SE-PROTONET: PROTOTYPICAL NETWORK WITH SQUEEZE-AND-EXCITATION BLOCKS FOR BIOACOUSTIC EVENT DETECTION

Technical Report

Junyan Liu[1], *Zikai Zhou*[2,3], *Mengkai Sun*[2,3],
Kele Xu[1,*], *Kun Qian*[2,3,*], *and Bin Hu*[2,3,*]

[1] National University of Defense Technology,
1838060124@qq.com, xukelele@163.com
[2] Key Laboratory of Brain Health Intelligent Evaluation and Intervention,
Ministry of Education (Beijing Institute of Technology), P. R. China
{zikaizhou, smk, qian, bh}@bit.edu,cn
[3] School of Medical Technology, Beijing Institute of Technology, P. R. China

ABSTRACT

In this technical reprot, we describe our submission system for DCASE2023 Task5: Few-shot Bioacoustic Event Detection. We propose a metric learning method to construct a novel prototypical network, based on adaptive segment-level learning and Squeeze-and-Excitation (SE) blocks. We make better utilization of the negative data, which can be used to construct the loss function and provide much more semantic information. Most importantly, we propose to use SE blocks to adaptively recalibrate channel-wise feature response, by explicitly modeling interdependencies between channels, which improves f-measure to 63.94 %. For the input feature, we use combination of per-channel energy normalization (PCEN) and delta mel-frequency cepstral coefficients (ΔMFCC). Our system performs better than the baseline given by the officials, on the DCASE task 5 validation set. Our final score reaches an f-measure of 65.49 %, outperforming the baseline performance by 30.18 %.

Index Terms— DCASE, few-shot bioacoustic event detection, prototypical network, adaptive segment-level learning, data augmentation

1. INTRODUCTION

Few-shot classification [1, 2, 3, 4] is a task in which a classifier must be adapted to accommodate new classes not seen in training, when given only a few examples. Using a naive approach, such as training the model on a few data, would lead to severe overfitting, which causes a bad generalization[5]. Sound event detection [6] is a task that needs to locate the onset and offset of certain sound classes. In order to solve the few data problem in the audio field, Wang *et al.* combine the idea of few-shot learning with sound event detection, which can detect a new sound event with only a few labeled samples. This makes it highly suitable for tasks where labeling the data may be costly to annotate, such as monitoring the animal population through their vocalizations.

In the previous DCASE 2021 task 5, most of the participants used a prototypical network[7]. Anderson *et al.* [8] proposed to use the prototypical network combined with various data augmentation methods, combining with per-channel energy normalization (PCEN) feature. Yang *et al.* [9] proposed a transductive inference method to maximize the mutual information between query features and their label predictions. Tang *et al.* [10] proposed to use embedding propagation and attention similarity approaches to improve the model performance. Various data augmentation methods are used in the system described in [11, 12].

In the DCASE 2022 task 5, Liu *et al.* [13] mentioned that in the previous works, the negative segments in each audio file are not fully used. So they proposed to use both positive segments and negative segments to construct the system, which outperformed the baseline by a large margin. Our system is based on their main idea, and we propose a new metric learning architecture, called **SE-prototypical network**, which can better utilize the information from different channels to improve the model performance and model generalization.

Metric learning [14, 15, 16] is a machine learning method aimed at learning a distance metric function, so that similar samples are closer and un-similar samples are farther under this metric. Metric learning is commonly used for tasks such as classification, clustering, and retrieval, which can improve model performance by learning a better distance. In the previous task 5 challenges, most of the studies [8, 10, 9] only use the positively labeled data to make the features closer. However, the positive data also need to be distinguishable from the negative data within the same audio file. We utilize better both positive segments and negative segments to solve the problem.

Because no external dataset is allowed unless permission is granted, we do not use the AudioSet [17]. We also have studied different audio features to choose the best feature for this task, including log-mel spectrogram (MEL), PCEN[18], mel-frequency cepstral coefficients (MFCC), and delta-MFCC (ΔMFCC). Finally, we tend to use the combination proposed by Liu *et al.* [13], using PCEN and delta-MFCC together as our input features.

This technical report is organized as follows. Section 2 pro-

This work was partially supported by the Ministry of Science and Technology of the People's Republic of China with the STI2030-Major Projects (No. 2021ZD0201900), the National Natural Science Foundation of China (No. 62227807 and 62272044), and the Teli Young Fellow Program from the Beijing Institute of Technology, China

Corresponding authors: K. Xu, K. Qian, and B. Hu vides an overview of our system. Section 3 introduces the methods we proposed and used to improve our system. Section 4 provides the experiments and results. Section 5 summarizes this work and provides a conclusion.

2. SYSTEM OVERVIEW

2.1. Dataset

Challenge official dataset DCASE 2023 task 5 dataset contains a development set, which includes a training set and an validation set, and an evaluation set. The training and validation sets are both fully labeled. The evaluation set is provided only with the labels of the first five positive events.

We use the training set and the validation set from the development set provided by DCASE for training. For the validation set, we only use the first five annotations for training, and the remaining part is used to verify the training effect.

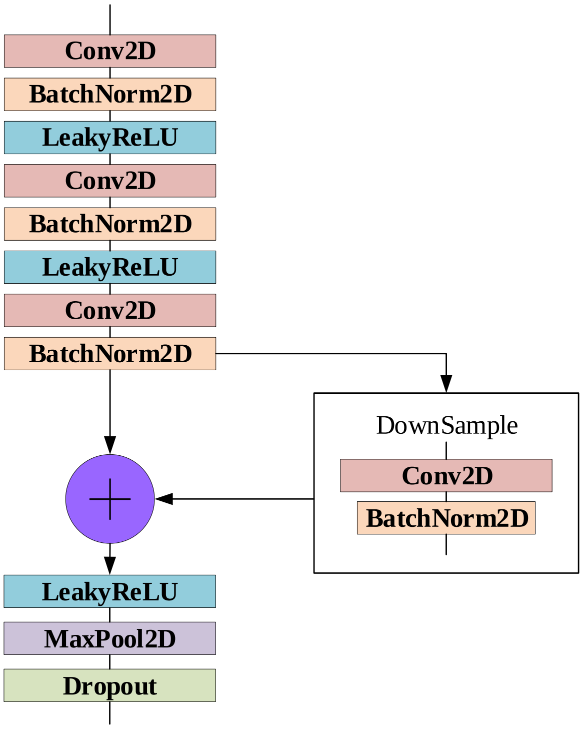

Figure 1: Basic Block

2.2. Model Architecture

The original baseline system contains an encoder, which made up of 4 ConvBlocks, each of which contains a Conv2d layer, a BatchNorm2d layer, a ReLU function, and a Maxpool2d layer. The newly revised baseline system is constructed on the basis of ResNet framework, which also contains 4 Basic Blocks, and uses a downsampled feature to act as a residual feature. The architecture of Basic Block is shown in Figure 1. For our novel prototypical network architecture, we have made some changes on the original framework. We introduce the Squeeze-and-Excitation mechanism, which will be discussed later in Section 3. The whole network architecture is shown Figure 2. We use several SE blocks to enhance the important feature in order to get better performance. The more details about the architecture will be introduced in Section 4.

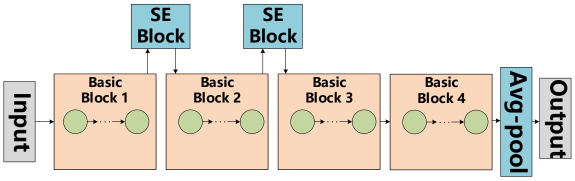

Figure 2: Network Architecture

2.3. Evaluation metric

We use the event-based f-measure as the evaluation metric for all the experiments. Meanwhile, we calculate and record the precision and recall for each epoch. To determine the optimal choice for threshold of the evaluation set in 2023, we calculate the f-measure of the Full version validation set in 2022.

3. METHOD

3.1. Feature extraction

As described above, delta MFCC and PCEN are mixed as feature in our system.

Delta MFCC Perform cepstrum analysis (taking logarithms and performing DCT transformation) on the Mel-spectrogram to obtain the Mel-scale Frequency Cepstral Coefficients (MFCC). MFCC is derive and mixed with the original MFCC to obtain delta MFCC.

PCEN Per channel energy normalization introduces a normalization mechanism for each channel based on FFT or Fbank features to suppress the impact of input signal amplitude changes on recognition results

3.2. Prototype network

A prototypical network[7] is a type of neural network that uses a similarity-based approach to classify input data. The basic idea behind it is to learn a prototype for each class in the training data. A prototype is a representative example of a class that captures the essential features of the class.

To classify a new input, the prototypical network computes the similarity between the input and each prototype, The similarity is typically measured using a distance metric, such as Euclidean distance or cosine similarity. The input is then classified as belonging to the class with the closest prototype.

During training, the prototypical network is given a set of labeled training data. For each class, the networks learns a prototype by computing the mean of all the training examples in that class. It Uses a distance metric to measure how similar the input is to the prototype. Then the input is classified as belonging to the class with the closest prototype, which is typically done using a nearest neighbor algorithm. The prototypical network can be trained using gradient descent or other optimization algorithms to minimize a loss function that measures the distance between the input and its assigned prototype.

From the official baseline system, we find that it uses the average embedding of the entire audio set as the negative prototype, because of no negative annotation given. However, it is based on the assumption of the positive event is sparse. In most of the evaluation files, the positive events are very dense. Building a negative prototype in this way can lead to a degraded result.

In order to better construct the positive prototype and the negative prototype, we propose two assumptions:

1. The positive events do not vary a lot. So the positive prototype is calculated by simply averaging the embeddings of the labeled positive segments.

2. The negative prototype are built by the negative sample searching algorithm, proposed by Liu *et al.* [13]. The algorithm includes a frequency bins weighting step and a frequency pattern matching step.

   - The frequency bins weighing operation is proposed to help us find the negative event more accurately, by getting the frequency band that is most likely to contain the target sound event.
   - The frequency pattern matching aims to locate possible negative samples, by using a threshold calculated using the minimum SISNR [19] value.

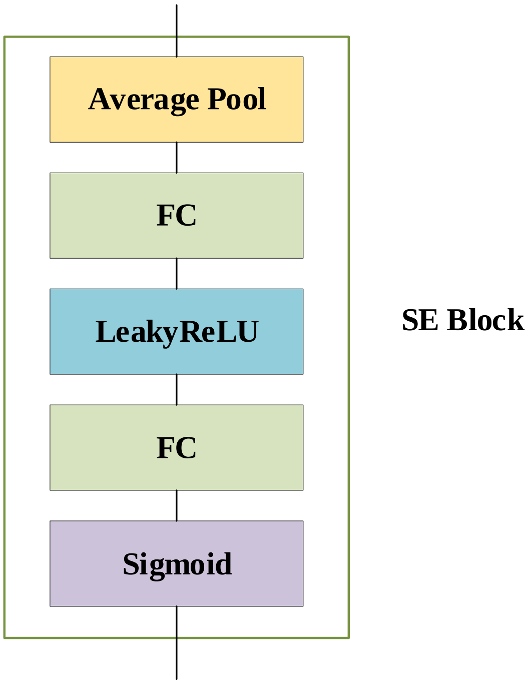

Figure 3: SE block

3.3. SE Block

Squeeze-and-Excitation block [20], as shown in Figure 3, uses an adaptive mechanism to assign different weights to different channels of the feature map, enhancing important features and weakening less important ones. Assuming that the input feature map of the squeezing excitation block is $X \in R^{C\times H\times W}$, the squeezing excitation block first uses a global average pooling to compress the feature map into a channel descriptor z of size $C \times 1 \times 1$.Then, this channel descriptor is predicted for the importance of each channel through two fully connected layers. Specifically Represented as $Weight = \sigma\left(W_2\delta\left(W_1 z\right)\right)$, where δ represents the ReLU function,σ represents the Sigmoid function, $W_1 \in R^{\frac{C}{r}\times C}$, $W_2 \in R^{C\times\frac{C}{r}}, Weight \in R^{C\times 1\times 1}$. Finally, the obtained weight is excited onto the corresponding channel of the feature map, obtaining $U = X \times Weight, U \in R^{C\times H\times W}$. The working mechanism is shown in Figure 4.

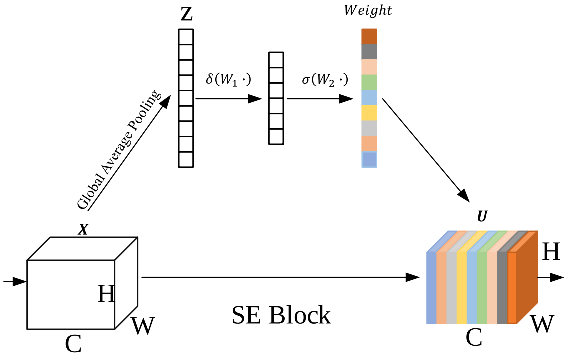

Figure 4: Squeeze-and-Excitation mechanism

3.4. Post-processing

For each audio file to be predicted, we only retain the predicted results that meet $Starttime - Endtime >= threshold * min\_duration$,where starttime is the start time of the detection event, endtime is the end time of the detection event, $min\_duration$ is the minimum duration of the first five given positive events in each audio file that needs to be detected, and threshold $\in [0, 1]$ set by as. We calculated the f-measure of the validation set under different thresholds and selected the threshold with the best performance as our submission option.

4. EXPERIMENTS AND RESULTS

Among various acoustic features, such as log-MEL, PCEN, MFCC, ΔMFCC and so on, we finally choose delta MFCC and PCEN as our input features because of their optimal performance.

During the training process, we calculate the f-measures of each epoch and select the checkpoint corresponding to the largest f-measure as the best checkpoint to predict the full version validation set for 2022 and evaluation set for 2023 under different thresholds. We choose the threshold corresponding to the highest f-measures as our submission option. At the same time, we will also use SE block as one of the submission options. Above all, we obtained

Table 1: Model Results. The F-measure (%) with different setting. SMP stands for splitting, merging, and padding.

No. of SE	SMP	Threshold	F-measure	Submission
None	False	0.1	**65.49**	System 1
None	False	0.05	63.06	System 2
1 after layer 3	False	0.3	63.94	System 3
1 after layer 1	False	0.15	62.14	System 4
4	False	0.15	51.43	×
None	True	0.1	39.41	×
1 after layer 2	False	0.15	55.43	×

the systems we submitted, and the specific performance is shown in Table1, and select the four systems with the highest f-measure to submit.

5. CONCLUSION

We have improved the prototype network on the basis of the baseline system, incorporating SE blocks into the model, and post-processing the obtained prediction results. Through experimental results, it can be found that our system performance has been greatly improved compared to baseline, with the highest f-measure reaching 65.491 on the validation set.

6. ACKNOWLEDGMENT

The authors would like to thank the organizers of this challenge, for proposing an interesting and novel problem which is relevent to our own research.This work was partially supported by the Ministry of Science and Technology of the People's Republic of China with the STI2030-Major Projects (No. 2021ZD0201900), the National Natural Science Foundation of China (No. 62227807 and 62272044), the Teli Young Fellow Program from the Beijing Institute of Technology, China, and the Grants-in-Aid for Scientific Research (No. 20H00569) from the Ministry of Education, Culture, Sports, Science and Technology (MEXT), Japan.

7. REFERENCES

[1] E. G. Miller, N. E. Matsakis, and P. A. Viola, "Learning from one example through shared densities on transforms," *Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No.PR00662)*, vol. 1, pp. 464–471 vol.1, 2000.

[2] B. M. Lake, R. Salakhutdinov, J. Gross, and J. B. Tenenbaum, "One shot learning of simple visual concepts," *Cognitive Science*, vol. 33, 2011.

[3] G. R. Koch, "Siamese neural networks for one-shot image recognition," Master's thesis, University of Toronto, 2015.

[4] Y. Wang, Q. Yao, J. T.-Y. Kwok, and L. M. shuan Ni, "Generalizing from a few examples: A survey on few-shot learning," *arXiv: Learning*, 2019.

[5] H. Chen, S. Shao, Z. Wang, Z. Shang, J. Chen, X. Ji, and X. Wu, "Bootstrap generalization ability from loss landscape perspective," in *ECCV Workshops*, 2022.

[6] A. Mesaros, T. Heittola, T. Virtanen, and M. D. Plumbley, "Sound event detection: A tutorial," *IEEE Signal Processing Magazine*, vol. 38, pp. 67–83, 2021.

[7] J. Snell, K. Swersky, and R. S. Zemel, "Prototypical networks for few-shot learning," in *NIPS*, 2017.

[8] M. Anderson and N. Harte, "Bioacoustic event detection with prototypical networks and data augmentation," *arXiv preprint arXiv:2112.09006*, 2021.

[9] D. Yang, H. Wang, Z. Ye, and Y. Zou, "Few-shot bioacoustic event detection= a good transductive inference is all you need," DCASE2021 Challenge, Tech. Rep, Tech. Rep., 2021.

[10] T. Tang, Y. Liang, and Y. Long, "Two improved architectures based on prototype network for few-shot bioacoustic event detection," DCASE2021 Challenge, Tech. Rep, Tech. Rep., 2021.

[11] M. Lasseck, "Acoustic bird detection with deep convolutional neural networks," in *Workshop on Detection and Classification of Acoustic Scenes and Events*, 2018.

[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," in *Interspeech*, 2019.

[13] H. Liu, X. Liu, X. Mei, Q. Kong, W. Wang, and M. . Plumbley, "Surrey system for dcase 2022 task 5: Few-shot bioacoustic event detection with segment-level metric learning," *ArXiv*, vol. abs/2207.10547, 2022.

[14] E. Hoffer and N. Ailon, "Deep metric learning using triplet network," in *International Workshop on Similarity-Based Pattern Recognition*, 2014.

[15] A. V. Patil and P. Rabha, "A survey on joint object detection and pose estimation using monocular vision," *ArXiv*, vol. abs/1811.10216, 2018.

[16] W. Ge, W. Huang, D. Dong, and M. R. Scott, "Deep metric learning with hierarchical triplet loss," in *European Conference on Computer Vision*, 2018.

[17] 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," *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 776–780, 2017.

[18] V. Lostanlen, J. Salamon, A. Farnsworth, S. Kelling, and J. P. Bello, "Robust sound event detection in bioacoustic sensor networks," *PLoS ONE*, vol. 14, 2019.

[19] J. L. Roux, S. Wisdom, H. Erdogan, and J. R. Hershey, "Sdr – half-baked or well done?" *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 626–630, 2018.

[20] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-excitation networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, pp. 2011–2023, 2017.