detection in domestic environments," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018)*, November 2018, pp. 19–23.

[16] N. Turpault, R. Serizel, J. Salamon, and A. P. Shah, "Sound event detection in domestic environments with weakly labeled data and soundscape synthesis," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2019 Workshop (DCASE2019)*, New York University, NY, USA, October 2019, pp. 253–257.

[17] N. Turpault, R. Serizel, S. Wisdom, H. Erdogan, J. R. Hershey, E. Fonseca, P. Seetharaman, and J. Salamon, "Sound event detection and separation: a benchmark on desed synthetic soundscapes," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 840–844.

[18] J. Ebbers and R. Haeb-Umbach, "Forward-backward convolutional recurrent neural networks and tag-conditioned convolutional neural networks for weakly labeled semi-supervised sound event detection," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2020 Workshop (DCASE2020)*, Tokyo, Japan, November 2020, pp. 41–45.

[19] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *arXiv preprint arXiv:1706.03762*, 2017.

[20] J. Ebbers and R. Hb-Umbach, "Convolutional recurrent neural network and data augmentation for audio tagging with noisy labels and minimal supervision," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2019 Workshop (DCASE2019)*, New York University, NY, USA, October 2019, pp. 64–68.

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

[22] N. Jaitly and G. E. Hinton, "Vocal tract length perturbation (vtlp) improves speech recognition," in *Proceedings of ICML Workshop on Deep Learning for Audio, Speech and Language*, vol. 117, 2013.

[23] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "Specaugment: A simple data augmentation method for automatic speech recognition," *arXiv preprint arXiv:1904.08779*, 2019.

[24] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[25] Ç. Bilen, G. Ferroni, F. Tuveri, J. Azcarreta, and S. Krstulović, "A framework for the robust evaluation of sound event detection," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 61–65.

[26] N. Turpault, R. Serizel, A. Shah, and J. Salamon, "Desed_public_eval," Dec. 2019. [Online]. Available: https://doi.org/10.5281/zenodo.3588172

[27] A. Mesaros, T. Heittola, and T. Virtanen, "Metrics for polyphonic sound event detection," *Applied Sciences*, vol. 6, no. 6, p. 162, 2016.

[28] X. Zheng, H. Chen, and Y. Song, "Zheng ustc teams submission for dcase2021 task4 semi-supervised sound event detection," DCASE2021 Challenge, Tech. Rep., June 2021.