ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 978-1-7281-7605-5/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICASSP39728.2021.9414450

SELF-TRAINING FOR SOUND EVENT DETECTION IN AUDIO MIXTURES

Sangwook Park[1]*, Ashwin Bellur*[1]*, David K. Han*[2]*, and Mounya Elhilali*[1]

[1]Department of Electrical and Computer Engineering, Johns Hopkins University, Baltimore, MD, USA
[2]Department of Electrical and Computer Engineering, Drexel University, Philadelphia, PA, USA

ABSTRACT

Sound event detection (SED) takes on the task of identifying presence of specific sound events in a complex audio recording. SED has tremendous implications in video analytics, smart speaker algorithms and audio tagging. Recent advances in deep learning have afforded remarkable advances in performance of SED systems; albeit at the cost of extensive labeling efforts to train supervised methods using fully described sound class labels and timestamps. In order to address limitations in availability of training data, this work proposes a self-training technique to leverage unlabeled datasets in supervised learning using pseudo label estimation. This approach proposes a dual-term objective function: a classification loss for the original labels and expectation loss for pseudo labels. The proposed self training technique is applied to sound event detection in the context of the DCASE 2020 challenge, and reports a notable improvement over the baseline system for this task. The self-training approach is particularly effective in extending the labeled database with concurrent sound events.

Index Terms— Semi-supervised learning, sound event detection, pseudo label, reliability, DCASE2020

1. INTRODUCTION

Analysis of sound events in audio recordings enables the detection of presence of different sound classes including human voice, animal vocalizations, man-made objects, etc. Identification of such sounds is critical in a number of applications spanning video analytics, multimedia tagging, baby or pets monitoring, or other surveillance applications [1, 2, 3]. These applications generally specify the types of events of interest that a sound event detection (SED) system aims to identify.

Recent approaches based on deep networks have shown tremendous improvement in SED task performance [4, 5, 6, 7]. With fully described labels for time boundaries and associated sound event classes, these models are able to learn temporal characteristics across target sound events. However, as is the case with other deep learning networks, these methods require a large amount of labeled data to optimize the system parameters. As curation of large labeled data is expensive and time consuming, alternative ideas have been explored particularly semi-supervised learning which leverages extensive unlabeled data in combination with small amounts of labeled data [8, 6, 9]. The Detection and Classification of Acoustic Scenes and Events (DCASE) task 4 challenge focuses on use of unlabeled data along with weakly labeled data which includes the identity of sound event classes without time boundary markings in order to train SED system [10]. In this challenge, a variety of network architectures have been suggested for the SED task with some notable improvements over the baseline [11, 12, 13, 14].

As in the case of DCASE task 4, the problem scenario in the current work is also to leverage unlabeled and weakly labeled data. However, this paper focuses on developing a new semi-supervised learning approach rather than optimizing the network structure. The novel learning method proposed here is based on pseudo label estimation and a weighted objective function computed from the pseudo label for self-training. A pseudo label is estimated based on expectations of potential labels. A probability for each potential label is calculated based on a Bernoulli process with class posteriors produced by an averaging network. The proposed objective function consists of classification loss between true labels and network predictions for labeled data and expectation loss between the pseudo label and the network prediction for unlabeled and weakly labeled data. To balance the contributions of the two loss terms on training, a weight designed by a cross entropy between true and pseudo labels for labeled data is multiplied by the expectation loss. To demonstrate effectiveness of the proposed method, experiments are performed following the protocol for the SED task in the recent DCASE challenge (DCASE2020). As a result, the proposed method shows statistically-significant improvement in SED performance. It also highlights advantages of the proposed approach when dealing with concurrent sound events.

2. RELATED WORK

There has been a growing body of work exploring use of unlabeled data in supervised learning [15]. Among the ap-

This work was supported by NIH U01AG058532 and R01HL133043, ONR N00014-19-1-2014, and N00014-19-1-2689.

proaches worth noting, the MeanTeacher model has been instrumental in pushing forth the state of the art in image classification [16]. The objective function for the MeanTeacher model consists of a classification loss and a consistency loss as

$$f_L = BCE(\hat{y}, y) + \alpha MSE(\hat{y}, \hat{y}'), \quad (1)$$

where the classification loss $BCE(\hat{y}, y)$ is designed by a binary cross entropy between *student* network prediction $\hat{y}$ and true label y. The consistency loss $MSE(\hat{y}, \hat{y}')$ is calculated by mean squared error between two predictions by *student* and *teacher* networks $\hat{y}'$. α is a coefficient to adjust contribution of the second term in training. The *student* network parameters are updated by gradient descent while the *teacher* network parameters are updated by exponential moving average of *student* network parameters over the training step (Fig.1(a)). Note that both networks have the same structure while random perturbations such rotating, shifting, or adding noise is independently performed on each network's mid-layer. The motivation for this configuration is to enable the *student* network to produce the same outputs even in presence of various perturbations. As a result, the model is able to map any data point within a manifold into similar predictions.

In an alternative approach known as self-training, a network is iteratively trained with both labeled data and unlabeled data with pseudo label estimated in previous iterations [17]. Thus, the concept is quite similar in nature with semi-supervised methods by exploiting its ability to generate pseudo labels for self training. It is, therefore, critically important to estimate pseudo label accurately to prevent confusion due to the label estimate.

Other approaches fall in the continuum between these methods, among them is pseudo-label training which is explored in this paper. The consistency loss in the MeanTeacher model trains the network to produce the same outputs from any data point within a manifold. However, the model has no way to provide any mitigation measures when the outputs are not accurate. These incorrect outputs may affect the overall model performance when a large amount of unlabeled data is used compared to the amount of labeled data [18]. To combat this issue, we propose a method of pseudo label estimation and an objective function designed to adjust its influence in training according to its label estimation accuracy (Fig.1(b)).

3. PROPOSED METHOD

3.1. Pre-processing and Network architecture

Each input audio clip is modified to a 16kHz mono-channel waveform by resampling and averaging left and right channels for multi-channel recordings. Then, the waveform is converted to a log mel-spectrogram with 2048 FFT window, 255 hop size, and 128 mel frequency channels.

Since optimizing the network structure is out of focus in this work, a Convolutional Recurrent Neural Network (CRNN), which is used for SED task in DCASE2020 challenge as a baseline, is adopted. The network is composed of two stages: Convolutional Neural Network (CNN) to compress acoustic features within the log mel-spectrogram and Gating Recurrent Units (GRUs) to capture temporal relations among the compressed codes by the CNN. The core element of the CNN consists of a convolutional layer, an average pooling layer, and a fully connected layer with a relu activation, batch normalization, and dropout techniques. A two-layered bidirectional GRUs is composing for the following stage. At the final output layer of the network, a sigmoid function is applied to represent posterior probability for each target event. More details can be found in [10].

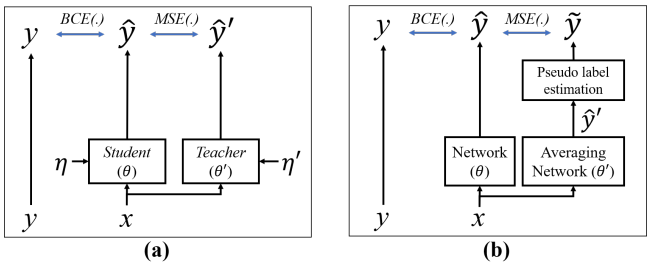

Fig. 1. Diagrams for semi-supervised learning where x is input data. θ and θ' mean parameters for network (Student) and averaging network (Teacher), respectively; (a) Mean Teacher model where η and η' are random perturbations in each network's mid-layer, (b) Proposed method where $\tilde{y}$ is the pseudo label.

3.2. Objective function

The objective function in the proposed method is designed as

$$f_L = BCE(\hat{y}, y) + \gamma MSE(\hat{y}, \tilde{y}). \quad (2)$$

The equation is nearly identical to equation (1) except that we define the expectation loss $MSE(\hat{y}, \tilde{y})$ as a mean squared error between the network prediction and the pseudo label $\tilde{y}$ with a reliability γ of the pseudo label.

3.3. Pseudo label estimation

l_n^k represents a label vector where k is the number of possible concurrent events in each frame and n is an index to represent all possible combination of concurrent events under k. The label vector l_n^k can be expressed by a summation of one-hot vector expressed as delta function as $l^2_{n:\{i,j\}} = \delta_i + \delta_j$ for events i and j ($k = 2$). Then, a pseudo label can be estimated by expectation of all possible labels as

$$\tilde{y} = \Sigma_k^K \, \Sigma_n^{N_k} \, p_n^k \, l_n^k, \quad (3)$$

where p_n^k is a probability for a label l_n^k, K is maximum number of concurrent events, and $N_k = C!/(k! \times (C-k)!)$ is the

number of possible labels under k and total number of target event categories C.

Since the averaging network, which is built by exponential moving average of the network weights over the training steps, tends to produce more accurate predictions [16], for a prediction $\hat{y}'$ by the averaging network, the probability for each possible label is calculated based on Bernoulli processing. For example, the probabilities are calculated depending on the k as

$$\begin{aligned} k=0, &\quad p^0_{n:\{\}} = \frac{1}{N}\Pi_q\,(1-\hat{y}'_q), \\ k=1, &\quad p^1_{n:\{i\}} = \frac{1}{N}\hat{y}'_i\Pi_{q\neq i}\,(1-\hat{y}'_q), \\ k=2, &\quad p^2_{n:\{i,j\}} = \frac{1}{N}\hat{y}'_i\hat{y}'_j\Pi_{q\neq i,j}\,(1-\hat{y}'_q), \\ k=3, &\quad p^3_{n:\{i,j,h\}} = \frac{1}{N}\hat{y}'_i\hat{y}'_j\hat{y}'_h\Pi_{q\neq i,j,h}\,(1-\hat{y}'_q), \\ &\quad \ldots, \end{aligned} \tag{4}$$

where N is a normalization factor as $N = \Sigma_k^K\,\Sigma_n^{N_k}\,p_n^k$.

This formulation introduces a heavy computational load to calculate probabilities for all possible labels in every frame. To resolve this issue, the number of concurrent events k is considered up to 3 and the probabilities for multi-event labels are calculated by a dynamic programming technique (5). In addition, the computing time can be dramatically reduced by parallel processing with GPUs.

$$\begin{aligned} k=0, &\quad P^0 = log(p^0_{n:\{\}}), \\ k=1, &\quad P^1_i = P^0 + log(\hat{y}'_i) - log(1-\hat{y}'_i), \\ k=2, &\quad P^2_{\{i,j\}} = P^1_i + P^1_j - P^0, \\ k=3, &\quad P^3_{\{i,j,h\}} = P^2_{i,j} + P^1_h - P^0. \end{aligned} \tag{5}$$

3.4. Reliability

The outputs by the averaging network are obviously unreliable at the beginning of training. Even at later stages of training, pseudo labels are still generating expectation values based on predictions. Thus, a weight γ is designed to estimate the reliability of the pseudo label (6).

$$\gamma = min(\frac{3.0\;e^{-5(1-m/M)^2}}{BCE(\tilde{y},y)},\;5.0). \tag{6}$$

where m is an index for training step and M is the maximum ramp up value. The reliability consists of exponential ramp up value and binary cross entropy between the pseudo and true label. At the beginning of training, the contribution of expectation loss (2) remains small due to the ramp up, and is increased as the training progresses. Additionally, the reliability is clipped to prevent the expectation value dominating the loss function over labeled data. Note that reliability is calculated with labeled data only. The implementation of proposed method can be found in *http://github.com/JHU-LCAP/Self-training*.

4. EXPERIMENTS

4.1. Database

A database for the SED task in DCASE2020 challenge is used [19]. Among the subsets in the database, *Synthetic:training*, *Real:weakly labeled*, and *Real:unlabeled* are used for network training via semi-supervised learning and *Real: validation* is used for evaluation. *Synthetic:training* has strong label that includes both sound event class and its timestamps when it happens and terminates and *Real:weakly labeled* has a truth for event class only. On the other hand, both event class and its timestamps are missed in *Real:unlabeled*.

4.2. Experimental setting

The challenge baseline, which adopts a version of Mean-Teacher model for semi-supervised learning, is considered as a counterpart of the proposed method in assessment [10]. Note that all settings such the CRNN structure and training parameters (optimizer, batch size, and learning rate) are set exactly the same way in both methods except loss function. For the proposed method, additional evaluations are performed depending on the parameter K (the maximum number of concurrent events) to investigate the effect of approximation (Eq. (5)).

In both methods, one batch for training consists of 6 weakly labeled data, 12 unlabeled data, and 6 synthetic data by random sampling on each data pool. To calculate loss (Eq. (2)), synthetic data are only applied for BCE while weakly labeled and unlabeled data are applied for MSE in every frame. Also, the synthetic data is used to calculate reliability of the pseudo label (6). Due to the randomness in batch, evaluation of each method is performed 5-times and the results are summarized by mean and standard deviation for comparison.

Each method is evaluated by event based class averaging F-score with the protocol for SED task in DCASE2020 challenge. Briefly, an event interval is detected by applying a fixed threshold to event posterior (i.e. network output) over the time. The interval would be considered as true positive if its time boundaries are close enough to true timestamps (less than 200ms) in the same event otherwise it is flagged as false positive. Then, precision, recall, and F-score are calculated for each class. More details are described in [20].

4.3. Results

In the baseline, class averaging F-score is 34.04 $\pm$ 1.48% marked as the black line with grey area for its variation (Fig. 2). The results for proposed method are represented as the red line depending on a constraint of maximum number of concurrent events. With the strict assumption of non-overlapping sound event detection ($K = 1$), the proposed method flags a single sound for overlaid sound events. Since sound event usually overlaid with other sounds in practical

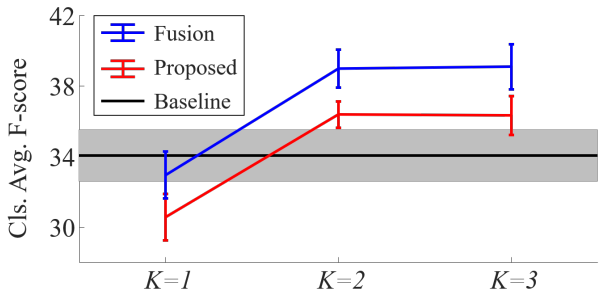

Fig. 2. Class averaging F-scores for baseline method, proposed method, and fusion of both methods. Black line with gray area represents mean and variation for baseline method. The proposed method and fusion result are represented as red and blue line, respectively.

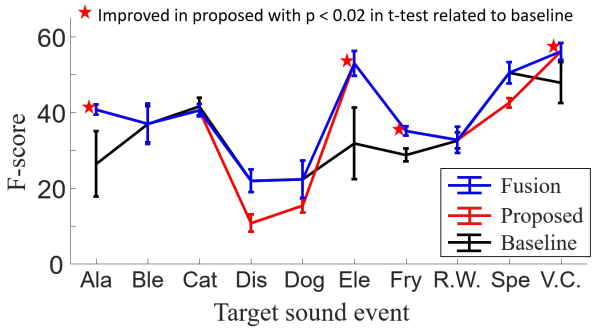

Fig. 3. Class-wise f-score in baseline method (black), proposed method (red), and fusion of both methods (blue)

environment, the performance is below the baseline (in t-test, $z = 3.95, p = 0.0042$). On the other hand, the performances of proposed method are notably improved relative to the baseline when concurrent event scenarios ($K = 2$ or 3) are considered in pseudo label estimation. The performance in F-score is saturated at $K = 2$. The case, which three or more sound events among the targets happen at a time, is unusual in real environments. The proposed method shows improvement relative to the baseline by about 2.3% in class averaging f-score (in t-test for $K = 2 : z = 3.14, p = 0.0137$ and $K = 3 : z = 2.77, p = 0.0243$).

Fig. 3 shows the breakdown of F-scores for all 10 sound events to allow a class-wise comparison. As noted in the figure, the proposed method yields improvement in four classes: *Alarm/bell/ringing*, *Electrical shaver/toothbrush*, *Frying*, and *Vacuum cleaner*. On the other hand, the F-scores are lower in *Dishes*, *Dog*, and *Speech*, as we will expand on in the following section. Additionally, as represented by error bar in each class, the proposed method shows consistent F-scores over the repetitions compared to baseline results. The maximum variance is founded in *Electrical shaver/tooth brush* as 9.50% for baseline and *Blender* as 5.33% for proposed method.

Given the limited improvement on some classes using this self-training method, we suggest a fusion system by combining both the self-trained and original baseline based on maximum performance on a per-class basis. The blue line in Fig 2 and 3 depicts the performance of this fusion system and shows that the fusion approach results in the best performance across both systems. In case of ($K = 2$) fusion, the class averaging f-score has been reached to 38.97 ± 1.08 %.

5. DISCUSSION AND CONCLUSION

In the recent DCASE challenge, many systems for SED have been introduced and show a much higher performance than this proposed method. They used alternative CRNN structure designed by themselves and their networks were trained in the same manner with the baseline. On the other hand, this work focuses on training strategy for usage of both weakly labeled and unlabeled data in network training for the same structure with the baseline. If alternative structure for CRNN is used in this work, the performance might be further improved because pseudo label would be estimated with more accurate class posterior. However, the baseline CRNN was used in this work for a fair comparison. The evaluation with alternative CRNN will be performed in future.

In class-wise comparison, the proposed method shows an issue in three classes, *Dishes*, *Dog*, and *Speech*. In the baseline, augmented input by transformations such shifting, rotating, and adding noise from original input was used in training so that the baseline network projects any data points within a manifold into similar predictions. On the other hand, the proposed network performs a point-wise projection since it has no way for manifold projection like the baseline. It seems that these classes, especially in *Dishes* and *Dog*, show this limitation of the proposed method. Additionally, the best f-score is represented on *Speech* class for baseline. It might be related to unbalancing issue in training data. The *Speech* has a large portion of the training set: unlabeled as well as weakly labeled data compared to other classes (about 40 % in *Real:validation* set for frequency). In case of supervised learning, this issue can be resolved by dynamic sampling on training dataset because all labels are available [21]. However, it is difficult to apply the method to semi-supervised learning because the frequencies of each class are unknown in unlabeled data. This issue remains an open question that will be explored in future.

This paper proposes self-training method with pseudo label and its reliability for supervised learning using unlabeled and/or partially labeled data in combination with fully labeled data. The self-training approach has shown effectiveness in experiments for concurrent sound event detection. In future, two issues in this approach will be considered: one is performance degradation in several classes and the other is the class unbalancing problem in training data.

6. REFERENCES

[1] Kashif Ahmad and Nicola Conci, "How deep features have improved event recognition in multimedia: A survey," *ACM Trans. on Multimedia Compu., Comm. and App.*, vol. 15, no. 2, 2019.

[2] Yizhar Lavner, Rami Cohen, Dima Ruinskiy, and Hans Ijzerman, "Baby cry detection in domestic environment using deep learning," *IEEE International Conference on the Science of Electrical Engineering,*, 2017.

[3] Sangwook Park, Woohyun Choi, David K. Han, and Hanseok Ko, "Acoustic event filterbank for enabling robust event recognition by cleaning robot," *IEEE Trans. Cons. Electron.*, vol. 61, no. 2, pp. 189–196, 2015.

[4] Emre Cakir, Giambattista Parascandolo, Toni Heittola, Heikki Huttunen, and Tuomas Virtanen, "Convolutional Recurrent Neural Networks for Polyphonic Sound Event Detection," *IEEE/ACM Trans. on Audio, Speech, and Lang. Proc.*, vol. 25, no. 6, pp. 1291–1303, 6 2017.

[5] Fabio Vesperini, Leonardo Gabrielli, Emanuele Principi, and Stefano Squartini, "Polyphonic Sound Event Detection by Using Capsule Neural Networks," *IEEE J. Selected Topics in Signal Proc.*, vol. 13, no. 2, pp. 310–322, 2019.

[6] Sandeep Kothinti, Keisuke Imoto, Debmalya Chakrabarty, Gregory Sell, Shinji Watanabe, and Mounya Elhilali, "Joint Acoustic and Class Inference for Weakly Supervised Sound Event Detection," in *IEEE International Conference on Acoustics, Speech and Signal Processing*. 5 2019, pp. 36–40, IEEE.

[7] Qiuqiang Kong, Yong Xu, Wenwu Wang, and Mark D. Plumbley, "Sound Event Detection of Weakly Labelled Data with CNN-Transformer and Automatic Threshold Optimization," *IEEE/ACM Trans. Audio Speech and Lang. Proc.*, vol. 28, pp. 2450–2460, 2020.

[8] Bowen Shi, Ming Sun, Chieh-chi Kao, Viktor Rozgic, Spyros Matsoukas, and Chao Wang, "Semi-supervised Acoustic Event Detection Based on Tri-training," in *IEEE International Conference on Acoustics, Speech and Signal Processing*. 5 2019, pp. 750–754, IEEE.

[9] Liwei Lin, Xiangdong Wang, Hong Liu, and Yueliang Qian, "Guided Learning for Weakly-Labeled Semi-Supervised Sound Event Detection," in *IEEE International Conference on Acoustics, Speech and Signal Processing*. 5 2020, pp. 626–630, IEEE.

[10] Nicolas Turpault and Romain Serizel, "Training Sound Event Detection On A Heterogeneous Dataset," in *DCASE workshop*, 2020.

[11] Koichi Miyazaki, Tatsuya Komatsu, Tomoki Hayashi, Shinji Watanabe, Tomoki Toda, and Kazuya Takeda, "Convolution-Augmented Transformer for Semi-Supervised Sound Event Detection," Tech. Rep., 2020.

[12] Tianchu Yao, Chuang Shi, and Huiyong Li, "Sound Event Detection In Domestic Environments Using Dense Recurrent Neural Network," Tech. Rep., 2020.

[13] Yuzhuo Liu, Chengxin Chen, Jianzhong Kuang, and Pengyuan Zhang, "Semi-supervised Sound Event Detection Based on Mean Teacher with Power Pooling and Data Augmentation," Tech. Rep. Mil, 2020.

[14] Chih-yuan Koh, You-siang Chen, Shang-en Li, Yi-wen Liu, Jen-tzung Chien, Mingsian R Bai, and National Tsing, "Sound Event Detection By Consistency Training and Pseudo-Labeling With Feature-Pyramid Convolutional Recurrent Neural Networks," Tech. Rep., 2020.

[15] Jesper E. van Engelen and Holger H. Hoos, "A survey on semi-supervised learning," *Machine Learning*, vol. 109, no. 2, pp. 373–440, 2020.

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

[17] Dong-Hyun Lee, "Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks," in *International Conference on Machine Learning Workshop*, 2013, pp. 1–6.

[18] Avital Oliver, Augustus Odena, Colin Raffel, Ekin D. Cubuk, and Ian J. Goodfellow, "Realistic evaluation of semi-supervised learning algorithms," in *Advances in Neural Information Processing Systems*, 2018, pp. 3235–3246.

[19] Nicolas Turpault, Romain Serizel, Justin Salamon, and Ankit Parag Shah, "Sound Event Detection in Domestic Environments with Weakly Labeled Data and Soundscape Synthesis," in *DCASE Workshop*. 2019, pp. 253–257, New York University.

[20] Annamaria Mesaros, Toni Heittola, and Tuomas Virtanen, "Metrics for polyphonic sound event detection," *Applied Sciences (Switzerland)*, vol. 6, no. 6, 2016.

[21] Samira Pouyanfar, Yudong Tao, Anup Mohan, Haiman Tian, Ahmed S. Kaseb, Kent Gauen, Ryan Dailey, Sarah Aghajanzadeh, Yung Hsiang Lu, Shu Ching Chen, and Mei Ling Shyu, "Dynamic Sampling in Convolutional Neural Networks for Imbalanced Data Classification," in *IEEE Conference on Multimedia Inform. Proc. and Retrieval*, 2018, pp. 112–117.