CONVOLUTION-AUGMENTED CONFORMER FOR SOUND EVENT DETECTION

Technical Report

YuHang Chen

Royal Flush
18, Tongshun Street, Wuchang Street,
Yuhang District.
HangZhou 310000, CHINA

ABSTRACT

In this technical report, we describe our submission system for DCASE2021 Task4: sound event detection and separation in domestic environments. Our model employs conformer blocks, which combine the self-attention and depth-wise convolution networks, to efficiently capture the global and local context information of an audio feature sequence. In addition to this novel architecture, we further improve the performance by utilizing a mean teacher semi-supervised learning technique, data augmentation for each sound event class. We demonstrate that the proposed method achieves the PSDS-1 and PSDS-2 score of 34%,55.7% on the validation set, outperforming that of the baseline score.

Index Terms— One, two, three, four, five

1. INTRODUCTION

This technical report describes our submission system for DCASE2021 Challenge Task4: sound event detection (SED) and separation in domestic environments [1]. The goal of this task is to build a system for the detection of sound events using real data either weakly labeled or unlabeled and simulated data that is strongly labeled (with timestamps). To address this task, we propose two neural network models that utilize the self-attention mechanism;

- Conformer-based model [4].
- CRNN model.

These models can efficiently capture both local and global context information of an audio feature sequence through the stack of CNN and self-attention layers. Besides, to further improve the performance, we implement

- semi-supervised learning based on mean teacher [5],
- data augmentation techniques, such as add-noise [6] and mixup [7],
- post-processing refinement,

We conduct experimental evaluations on the DCASE2021Task4 validation set to investigate the effectiveness of the proposed network architecture and each of the implemented techniques. The experimental results show that the proposed models outperform the baseline system, achieving the PSDS-1 and PSDS-2 score of 34%,55.7% on the validation set with the best single system.

2. FROPOSED METHOD

2.1. Feature extraction

Subheadings should appear in lower case (initial word capitalized) in boldface. They should start at the left margin on a separate line. We extract 64-dimensional log-Mel filterbanks from the input audio. The window size and the hop size are 2048 points and 156 points, respectively, in 16 kHz sampling. We fix the length of the feature sequence to 256 frames (corresponding to around 10 seconds). To make the length of feature sequences the same, we perform zero-padding for shorter sequences and truncation for longer sequences from their last frames.

2.2. Network architecture

Inspired by the great success of the self-attention architectures in various fields [2]–[4], [8], [9], we propose a neural network models for SED; Transformer-based model and Conformer-based model. The Transformer-based model consists of three modules; a CNN-based feature extractor, Transformer blocks, and a position-wise classifier [3]. The architecture of the CNN-based feature extractor follows the baseline system of DCASE2021 Task4 [1], which consists of three or seven convolution layers. To match with the input size, we slightly modify the network, which add a liner layer to convert the dims of predicted into labels. The Transformer block follows the architecture in [2], which consists of a multi-head self-attention layer, a layer-normalization layer, and a linear layer with a rectified linear unit (ReLU) activation function followed by another layer normalization. The final position-wise classifier is a simple linear layer to calculate the final outputs that correspond to the sound event types.

The designed CRNN system and the baseline system are basically the same in the framework. The main difference is reflected in the addition of an additional layer of LSTM module between the CNN module and the GRUmodule. The input and output dimensions of the LSTM module are consistent with the GRU module, but The number of neurons in the hidden layer is twice that of the GRU module, and the number of layers is half that of the GRU module.

2.3. Semi-supervised learning

To further improve the performance, we employ the mean teacher technique [5] as one of the typical semi-supervised training methods capable of using unlabeled data in training. We use a mean square error function as the consistency criterion, and set the exponential ramp-up steps [13] and the consistency cost to 10,000 and 2.0, respectively.

2.4. Data augmentation

For data augmentation, we employ time-shifting [6] and mixup [7]. The time-shifting shifts a feature sequence on the time axis, and overrun frames are concatenated with the opposite side of the sequence. We randomly choose the shift size by sampling from a normal distribution with a zero mean and a standard deviation of 90. The use of the time-shifting is helpful for preventing the network from inappropriately learning the location information over the sequence.
The mixup smoothes out the decision boundary by adding pseudo data generated by mixing different data points (x1 , x2) and the corresponding labels (y1 , y2). The mixup is same with the [7].

2.5. Post-processing

To determine the sound event activation, we perform thresholding for the network output posterior. Then, we perform median filter ing as post-processing to smooth the detected activation sequence. Since each sound event has different characteristics, such as temporal structures, the optimal post-processing parameters depend on the individual sound events. Hence, we determine the optimal post- processing parameters for each sound event using the validation set. In our system we set the median filter size from 7,13,41 respectively.

3. EXPERIMENTAL EVALUATION

3.1. Experimental conditions

We conducted experimental evaluations using DCASE2021 Task4 dataset [1]. The dataset included 2,584 audio clips with a strong label, 1578 audio clips with a weak label, and 14,412 unlabeled audio clips. Since the audio clips were collected from YouTube, the dataset included various audio clips with different recording settings (e.g., 16 kHz sampling rate vs. 44.1 kHz sampling rate). To address this issue, we first converted all of the audio clips to be 1 ch, 16 bit, and 16 kHz sampling rate using sox [14]. To verify the performance, we compared the following models:

Baseline: The DCASE2021 Task4 official baseline system [15]. The architecture was a convolutional recurrent neural net- work (CRNN), and it was trained with the mean-teacher semi-supervised learning technique [5]. We used the num- bers provided in the official HP.

Conformer (Ours): The proposed Conformer-based model. The number of attention units and that of the attention heads were 256 and 4, respectively. The dropout rate was set to 0.1.

CRNN (Ours): The proposed CRNN-based model. The detailed network configuration information can be seen in chapter 2.2
We used RAdam [16] optimizer with a batch size of 24 and a learning rate of 0.001. We used a GPU (NVIDIA 1080ti) to train the models. It took around 12 hours to finish the training. The detailed training condition is shown in Table 1.
The evaluation metrics were the poly- phonic sound event detection score (PSDS) [17]. These metrics were calculated using sed eval toolkit [18]. The segment length in the segment-based evaluation was set to 1 second. We computed PSDS using 50 thresholds from 0.01 to 0.99.

Table 1: *Network training configuration.*

Training Samples	strong =2584
	weak = 1578
	unlabeled = 14412

Batch size	6,6,12
Epochs	200
Optimizer	RAdam
Learning rate	0.001
Consistency cost	2.0

3.2. Experimental Result

We investigated the effects of the model architecture. In a comparison of the model architectures, we used the post-processing and the mean teacher learning but we did not use the data augmentation for the proposed method. From the result shown in Table 2, we can observe that both the proposed models outperform the baseline even if we do not use data augmentation, revealing the effectiveness of the self-attention architecture for SED.
Next, we investigated the effects of the number of Conformer blocks. The result in Table 3 shows that the number of blocks affects the performance and 4 was the best. We used this configuration in the following experiments.

Table2: Effects of model architectures.

Method	PSDS-1[%]	PSDS-2[%]
Baseline	33.6	52.7
CRNN(our)	34.0	52.3
Conformer	13.3	55.7

4. COLCLUSION

In this technical report, we have described our submission system for DCASE2021 Task4. Our system has been developed by using the self-attention architecture including the Conformer blocks, the data augmentation techniques, the class-dependent post-processing. The experimental results using the validation set have demonstrated that our system outperforms the baseline. In future work, we will investigate the class-wise performance more carefully to develop more effective model ensemble technique, and further integrate source separation techniques to sound event detection.

5. REFERENCES

[1] http://dcase.community/workshop2021/.

[2] http://www.ieee.org/web/publications/rights/copyright-main.html

[3] E. G. Williams, *Fourier Acoustics: Sound Radiation and Nearfield Acoustic Holography*, London, UK: Academic Press, 1999.

[4] C. D. Jones, A. B. Smith, and E. F. Roberts, “A sample paper in conference proceedings,” in *Proc. IEEE ICASSP*, 2003, vol. II, pp. 803-806.

[5] A. B. Smith, C. D. Jones, and E. F. Roberts, “A sample paper in journals,” *IEEE Trans. Signal Process.*, vol. 62, pp. 291-294, Jan. 2000.

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

[7] H. Zhang, M. Cisse, Y. N. Dauphin, and D. LopezPaz, “Mixup: Beyond empirical risk minimization,” arXiv preprint arXiv:1710.09412, 2017

[8] S. Karita, N. Chen, T. Hayashi, et al., “A comparative study on transformer vs RNN in speech applications,” in Proc. ASRU, 2019, pp. 449–456.

[9] Y. Fujita, N. Kanda, S. Horiguchi, et al., “End-to-end neural speaker diarization with self-attention,” in Proc. ASRU, 2019, pp. 296–303.

[10] P. Ramachandran, B. Zoph, and Q. V. Le, “Searching for activation functions,” arXiv preprint arXiv:1710.05941, 2017.

[11] Y. N. Dauphin, A. Fan, M. Auli, and D. Grangier, “Language modeling with gated convolutional networks,” in ICML, 2017, pp. 933–941.

[12] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” in NAACL, 2019, pp. 4171–4186.

[13] S. Laine and T. Aila, “Temporal ensembling for semisupervised learning,” arXiv preprint arXiv:1610.02242, 2016.

[14] http://sox.sourceforge.net/.

[15] https : / / github . com / turpaultn / dcase20 _ task4/tree/public_branch/baseline.

[16] L. Liu, H. Jiang, P. He, et al., “On the variance of the adaptive learning rate and beyond,” in ICLR, 2020.

[17] A. Mesaros, T. Heittola, and T. Virtanen, “Metrics for polyphonic sound event detection,” Applied Sciences, vol. 6, no. 6, p. 162, 2016.

[18] 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.