MIZOBUCHI PCO TEAM'S SUBMISSION FOR DCASE2022 TASK4 -SOUND EVENT DETECTION USING EXTERNAL RESOURCES-

Technical Report

Shohei Mizobuchi, Hiromasa Ohashi, Akitoshi Izumi, Nobutaka Kodama,

Advanced Research Lab., R&D Division, Panasonic Connect Co., Ltd.
{mizobuchi.shohei, ohashi.hiromasa, izumi.akitoshi, kodama.nobutaka}@jp.panasonic.com

ABSTRACT

In this Technical report, we describe an overview and performance of the system we submitted for DCASE 2022 Task 4. We submitted the following 4 systems. System 1 is aimed to improve the performance of PSDS1 under the condition that external resources are not used. System 2 uses AudioSet as additional training dataset on System 1. System 3 uses System 1 with additional training dataset including not only AudioSet dataset but also synthetic dataset generated by ourselves, and changes the training conditions to improve the performance of PSDS2. System 4 adds PANNs pretrained model to System 3. The highest performance evaluated using "development dataset" in these systems is 0.4489 for PSDS1 and 0.8519 for PSDS2. Details will be described below.

Index Terms— Sound Event detection, Pretrained models, External data, PSDS

1. INTRODUCTION

Sound event detection (SED) is a technology designed to detect and categorize segments of each sound event from a variety of sound environments. This technology can be applied to a variety of applications. For example, it can be used to detect the sound of gunshots or breaking glass for security purposes. It is especially important in situations where image information is not available due to camera blind spots or other reasons.

Compared to DCASE 2021 Task 4, DCASE 2022 Task 4 allows the use of external data and pretrained models. Therefore, we examined the change in accuracy by adding external data and using pretrained models to our SED model. Our SED model is based on a convolutional recurrent neural network (CRNN)-based mean-teacher model[1]. To improve accuracy by adding external data, we examined the use of data expanded using AudioSet and the use of additional data synthesized from FSD50K, SINS, and MUSAN data. For the pretrained models, we compared their performance when using embeddings obtained from PANNs.

Following this introduction, Section 2 describes the SED model constructed, and Section 3 describes the external data and pretrained models used in the experiments. Finally, We discuss the experimental results.

2. PROPOSED METHOD

2.1. Feature Extraction

Each of the prepared datasets contains 10 seconds of sound signal, resampled to 16000 Hz. The constructed SED model uses log-mel spectrograms as input features. Log-mel spectrograms are computed with a hop size of 256, 2048 STFT windows, and 128 Mel-scale filters. Finally, 10 second of sound signals are represented by (626 x 128) 2D time-frequency.

2.2. Data augmentation

Our system uses data augmentations named filter augmentation[2], MixUp[3], Frame shift[4], and Time mask[5].
filter augmentation: In this method, N to M frequency bands are randomly selected and an $\alpha\sim\beta$dB filter is applied to each of these frequency bands.
MixUp: In this case, the coefficient of beta distribution was set to 50, and mix-up was applied to 80 % of the training data.
Frame shift: In this method, a control point is randomly selected on the time axis of the spectrogram, and the entire spectrogram is shifted so that the control point can move by distance w on the time axis. The distance w is randomly selected from range of -W to W. In this experiment, W was set to 90.
Time mask: In this method, a point on the time axis is randomly selected and the training data and the label are masked with a random range of 0 to T. In this experiment, T was set to 31.

2.3. Network Architecture

Our SED system is based on the CRNN-based architecture used in the DCASE 2021 baseline model. The number of CNN channels consists of 16, 32, 64, 128, 128, 128, and 128. Context Gating is applied to the activation function of the CNN layer and bce is used for the loss function. The RNN consists of two bi-directional gated recurrent units (BiGRUs), which learn temporal context information. The activation function for each GRU uses RELU, and zavier is applied to initialize the RNN. During training, unsupervised learning is performed using the mean teacher model [1]. Teacher model and Student model use the same data augmentation. mse is used to update the Teacher model.

Table 1: Combination of post-processing applied to the submitted system

class name	System1 and System2		System3 and System4	
-	filter length	Probability value correction	filter length	Probability value correction
Alarm Bell Ringing	64ms	1.0	112ms	1.0
Blender	144ms	1.0	112ms	1.0
Cat	16ms	1.0	112ms	1.5
Dishes	64ms	1.0	112ms	1.0
Dog	48ms	3.5	112ms	1.5
Electric Shaver/Tooth brush	16ms	1.0	112ms	1.0
Frying	144ms	1.0	112ms	1.0
RunningWater	144ms	1.0	112ms	1.0
Speech	128ms	4.0	112ms	1.5
VacuumCleaner	80ms	1.0	112ms	1.0

3. EXPERIMENTS

3.1. Dataset

The training data in the Development Dataset provided in The DCASE 2022 Challenge Task 4 consists of three datasets(DESED) [6, 7]. The three datasets are: weakly labeled dataset (1578 clips), unlabeled-in-domain dataset (14412 clips), and strongly labeled synthetic dataset (10000 clips). The weakly labeled dataset and the unlabeled-in-domain dataset are taken from AudioSet [8]. The strongly labeled synthetic dataset is generated using the Scaper soundscape synthesis and extension library [9]. In the Development Dataset, the validation dataset (1168 clips) is provided for validation of trained models. In The DCASE 2022 Challenge Task 4, the evaluation dataset[10] is also provided to submit final results.

3.2. Experimental Settings

In each experiment, the models were trained and saved as appropriate to improve PSDS1 and PSDS2. Specifically, System1 and System2 were trained with both weak and strong labels, while System3 and System4 were trained with strong labels as weak labels. The saved models were used in the ensemble described below. In addition, addabelief was used as the optimizer for System1 and System2, and Adam was used as the optimizer for System3 and System4.

3.3. External Dataset

In this experiment, the strongly labeled AudioSet is used as a sub-training dataset for training.This dataset is the 3470 clips used in the DCASE 2022 baseline script[11] when the strong_real argument is specified. This strong label data is obtained from [12] in the dataset download script.

We also generated synthetic data using an external dataset that allowed us to extend our training data. Using the metadata dev_clips_info_FSD50K.json included in the FSD50K[13], we excluded wave files that did not contain single event tones from the files tagged with the equivalent of the three classes Dishes, Electric Shaver and tooth blushing, Blender. In this way, we extracted 31, 48, and 47 clips of wave files for each event, which we used as the foreground files. As background files, we used the audio files included in SINS[14] and MUSAN[15]. For SINS, we used the background files already split for generating the synth dataset included in the DCASE 2022 Development Dataset. For MUSAN, among the files classified as noise, wave files listed as background in the ANNOTATION file in the data obtained from free-sound were used as background files. A total of 3000 clips of these foreground and background files were generated by using Scaper[9]. For metadata such as label distribution, we used the meta_info_2021.tar.gz file[16] used in the script[17] for generating the DCASE 2021 Development Dataset. In System 2, we added the generated data to the strong label training data and performed training. For System3 and System4, we trained the models after adding the generated data to the weak label training data.

3.4. Pre-trained Model

For the pretrained model, we used the trained CNN-14 from PANNs[18]. Data augmentation similar to CRNNs was applied as input to the pretrained model during training, but gradient computation was not performed. The 2048 elements of the CNN-14 after performing global pooling were combined with the output layer of the constructed CNN and used as input for the RNN with 128 x 2 elements. The pretrained model was applied only to system4 of the submitted systems.

3.5. Post Processing

We applied a median filter for the probabilities which proposed system outputs for each target classes as a post processing. Table 1 shows the post-processing parameters of the submitted System. The ”filter length” indicates the length of the median filter, which is modified according to the class to be detected. The ”probability value correction” indicates the magnification factor for the correction of the existence probability for each class in the final output. The correction of the presence probability is performed with a maximum of 1.0. These parameters were determined heuristically.

Table 2: Description for our submitted system.

system name	model count	pretrained model	external dataset	PSDS1	PSDS2
baseline1	-	-	-	0.336	0.536
baseline2	-	-	AudioSet	0.351	0.552
baseline3	-	AST	-	0.313	0.722
System1	8	-	-	0.425	0.625
System2	16	-	AudioSet	0.449	0.662
System3	10	-	AudioSet & FSD50K with SINS, MUSAN	0.231	0.714
System4	16	PANNs	AudioSet & FSD50K with SINS, MUSAN	0.075	0.852

3.6. Ensemble

The SED models were evaluated using the development dataset, and the top performing model was used in Ensemble. In Ensemble, the existence probabilities of the discriminant classes calculated for each model are averaged and used as the final output.

4. RESULTS

The systems we submitted are shown in Table 2. System name describes the model to be compared, and model count describes the number of models used for ensemble. The external data used for each model is described in the ”pretrained model” and ”external dataset” sections. Finally, the PSDS values for the devropment dataset for each model are described. The change in PSDS due to the external data in the baseline model shows that PSDS1 and PSDS2 increased by about 0.016 due to the AudioSet data expansion, and we can see that it has an effect on improving the accuracy. Comparing the submitted System1 and System2, we can confirm that PSDS1 and PSDS2 increased by 0.024 to 0.037 as well as baseline. We performed an internal evaluation by adding the strong label data using the FSD50K to System2 as training data, but could not confirm the improvement in accuracy of PSDS1 in System2. We assume that this is ineffective in improving accuracy because the labels assigned to the FSD50K are inaccurate for events. System 3 is the result of adding an external dataset to System 1 and training strong label data as weak label data. In System 3, PSDS1 is 0.194 lower than in System 1, but PSDS2 is 0.089 higher. System 4 is the addition of a pretrained model to System 3. Comparing Baseline2 and Baseline3, PSDS2 increased by 0.17, and PSDS2 increased by 0.138 for System4 compared to System3.

5. REFERENCES

[1] A. Tarvainen and H. Valpola, “Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results,” 2017. [Online]. Available: https://arxiv.org/abs/1703.01780

[2] H. Nam, B.-Y. Ko, G.-T. Lee, S.-H. Kim, W.-H. Jung, S.-M. Choi, and Y.-H. Park, “Heavily augmented sound event detection utilizing weak predictions,” DCASE2021 Challenge, Tech. Rep., June 2021.

[3] H. Zhang, M. Cissé, Y. N. Dauphin, and D. Lopez-Paz, “mixup: Beyond empirical risk minimization,” *CoRR*, vol. abs/1710.09412, 2017. [Online]. Available: http://arxiv.org/abs/1710.09412

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

[5] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, “SpecAugment: A simple data augmentation method for automatic speech recognition,” in *Interspeech 2019*. ISCA, sep 2019. [Online]. Available: https://doi.org/10.21437%2Finterspeech.2019-2680

[6] N. Turpault, R. Serizel, A. Parag Shah, and J. Salamon, “Sound event detection in domestic environments with weakly labeled data and soundscape synthesis,” in *Workshop on Detection and Classification of Acoustic Scenes and Events*, New York City, United States, October 2019. [Online]. Available: https://hal.inria.fr/hal-02160855

[7] R. Serizel, N. Turpault, A. Shah, and J. Salamon, “Sound event detection in synthetic domestic environments,” in *ICASSP 2020 - 45th International Conference on Acoustics, Speech, and Signal Processing*, Barcelona, Spain, 2020. [Online]. Available: https://hal.inria.fr/hal-02355573

[8] J. F. Gemmeke, D. P. W. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, “Audio set: An ontology and human-labeled dataset for audio events,” in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 776–780.

[9] J. Salamon, D. MacConnell, M. Cartwright, P. Li, and J. P. Bello, “Scaper: A library for soundscape synthesis and augmentation,” in *2017 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2017, pp. 344–348.

[10] F. Ronchini *et al.*, “Evaluation set dcase 2021 task 4 (for submissions),” June 2021. [Online]. Available: https://doi.org/10.5281/zenodo.5524373

[11] “dcase task4 2022 baseline python script,” https://github.com/DCASE-REPO/DESED_task/blob/master/recipes/dcase2022_task4_baseline/train_sed.py.

[12] F. Ronchini, N. Turpault, R. Serizel, A. Shah, and J. Salamon, “Desed_real,” Mar. 2021. [Online]. Available: https://doi.org/10.5281/zenodo.6444477

[13] E. Fonseca, X. Favory, J. Pons, F. Font, and X. Serra, “FSD50K: an open dataset of human-labeled sound events,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 829–852, 2022.

[14] G. Dekkers, S. Lauwereins, B. Thoen, M. W. Adhana, H. Brouckxon, B. Van den Bergh, T. Van Waterschoot, B. Vanrumste, M. Verhelst, and P. Karsmakers, “The sins database for detection of daily activities in a home environment using an acoustic sensor network,” *Detection and Classification of Acoustic Scenes and Events 2017*, pp. 1–5, 2017.

[15] D. Snyder, G. Chen, and D. Povey, “MUSAN: A Music, Speech, and Noise Corpus,” 2015, arXiv:1510.08484v1.

[16] N. Turpault and R. Serizel, “Desed_synthetic,” Mar. 2020. [Online]. Available: https://doi.org/10.5281/zenodo.4569096

[17] “dcase task4 2021 dataset generation python script,” https://github.com/turpaultn/DESED/blob/master/examples/generate_dcase_task4_2021.py.

[18] Q. Kong, Y. Cao, T. Iqbal, Y. Wang, W. Wang, and M. D. Plumbley, “Panns: Large-scale pretrained audio neural networks for audio pattern recognition,” 2019. [Online]. Available: https://arxiv.org/abs/1912.10211