Power Pooling Operators and Confidence Learning for Semi-Supervised Sound Event Detection

Yuzhuo Liu¹,², Hangting Chen¹,², Pengyuan Zhang¹,²

¹Key Laboratory of Speech Acoustics & Content Understanding, Institute of Acoustics, CAS, China
²University of Chinese Academy of Sciences, Beijing, China

liuyuzhuo@hccl.ioa.ac.cn, chenhangting@hccl.ioa.ac.cn, zhangpengyuan@hccl.ioa.ac.cn

Abstract

In recent years, the involvement of synthetic strongly labeled data, weakly labeled data and unlabeled data has drawn much research attention in semi-supervised sound event detection (SSED). Self-training models carry out predictions without strong annotations and then take predictions with high probabilities as pseudo-labels for retraining. Such models have shown its effectiveness in SSED. However, probabilities are poorly calibrated confidence estimates, and samples with low probabilities are ignored. Hence, we introduce a method of learning confidence deliberately and retaining all data distinctly by applying confidence as weights. Additionally, linear pooling has been considered as a state-of-the-art aggregation function for SSED with weak labeling. In this paper, we propose a power pooling function whose coefficient can be automatically learned and attention-based pooling are boosted and many false positive results are produced for positive detections. The drawback of max pooling is that only one frame receives the predictions of target sound events in continuous audio recordings. As modern neural network classifiers are designed to produce output probabilities prone to extreme values, incorrect predictions can be generated with high probabilities. However, ratio 1/2 may not be the optimal value for various models and data. Hence, we propose an adaptive pooling function termed as power pooling, which can automatically learn the proper ratio.

SSED is to complete SED task with data partially annotated. Most SSED research is based on two classic semi-supervised learning methods: mean teacher and self-training. The mean teacher averages model outputs of all samples and then uses them as pseudo-labels for retraining, while self-training employs both frame-level and clip-level consistency loss. Self-training [20], a simple but effective bootstrapping semi-supervised method, cycles retraining the model with part of its own predictions as pseudo-labels. Self-training methods adopted in SSED [17,22,29] retrained only once and employed a small part unlabeled data with high probabilities. These approaches filtered unlabeled data by posterior to ensure the quality of the pseudo-labels, but caused three problems. First, Probability is not a calibrated indicator for evaluating the correctness of the model predictions. As modern neural network classifiers are designed to produce output probabilities prone to extreme values, incorrect predictions can be generated with high probabilities [24]. Second, Simplified self-training methods [21,22,23] lose considerable information. Third, These methods ignore true negative predictions. Nevertheless, data for SSED is extremely imbalanced. Massive correct negative predictions are beneficial to the retraining process.

Aiming at solving the above problems, this paper proposes a confidence-based semi-supervised sound event detection (C-SSED) framework (Figure 1). C-SSED includes two stages of prediction and self-training with weak labeling and semi-supervised sound event detection (SSED) with the above data.

1. Introduction

Sound event detection (SED) is a task for identifying the categories and timestamps of target sound events in continuous audio recordings. As one of the core technologies in non-verbal sound perception and understanding, SED is widely deployed in various applications, such as noise monitoring for smart cities [1], nocturnally migrating bird detection [2], surveillance systems [3] and multimedia indexing [4]. Time consuming to add high-quality labels to SED data manually. In comparison, synthetic strongly labeled data, weakly labeled data with clip-level categories only and unlabeled data are widely available. Therefore, research and competitions [5,6] are turned to multiple instance learning (MIL) for SED with weak labeling and semi-supervised sound event detection (SSED) with the above data.

MIL [7] in SED permits models to learn frame-level classification from clip-level class labels. At each frame, a SED model predicts the probability of each sound event class being active. Then, a pooling function aggregates the frame-level predictions into a clip-level prediction for each sound event class. The derivative of the pooling function determines the direction of the frame-level gradient during backpropagation. The drawback of max pooling is that only one frame receives a non-zero gradient. The fact may cause many frame-level false negatives. The gradients of mean pooling and exponential softmax pooling [10] are always positive. Consequently, all frame-level predictions are boosted and many false positive results are produced for positive clips. The auto-pooling introduces a trainable parameter and can interpolate between the above three functions. A group of frame-wise trainable pooling weights are proposed in attention-based pooling [13]. However, the gradients of auto-pooling and attention-based pooling are positive too. Linear pooling function [13] has been confirmed to work best for frame-level classification [14], since the gradient is positive where the frame-level probabilities are larger than half of the clip-level probability. However, ratio 1/2 may not be the optimal value for various models and data. Hence, we propose an adaptive pooling function termed as power pooling, which can automatically learn the proper ratio.

1.1. Baseline

In recent years, the involvement of synthetic strongly labeled data, weakly labeled data and unlabeled data has drawn much research attention in semi-supervised sound event detection (SSED). Self-training models carry out predictions without strong annotations and then take predictions with high probabilities as pseudo-labels for retraining. Such models have shown its effectiveness in SSED. However, probabilities are poorly calibrated confidence estimates, and samples with low probabilities are ignored. Hence, we introduce a method of learning confidence deliberately and retaining all data distinctly by applying confidence as weights. Additionally, linear pooling has been considered as a state-of-the-art aggregation function for SSED with weak labeling. In this paper, we propose a power pooling function whose coefficient can be automatically learned and attention-based pooling are boosted and many false positive results are produced for positive detections. The drawback of max pooling is that only one frame receives the predictions of target sound events in continuous audio recordings. As modern neural network classifiers are designed to produce output probabilities prone to extreme values, incorrect predictions can be generated with high probabilities. However, ratio 1/2 may not be the optimal value for various models and data. Hence, we propose an adaptive pooling function termed as power pooling, which can automatically learn the proper ratio.

SSED is to complete SED task with data partially annotated. Most SSED research is based on two classic semi-supervised learning methods: mean teacher and self-training. The mean teacher averages model outputs of all samples and then uses them as pseudo-labels for retraining, while self-training employs both frame-level and clip-level consistency loss [16,17,18,19]. Self-training [20], a simple but effective bootstrapping semi-supervised method, cycles retraining the model with part of its own predictions as pseudo-labels. Self-training methods adopted in SSED [17,22,29] retrained only once and employed a small part unlabeled data with high probabilities. These approaches filtered unlabeled data by posterior to ensure the quality of the pseudo-labels, but caused three problems. First, Probability is not a calibrated indicator for evaluating the correctness of the model predictions. As modern neural network classifiers are designed to produce output probabilities prone to extreme values, incorrect predictions can be generated with high probabilities [24]. Second, Simplified self-training methods [21,22,23] lose considerable information. Third, These methods ignore true negative predictions. Nevertheless, data for SSED is extremely imbalanced. Massive correct negative predictions are beneficial to the retraining process.

Aiming at solving the above problems, this paper proposes a confidence-based semi-supervised sound event detection (C-SSED) framework (Figure 1). C-SSED includes two stages of prediction and self-training with weak labeling and semi-supervised sound event detection (SSED) with the above data.

In this paper, we propose a power pooling function whose coefficient can be automatically learned and attention-based pooling are boosted and many false positive results are produced for positive detections. The drawback of max pooling is that only one frame receives the predictions of target sound events in continuous audio recordings. As modern neural network classifiers are designed to produce output probabilities prone to extreme values, incorrect predictions can be generated with high probabilities. However, ratio 1/2 may not be the optimal value for various models and data. Hence, we propose an adaptive pooling function termed as power pooling, which can automatically learn the proper ratio.

SSED is to complete SED task with data partially annotated. Most SSED research is based on two classic semi-supervised learning methods: mean teacher and self-training. Mean teacher [15] averages model outputs of all samples and then uses them as pseudo-labels for retraining, while self-training employs both frame-level and clip-level consistency loss [16,17,18,19]. Self-training [20], a simple but effective bootstrapping semi-supervised method, cycles retraining the model with part of its own predictions as pseudo-labels. Self-training methods adopted in SSED [17,22,29] retrained only once and employed a small part unlabeled data with high probabilities. These approaches filtered unlabeled data by posterior to ensure the quality of the pseudo-labels, but caused three problems. First, Probability is not a calibrated indicator for evaluating the correctness of the model predictions. As modern neural network classifiers are designed to produce output probabilities prone to extreme values, incorrect predictions can be generated with high probabilities. However, ratio 1/2 may not be the optimal value for various models and data. Hence, we propose an adaptive pooling function termed as power pooling, which can automatically learn the proper ratio.

2. Proposed System

2.1. Baseline

Mean teacher is a consistency regularization method that evaluates unlabeled data with two different noises, and then apply a consistency...
The clip-level and frame-level gradient directions are illustrated in Table 1. As a result, when $t = 1$, larger $y_f$ is driven to 0, benefiting the timestamps detection. When $t = 0$, $y_f$ is pushed towards $y_c/2$. Considering that $y_c$ is a weighted average of the $y_f$, all the $y_f$ will converge to 0 as desired after enough iterations.

2.2.2. Power Pooling

For positive recordings ($t = 1$), linear pooling function leads larger $y_f$ to be boosted. Limited by the form of linear function, the threshold of larger $y_f$ is defined as $y_c/2$. However, the threshold is supposed to be adjusted dynamically according to the value of $y_c$ and the number of positive frame-level samples. This issue can be addressed by applying a power pooling function:

$$y_c = \frac{1}{n} \sum_i y_f(i) \times y_f^\theta(i),$$  

and the gradient can be written as:

$$\frac{\partial y_c}{\partial y_f(i)} = \frac{(n + 1) \times y_f^\theta(i) - n \times y_c^{n-1} \times y_c}{\sum_j y_f^\theta(j)}.$$  

parameter $n \neq -1$ and threshold $\theta = n/(n + 1)$ ($\theta \neq 1$). Treating $n$ as a free parameter to be learned along-side the model parameters allows eq. (3) to automatically adapt to and interpolate between separate pooling functions. For instance, when $n = 0$, eq. (3) reduces to mean pooling. When $n \rightarrow \infty$, eq. (3) approaches the max aggregation.

We discuss the diversification of $y_c$ and $y_f$ more specifically in three cases according to the value of $n$. For $n \in (0, +\infty)$, $\theta \in (0, 1)$, weight $y_f$ increases where $y_f$ increases, leading to $y_c$ being generated under the standard multiple instance (SMI) assumption: the bag label is positive if and only if the bag contains at least one positive instance. During backpropagation, parameters for producing frame-level outputs upgrade as linear pooling pattern (Table 1) with dynamic threshold $\theta$ instead of value $1/2$. For $n \in (-1, 0)$, $\theta \in (-\infty, 0)$, ($\theta \times y_c$) $\leq 0$. Therefore, for negative clips ($t = 0$), all $y_f$ are pushed towards 0, as
desired. Nevertheless, for positive clips \((t = 1)\), all \(y_f\) increase towards 1 as the mean pooling mode. For \(n \in (-\infty, -1)\), \(\theta \in (1, +\infty)\), weight \(y_f^n\) increases where \(y_f^n\) decreases. This behaviour violates the SMI assumption. A positive \(y_c\) can only be produced when the vast majority \(y_f\) is positive.

2.3. C-SSED

2.3.1. Stage One: Multi-task SSED Model (MT-SSED)

To get reliable confidence, we added a branch to train confidence in MT-SSED. When solving the issue of simultaneously generating SED predictions and their corresponding confidence without the confidence label, we draw on the successful experience in the field of out-of-distribution detection [23]. The motivation is equivalent to a special test that permits giving hints. Candidates are allowed to ask for hints according to their confidence of the questions. Furthermore, a certain penalty is carried out in order to prevent candidates from tending to ask for hints for all questions. For obtaining the highest score, candidates must improve their ability to answer questions and self-assess at the same time.

MT-SSED is constructed based on the baseline model. There are four outputs in the baseline model, the frame-level output \(y_{fs}\) and clip-level output \(y_{cs}\) of the teacher model, the frame-level output \(y_{fs}\) and clip-level output \(y_{cs}\) of the student model. We choose power pooling as the aggregation function in MT-SSED. To make the model self-assess, we add a confidence branch in parallel with the original class prediction branch. The confidence branch, which shares the same structure with the frame-level classification branch, applies a fully-connected layer followed by sigmoid. The confidence branch generates corresponding confidence values \(c\) for the classification results of each sound event at every frame. Output \(c\) takes values between 0 and 1. If the model is confident about the classification, output \(c\) will be closer to 1. Conversely, if the model is uncertain about the correctness of classification predictions, the value of \(c\) will be closer to 0.

A crucial issue of confidence is how to achieve the training of two tasks with just the confidence labels. Following the main idea of giving hints, we construct a new frame-level output of student model \(y_{fs}^{\prime}\) with the label \(t_f\) and two outputs \(y_{fs}\) and \(c\):

\[
y_{fs}^{\prime} = (1 - c) \times t_f + c \times y_{fs}.
\]

The outputs of the student model \(y_{fs}^{\prime}\) and \(y_{cs}\) are in comparison with strong labels \(t_f\) and \(t_c\) utilizing the binary cross entropy (BCE) loss. The classification loss can be written as

\[
L_{\text{class}} = L_{\text{class}}^{\text{cls}} + L_{\text{class}}^{\text{fs}}.
\]

(6)

The output \(y_{fs}\) and \(y_{cs}\) are compared with the outputs \(y_{fs}\) and \(y_{cs}\) by applying the mean square error (MSE) loss. The consistency loss is

\[
L_{\text{cons}} = L_{\text{MSE}}(y_{fs}, t_f) + L_{\text{MSE}}(y_{cs}, t_c).
\]

(7)

Training with \(L_{\text{class}}\) and \(L_{\text{cons}}\) loss functions, the network will be lazy to learn the differences between classes. Instead, the model tends to make \(c\) approach 0 and receives ground truth for every sample. Thus, a log penalty is added to the loss function. The confidence loss can be interpreted as a BCE loss:

\[
L_c = -\log(c).
\]

(8)

The loss function of the multi-task system is

\[
L = L_{\text{class}} + \mu \times L_{\text{cons}} + \lambda \times L_c.
\]

(9)

parameter \(\mu\) increases with epochs and \(\lambda\) is a hyperparameter. When \(\lambda\) is too small, MT-SSED model tends to ask for hints and performs poorly in classification. When \(\lambda\) is too large, the confidence \(c \to 1\) and lose the distinction. To ensure the effects of both classification and confidence estimation, we first optimize the mean teacher model and classification branch without \(L_c\). Then, the trained parameters are fixed, and \(L\) is deployed to train the confidence branch separately for 5 epochs.

The outputs of the student model \(y_{fs}^{\prime}\) and \(y_{cs}\) of the student model. We choose power pooling as the aggregation function in MT-SSED. To make the model self-assess, we add a confidence branch in parallel with the original class prediction branch. The confidence branch, which shares the same structure with the frame-level classification branch, applies a fully-connected layer followed by sigmoid. The confidence branch generates corresponding confidence values \(c\) for the classification results of each sound event at every frame. Output \(c\) takes values between 0 and 1. If the model is confident about the classification, output \(c\) will be closer to 1. Conversely, if the model is uncertain about the correctness of classification predictions, the value of \(c\) will be closer to 0.

A crucial issue of confidence is how to achieve the training of two tasks with just the confidence labels. Following the main idea of giving hints, we construct a new frame-level output of student model \(y_{fs}^{\prime}\) with the label \(t_f\) and two outputs \(y_{fs}\) and \(c\):

\[
y_{fs}^{\prime} = (1 - c) \times t_f + c \times y_{fs}.
\]

The outputs of the student model \(y_{fs}^{\prime}\) and \(y_{cs}\) are in comparison with strong labels \(t_f\) and \(t_c\) utilizing the binary cross entropy (BCE) loss. The classification loss can be written as

\[
L_{\text{class}} = L_{\text{class}}^{\text{cls}} + L_{\text{class}}^{\text{fs}}.
\]

(6)

The output \(y_{fs}\) and \(y_{cs}\) are compared with the outputs \(y_{fs}\) and \(y_{cs}\) by applying the mean square error (MSE) loss. The consistency loss is

\[
L_{\text{cons}} = L_{\text{MSE}}(y_{fs}, t_f) + L_{\text{MSE}}(y_{cs}, t_c).
\]

(7)

Training with \(L_{\text{class}}\) and \(L_{\text{cons}}\) loss functions, the network will be lazy to learn the differences between classes. Instead, the model tends to make \(c\) approach 0 and receives ground truth for every sample. Thus, a log penalty is added to the loss function. The confidence loss can be interpreted as a BCE loss:

\[
L_c = -\log(c).
\]

(8)

The loss function of the multi-task system is

\[
L = L_{\text{class}} + \mu \times L_{\text{cons}} + \lambda \times L_c.
\]

(9)

parameter \(\mu\) increases with epochs and \(\lambda\) is a hyperparameter. When \(\lambda\) is too small, MT-SSED model tends to ask for hints and performs poorly in classification. When \(\lambda\) is too large, the confidence \(c \to 1\) and lose the distinction. To ensure the effects of both classification and confidence estimation, we first optimize the mean teacher model and classification branch without \(L_c\). Then, the trained parameters are fixed, and \(L\) is deployed to train the confidence branch separately for 5 epochs.

3. Experiments and Discussion

3.1. Dataset and Metrics

We carried our experiments on the DCASE2019 task4 dataset [5]. The dataset can be divided into four subsets, including training sets (synthetic strongly labeled: 2,045 clips, weakly labeled: 1,578 clips, unlabeled: 14,412 clips) and validation set (1,168 clips). In addition, the evaluation set of DCASE2018 (800 clips) [5] was tested in the C-SSED experiment. The duration of each audio clip is 10 seconds, and multiple audio events may occur at the same time. The sampling rate is 44100 Hz.

Experiments were evaluated mainly with event-based macro-average error rate (ER) with a 200 ms collar on onsets and a 200 ms/20% of the events length collar on offsets. Both event-based and segment-based (1 s) ER and \(F_1\) were applied in the pooling function experiments. A smaller ER is better and a larger \(F_1\) is better. The specific evaluation details can be found in [24].

3.2. Results and Analysis

3.2.1. Pooling Function

We observe that the initialization of the parameter \(n\) influenced the training process. If the value of \(n\) was too large (for instance, the initial value was 10), only very few frame-level samples could be updated towards positive. The change for clip-level predictions was slow as they were generated from frame-level predictions. The training was trapped

![Figure 2: The parameter \(n\) of power pooling in different epochs with different initial values (init).](image-url)
Table 2: Detailed performance of four pooling functions. Initial value of power pooling is 1.2.

| Pooling Function | Event-Based | Segment-Based |
|------------------|-------------|---------------|
|                  | ER (%)/ Pre (%) | ER (%)/ Pre (%) |
| Attention        | 1.26/32.04       | 0.72/65.75       |
| Auto             | 1.16/26.15       | 0.67/63.14       |
| Linear           | 1.05/34.07       | 0.68/62.59       |
| Power            | 1.87/37.04       | 0.64/63.57       |

Figure 3: Accuracy of frame-level predictions changes with confidence estimates and probabilities. (a) The accuracy of all labels was obtained by the DCASE2019 validation set as tested by the MT-SSED model.

Table 3: Comparison of models in terms of ER (in %). Retraining with $\alpha = 0$ is equal to retrain without confidence.

| Model               | Evaluation 2018 | Validation 2019 |
|---------------------|-----------------|-----------------|
|                     | ER/DEL/INS      | ER/DEL/INS      |
| MT18                | 0.65/0.76/0.87  | 1.15/0.76/0.80  |
| Baseline            | 1.34/0.72/0.62  | 1.26/0.70/0.56  |
| MT-SSED             | 1.11/0.69/0.44  | 1.07/0.67/0.40  |
| Retrain $\alpha$    |                 |                 |
| Prob0.9             | 0.72/0.30/0.34  | 0.69/0.27/0.22  |
| Prob0.5             | 0.76/0.47/0.69  | 0.66/0.42/0.35  |
| C-SSED              | 0.70/0.39/0.39  | 0.68/0.35/0.35  |
| C-SSED $\alpha = 0.01$ | 0.70/0.36/0.36  | 0.69/0.32/0.32  |
| C-SSED $\alpha = 0.01$ | 0.86/0.14/0.08  | 0.85/0.13/0.13  |

Figure 3: Accuracy of frame-level predictions changes with confidence estimates and probabilities. (a) The accuracy of all labels was obtained by the DCASE2019 validation set as tested by the MT-SSED model.
C-SSED framework can be extended to other semi-supervised tasks. In addition, this paper introduces a confidence training method to SSED, but confidence can also be applied in other scenes, such as optimizing focal loss.

5. References

[1] J. P. Bello, C. Silva, O. Nov, R. L. Dubois, A. Arora, J. Salamon, C. Mydlarz, and H. Doraiswamy, “Sonic: A system for monitoring, analysis and mitigation of urban noise pollution,” *Communications of the ACM*, vol. 62, no. 2, pp. 68–77, 2019.

[2] V. Lostanlen, J. Salamon, A. Farnsworth, S. Kelling, and J. P. Bello, “Birdvox-full-night: A dataset and benchmark for avian flight call detection,” in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 266–270.

[3] M. Crocco, M. Cristani, A. Trucco, and V. Murino, “Audio surveillance: A systematic review,” *Acm Computing Surveys*, vol. 48, no. 4, pp. 52.1–52.46, 2016.

[4] S. Hershey, S. Chaudhuri, D. P. W. Ellis, J. F. Gemmeke, A. Jansen, R. C. Moore, M. Ptakal, D. Platt, R. A. Saurous, B. Seybold, M. Slaney, R. J. Weiss, and K. Wilson, “Cnn architectures for large-scale audio classification,” in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 131–135.

[5] 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.

[6] N. Turpault, R. Serizel, A. Parag Shah, and J. Salamon, “Sound event detection in domestic environments with weakly labeled data and soundscape synthesis,” June 2019, working paper or preprint. [Online]. Available: https://hal.inria.fr/hal-02160855

[7] J. Amores, “Multiple instance classification: Review, taxonomy and comparative study,” *Artificial Intelligence*, vol. 201, no. aug., pp. 81–105.

[8] T. Su, J. Liu, and Y. Yang, “Weakly-supervised audio event detection using event-specific gaussian filters and fully convolutional networks,” in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 791–795.

[9] B. Raj, “A closer look at weak label learning for audio events,” 2018.

[10] S. J. M. B., and L. P., “Dcase 2017 submission: Multiple instance learning for sound event detection,” in *DCASE2017 Challenge*, 2017.

[11] 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.

[12] 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)*, 2018, pp. 121–125.

[13] D. A., V. T., H., and W. J., C., “Deep learning for dcase2017 challenge,” in *DCASE2017 Challenge*, 2017.

[14] 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)*, 2019, pp. 31–35.

[15] A. Tarvainen and H. Valpola, “Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results,” pp. 1195–1204, 2017.

[16] J. Wang, J. Xia, Q. Yang, and Y. Zhang, “Research on semi-supervised sound event detection based on mean teacher models using ml-lobcod-net,” *IEEE Access*, vol. 8, pp. 38032–38044, 2020.

[17] J. Yan, Y. Song, L. Dai, and I. McLoughlin, “Task-aware mean teacher method for large scale weakly labeled semi-supervised sound event detection,” in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 326–330.

[18] L. J. K., “Mean teacher convolution system for dcase 2018 task 4,” in *DCASE2018 Challenge*, 2018.

[19] D.-P. L. and P. C., “Mean teacher with data augmentation for dcase 2019 task 4,” in *DCASE2019 Challenge*, 2019.

[20] D. McClosky, E. Charniak, and M. Johnson, “Effective self-training for parsing,” in *Proceedings of the Main Conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, 2006, pp. 152–159.

[21] Y. L. Liu, J. Yan, and Y. Song, “Ustc-nelslip system for dcase 2018 challenge task 4,” in *DCASE2018 Challenge*, 2018.

[22] B. Elizalde, A. Shah, S. Dalmia, H. L. Min, and I. Lane, “An approach for self-training audio event detectors using web data,” in *European Signal Processing Conference*, 2017.

[23] S. Kothintu, K. Imoto, and D. Charkrabarty, “Joint acoustic and class inference for weakly supervised sound event detection,” in *DCASE2018 Challenge*, 2018.

[24] C. Guo, G. Pleiss, S. Yu, and K. Q. Weinberger, “On calibration of modern neural networks,” pp. 1321–1330, 2017.

[25] D. T. and T. Graham, W., “Learning confidence for out-of-distribution detection in neural network,” 2018.

[26] L. Lin, X. Wang, H. Liu, and Y. Qian, “Guided learning for weakly-labeled semi-supervised sound event detection,” 2020, pp. 626–630.

[27] A. Mesaros, T. Heittola, and T. Virtanen, “Metrics for polyphonic sound event detection,” *Applied Sciences*, vol. 6, no. 6, p. 162, 2016. [Online]. Available: http://www.mdpi.com/2076-3417/6/6/162