SEMI-SUPERVISED SOUND EVENT DETECTION BASED ON PRETRAINED MODELS FOR DCASE 2024 TASK 4

Technical Report

Jingxuan Chen, Xichang Cai, Ziyi Liu, Haiyue Zhang, Liangxiao Zuo, Menglong Wu*

North China University of Technology
Electronic and Communication Engineering
Beijing, China
caixc_ip@126.com

ABSTRACT

In this technical report, we present our submission system for DCASE 2024 Task 4: Sound Event Detection in Domestic Environments with Heterogeneous Training Dataset and Potentially Missing Labels. Firstly, our proposed system employs a full-frequency dynamic convolution (FFD-Conv) network based on the Mean Teacher semi-supervised learning framework. Secondly, we utilize a two-stage training framework, where in the first stage, a large unlabeled in-domain set is converted into pseudo-weak labels to balance the number of strongly labeled datasets in the second stage. Additionally, we employ various methods such as data augmentation, post-processing, and model ensembling to further enhance the generalization capability of the system. Ultimately, our system achieved a PSDS-scenario1 score of 0.535 and a macro-average pAUC score of 0.697 on the validation set.

***Index Terms*—** Sound event detection, DCASE2024, Mean Teacher, Semi-supervised learning, CRNN, two-stage framework

1. INTRODUCTION

This technical report describes our submitted systems for DCASE 2024 Task 4: Sound Event Detection in Domestic Environments with Heterogeneous Training Dataset and Potentially Missing Labels. The target of this task is to provide the event class along with the event time boundaries, given that multiple events can be present and may overlap in an audio recording.

This task aims to explore how to leverage training data with varying annotation granularity (temporal resolution, soft/hard labels). The baseline network model adopts a Convolutional Recurrent Neural Network (CRNN) [1] and uses the Mean Teacher (MT) [2] approach. Additionally, the baseline utilizes the pre-trained BEATs model to extract audio embeddings, which has helped the model achieve better performance compared to previous years.

In our proposed approach, there are several major improvements. First, we replaced the traditional CNN blocks with full-frequency dynamic convolution (FFD-Conv) [3] blocks. This structure processes the channel and spatial dimensions separately through two branches. These blocks can extract more features from the audio, improving the accuracy of classification and time localization. Second, we employed a two-stage framework to convert unlabeled data into pseudo-weak labeled data for training. This method addresses the issue of the limited number of weak labels and the insufficient proportion of strongly labeled data. Additionally, we trained a separate RNN model using BEATs pre-training and ensembled multiple high-performing models to further enhance the PSDS1 performance.

* Corresponding author.

2. PROPOSED METHODS

2.1. Network architecture

We used a total of three models, as described below:

Model 1: We adopted a Convolutional Recurrent Neural Network (CRNN), which follows the same structure as the baseline [4]. The CNN part consists of 7 layers, with 16, 32, 64, 128, 128, 128, and 128 filters in each layer, respectively. Each layer has a 3x3 kernel size and uses average pooling with sizes of [2,2], [2,2], [2,1], [2,1], [2,1], [2,1], and [2,1]. The RNN employs 2 layers of 128 bidirectional gated recurrent units (Bi-GRU) [5]. Additionally, we utilized the pre-trained BEATs model [Citation 3] in our system. Since the sequence length of the extracted frame-level features differs from that of the CNN features, adaptive average pooling is used to unify the sequence length. Finally, these features are fed into an RNN + MLP classifier.

Model 2: We used FFD-CRNN. This network employs a separate branch to predict kernels for each frequency band, with the kernel content based on the input feature. We replaced the last 6 layers of the 7-layer CNN in the baseline CRNN network with FFD-convolutional blocks and adjusted the number of filters in each layer as shown in Figure 1. Like Model 1, we utilized the BEATs pre-trained model for feature fusion.

Model 3: We completely removed the CNN part of the CRNN network and used 3 layers of 512 bidirectional gated recurrent units (Bi-GRU) along with the frame embeddings from the BEATs pre-trained model.

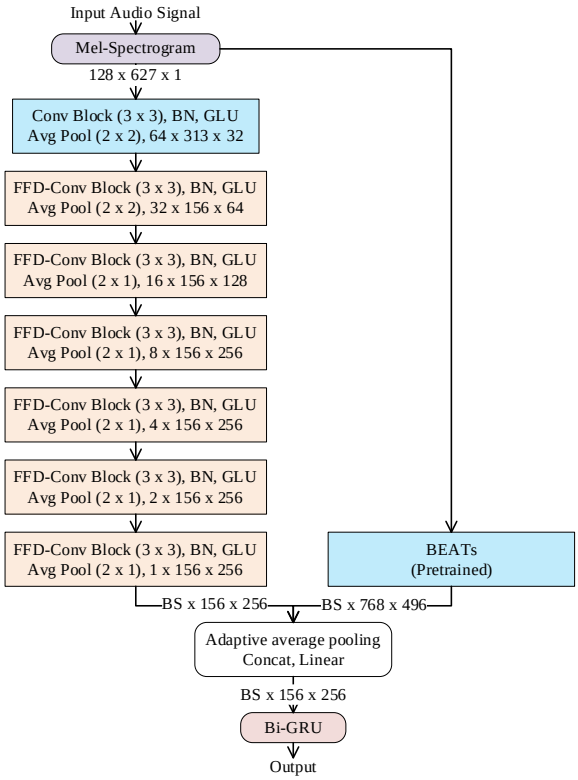

Figure 1: FFD-CRNN architecture with using pretrained BEATs, where BS denotes batch size.

2.2. Two-stage framework

During training, we found that the weakly labeled dataset was significantly smaller than the other datasets (only 1578 clips). This resulted in the weakly labeled data being exhausted while only a small portion of the strongly labeled dataset could be used. We believed that the insufficient proportion of the strongly labeled dataset would negatively impact the final PSDS1 score, so we adopted a two-stage framework.

Like [6], in the first stage, we transformed the strongly labeled data into weakly labeled data by removing onset and offset information. This data was then combined with other datasets and fed into the FFD-Conv network mentioned in Section 2.1 for training. During training, we employed the weak SED method [7], which involves making predictions solely on weak labels and setting the timestamp length to the duration of the entire audio clip to maximize the accuracy of weak label predictions. Subsequently, we used the unlabeled data with the model trained in the first stage to obtain pseudo-weak labels.

In the second stage, we trained the FFD-Conv network using strongly labeled data, soft labels, and the pseudo-weak labels obtained from stage one. Additionally, we employed weak prediction masking [7] to enhance strong predictions during training.

2.3. Data Augmentation

During the training process, data augmentation strategies including mixup[8], frameshift[9], and FilterAugment[7] were employed. Mixup randomly selects two samples-label pairs to generate new data for improving model generalization. Frameshift moves features and labels along the time axis, and FilterAugment applies random weights to different frequency bands of the Mel spectrogram by randomly dividing the frequency range into several sub-bands, which helps train SED models to recognize time frequency patterns from a wider frequency range.

3. EXPERIMENT

3.1. Dataset

We trained and evaluated the proposed model on the development dataset of DCASE2024 Task 4. Unlike DCASE2023 Task 4A, DCASE2024 Task 4 introduces the MAESTRO Real dataset [10] in addition to the DESED dataset [11]. The MAESTRO Real dataset includes soft-labeled strong annotations collected from crowdsourced annotators in various acoustic environments. Due to the approximately 3-minute duration of each audio segment in this dataset, the segments were sliced into 10-second chunks. The development set comprises several different datasets, including:

- **Weakly labeled training set:** 1578 clips
- **Unlabeled in domain training set:** 14412 clips
- **Synthetic strongly labeled training set:** 10000 clips
- **Synthetic strongly labeled validation set:** 2500 clips
- **Strongly labeled validation set:** 1168 clips
- **Strong-label Audioset dataset:** 3470 clips
- **Maestro real training set:** 7503 clips
- **Maestro real validation set:** 3474 clips

We used all unlabeled in-domain training set, synthetic strongly labeled training sets, partial weakly labeled training sets, and partial MAESTRO Real training set to train the model. All synthetic strongly labeled validation sets, partial weakly labeled training sets, and partial MAESTRO Real training sets were used as validation sets. Strongly labeled validation sets and MAESTRO Real validation set were used to evaluate the performance of the model.

3.2. Experiment setup

The log-mel spectrum is used as the input feature to the SED system. We trained the whole system for 200 epochs and the learning rate warms up in the first 50 epochs with the initial learning rate of 0.001. The batch size is set to 56.

3.3. Result and submissions

We evaluate the system using a threshold-independent implementation of PSDS[12] and macro-average pAUC. The best system achieves 0.535 for PSDS-scenario1 and 0.697 for macro-average pAUC on the validation set, outperforming the results of 0.495 and 0.652 in the baseline system. We submitted 4 systems and the results are shown in Table 1.

Table 2: Experimental results.

System	Model	Data Aug	Two-Stage	Ensemble	PSDS1	pAUCm
Baseline		Only mixup			0.503	0.662
1	1	✓			0.521	0.659
2	2	✓	✓		0.525	0.651
3	3	✓	✓		0.514	**0.697**
4	1+2+3	Only mixup		✓	**0.535**	0.677

4. CONCLUSION

In this technical report, we describe our system submission for DCASE 2023 Challenge Task 4A. We primarily employed FFD-Conv to replace conventional CNN layers and utilized a two-stage framework to augment the quantity of weak labels. During training, we employed techniques such as data augmentation and weak prediction masking to enhance system performance. The system achieved a PSDS1 score of 0.535 and a macro-average pAUC score of 0.697 on the validation set.

5. REFERENCES

[1] L. Delphin-Poulat and C. Plapous, “Mean teacher with data augmentation for DCASE 2019 task 4 technical report,” Detection and Classification of Acoustic Scenes and Events (DCASE)Challenge,2019.http://www.ieee.org/web/publications/rights/copyrightmain.html

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

[3] Yue H, Zhang Z, Mu D, et al. Full-frequency dynamic convolution: a physical frequency-dependent convolution for sound event detection[J]. arXiv preprint arXiv:2401.04976, 2024.

[4] L. Delphin-Poulat and C. Plapous, “Mean teacher with data augmentation for DCASE 2019 task 4 technical report,” Detection and Classification of Acoustic Scenes and Events (DCASE) Challenge, 2019.

[5] K. Cho, B. van Merri¨enboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, “Learning phrase representations using RNN encoder–decoder for statistical machine translation,” Empirical Methods in Natural Language Processing (EMNLP), pp. 1724–1734, 2014.

[6] Khandelwal T, Das R K, Koh A, et al. Leveraging audio-tagging assisted sound event detection using weakified strong labels and frequency dynamic convolutions[C]//2023 IEEE Statistical Signal Processing Workshop (SSP). IEEE, 2023: 329-333.

[7] Nam H, Ko B Y, Lee G T, et al. Heavily augmented sound event detection utilizing weak predictions[J]. arXiv preprint arXiv:2107.03649, 2021.

[8] Zhang H, Cisse M, Dauphin Y N, et al. mixup: Beyond empirical risk minimization[J]. arXiv preprint arXiv:1710.09412, 2017.

[9] L. Delphin-Poulat and C. Plapous, “Mean teacher with data augmentation for dcase 2019 task 4 technical report,” 2019.

[10] Martín-Morató I, Mesaros A. Strong labeling of sound events using crowdsourced weak labels and annotator competence estimation[J]. IEEE/ACM transactions on audio, speech, and language processing, 2023, 31: 902-914.

[11] Turpault N, Serizel R, Shah A P, et al. Sound event detection in domestic environments with weakly labeled data and soundscape synthesis[C]//Workshop on Detection and Classification of Acoustic Scenes and Events. 2019.

[12] Janek Ebbers, Reinhold Haeb-Umbach, and Romain Serizel. Threshold independent evaluation of sound event detection scores. In ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 1021–1025. IEEE, 2022.