SOUND EVENT DETECTION WITH WEAK PREDICTION FOR DCASE 2023 CHALLENGE TASK4A

Technical Report

Shengchang Xiao, Jiakun Shen, Aolin Hu, Xueshuai Zhang,Pengyuan Zhang, Yonghong Yan*

University of Chinese Academy of Sciences
Institute of Acoustics
Beijing, China
xiaoshengchang@hccl.ioa.ac.cn

ABSTRACT

In this technical report, we describe our submitted systems for dcase 2023 Challenge Task4A: Sound Event Detection with weak labels and synthetic soundscapes. Specifically, we design two different systems respectively for PSDS1 and PSDS2. As in previous editions of the Challenge, we also predict weak labels of clips to improve PSDS2. The difference is that this year we use shorter segments for specific classes. Moreover, we adopt the energy difference based log-mel spectrogram to improve feature representation. And we use the Multi-dimensional frequency dynamic convolution (MFDConv) to strengthen the feature extraction ability of convolutional kernels. And we use the confidence-wieghted BCE loss in self-training stage. In addition, we also set higher weight for those classes with worse performances. For post-processing, we optimize the probability values of intervals between events to obtain sharper boundaries.

Index Terms— Sound event detection, weak prediction, self-training

1. METHODS

1.1. Dataset

In our work, we mainly used the desed dataset (unlabeled in domain, synthetic strongly labeled, weakly labeled) and audioset to train our system. As for the strongly labeled data, in addition to the synthetic part and auioset strong (3470 clips), we picked up some clips from audioset by ourselves. In particular, we utilize the audioset strong label file and the audioset tsv file to map our desed classes to the 456 classes. The mapping relationship is shown in the table 1. Then we pick out the clips that contain at least one of these events from the audioset (excluding audio containing only speech). Finally, we get about 7000 external real strong labeled clips.

In addition, we calculate the duration length and occurrences of ten event classes in validation set and audioset strong set. The result of 4638 clips is shown in Table 2. It's shown that ten event classes can be roughly divided into two groups: long duration and short duration. The long duration group includes Blender, Electric_shaver_toothbrush, Frying, Running_water and Vacuum_Cleaner. The short duration groups includes Alarm_bell_ringing, Cat, Dishes, Dog, Speech. Moreover, we can conclude that the minimum length for each event is 250ms.

Table 1: Mapping Relationship

Desed classes	Audioset classes code
Alarm bell ringing	/m/0c3f7m, /m/01hnzm, /m/01y3hg, /m/03wwcy, /m/07pp8cl, /m/07pp_mv, /m/046dlr
Blender	/m/02pjr4
Cat	/m/01yrx
Dishes	/m/04brg2, /m/023pjk
Dog	/m/0bt9lr
Electric shaver toothbrush	/m/02g901, /m/04fgwm
Frying	/m/0dxrf, /m/07p9k1k
Running water	/m/01jt3m, /m/02f9f_, /m/02jz0l, /m/03dnzn, /m/0130jx
Speech	/m/0brhx, /m/0ytgt, /m/01h8n0, /m/02zsn, /m/05zppz, /m/09x0r
Vacuum cleaner	/m/0d31p

Table 2: Duration Length and Occurrences of Desed Event Classes

	Mean	**Mid**	**Min**	**Occurrences**
Alarm bell ringing	2.14	1.03	0.25	2143
Blender	5.25	4.80	0.25	313
Cat	1.1	0.74	0.25	781
Dishes	0.55	0.33	0.25	2576
Dog	1.00	0.56	0.25	1949
Electric shaver toothbrush	7.05	8.96	0.3	279
Frying	8.23	10	0.25	620
Running water	6.11	6.09	0.4	833
Speech	1.59	1.04	0.35	9998
Vacuum Cleaner	7.86	9.97	0.25	178

1.2. Weak Prediction

As PSDS2 focuses on avoiding confusion between classes rather than the localization of sound events, we only predict weak labels of clips and set timestamp to start and end of the entire duration of the audio [1] in Dcase2022 Task4. Because of the low Detection Tolerance criterion (DTC) [2], this method can greatly improve the PSDS2 scores. However, for those event class with short duration less than 1s, this didn't work. The winner of Dcase2022 Task4 use the 5 seconds' segments and achieve higher performance in PSDS2 [3]. But according to our analysis from the dataset, there are many dishes and dogs lasting only 250ms. Therefore, we select 2.5 seconds segments as the shortest segment. In addition, it's obvious that

*Corresponding author

different events have different optimal segment duration. Therefore, we train multiple tagging model with different segment durations. According to our experiment, the optimal duration of segments for different events are 2.5s, 5s and 10s. In particular, the results of different segments for all the events are shown in Table 3.

Table 3: Event-wise performance comparision with different durations

Events	2.5s	5s	10s
Alarm bell ringing	3	2	1
Blender	3	1	2
Cat	1	2	3
Dishes	1	2	3
Dog	3	2	1
Electric shaver toothbrush	3	2	1
Frying	3	2	1
Running water	3	1	2
Speech	1	2	3
Vacuum Cleaner	3	2	1

In the table, 1, 2 and 3 represent the highest, mid and lowest performance for the event respectively. It can be observed that for most events the results are to our expectation. The longer events perform better with longer segments and the shorter events perform better with shorter segments. But for the dog, the 10s segments perform the best while 2.5s segments perform worst. This is because in most cases, there are commonly multiple dogs in an audio clip. If the sum of their duration length is more than 1 seconds, all the dogs in the clip will be considered as True Positive.

In our weak prediction system, we train 3 tagging models with 2.5s, 5s, 10s segments. Then we weighted the best, second-best and worst model results according to event-specific parameters and fuse the three results. This method can greatly promote the PSDS2 performance.

1.3. Improved Features

Since frame-level labels are first generated in the sound event detection task, we use the information between frames in the following ways. First, similar to [4], the energy difference between adjacent frames is calculated as Equation 1 to describe the dynamic characteristics of different sound events over time.

$$\Delta F(\mathtt{t},\mathtt{f}) = \mathtt{F}(\mathtt{t}+1,\mathtt{f}) - \mathtt{F}(\mathtt{t},\mathtt{f}) \tag{1}$$

Where F is the log-mel spectrogram, t is the index in the time dimension, f is the index in the frequency dimension, ΔF is the generated difference feature and attached to the original feature along the channel dimension.

Previous studies have shown that adding positional encodings can improve performance in classification tasks [5, 6]. In sound event detection, each event in audio has a continuous period. To learn the relationship between frames at different positions in an event, we add a position matrix P to represent the position of each frame in the audio and then P will be added directly to the log-mel spectrogram.

$$\mathrm{P}(\mathtt{t},\mathtt{f}) = \mathtt{t}/\mathrm{T} \tag{2}$$

Where T is the number of frames.

Finally, we propose an energy-weighted log-mel spectrogram feature used for data augmentation during both the training and testing stage. The energy distributions vary in different sound events, so we use the energy of each frame to weigh the frame features. In this way, frames with sound events usually get higher weights, and frames without sound events get lower weights. In the training stage, the original and weighted features are trained as separate samples. While in the testing stage, the model achieves test-time augmentation by averaging the results of the original features and energy-weighted features. The energy-weighted feature is calculated as follows:

$$\mathrm{weight}_t = \mathrm{Sigmod}\left(\ \tfrac{1}{F}\sum_{f=1}^{F} F(t,f)\ \right) \quad t = 1\ldots\ldots T \tag{3}$$

$$F'(t,f) = F(t,f) * (1 + weight_t) \tag{4}$$

1.4. Model Architecture

For weak prediction, we adopt the FBCRNN [3] and efficient CNN MobileNetV3 [7] pretrained on audioset. In particular, we improve the CNN structure with our proposed multi-dimensional frequency dynamic convolution [8]. For the fused tagging model, we train three times with 2.5s, 5s, 10s segments respectively.

For strong prediction, we adopt the CRNN and Beats for model ensemble. The CRNN is also improved with multi-dimensional frequency dynamic convolution. The Beats are used for embeddings extractor and the model paremters are freezed. The CRNN is trained with self-training methods similar to [3]. We use the stongly labeled data in DESED dataset and pseudo labels for unlabeled and weakly labeld data from [3], which train the CRNN fully supervised. Then we use the trained CRNN and Beats to infer the unlabeled and weakly labeled data again.

It's noting that for our pseudo strong labels, we get a confidence parameter for each event. The confidence parameter is obtained from the inference probabilities. In the next supervised training iteration, we add the confidence parameter in the loss calculation. In particular, when we encode the strong labels to the feature maps, we calculate a confidence weighted map. For the strongly labeled data, the confidence weight is set to 1. For the pseudo labeled data, the confidence weight is set to the corresponding event specific confidence parameters. This can reduce the impact of inaccurate pseudo labels and alleviate the confirmation bias problems.

In addition, because some short classes are difficult for SED such as dishes and dogs, we optimize the BCE loss in the supervised training. Under normal circumstances, all the 10 classes have the same weight in the loss function. To improve the performance of difficult classes, we set higher weight of BCE loss for these class. That means the model will tend to learn the difficult classes better. From the experiments results, we observe the improvement in the high weight classes despite of the overall performance degradation. In fact, we try the higher loss weight for each class and only the worse classes (dishes and dog) can be improved.

1.5. Data augmentation and Post-processing

In our system, we utilize various data augmentation techniques including specaugment [9], mixup [10], frame shift, , FilterAug [1] and add background noise to expand provided data. For specaugment, we apply frequency masking and time masking. The mixup and frame shift strategies is used to enhance the generalization ability. FilterAugment is proposed to consider various acoustic conditions and simulate them. It splits the whole frequency range

Table 4: Experiments Results for submitted systems

system	external	model ensemble	weak prediction	pertrained model	PSDS1	PSDS2
1	✓	25		✓	0.598	0.837
2	✓	16	✓	✓	0.071	0.921
3	✓	50		✓	0.601	0.847
4	✓	25		✓	0.602	0.841
5		1			0.498	0.746
6	✓	1		✓	0.552	0.794
7	✓	1	✓	✓	0.065	0.865

into several frequency bands and multiplies random factors to these bands. The background noise includes Gaussian white noise, pure music and other free sounds.

Because each event class differs in duration length, we use the class-wise median filter. For each sound event, we search for the optimal median filter length. In addition, We also find that some event classes are easily confused by the models on account of their similar spectrograms, i.e. Blender and Vacuum cleaner, Frying and Running water. To compensate for the model's low ability in distinguishing these classes, we train extra models separately to make a further classification. The training segments are cut from the audio files with the strong labels provided by the challenge and the pseudo strong labels with high confidence obtained by our detection model. Each detected event of the easily confused classes are double classified by the classification model, and we increase the probability value of the class verified by both models and decrease the opposite one.

Moreover, because dish has the shortest duration and is most difficult to detect, We consider post processing for this single class. Since all the annotation is longer than 250ms, we place a limit on the length of the detected events. First, the pseudo strong label is obtained with the optimal probability threshold. Then, for each detected 'Dishes' event shorter than 250ms, we extend its high probability such that the duration of the pseudo label extends to 250ms. And we raise the above-mentioned probability with a square root function and we observe further improvement on the PSDS1 metric.

2. EXPERIMENTS RESULTS

Experiments results for our submitted systems are shown in Tabel 4. System 1-4 uses model ensemble and system 5-7 are single models. And system 5 is the base system without external data and pretrained models. System 2 and 7 adopt the weak prediction and obtain low PSDS1 and high PSDS2. We achieve the best PSDS1 of 0.602 and best PSDS2 of 0.921.

3. REFERENCES

[1] 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," *arXiv preprint arXiv:2107.03649*, 2021.

[2] Ç. Bilen, G. Ferroni, F. Tuveri, J. Azcarreta, and S. Krstulović, "A framework for the robust evaluation of sound event detection," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* IEEE, 2020, pp. 61–65.

[3] J. Ebbers and R. Haeb-Umbach, "Pre-training and self-training for sound event detection in domestic environments," Paderborn University, Tech. Rep, Tech. Rep., 2022.

[4] X. Zhang, J. Shen, J. Zhou, P. Zhang, Y. Yan, Z. Huang, Y. Tang, Y. Wang, F. Zhang, S. Zhang, *et al.*, "Robust cough feature extraction and classification method for covid-19 cough detection based on vocalization characteristics," in *23rd Annual Conference of the International Speech Communication Association, INTERSPEECH 2022*, 2022, pp. 2168–2172.

[5] J. Shen, X. Zhang, P. Zhang, Y. Yan, S. Zhang, Z. Huang, Y. Tang, Y. Wang, F. Zhang, and A. Sun, "Piecewise position encoding in convolutional neural network for cough-based covid-19 detection," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* IEEE, 2023, pp. 1–5.

[6] K. Koutini, H. Eghbal-zadeh, and G. Widmer, "Receptive-field-regularized cnn variants for acoustic scene classification," *arXiv preprint arXiv:1909.02859*, 2019.

[7] F. Schmid, K. Koutini, and G. Widmer, "Efficient large-scale audio tagging via transformer-to-cnn knowledge distillation," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* IEEE, 2023, pp. 1–5.

[8] S. Xiao, X. Zhang, and P. Zhang, "Multi-dimensional frequency dynamic convolution with confident mean teacher for sound event detection," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* IEEE, 2023, pp. 1–5.

[9] 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," *arXiv preprint arXiv:1904.08779*, 2019.

[10] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," *arXiv preprint arXiv:1710.09412*, 2017.