CHT+NSYSU SOUND EVENT DETECTION SYSTEM WITH PRETRAINED EMBEDDINGS EXTRACTED FROM BEATS MODEL FOR DCASE 2023 TASK 4

Technical Report

Chia-Chuan Liu[1]*, Tzu-Hao Kuo*[1]*, Chia-Ping Chen*[1]*, Chung-Li Lu*[2]*, Bo-Cheng Chan*[2]*,*
Yu-Han Cheng[2]*, Hsiang-Feng Chuang*[2]

[1] National Sun Yat-Sen University, Taiwan,
{m103040063,m113040053}@student.nsysu.edu.tw,
cpchen@cse.nsysu.edu.tw
[2] Chunghwa Telecom Laboratories, Taiwan,
{chungli,cbc,henacheng, gotop}@cht.com.tw

ABSTRACT

In this technical report, we describe our submission system for DCASE 2023 Task4: sound event detection in domestic environments. We propose FDY CRNN systems using BEATs embeddings. The system adapted late-fusion to concate the feature maps from Frequency Dynamic Convolution and the frame-level embeddings from BEATs. After that, a classification layer produces the prediction from the late-fusion features. The system is trained by the mean teacher framework. We utilize Asymmetric Focal Loss as the supervised loss to alleviate the imbalance between positive and negative samples. Furthermore, we apply two-stage mean teacher training to utilize training data adequateately. Compared to PSDS-scenario 1 of 50% and PSDS-scenario 2 of 76.2% of the baseline system using BEATs embeddings. Our FDY CRNN system achieves 50.1% and 79.8%, respectively. The ensemble of the FDY CRNN system further improves the PSDS-scenario 1 to 52.5% and the PSDS-scenario 2 to 80.4%.

Index Terms— BEATs, FDY-CRNN, Selective kernel unit

1. INTRODUCTION

In this technical report, we describe our submission systems for Detection and Classification of Acoustic Scenes and Events (DCASE) 2023 Task 4 A. In this year, a new baseline system [1] using pretrained embeddings extracted from Bidirectional Encoder representation from Audio Transformers (BEATs) [2] is proposed. BEATs is a transformer-based model trained on the large-scale Audioset [3] dataset. The model is the state-of-the-art system for audio classification tasks. As the output of BEATs, the frame-level embedding contains high-level classification information along temporal dimension, which is useful for SED systems. The new baseline system takes fusion features, which is the concatenation of frame-level embedding extracted from BEATs and feature maps produced by CNN, as input. Due to the temporal resolution of the frame-level embedding is finer, it needs to be downsampled to match the feature maps. There are three different downsampling methods, including average pooling, interpolation and RNN encoding. By fusing with the embeddings of BEATs, the newly baseline system has a huge progress in performance. Based on the baseline, we proposed our improvements to the architecture and training process. Firstly, we used frequency dynamic convolution [4] and selective kernel convolution [5] to implement two different SED systems, called FDY-CRNN and VGGSK respectively. The former improves feature extraction ability in the frequency domain, and the latter uses different kernel sizes with an attention mechanism to enhance discrimination between different event classes. Secondly, we applied noisy student [6], random consistency training [7] and asymmetric focal loss [8] during training. Thirdly, we employ an adaptive median filter to smooth strong label predictions. Our improved system achieves true PSDS-scenario 1 of 50.1% and true PSDS-scenario 2 of 79.8%, which outperform the baseline system of 50% and 76.4%. Furthermore, we ensemble multiple systems with different architectures and training methods. The ensemble system further increases true PSDS-1 to 52.5% and true PSDS-2 to 80.4%.

2. METHODS

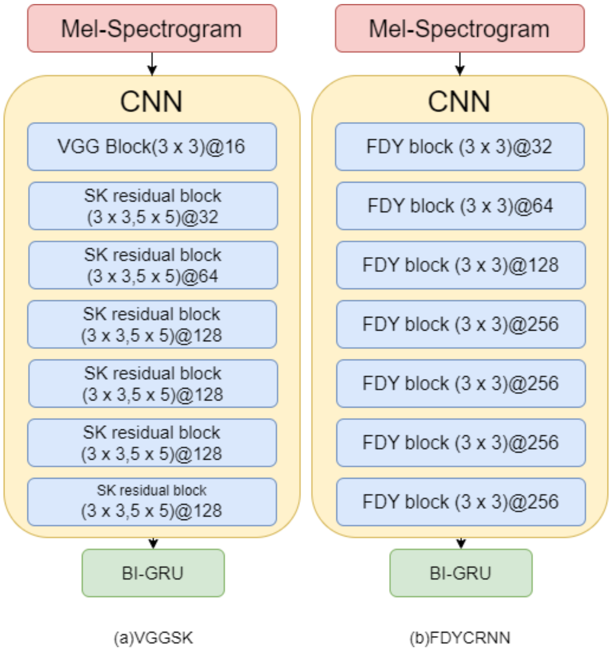
2.1. Model architectures

We propose the VGGSK structure, consisting of selective kernel, residual structure [9] and Visual Geometry Group(VGG) convolution block [10]. The selective kernel(SK) is a dynamic selection mechanism that dynamically adjusts the size of the receptive field of each of the neurons in a CNN network.The residual structure is designed to solve the network degradation problems that can occur in deep networks.However, 2D convolution enforces translational isochronism on the time and frequency axes for sound events, whereas Frequency is not a shift-invariant dimension.To solve this problem we use FDYCRNN as a second network architecture.

The CNN part is made up of 7 convolutional layers, and we replace the normal convolution in the baseline model with SKunit and VGG block.And add a shortcut to make it a residual structure.The RNN part is the same as the baseline with a bi-directional gated recurrent unit (Bi-GRU).Fig. 1 shows the proposed VGGSK and FDYCRNN architecture.

2.2. Pretrained model BEATs

Audioset classification is a task very similar to DCASE TASK4. BEATs are models that perform extremely well in this task. BEATs is an iterative audio pre-training framework to learn Bidirectional



(a)VGGSK

(b)FDYCRNN

Figure 1: (a)VGGSK architecture (b)FDYCRNN architecture

Encoder representation from Audio Transformers, where an acoustic tokenizer and an audio SSL model are optimized by iterations.First, the audio self-supervised learning (SSL) model is trained using random projection as the acoustic tagger. The audio model is then augmented by pre-training or fine-tuning as the acoustic tagger for the next iteration. The main goal is to allow the acoustic tagger and the audio model to promote each other.

In our proposed system, we capture the frame-level embeddings of BEATs and fuse them with the features of the CNN. The frame-level embeddings are matched with the CNN through an adaptive averaging pool and finally fed together into the RNN and MLP classifiers.

2.3. Training process

How to effectively utilize training data affects the performance dramatically. In this section, we will describe our training methods. Since the new baseline system uses the fusion features of the pre-trained embedding extracted from BEATs and the features of CNN as input of the classifier, we also explore training methods that can further improve the performance of the system. We apply noise student, random consistency training, asymmetric loss to improve the data utility.

	training method	PSDS1	PSDS2
FDY-CRNN	MT	0.499	0.794
FDY-CRNN	NS	0. 501	0.798
FDY-CRNN	MT+AFL($\gamma = 0, \zeta = 1$)	0.506	0.785

Table 1: Impact of noisy student and asymmetric focal loss on FDY-CRNN using BEATs embeddings

	training method	PSDS1	PSDS2
FDY-CRNN	MT	0.413	0.647
FDY-CRNN	MT+RCT	0. 435	0.662

Table 2: Performance of FDY-CRNN trained with RCT.

- **Noisy student**
  Noisy student (NS) widely used in DCASE 2022 Task 4, and we will demonstrate that the method also bring positive impact on training SED system using BEATs embeddings. The training method consists of two stages. In the first stage, the SED system is trained from scratch in the mean-teacher framework, while we did not used data augmentation and dropout at this training stage. Next, the teacher model trained from the first stage provide pesudo-labels on the unlabeled data during second stage, and the student model is trained on these.

  Furthermore, we apply mix-up, filter augment, SpecAugment, frame/frequency shift and Gaussian noise to the data sample as the input of the student model and dropout with 0.5 is used in the student model to increase the generalization of the student model.

- **Asymmetric Focal Loss**
  Imbalances between active and inactive samples are commonly found in multi-class tasks. A large number of inactive cases will affect the system's learning on active cases. Thus, we introduce AFL (Asymmetric Focal Loss) instead of BCE(Binary Cross-Entropy) to migrate the imbalance. AFL decouples the active and inactive loss of BCE and applied focal function with different focusing weight.

  $$AFL(\hat{y}) = (1-\hat{y})^{r} y log(\hat{y}) + \hat{y}^{\zeta}(1-y) log(1-\hat{y}) \quad (1)$$

  Here, $\hat{y}$ is the output probability and y is the ground-truth. γ, ζ are focusing parameters applied on active and inactive loss, respectively. We set $\gamma < \zeta$ to suppress inactive loss to enhence the training on active cases. We will show the impact of AFL in experiment results.

- **Random consistency training**
  Self-consistency training between teacher and student model in mean-teacher is an effective way to increase the utilization of training data. RCT(Random consistency training) further leverage the unlabeled data with the concept of self-supervised learning. The self-consistency loss L_{SC} proposed in RCT is the MSE loss between the output of student model that take original and augmented data as input respectively.

  $$L_{SC} = w * MSE(D_{aug}(\hat{y}), \bar{y}) \quad (2)$$

  $\hat{y}, \bar{y}$ denote the output of original and augmented samples of student model. D_{aug} is a transformation function that applied on $\hat{y}$ to maintain the consistency to $\bar{y}$. w is a ramp-up factor increasing during training, and the maximum value of w is 2.

Table 2 shows the impact of appling NS and AFL. The first row is the result of FDY-CRNN using embeddings trained in mean-teacher. When the system was trained with NS, PSDS-1 and 2 were improved slightly. We further replace BCE loss function with AFL. On PSDS-1, the system improved, but on PSDS-2, it decreased. Next, we compare the impact of FDY-CRNN without using embeddings trained with RCT. As shown in Table 2, the performance of FDY-CRNN was improved on PSDS-1 and 2.

2.4. Data augmentation and Post-processing

We apply various data augmentations, including mix-up[11], SpecAugment[12], frame-shift, pitch-shift, filter augment[13] to improve the regulerization of the system. In addition, the duration time of each event class being quite different, we applied an adaptive median filter to smooth the predictions for a better fit. The window length of different classes is the optimal length found by grid search on the validation set.

Due to the DTC (Detection Tolerance Criteria) and GTC (Ground Truth Criterion) thresholds of PSDS-scenario 2 are 0.1, we perform weak SED to minimize false negative cases and elimate cross-trigger predictions. When model inference, we take weak predictions as strong predictions.

3. DATASET AND SIGNAL PROCESSING

3.1. Dataset

We used Domestic Environment Sound Event Detection Dataset (DESED dataset) for model training and evaluation. The dataset contains strong labeled, weak labeled, and unlabeled data. Strong labels provide sound event classes and corresponding time stampes (onset and offset times). And a weak label providess only the classes of an audio clip. The DESED dataset contains 10 classes of sound events, including alarms, bells, and ringing, blenders, cats, dishes, dogs, electric shaver/toothbrushes, frying, running water, speech, and vacuum cleaners. The training set consists of 10,000 synthetic and 3470 sound clips with strong labels, 1,578 sound clips with weak labels, and 14,412 unlabeled samples. The validation set contains 1,168 real samples with strong annotations.

3.2. Signal processing and model training

We take mel-spectrogram as input to the system. Audio clips in the dataset are resampled to 16kHz and mono-channel. Then, mel-spectrogram is extracted using Fourier transform with window size of 2048, hop length of 256 and 128 mel-scale filters. As a consequence, the input acoustic features were represented with 626 frames and 128 frequencies.

The model was trained with ADAM optimizer. The learning rate was 0.001, and we applied ramp-up scheduler in first 50 epochs.

4. SUBMISSION SYSTEMS

We submit several systems, listed in Table 3. The main difference between them is that they use different CNN architectures, VGGSK and FDYCRNN respectively, and whether they are trained using pre-trained models.

PSDS1 and PSDS2 are used to measure the performance of each system. All of our proposed systems have improved performance compared to the baseline System1/system3 increases 14.89% / 21.80% in total of PSDS1 and PSDS2. And system2/system4 increases 1.4% / 2.29% in total PSDS1 and PSDS2. System5 is our ensemble system,which increases 5.14% in total of PSDS1 and PSDS2

Table 3: Description for submitted system

no.	system	pretrained model	PSDS 1	PSDS 2
1	VGGSK	-	0.413	0.667
2	VGGSK	BEATs	0.491	0.791
3	FDYCRNN	-	0.448	0.697
4	FDYCRNN	BEATs	0.499	0.794
5	FDYCRNN(ensemble)	BEATs	0.525	0.804
-	Baseline	-	0.364	0.576
-	baseline	BEATs	0.500	0.764

5. CONCLUSION

In this technical report, we implemented two different systems based on DCASE 2023 Task 4 A baseline, referred to as FDY-CRNN and VGGSK. These two systems were improved on PSDS compared to baseline in our experiment. Furthermore, we utilize training methods widely used in DCASE 2022 Task 4, including noisy student, asymmetric focal loss and random consistency training. The experiment results show that these training methods have an positive effect during training SED systems. Our submission system is an ensemble of SED systems with different architectures and training methods. We average the prediction of each SED system to get a better score.

6. REFERENCES

[1] L. Delphin-Poulat and C. Plapous, “Mean teacher with data augmentation for dcase 2019 task 4,” *Orange Labs Lannion, France, Tech. Rep*, 2019.

[2] S. Chen, Y. Wu, C. Wang, S. Liu, D. Tompkins, Z. Chen, and F. Wei, “Beats: Audio pre-training with acoustic tokenizers,” *arXiv preprint arXiv:2212.09058*, 2022.

[3] J. F. Gemmeke, D. P. 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)*. IEEE, 2017, pp. 776–780.

[4] H. Nam, S.-H. Kim, B.-Y. Ko, and Y.-H. Park, “Frequency dynamic convolution: Frequency-adaptive pattern recognition for sound event detection,” *arXiv preprint arXiv:2203.15296*, 2022.

[5] X. Li, W. Wang, X. Hu, and J. Yang, “Selective kernel networks,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 510–519.

[6] J. W. Kim, G. W. Lee, H. K. Kim, Y. S. Seo, and I. H. Song, “Semi-supervised learning-based sound event detection using frequency-channel-wise selective kernel for dcase challenge 2022 task 4 technical report.”

[7] N. Shao, E. Loweimi, and X. Li, “Rct: Random consistency training for semi-supervised sound event detection,” *arXiv preprint arXiv:2110.11144*, 2021.

[8] T. Ridnik, E. Ben-Baruch, N. Zamir, A. Noy, I. Friedman, M. Protter, and L. Zelnik-Manor, “Asymmetric loss for multi-label classification,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 82–91.

[9] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[10] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.

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

[12] 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.

[13] H. Nam, S.-H. Kim, and Y.-H. Park, “Filteraugment: An acoustic environmental data augmentation method,” in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 4308–4312.