Sound Event Detection with Cross-Referencing Self-Training

Sangwook Park[1], *Woohyun Choi*[2], *Mounya Elhilali*[1]

[1]Department of Electrical and Computer Engineering, Johns Hopkins University
[2]LG electronics

{spark190, mounya}@jh.edu, settler12@gmail.com

Abstract

This report describes a sound event detection method submitted to the DCASE2021 challenge, task 4. In this approach, we design a residual convolutional recurrent neural network and train this network with a cross-referencing self-training approach that leverages an extensive unlabeled data in combination with labeled data. This approach takes advantage of semi-supervised training using pseudo-labels from a balanced student-teacher model, and outperforms DCASE2021 challenge baseline in terms of Poly-phonic Sound event Detection Score. Additionally, the proposed network has more accurate predictions in class-wise collar-based-F1, compared to the baseline.

Index Terms: self-training, few-shot learning, sound event detection, multi-target detection

1. Introduction

This report describes a Sound Event Detection (SED) system which is submitted to the task 4 for Detection and Classification of Acoustic Scenes and Events (DCASE) 2021 challenge. The goal of SED is to identify sounds of interests in an audio clip as sound class and time boundaries as well. Its applications include audio surveillance [1, 2] in various location such as smart home and cities [3], health monitoring, life logging and multimedia retrieval. Since DCASE 2017, SED task has featured as task 4 that deals with weakly labeled data. In task 4 of this challenge, 10-domestic sound events are considered as the target events such Alarm/bell/ringing, Blender, Cat, Dishes, Dog, Electric shaver/tooth brush, Frying, Running water, Speech, and Vacuum cleaner. And a question has been introduced: How to apply weakly labeled and/or unlabeled data in combination with synthesized strong labeled data in network training?

We propose a Cross-Referencing Self-Training (CRST) approach that leverages weakly labeled and unlabeled data in supervised fashion [4]. In case of the challenge baseline, a Mean Teacher (MT) approach is used to train the network [5, 6]. The MT approach is composed of *student* and *teacher* networks. For the *student* network, the parameters are optimized by using gradient descent method. On the other hand *teacher* parameters are updated by moving average of *student* parameters over the training. In the proposed method, we incorporate two model as depicted in Fig. 1 (*Model I* and *Model II*) which estimate pseudo labels by themselves and pass the estimate to the other model. In parallel, *Model II* is trained on different version of data produced by a transformation function $T(.)$. By independently training these two models, the proposed framework resolves issues of self-biasing that arise from self-referencing schemes [7].

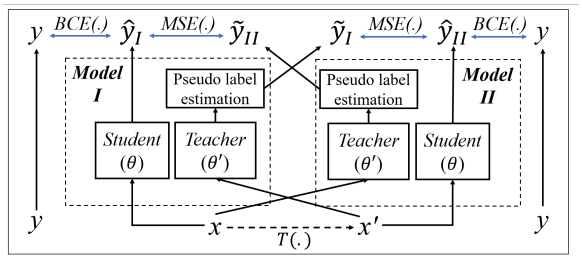

Figure 1: *Block diagrams of cross-referencing self-training.*

2. Method

The proposed approach extends the baseline system for this challenge [8] by incorporating modifications in network architecture and using a semi-supervised learning strategy, as detailed next. All other components of system design and training are similar to the baseline implementation: batch size (48), batch composition of strong labeled data (12), weakly labeled data (12), and unlabeled data (24), learning rate(0.001 for maximum), exponential ramp-up function, validation threshold (0.5), data augmentation based on soft-mixup with a 50 % change, and min-max normalization for network input.

2.1. Pre-processing

Each audio clip is resampled to a 16kHz mono-channel audio waveform and it is converted to a spectrogram by performing Short Time Fourier Transform (STFT) with 2048-points frame length and 255-points hop size. Then, a log-Mel spectrogram is obtained by performing frequency integration with 128 Mel-filters spanned 0 to 8kHz frequency domain and logarithm function. Note that audio length is set to 10 second with zero-padding or cutting for shorter or longer audios than 10 second, respectively.

2.2. Network architecture

Inspired by a Residual Convolutional Recurrent Neural Network (RCRNN) proposed in [9], the proposed network architecture is designed with residual convolutional layer. As shown in Fig. 2, the network is composed of two parts: Convolutional Neural Network (CNN) and Bidirectional Gate Recurrent Unit (BGRU). In the CNN section, stem-block consists of convolutional layer (Conv) with 3×3 kernel, 1×1 stride, Batch Normalization (BN), Gate Linear Unit (GLU), and 2D Average pooling (AvgPool) along to time-frequency axes. The residual convolution block (R-Conv) consists of one convolutional layer for skip connection and two convolutional layers, BN, and ReLU activation as in Fig. 3. Note that all convolutional layers in R-Conv use 3×3 kernel and 1×1 stride. Then, the Convolutional Block Attention Module (CBAM) proposed in [10] and AvgPool along

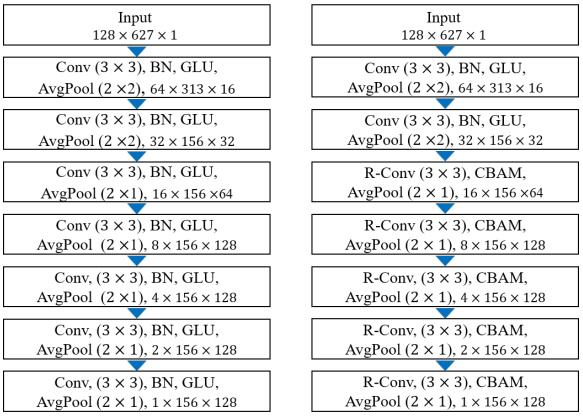

Figure 2: *Block diagrams for network architecture. (a) Convolutional Recurrent Neural Network (CRNN) used in baseline, (b) Residual CRNN*

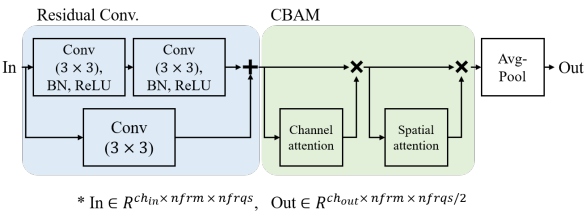

Figure 3: *Block description for R-Conv*

to frequency axis only are followed by each R-Conv.

2.3. Cross-Referencing Self-Training

To leverage both weakly labeled data and unlabeled data in supervised learning, a Cross-Referencing Self-Training (CRST) approach is applied for network training [4]. In this work, the *student* and *teacher* in both models are designed by using the RCRNN network, and both have the same architecture to each other. The objective function to train each model is defined as

$$\begin{aligned} L_I =& \sum_{x\in S} BCE(f_I(x), y^s) + \sum_{x\in W} BCE(E[f_I(x)], y^w) \\ &+\gamma^s_{II} \sum_{x\in U} (MSE(f_I(x), \tilde{y}_{II}) + \gamma^w_{II} \sum_{x\in W} (MSE(f_I(x), \tilde{y}_{II}), \\ L_{II} =& \sum_{x'\in S} BCE(f_{II}(x'), y^s) + \sum_{x'\in W} BCE(E[f_{II}(x')], y^w) \\ &+\gamma^s_{I} \sum_{x'\in U} (MSE(f_{II}(x'), \tilde{y}_{I}) + \gamma^w_{I} \sum_{x'\in W} (MSE(f_{II}(x'), \tilde{y}_{I}), \end{aligned} \tag{1}$$

where L_i is the objective function for training *Model i* whose prediction is denoted as $f_i(.)$. x' is produced by performing frame shifting on original log-Mel spectrogram x with a random delay factor generated by Gaussian distribution $G(0, 40)$. $E[.]$ is an averaging operator over the frames. y^w and y^s are labels for weak labeled data and strong labeled data, respectively. γ^s_i and γ^w_i are the reliability of pseudo label estimated in *Model i* with strong labeled data and weakly labeled data, respectively.

With an assumption that *teacher* prediction represents a posterior probability of each target class, the pseudo label is estimated to a probabilistic expectation of potential labels as

$$\tilde{y} = \Sigma_k^K \, \Sigma_n^{N_k} \, p_n^k \, l_n^k, \tag{2}$$

where k is the number of concurrent events in each frame and n is an index for the case of choosing k-sounds of total target sounds. l_n^k is a potential label vector expressed by a summation of delta functions like $l^2_{n:\{i,j\}} = \delta_i + \delta_j$ for events i and j ($(k = 2)$. p_n^k is a probability of the label l_n^k. With Bernoulli process, the probability is calculated by multiplying posterior for potential sound class and complementary for others. For the case of a potential label $l^2_{n:i,j}$, as an instance, the probability is $p^2_{n:\{i,j\}} = \frac{1}{N} \hat{y}'_i \hat{y}'_j \Pi_{q\neq i,j} \, (1 - \hat{y}'_q)$. Note that $\hat{y}'$ is *teacher* prediction. K is maximum number of concurrent events, and $N_k = C!/(k! \times (C-k)!)$ is the number of potential labels under the k and total number of target sound classes C. In this work, the K is set to 3 due to the mixup with a 50% chance for data augmentation.

The reliability of pseudo label is designed with Jensen Shannon Divergence (JSD), which is bounded in [0,1].

$$\begin{aligned} \gamma^s &= \omega \times \frac{1}{N^s} \sum_{x\in S} (1 - JSD(\tilde{y}||y^s)), \\ \gamma^w &= \omega \times \frac{1}{N^w} \sum_{x\in W} (1 - JSD(E[\tilde{y}]||y^w)), \\ &where \; \omega = 3.0exp(-5(1 - t/T)^2), \\ &JSD(a||b) = KLD(a||m)/2 + KLD(b||m)/2, \\ &m = (a + b)/2, \end{aligned} \tag{3}$$

where γ^s and γ^w is reliability with respect to strong labeled and weakly labeled data, respectively. ω is a ramp-up parameter with an index of training step t and maximum number of the steps T, N^s and N^w is the number of strong labeled data and weakly labeled data, respectively. KLD is a Kullback Leibler Divergence. Note that a ramp-up function for ω is same with a function used in baseline.

While both *Model I* and *Model II* are separately optimized in the training phase, predictions during the validation and test phases are produced by averaging two outputs from each *student* and *teacher* network.

2.4. Post-processing

Imbalance in the number of training data for each target class introduces a bias toward a dominant class. This framework is not free from this issue because the network is trained on real recordings. However, it is hard to use one of methods such as dynamic sampling [11] or data augmentation [12] for minority class data since those methods need class label for all training data. Instead, class-wise parameters of threshold and smoothing are used in post-processing. With weakly labeled data, we collect samples for network prediction of each class, and we applied Extreme Value Theory (EVT) to the samples to estimate threshold. For smoothing length, we calculated average length of sound duration for each target class, then a 25% of the average is decided to the smoothing length (for more detail about post-processing, please find [4]).

3. Experiment

3.1. Database

Domestic Environment Sound Event Detection (DESED) database is used for evaluation. The database has three types

Table 1: *Performance comparison with global post-processing*

		PSDS1	PSDS2	Intersection based F1	Collar-based F1
MT (baseline) $w/CRNN$	Student	0.3400	0.5346	0.7635	0.4009
	Teacher	0.3424	0.5381	0.7753	0.4200
MT $w/RCRNN$	Student	0.2700	0.5036	0.8130	0.3923
	Teacher	0.2613	0.5083	**0.8263**	0.4072
CRST $w/CRNN$	Student	0.3305	0.5304	0.6912	0.3842
	Teacher	0.3381	0.5327	0.6907	0.3814
CRST $w/RCRNN$	Student	0.3406	0.6323	0.7445	0.3825
	Teacher	**0.3600**	**0.6421**	0.7758	**0.4284**

Table 2: *Effect of classwise post-processing in CRST w/ RCRNN*

		PSDS1	PSDS2	Intersection based F1	Collar-based F1
global thresholding & smoothing	Student	0.3406	0.6323	0.7445	0.3825
	Teacher	0.3600	0.6421	**0.7758**	0.4284
global thresholding & classwise smoothing	Student[1]	0.5082	0.6684	0.7572	0.4131
	Teacher[2]	**0.5237**	**0.6737**	0.7740	0.4389
classwise thresholding & smoothing	Student[3]	0.4564	0.5957	0.5864	0.4289
	Teacher[4]	0.4464	0.5609	0.7737	**0.4558**

* Superscription indicates submission ID

of training dataset: strong labeled, weakly labeled, and unlabeled dataset [13]. The strong labeled data let us know sound class and timestamps for each target sound interval. For weakly labeled data, the label let us know sound class only while unlabeled data has no information about the truth. The weakly labeled and the unlabeled dataset contain 1,578 and 14,412 audio clips, respectively. We generate about 10,000 audio clips by using the Scaper soundscape library for synthesis and augmentation [14], and SINS database and TUT Acoustic scenes 2017 database for background sounds. The validation dataset, including 1,168 audio clips, is taken from the DESED dataset for evaluation.

3.2. Result

A Poly-phonic Sound event Detection Scores (PSDS) [PSDS] is evaluated in two different scenarios: 1) The system needs to react fast upon an event detection. For this case, PSDS1 is sensitive to time accuracy. 2) The system has to avoid confusing between classes but reaction time is less crucial than in the first scenario. For this scenario, PSDS2 is calculated. To calculate PSDS metric, 0.01 to 0.99 with 0.02 step are used as a threshold for all targets. Additionally, intersection- and collar- based F1 measures are used as contrastive metric. In a collar based F1 measure, 200 ms and 200 ms / 20% of the event length is applied for a collar on onsets and offsets, respectively. Note that 0.5 and about 450 ms are applied to all target classes as a threshold and smoothing length, respectively.

Table 1 shows the results of *Student* and *Teacher* networks in baseline and CRST. In PSDS results, the CRST (with RCRNN) shows an improvement in the second scenario while its time accuracy is comparable to the baseline. Particularly, *teacher* network shows better than *student* network. It is consistent with the idea of using averaging model which tends to produce more accurate predictions. On the other hand, the effectiveness of RCRNN architecture is inconsistent depending on semi-supervised learning method. CRST approach is more effective on RCRNN structure while MT approach prefers CRNN structure to improve the performance. It is remained to explore the best network architecture as a future work,

Table 2 shows the effect of classwise post-processing in the RCRNN based CRST. To calculate PSDS metric for classwise thresholding, 1% to 99 % with 2% step of classwise threshold estimated based on EVT are used. In PSDS results, "global thresholding & classwise smoothing" shows the best among three cases. Especially, classwise smoothing improves time accuracy compared to other two cases in PSDS1 metric. In collar-based F1 results, "classwise thresholding & smoothing" shows the best among them. Table 3 shows the collar-based F1 metric in each target class. According to the result, the CRST has an issue to detect *Dishes* sound. In case of *Dishes*, its duration is relatively too short and its frequency is not much. Due to this imbalance, the model has seen non-dishes sounds much more than dishes sounds. This limitation can be resolved by performing classwise thresholding and smoothing.

4. Summary

In this report, we explored semi-supervised learning approach that leverages unlabeled data in supervised training, and applied a cross-referencing self-training approach to network training for the sound event detection task of DCASE2021 challenge. For this challenge, we built two models *Model I* and *Model II*, where each model estimates pseudo labels by itself and passes the estimate to the other model, with an expectation that resolves the self-biasing issue in a self-referencing framework. The effectiveness of this approach was demonstrated in experiments. In evaluation with PSDS metric, our best model achieves 0.5237 and 0.6737 for PSDS1 and PSDS2, respectively on the validation set (1,168 real audio clips). Additionally, "classwise thresholding and smoothing" further improved in evaluation with collar-based F1 metric.

Table 3: *Classwise collar-based F1 measure*

post-processing network	**Baseline**$_{w/CRNN}$ global		**CRST**$_{w/RCRNN}$ global		classwise smoothing		classwise	
	student	*teacher*	*student*	*teacher*	*student*	*teacher*	*student*	*teacher*
Alarm	0.422	0.430	0.262	0.433	0.259	0.432	0.320	0.455
Blend	0.433	0.430	0.465	0.480	0.497	0.480	0.448	0.439
Cat	0.417	0.454	0.324	0.360	0.352	0.406	0.318	0.358
Dish	0.236	0.248	0.109	0.134	0.181	0.112	0.316	0.313
Dog	0.208	0.228	0.201	0.217	0.213	0.267	0.226	0.257
Elec	0.515	0.538	0.528	0.640	0.571	0.607	0.609	0.600
Fry	0.392	0.362	0.497	0.543	0.515	0.495	0.506	0.525
R.W.	0.362	0.379	0.345	0.382	0.354	0.381	0.361	0.364
Speech	0.511	0.527	0.547	0.546	0.606	0.637	0.615	0.626
V.C.	0.512	0.604	0.547	0.549	0.583	0.573	0.571	0.620
Avg	0.4009	0.4200	0.3825	0.4284	0.4131	0.4389	0.4289	0.4558

5. References

[1] W. Choi, J. Rho, D. K. Han, and H. Ko, "Selective background adaptation based abnormal acoustic event recognition for audio surveillance," in *2012 IEEE Ninth International Conference on Advanced Video and Signal-Based Surveillance*, 2012, pp. 118–123.

[2] S. Park, W. Choi, D. K. Han, and H. Ko, "Acoustic event filterbank for enabling robust event recognition by cleaning robot," *IEEE Transactions on Consumer Electronics*, vol. 61, no. 2, pp. 189–196, 2015.

[3] C. Debes, A. Merentitis, S. Sukhanov, M. Niessen, N. Frangiadakis, and A. Bauer, "Monitoring activities of daily living in smart homes: Understanding human behavior," *IEEE Signal Processing Magazine*, vol. 33, no. 2, pp. 81–94, 2016.

[4] S. Park, D. K. Han, and M. Elhilali, "Cross-referencing self-training network for sound event detection in audio mixtures," *CoRR*, vol. abs/2105.13392, 2021. [Online]. Available: https://arxiv.org/abs/2105.13392

[5] N. Turpault and R. Serizel, "Training Sound Event Detection On A Heterogeneous Dataset," in *DCASE workshop*, 2020. [Online]. Available: http://arxiv.org/abs/2007.03931

[6] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," in *Advances in Neural Information Processing Systems*, vol. 2017-Decem, no. Nips, 2017, pp. 1196–1205.

[7] S. Park, A. Bellur, D. K. Han, and M. Elhilali, "Self-Training for Sound Event Detection in Audio Mixtures," in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, jun 2021, pp. 341–345. [Online]. Available: https://ieeexplore.ieee.org/document/9414450/

[8] https://github.com/DCASE-REPO/DESED_task/tree/master /recipes/dcase2021_task4_baseline.

[9] N. K. Kim and H. K. Kim, "Polyphonic Sound Event Detection Based on Residual Convolutional Recurrent Neural Network with Semi-Supervised Loss Function," *IEEE Access*, vol. 9, pp. 7564–7575, 2021. [Online]. Available: https://ieeexplore.ieee.org/document/9312148/

[10] S. Woo, J. Park, J. Lee, and I. S. Kweon, "CBAM: convolutional block attention module," *CoRR*, vol. abs/1807.06521, 2018. [Online]. Available: http://arxiv.org/abs/1807.06521

[11] S. Pouyanfar, Y. Tao, A. Mohan, H. Tian, A. S. Kaseb, K. Gauen, R. Dailey, S. Aghajanzadeh, Y.-H. Lu, S.-C. Chen, and M.-L. Shyu, "Dynamic sampling in convolutional neural networks for imbalanced data classification," in *2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, 2018, pp. 112–117.

[12] H. Lee, M. Park, and J. Kim, "Plankton classification on imbalanced large scale database via convolutional neural networks with transfer learning," in *2016 IEEE International Conference on Image Processing (ICIP)*, 2016, pp. 3713–3717.

[13] https://project.inria.fr/desed/.

[14] J. Salamon, D. MacConnell, M. Cartwright, P. Li, and J. Bello, "Scaper: A library for soundscape synthesis and augmentation," in *2017 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, WASPAA 2017*, Dec. 2017, pp. 344–348.