THE ACADEMIA SINICA SYSTEM OF SOUND EVENT DETECTION AND SEPARATION FOR DCASE 2020

Technical Report

Hao Yen[1,2], *Pin-Jui Ku*[1,2], *Ming-Chi Yen*[1], *Hung-Shin Lee*[1,2], *Hsin-Min Wang*[1]

[1] Institute of Information Science, Academia Sinica, Taiwan
[2] Dept. Electrical Engineering, National Taiwan University, Taiwan
{b05901090, b05901107}@ntu.edu.tw

ABSTRACT

In this report, we present the system of sound event detection and separation in domestic environments for DCASE 2020. The goal of the task aims to determine which sound events appear in a clip and detailed temporal ranges they occupy. The system is trained by using real data, which are either weakly-labeled or unlabeled, and synthesized data with a strongly annotated label. Our proposed model structure starts with a feature-level front-end based on convolution neural networks (CNN) followed by both embedding-level and instance-level back-end attention modules. To take full advantage of a large amount of unlabeled data, we jointly adopt guided learning mechanism and Mean Teacher, which averages model weights instead of label predictions, to carry out weakly-supervised and semi-supervised learning. A group of adaptive median windows for each sound event is also utilized in post-processing for smoothing frame-level predictions. In the public evaluation set of DCASE 2019, our best system achieves 48.50% event-based F-score, much better than the official baseline performance (38.14%) with a relative improvement of 27.16%. Moreover, in the development set of DCASE 2020, our system is also superior to the baseline while using the student model as the back-end classifier. The F_1-score is relatively improved by 32.91%.

Index Terms— Guided learning, Mean teacher, Semi-supervised learning

1. INTRODUCTION

DCASE 2020 Task 4 is the follow-up of DCASE 2019 Task 4, which aims at developing sound event detection (SED) system that not only predicts the presence of event classes, but also the onset and offset positions of each event. The challenge provides three kinds of data, namely, weakly-labeled data (without timestamps), unlabeled data, and an additional strongly annotated synthetic data (with timestamps). Each 10-second audio clip in the dataset contains one or more (or none) of 10 events, i.e., alarm bell ringing, blender, cat, dishes, dog, electric shaver, frying, running water, speech, and vacuum cleaner. The training set contains much more unlabeled data and fewer labeled data, where the distributions of labels are unbalanced with respect to training clips. Due to the above challenging situation, the primary focus of the task is to efficiently exploit unlabeled training data and to mitigate the influence caused by label preference while training for better test performance.

To deal with the aforementioned problems, previous methods tend to adopt weakly-supervised and semi-supervised learning technique with teacher-student model structure. In DCASE 2019 Task 4, Guided Learning [1] introduced us a brand new weakly-labeled semi-supervised learning algorithm which utilized a more professional teacher model aiming at audio tagging to guide the student model aiming at boundary detection to learn from unlabeled data. The system, however, did not involve learning from timestamps information.

Meanwhile, Mean Teacher [2], a state-of-the semi-supervised learning approach was commonly adopted for this task, i.e., model from second place of DCASE 2019 [3], and the baseline system of DCASE 2020 Task 4. With the help of consistency loss, Mean Teacher can learn from both weakly and strongly annotated data. The input of Mean Teacher, however, as we observe, often require a robust representation.

In this paper, we describe a unified approach to sound event detection that combines the best of the previous approaches: a well-trained feature extractor trained by Guided Learning that generates informative high-level representation, followed by a recurrent neural network (RNN) structure and classifier trained by Mean Teacher to fully exploit strongly annotated information. Trained directly on normalized log Mel-spectrogram and corresponding weakly and strongly annotated labels, our model achieves competitive results on both audio tagging and boundary detection.

2. PROPOSED METHOD

Our system is inspired by the officially provided baseline, which is based on Mean Teacher [2, 3] and Guided Learning convolution system [1, 4, 5] proposed by the winner of DCASE 2019 Task 4.

2.1. Model structure

As shown in Figure 1, the model consists of three parts: a feature extractor, an embedding-level attention pooling module (eATP), and an instance-level attention pooling module (iATP). Each of the pooling modules generates both clip-level and frame-level probabilities. We utilize different training algorithms for embedding-level and instance-level pooling module. The structures of both models of each step are shown in Figures 2 (a) and 3 (b). For step 1, we follow the Guided Learning framework in [4] and use a more professional teacher model to carry out weakly-supervised learning. As for step 2, we apply Mean Teacher [2] method for semi-supervised learning.

2.1.1. Feature extractor

The feature extractor adopts the same structure in [1] as shown in Figure 2 (c), which consists of 1 batch normalization layer, followed

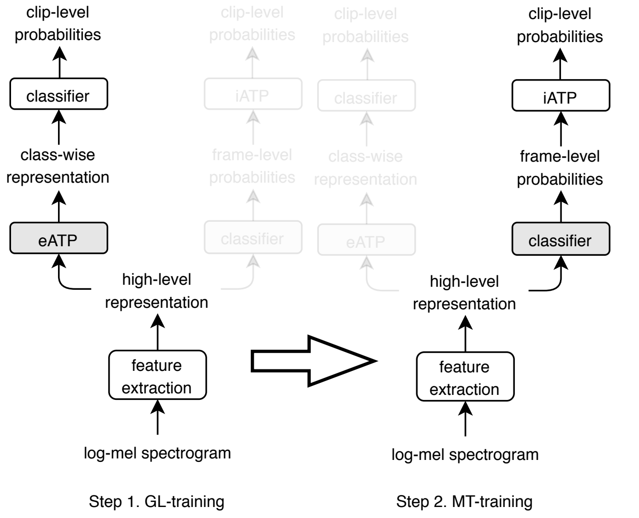


Figure 1: *Flowchart of our system. The feature extractor is pre-trained by the guided learning algorithm in Step 1. In Step 2, Mean Teacher learning is adopted with the pre-trained feature extractor in Step 1.*

by 3 CNN blocks (Figure 2 (d)). Each CNN block includes a single 2-dimensional CNN layer, a batch normalization layer and a ReLU activation layer. A Max-pooling layer comes after each CNN block. The input log Mel-spectrogram is further converted into a high-level representation, which will then be pass on to the pooling module. In our approach, we pre-train our feature extractor using the Guided Learning system.

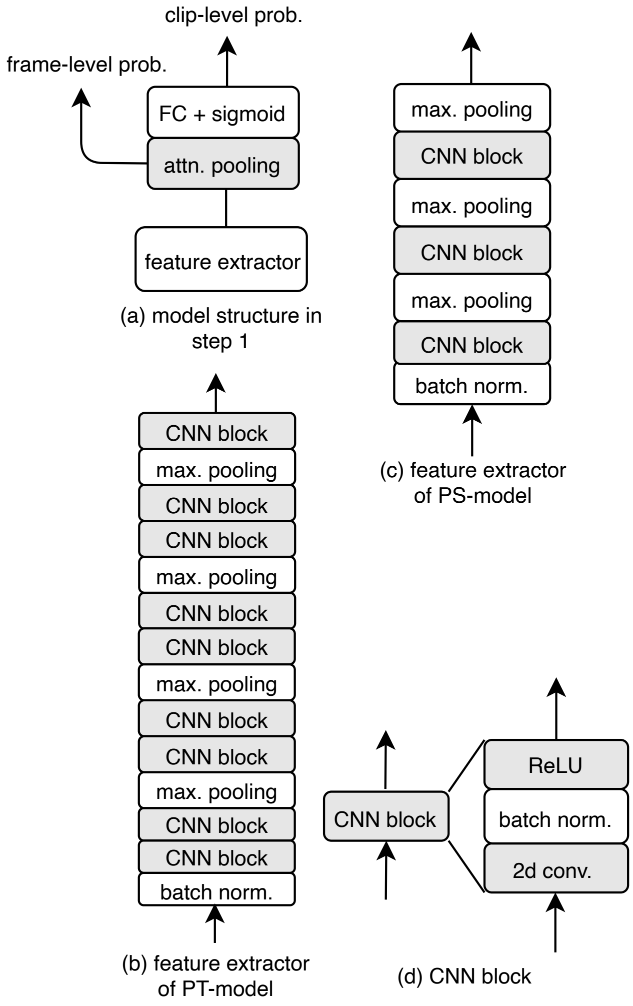
2.1.2. Pooling module

In [5], the influence of the pooling module for SED tasks is highlighted. Though the embedding-level pooling approach is claimed to be superior to instance-level pooling in general, the strongly annotated label is not included in the training process. That is, it relies heavily on the feature extractor to learn frame-level prediction by itself, which results in a better feature extractor.

On the other hand, instance-level pooling can utilize timestamps information, but it often requires a more sophisticated high-level representation. We argue that by training both pooling modules in turn, i.e., using embedding-level pooling to obtain robust high-level representation for instance-level pooling, and adding strong label information through instance-level pooling to further fine-tune the feature extractor, the overall performance on both sides can be improved. As shown in Figure 2 (a) and Figure 3 (b), we adopt the same eATP structure in [5], and an identical RNN structure provided in baseline system for iATP in our model.

2.2. Learning process

In this section, we demonstrate the procedure to train our model. First we introduce two learning techniques, i.e., Guided Learning and Mean Teacher. Our system is based on the two methods. Then in the following section, we show how to apply these techniques to our proposed learning process.



(a) model structure in step 1

(c) feature extractor of PS-model

(b) feature extractor of PT-model

(d) CNN block

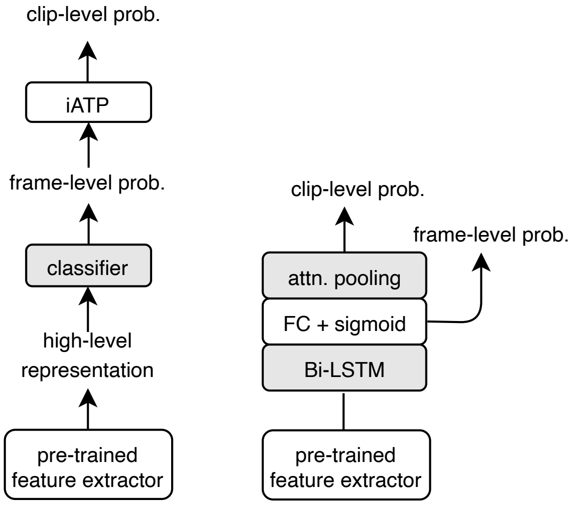
Figure 2: *Key components of the model structure used in our proposed method, where FC denotes the full-connected layer.*

2.2.1. Guided Learning

As proposed in [1], Guided Learning consists of a teacher model (PT-model) and a student model (PS-model), which are shown in Figure 2 (b) and (c). Since the feature extractor of the PT-model has a deeper CNN structure and larger receptive field than the PS-model, we can foresee better audio tagging performance in the PT-model. Nevertheless, the larger receptive field comes with bigger time compression in the PT-model, and therefore reduces model ability to see finer information hidden in time dimension. For this reason, the PS-model is designed with no time compression in order to achieve better performance on frame-level prediction. With the difference in their abilities on clip-level and frame-level predictions, we can exploit unlabeled data by making the PS-model learn from pseudo labels generated by the PT-model.

2.2.2. Mean Teacher

As proposed in [2], the main purpose of Mean Teacher technique is to average model weights from every training step, i.e., exponential moving average, and to produce a more accurate model instead of using the latest model weight directly. We then call the averaging



(a) instance-level approach

(b) model structure in step 2

Figure 3: *Step 2 framework, where pre-trained feature extractor adopts the same structure in Figure 2 (c).*

model as Mean Teacher model (MT-model) and the latest model as Mean Student model (MS-model). In each training step, we calculate two kinds of losses: the classification loss and consistency loss. For the classification loss, we compute binary cross entropy (BCE) loss from those predictions of MS-model that have labels to be corrected. As for the consistency loss, it can be obtained by comparing the clip-level and frame-level predictions of both the MS-model and the MT-model for all labeled and unlabeled data. Both losses then are summed up to update the MS-model, so we can compute new average weights to update the MT-model.

2.2.3. The GL-MT learning algorithm

The aforementioned feature extractor is first pre-trained using Guided Learning (Step 1 in Figure 1). After normalizing our input log Mel-spectrogram separately on real and synthetic training data, we follow the process in 2.2.1 to train our feature extractor. Note that we do not adopt the disentangle feature method proposed in [5]. That is, all categories share the same feature space of the extracted high-level representation.

After the feature extractor has been well-trained enough to extract robust representation, Mean Teacher with iATP is used simultaneously to help the model learn strongly annotated information and to fine-tune the feature extractor (Step 2 in Figure 1). We choose the outputs generated from instance-level attention pooling as our final prediction, i.e., the frame-level probability in 3. The frame-level 0/1 prediction at time t is determined by

$$F(\mathbf{x}, t) = p(x_t) \cdot C(\mathbf{x}), \tag{1}$$

where $p(x_t)$ denotes the frame-level probability in Figure 3 and $C(\mathbf{x})$ represents the 0/1 clip-level prediction. If $F(\mathbf{x}, t)$ is greater than the threshold, then the output will be 1. In our system, we take the mean of two clip-level probabilities from both eATP and iATP to generate our clip-level 0/1 prediction, and set the threshold to 0.5.

Table 1: *Median window sizes with respect to sound events*

Event	Window size (frame)
Alarm bell ringing	18
Blender	52
Cat	29
Dishes	11
Dog	15
Electric shaver	161
Frying	196
Running water	80
Speech	18
Vacuum cleaner	177

2.3. Adaptive median window

The median filter is utilized to post-process the frame level output. Once the frame-level output is generated from our system, it will be smoothed by a group of median windows before being converted into 0/1 prediction with a threshold of 0.5. We will then smooth the prediction one more time with the same group of windows. In [5], the importance of median window is underlined. Instead of using a fixed-sized window for every class as the baseline utilized, we design a group of median windows for each individual event so that each class has its own unique window. The idea is to acquire more accurate boundaries by providing suitable length of filter considering the varying duration of each category in the dataset. To decide the sizes of median windows, we analyze the average duration of each category in the validation set and synthetic set. We follow [1] and calculate window sizes S_{win} with the following equation:

$$S_{win} = D_{avg} \times \beta, \tag{2}$$

where we take $\beta = 1/3$, and D_{avg} denotes the average duration of each class in the dataset. Note that we do make some small adjustments according to the validation results. Table 1 shows the corresponding window size for each event.

3. EXPERIMENT RESULTS

In DCASE 2020 Task 4, the event-based F_1-score (macro-average) is used to evaluate the performance. We take the 1,168 clips from the validation set provided by DCASE 2020 as our development set and the 692 clips from the public evaluation set provided in DCASE 2019 as our evaluation set. We report both event-based and segment-based (1s) results.

The original Guided Learning system is named GL-ps, using PS-model as detector, and our approach of combining Guided Learning and Mean Teacher is named GL-MT-ms, using MS-model as our final detector. In addition, we find that by choosing the exponential moving average (EMA) model from mean teacher to be our final detector, i.e., GL-MT-ema, can yield a 1.03% improvement in the event-based evaluation result.

By training both systems in Figure 1, the overall performance of both eATP and iATP can be improved. As shown in Table 2, the performance of GL-MT-ps improves by 2.88% on the evaluation set compare to GL-ps which only trained on embedding-level. Furthermore, GL-MT-ms also performs better than baseline system by 9.33%. Since we use the same RNN model as the provided baseline system for instance-level pooling, but with a much robust feature extractor, it supports our argument that a better high-level representation can help improve the instance-level pooling system.

Table 2: *Macro F_1-scores with respect to various models.*

	Event-based		**Segment-based**	
Model	**Dev**	**Eval**	**Dev**	**Eval**
Baseline	34.37	38.14	69.07	71.68
GL-ps	45.05	42.53	70.81	72.34
GL-MT-ps	45.42	45.41	69.04	70.93
GL-MT-ms	**45.68**	47.47	**71.96**	74.63
GL-MT-ema	45.65	**48.50**	71.87	**75.83**

As shown in Table 2, our approach GL-MT-ema achieves the best performance on event-based F_1-score with a relative improvement of 30.20% from the baseline on the evaluation set. We submit the results of GL-MT-ps, GL-MT-ms, and GL-MT-ema to the challenge.

4. CONCLUSIONS

In this technical report, we present a system for DCASE 2020 Task 4. We utilize a CNN model with both embedding-level and instance-level attention pooling module to carry out weakly-supervised learning. We also adopt Guided Learning and Mean Teacher method to carry out semi-supervised learning. In addition, the adaptive median window post-processing is able to get more accurate detection boundaries. We evaluate different frame-level predictions generated by both embedding-level and instance-level pooling modules. As a result, we achieve 48.50% on the public evaluation set, improving the performance by 10.36% from the baseline.

5. REFERENCES

[1] L. Lin, X. Wang, H. Liu, and Y. Qian, “Guided learning convolution system for DCASE 2019 Task 4,” in *Proc. DCASE*, 2019.

[2] A. Tarvainen and H. Valpola, “Mean teachers are better role models: weight-averaged consistency targets improve semi-supervised deep learning results,” in *Proc. NIPS*, 2017.

[3] L. Delphin-Poulat and C. Plapous, “Mean teacher with data augmentation for DCASE 2019 Task 4,” in *Proc. DCASE*, 2019.

[4] L. Lin, X. Wang, H. Liu, and Y. Qian, “Guided learning for weakly-labeled semi-supervised sound event detection,” in *Proc. ICASSP*, 2020.

[5] ——, “Specialized decision surface and disentangled feature for weakly-supervised polyphonic sound event detection,” *Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 1466 – 1478, 2020.