LIF-PROTONET: PROTOTYPICAL NETWORK WITH LEAKY INTEGRATE-AND-FIRE NEURON AND SQUEEZE-AND-EXCITATION BLOCKS FOR BIOACOUSTIC EVENT DETECTION

Technical Report

Mengkai Sun[1,2]*, Haojie Zhang*[1,2]*, Kun Qian*[1,2,*]*, and Bin Hu*[1,2,*]

[1] Key Laboratory of Brain Health Intelligent Evaluation and Intervention, Ministry of Education (Beijing Institute of Technology), P. R. China
[2] School of Medical Technology, Beijing Institute of Technology, P. R. China
{smk, zhj, qian, bh}@bit.edu,cn

ABSTRACT

In this technical repot, we describe our submission system for DCASE2024 Task5: Few-shot Bioacoustic Event Detection. We propose a metric learning method to construct a novel prototypical network, based on Leaky Integrate-and-Fire Neuron 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 53.72 %. For the input feature, we use combination of per-channel energy normalization (PCEN) and delta mel-frequency cepstral coefficients (ΔMFCC), then the features were first transformed through Leaky Integrate-and-Fire Neuron to mimic brain function. Our system performs better than the baseline given by the officials, on the DCASE2024 task 5 validation set. Our final score reaches an f-measure of 55.49 %, outperforming the baseline performance.

***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, 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 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 such as monitoring the animal population through their vocalizations, where labeling the data may be costly to annotate.

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, inputting the 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 segment in each audio file is not fully used. So they proposed to use both positive segment and negative segment to construct the system, which outperformed the baseline by a large margin. In DCASE 2023, we presented our initial implantation with called SE-prototypical network [14]. Our system is based on their main idea and our previous work, and we propose a new metric learning architecture, called **LIF-prototypical network**, which can better utilize the information from different channels to improve the model performance and model generalization.

Metric learning [15, 16, 17] 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 challenge, most of the studies [8, 9, 10] only use the labeled positive 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, we do not use the AudioSet [18]. We also have studied different audio features to choose the best feature for this task, including log-mel spectrogram (MEL), per-channel energy normalization (PCEN) [19], 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 will be organized as follows: Section 2 provides 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 discusses the difficulties we met during the experiment. Section 6 summarizes this work and provides a conclusion.

Corresponding authors: Kun Qian and Bin Hu.

2. SYSTEM OVERVIEW

2.1. Dataset

Challenge official dataset DCASE 2024 task 5 dataset contains a training set, an validation set from the development set, and an evaluation set. The training and validation set 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.

2.2. Model Architecture

The original baseline system contains a encoder, made up of 4 ConvBlock, each of which contains a Conv2d block, a BatchNorm2d layer, ReLU function and a Maxpool2d. The newly revised baseline system is constructed on the basis of ResNet framework, which also contains 4 Basic Block, and uses a downsample 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 in later Section 3. The whole network architecture is like Figure 2. We use several SE block to enhance the important feature in order to get better performance. The more details about the architecture will be introduced in Section 4.

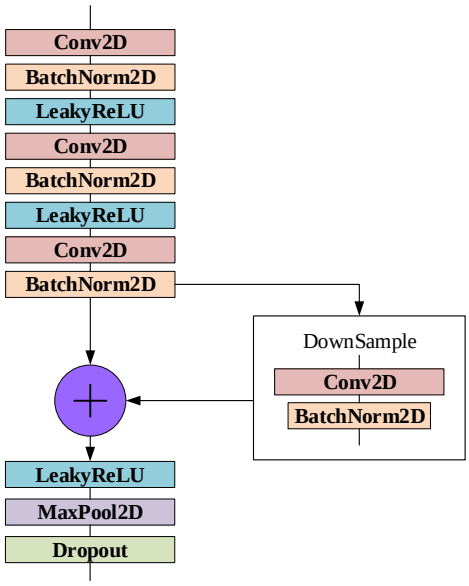

Figure 1: Basic Block

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 2024, we calculate the f-measure of the Full version validation set in 2023.

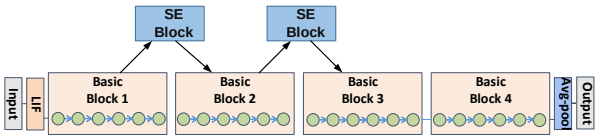

Figure 2: Network Architecture

3. METHOD

3.1. Feature extraction

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

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

3.2. Leaky Integrate-and-Fire Neuron

The Leaky Integrate-and-Fire (LIF) neuron model is a cornerstone of computational neuroscience, providing a simplified yet insightful representation of neuronal dynamics. This model is pivotal for understanding the fundamental mechanisms of neuronal behavior and synaptic integration. In this article, we explore the mathematical formulation, biophysical interpretation, and applications of the LIF neuron model in both theoretical and practical contexts. The core equation is given by:

$$\tau_m \frac{dV(t)}{dt} = -(V(t) - V_{rest}) + R_m I(t) \quad (1)$$

where τ_m is the membrane time constant, V_{rest} is the resting membrane potential, $R_m I(t)$ is the membrane resistance, and $I(t)$ is the input current. When the membrane potential τ_m reaches a threshold, the neuron fires an action potential (spike) and the membrane potential is reset to a specified value V_{rest}.

3.3. 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 [20] value.

3.4. SE Block

Squeeze-and-Excitation block [21], as shown in Figure 4, 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 3.

3.5. Post-processing

4. EXPERIMENTS AND RESULTS

Among various acoustic features, such as log-mel spectrogram(MEL), per-channel energy normalization(PCEN) [19], mel-frequency cepstral coefficients (MFCC), delta-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

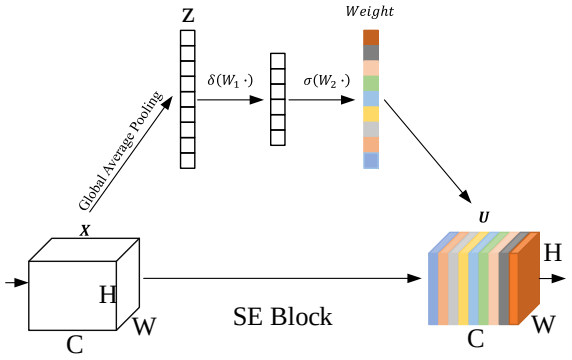

Figure 3: Squeeze-and-Excitation

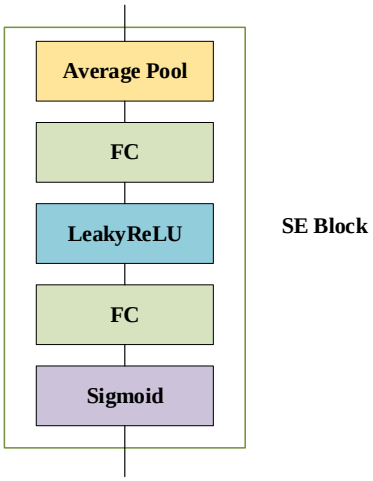

Figure 4: SE block

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 the four systems we submitted, and the specific performance is shown in Table 1.

5. DISCUSSION

Compared to the challenge in 2023, the data volume of training set has a huge increment, which makes the experiment more difficult due to the lack of high performance computing devices.

6. 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

Table 1: Submission Overview

No. of SE	LIF Beta	Threshold	F-measure	Submission
None	0.85	0.15	51.43	1
1 after layer 1	0.9	0.15	53.72	2
2	0.85	0.16	51.83	3
1 after layer 3	0.9	0.15	**55.49**	4

improved compared to baseline, with the highest f-measure reaching 55.49 % on the validation set.

7. ACKNOWLEDGMENT

The authors would like to thank the organisers of this challenge, for proposing an interesting and novel problem which is relevent to our own research. This work was partially supported by the National Natural Science Foundation of China (Nos. 62272044 and 62227807), the National Key R&D Program of China (No. 2023YFC2506804), the Ministry of Science and Technology of the People's Republic of China with the STI2030-Major Projects (Nos. 2021ZD0201900 and 2021ZD0200601), and the Teli Young Fellow Program from the Beijing Institute of Technology, China.

8. REFERENCES

[1] E. G. Miller, N. E. Matsakis, and P. A. Viola, "Learning from one example through shared densities on transforms," *Proc. CVPR, Hilton Head, SC, USA*, 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, Tel Aviv, Israel*, 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 *Proc. NeurIPS, Long Beach, CA, USA*, 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 DCASE*, 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 *Proc. Interspeech, Graz, Austria*, 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] J. Liu, Z. Zhou, M. Sun, X. Kele, K. Qian, and H. Bian, "Seprotonet: Prototypical network with squeeze-and-excitation blocks for bioacousti c event detection," DCASE2023 Challenge, Tech. Rep, Tech. Rep., 2023.

[15] E. Hoffer and N. Ailon, "Deep metric learning using triplet network," in *Proc. SIMBAD, Copenhagen, Denmark*. Springer, 2015, pp. 84–92.

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

[17] W. Ge, W. Huang, D. Dong, and M. R. Scott, "Deep metric learning with hierarchical triplet loss," in *Proc. ECCV, Munich, Germany*, 2018.

[18] 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," *Proc. ICASSP, New Orleans, LA, USA*, pp. 776–780, 2017.

[19] 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.

[20] J. L. Roux, S. Wisdom, H. Erdogan, and J. R. Hershey, "Sdr – half-baked or well done?" *Proc. ICASSP, Calgary, Alberta, Canada*, pp. 626–630, 2018.

[21] 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.