SOUND EVENT DETECTION USING METRIC LEARNING AND FOCAL LOSS FOR DCASE 2021 TASK 4

Technical Report

Gangyi Tian[1], *Yuxin Huang*[1,2], *Zhirong Ye*[1,2], *Shuo Ma*[1,2], *Xiangdong Wang*[1*], *Hong Liu*[1], *Yueliang Qian*[1], *Rui Tao*[3], *Long Yan*[3], *Kazushige Ouchi*[3], *Janek Ebbers*[4], *Reinhold Haeb-Umbach*[4]

[1] Beijing Key Laboratory of Mobile Computing and Pervasive Device,
Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China,
gangyitian6@gmail.com, {huangyuxin18g, yezhirong19s, mashuo20g, xdwang, hliu, ylqian}@ict.ac.cn
[2] University of Chinese Academy of Sciences, Beijing, China
[3] Toshiba China R&D Center, Beijing, China,
{taorui, yanlong}@toshiba.com.cn, kazushige.ouchi@toshiba.co.jp
[4]Paderborn University, Germany
{ebbers, haeb}@nt.upb.de
*Corresponding author

ABSTRACT

In this paper, we describe in detail our systems for DCASE 2021 Task 4. The main module in our systems is named MLFL, which uses metric learning and focal loss, adopts the weakly-supervised learning framework with an attention-based embedding-level pooling module and the mean-teacher method for semi-supervised learning. To better utilize the synthetic data, the system adopts metric learning with inter-frame distance contrastive loss to perform domain adaptation. We also employ a sound event detection branch with focal loss to use the strong labels of synthetic data and pseudo strong labels of the weakly-labeled and unlabeled data. The pseudo labels are generated using the forward-backward convolutional recurrent neural network (FBCRNN) model. In addition, we also utilize the tag-conditioned CNN as predicting module, which is trained by the pseudo labels of the weakly-labeled and unlabeled data output by our model and conduct sound event detection. The experimental results prove that our system can achieve competitive results.

Index Terms— metric learning, focal loss, mean-teacher

1. INTRODUCTION

Task 4 of DCASE 2021 [1] is the follow-up to Task 4 of DCASE 2020 [2]. The goal of DCASE 2021 Task 4 is to explore the use of a large amount of unbalanced unlabeled data and synthetic data, as well as a small weakly annotated training set to improve the performance of the sound event detection (SED) system. DCASE 2021 task 4 contains three subtasks: SED with silent separation, SED with acoustic separation, and acoustic separation (using the SED baseline system). We focus on the first subtask, namely, SED without source separation preprocessing. The SED task not only needs to provide event categories, but also needs to provide the onset and offset of the event.

In this paper, we describe in detail the system participating in the first subtask in task 4 of DCASE2021. There are three kinds of data in this challenge, including weakly-labeled real data, unlabeled real data and strongly-labeld synthetic data. The main module in our model mainly uses metric learning and focal loss, and is named MLFL. To utilize the weakly-labeled real data, the module adopts the weakly-supervised learning framework, which uses the embedding-level attention pooling as the pooling method [3]. To utilize the unlabeled real data in our system and to better utilize the real weakly-labeled data, we adopts two semi-supvervised learning methods, including mean teacher and generating strong pesudo-labels. For the mean teacher method, it focuses on using the unlabeled data. For generating strong pesudo-labels, we use the pseudo-labels generated by the FBCRNN model [4] for weakly-labeled real data and unlabeled real data. To use the synthetic data, we use domain adaptation method, which is based on metric learning [5]. We also adopts a sound event detection branch (SEDB) [6] to make full use of all three kinds of strong labels, namely pseudo strong labels of the unlabeled and weakly-labeled real data and strong labels of the synthetic data. In addition, in order to balance the categories of samples, we adopts focal loss [7] as the loss function of the SEDB. The focal loss enables the model to focus more on difficult-to-classify samples during training. By reducing the weight of a large number of easy-to-classify negative samples, it helps the feature encoder to model the feature space more effectively. Finally, we use the tag-conditioned CNN model to conduct final sound event prediction. By using the method mentioned above, the system gains good performance.

2. METHOD

2.1. Overview of MLFL

As shown in Figure 1, The MLFL model adopts three branches. The first branch is the embedding-level attention pooling branch. The second branch is the sound event detection branch which uses the focal loss as the loss function. The third branch is the domain adaptation branch which use metric learning by inter-frame distance contrastive loss to conduct domain adaptation.

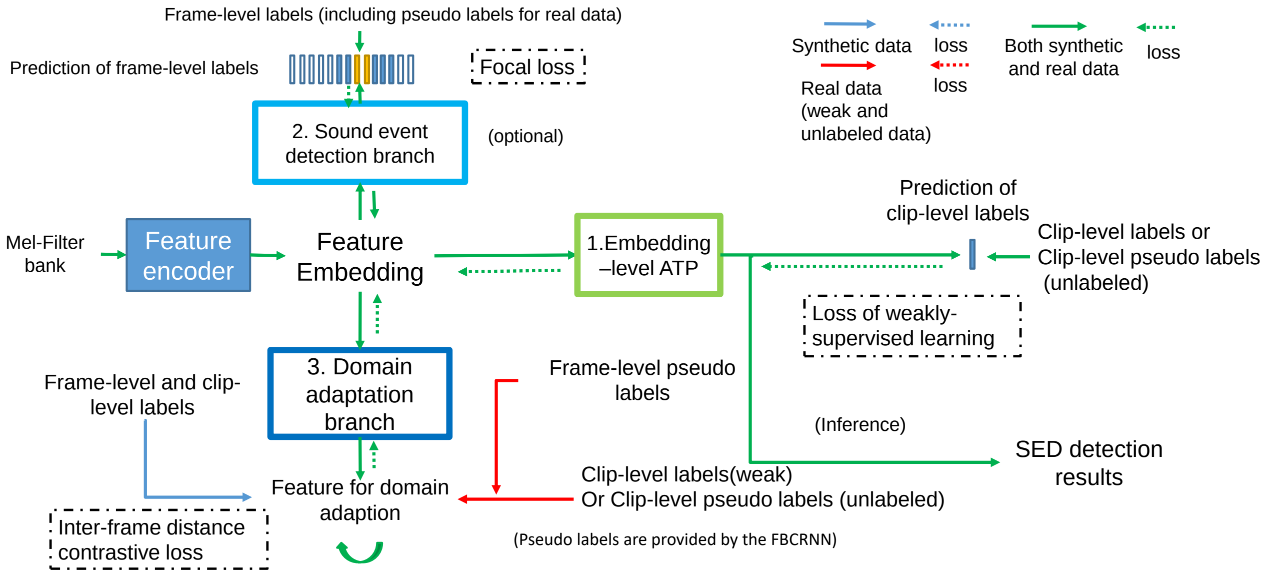
Frame-level labels (including pseudo labels for real data)
Prediction of frame-level labels
Focal loss
2. Sound event detection branch
(optional)
Synthetic data
loss
Real data (weak and unlabeled data)
loss
Both synthetic and real data
loss
Mel-Filter bank
Feature encoder
Feature Embedding
1.Embedding –level ATP
Prediction of clip-level labels
Clip-level labels or Clip-level pseudo labels (unlabeled)
Loss of weakly-supervised learning
Frame-level and clip-level labels
3. Domain adaptation branch
Frame-level pseudo labels
(Inference)
SED detection results
Feature for domain adaption
Clip-level labels(weak) Or Clip-level pseudo labels (unlabeled)
Inter-frame distance contrastive loss
(Pseudo labels are provided by the FBCRNN)

Figure 1: Overview of the MLFL

2.2. Embedding-level pooling

The embedding-level pooling branch is based on the multiple instance learning framework. It applies the embedding-level pooling strategy and use the attention pooling method. The implementation of the embedding-level pooling in MLFL model is same with the system that Lin et al. proposed in [3], which employs the specialized decision surface for inference.

2.3. Domain adaptation based on inter-frame distance loss

The reason for applying domain adaptation based on inter-frame distance loss is to make full use of data in different scenarios to achieve the purpose of improving the performance and adaptability of the algorithm. The real data and synthetic data are mapped to the embedding for domain adaptation through the common feature encoder and the domain adaptation branch. Then, the frame-level feature embeddings of real data and synthetic data are paired by one-to-one correspondence, and domain adaptation based on inter-frame distance contrastive (IFDC) loss is applied. The IFDC loss calculates the difference in the distribution of data feature embedding in two different scenarios. Between the synthetic data and the real data, it makes the distance of the frame-level feature embeddings with the same category closer, and the distance of the frame-level feature embeddings with different categories further. The implementation of domain adaptation branch is a dense projection layer. The input of this layer is frame-level feature embedding, and the output is domain embedding representation.More details of the method can be found in [5]

2.4. The sound event detection branch with focal loss

To make better use of the strong labels, including pseudo strong labels of the unlabeled and weakly-labeled data and strong labels of the synthetic data, we also added a sound event detection branch (SEDB). For the SEDB, we use the focal loss function to measure the contribution of hard-to-classify and easy-to-classify samples to the total loss. This function reduces the weight of easy-to-classify samples, so that the model focuses more on difficult-to-classify samples during training, and quickly achieves the purpose of sample balance. For all training data, the strong labels or pseudo strong labels are used to train the SEDB, and the output of the SEDB is the probability of each frame.

2.5. The tag-conditioned CNN

We use the tag-conditioned CNN module to help us complete the final predicting work of the system. For training this module, we use the log-mel spectrogrm and audio-tags output by the MLFL as input and use the strong label of synthetic data and the strong pseudo-label of real data output by the MLFL as label. And we use it to predict the final results by using the log-mel spectrogram and the audio-tags output by the MLFL as input.

2.6. Data augmentation

For all training data, including weakly labeled data, unlabeled data, and synthetic data, we use mixup method to generate augmented data. For mixup method, it generated augmented data by getting the weighted sum of the two pieces of data.

3. SYSTEM

3.1. System overview

We selected the four training models that performed best on the validation set and sorted them according to the results. We adopt an average weighting method to fuse the results generated by the four models, the top three models, and the top two models, the results on the validation set correspond to Result2, Result1 and Result3

Table 1: The PSDS and event-based F1 score on validation set

Model	PSDS-scenario1	PSDS-scenario2	Event-F1
Baseline	0.342	0.527	0.401
Result1	0.401	0.597	0.550
Result2	0.396	0.587	0.547
Result3	0.392	0.585	0.542
Result4	0.398	0.599	0.549

in Table 1. Finally, we also try to increase the weight of the first-ranked model to 0.4, and the result weights of 2, 3, and 4 were all 0.2 to generate the fusion result as Result4.

3.2. Model architecture

For each single system, we use three modules which are MLFL, FBCRNN and tag-conditioned CNN. The FBCRNN provides strong pseudo-labels for weakly-labeled real data and unlabeled real data to train the MLFL, and the MLFL provides audio-tags and strong pseudo labels of weakly-labeled real data and unlabeled real data to train the tag-conditioned CNN. And the tag-conditioned CNN conducts final sound event detection.

4. EXPERIMENT

4.1. Experimental setup

The training set of our SED system contains a weakly labeled training set (1578 clips), an unlabeled training set (14412 clips), and a synthetic strong labeled set (10,000 clips). The verification set contains 1211 strongly marked clips. All detection results are evaluated using the poly-phonic sound event detection scores (PSDS), which is calculated on the real recordings in the evaluation set (the performance of synthetic recordings is not considered in the indicator). The PSDS [8] value is calculated using 50 operating points (linearly distributed from 0.01 to 0.99). In order to better understand the behavior of each submission for two different scenarios that emphasize different system attributes,we report the PSDS results and event-based f1 [9] results of each model.

4.2. Experimental results

The experimental results are shown in Table 1. We use the four best-performing results for model integration. The best sys achieves a PSDS-scenario1 of 0.401 and a PSDS-scenario2 of 0.597 on the validation set, and achieved an F1 score of 0.55.

5. CONCLUSIONS

This article presents the system we submitted to DCASE 2021 Task 4. The system is based on the model which includes three modules: FBCRNN, MLFL and tag-conditioned CNN. For the main part MLFL, we use a weakly-supervised learning framework, with embedding-level attention pooling module. We also uses the mean-teacher architecture and strong pesudo labels for semi-supervised learning. To better use the synthetic data, we use metric learning by inter-frame distance contrastive loss to conduct domain adaptation. In addition, the sound detection branch with focal loss is added and retains more valuable feature information by using the strong labels of the data, of which the strong pesudo labels of real data (weakly-labeled and unlabeled) is provided by FBCRNN. Finally, we use tag-conditioned CNN to conduct prediction. We focus on the shortcomings of the SED task in different aspects, combines the solutions into the system, and obtained competitive results.

6. REFERENCES

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

[2] http://dcase.community/challenge2020/task-sound-event-detection-and-separation-in-domestic-environments.

[3] L. Lin, X. Wang, H. Liu, and Y. Qian, “Specialized decision surface and disentangled feature for weakly-supervised polyphonic sound event detection,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 1466–1478, 2020.

[4] 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.

[5] Y. Huang, L. Lin, S. Ma, X. Wang, H. Liu, Y. Qian, M. Liu, and K. Ouchi, “Guided multi-branch learning systems for sound event detection with sound separation,” in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2020 Workshop (DCASE2020)*, Tokyo, Japan, November 2020, pp. 61–65.

[6] Y. Huang, L. Lin, S. Ma, X. Wang, H. Liu, M. Liu, and K. Ouchi, “Guided multi-branch learning systems for dcase 2020 task 4,” *arXiv:2007.10638v1*, 2020.

[7] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, “Focal loss for dense object detection,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018.

[8] C. Bilen, G. Ferroni, F. Tuveri, J. Azcarreta, and S. Krstulovic, “A framework for the robust evaluation of sound event detection,” *arXiv preprint arXiv:1910.08440*, 2019. [Online]. Available: https://arxiv.org/abs/1910.08440

[9] 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