PROTOTYPICAL NETWORK FOR BIOACOUSTIC EVENT DETECTION VIA I-VECTORS

Technical Report

Hao Cheng, Chenguang Hu, Miao Liu,

Beijing Institute of Technology, School of Information and Electronics, Beijing, China,
alex_chenghao@163.com, {3220200551, 3120200795}@bit.edu.cn

ABSTRACT

In this technical report, we present our system for the task 5 of Detection and Classification of Acoustic Scenes and Events 2021 (DCASE2021) challenge, i.e. few-shot bioacoustic event detection. First, per-channel energy normalization (PCEN) and i-vectors are extracted as features. In order to improve the diversity of original audio, some data augmentation methods are adopted, for example, specaugment. Then, the prototypical network with convolutional neural networks (CNN) is used for few-shot detection. Finally, we use aforementioned features as inputs to train our CNN model. We evaluate the proposed systems with overall F-measure for the whole of the evaluation set, and our best F-measure score on the validation set is 46.28.

Index Terms— DCASE, few-shot bioacoustic event detection, PCEN, i-vectors, prototypical networks

1. INTRODUCTION

Bioacoustic event detection in audio is an important task for automatic wildlife monitoring, as well as in citizen science and audio library management [1]. Bioacoustic event detection is a very common required first step before further analysis, and makes it possible to conduct work with large datasets (e.g. continuous 24h monitoring) by filtering data down to regions of interest. Few-shot learning is a highly promising paradigm for scarce bioacoustic event detection. For the main assessment, we will use the F measure of detection performance.

In previous studies, the prototypical network as classifiers have recently shown improved performances over established methods in few-shot acoustic event detection [2]. And CNN has provided state-of-the-art results on various polyphonic sound event detection and audio tagging tasks [3].

In our proposed system, we used SpecAugment as one of data augmentation methods in few-shot bioacoustic event detection. Then, we extract PCEN and i-vector from the bioacoustic audio. Finally, we trained a prototypical network to overcome the difficulties of few shot problems.

The rest of the paper is organized as follows. In section 2, the dataset and features used in proposed system is described. In section3, we interpret the prototypical network and the corresponding configuration. Experiment result is presented in Section 4. Section 5 concludes our work.

2. DATASET AND FEATURES

2.1. Dataset

The development dataset [4] for task 5 consists of multi-class animal (mammal and bird) audio files. As the short-est event class is estimated to have a duration of 150 milliseconds [5], we chose 150 milliseconds as the model-ing unit. Audio recordings are down-sampled to a sampling rate of 22050Hz. We used the Librosa library to gener-ate the acoustic features. Similar to [6], we automatically construct a set of negative examples for inference, and adopt the inference-time data augmentation method to generate more positive examples without increasing the cost of manual marking. The query set is comprised of all audio clips after the fifth annotation.

2.2. Features

2.2.1. PCEN

In real world audio recording, especially outdoors, there are usually multiple sources. Recently, Per-channel energy normalization (PCEN) [7] has been proposed as an alter-native to MFCC, which aims to whiten the background of acoustic recordings and improve the robustness to channel distortion through temporal integration, adaptive gain control, and dynamic range compression.

2.2.2. Specaugment

We use SpecAugment [8] as our data augmentation method. SpecAugment, a simple data augmentation method, is applied to the feature inputs of a neural network. The augmentation policy consists of warping the features, masking blocks of frequency channels and masking blocks of time steps. In oursystems, SpecAugment is applied to the PCEN features using frequency masking and time masking. The frequency mask can improve the robustness of our systems to frequency distortion of audios [8]. Time masking is applied in the time domain, which is similar to frequency masking.

2.2.3. I-vector

In principle, we use the same i-vector [9] extraction pipeline as kaldi. A Universal Background Model (UBM) is first trained on the MFCCs from the training set. This UBM is then used to learn the i-vector space known as Total Variability Space (TVS). Using UBM and TVS, i-vectors are extracted from the development and test set. In our system, We repeat the ivector according to the time scale, and then splice the ivector and PCEN according to the frame as the input of the model.

Table 1: Network architectures of our system.

Input	bs × 1 × 17 × (128 + 128)
Prototypical network	conv,3×3@128 Batchnorm + Relu Maxpool 2×2 Dropout 0.4
	conv,3×3@128 Batchnorm + Relu Maxpool 2×2 Dropout 0.4
	conv,3×3@128 Batchnorm + Relu Maxpool 2×2 Dropout 0.4
	conv,3×3@128 Batchnorm + Relu Maxpool 2×2 Dropout 0.4
Output	bs × 2048 (reshaped)

3. PROTOTYPICAL NETWORKS

Meta-learning is often used in face of the problem of few-shot classification, where only limited exampled data is provided and a classifer must generalize to given classes. The idea of meta-learning can be summarized as "learn to learn", which is a classifer can use training tasks to train itself to optimize the function for mapping the data into intended label. As miraculously as it sounds, there has been a few methods to implement this idea. And prototypical network is a classic paradigm of meta-learning.

3.1. Structure

Prototypical network [10] consists of a feature extractor and a basic classifier. The classifier simply uses the euclidean distance of features, which can be seen as the similarity of features, to distinguish different types of data, so the performance of the network heavily depends on the generalization ability of the feature extractor. If the network is already trained with different few-shot classification tasks and as a result has a strong feature extractor. When given a new set of limited labeled data, the network will be able to recognize new classes by extracting the feature of labeled data as "prototype" and comparing the similarity of other features to the prototype. We build a Prototypical network based on CNN. The specific structure is shown in Table 1. The input is the feature after the splicing of PCEN and i-vectors.

3.2. Training

The network uses training tasks to obtain the ability of optimizing the mapping function between data and label. More specifically, the network uses different few-shot classfication tasks to optimize its feature extractor. When creating the training tasks, we should manually split the task into support set and query set, treating a training task as a separate dataset. Ideally, the network should perform well after several epochs of training. With a specific task, we only compute the cross entropy loss from the query set to optimize the network. Our goal is to train a strong feature extractor, and we will use different supplementary acoustic feature to boost the performance of the network.

4. EXPERIMENTAL RESULTS

4.1. Experiment setting

For few-shot setup, we experimented on 5-shot and 5-way setting. During the training, back-propagation and Adam optimizer with learning rate of 0.0001 are used. The total epoch is 15. During the evaluating, we set query batch size as 8 and negative set batch size as 16.

4.2. Experiment results

In this section, Table 2 shows the results of our systems on the validation set. We can see that the model that uses both PCEN and i-vector features achieves the best results, which means that i-vector features help to improve the accuracy of the few shot sound event detection system.

Table 2: The results of F-measure score on the evaluation dataset.

Features	Augmentation	P	R	F
PCEN	-	37.70%	31.57%	34.3%
PCEN	Specaug	45.76%	44.17%	44.96%
PCEN+i-vector	Specaug	45.96%	46.64%	46.28%

5. CONCLUSIONS

In this technical report, we propose using CNN-based prototypical network for few-shot bioacoustic event detection task. We use spliced PCEN and i-vector features and apply data enhancement methods such as specaugment to improve model performance At last, we get 46.28 under F-measure score on the validation set.

6. REFERENCES

[1] http://dcase.community/challenge2021/.

[2] B. Shi, M. Sun, K. C. Puvvada, C.-C. Kao, S. Matsoukas, and C. Wang, "Few-shot acoustic event detection via meta learning," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 76–80.

[3] Q. Kong, Y. Cao, T. Iqbal, Y. Wang, W. Wang, and M. D. Plumbley, "Panns: Large-scale pretrained audio neural networks for audio pattern recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2880–2894, 2020.

[4] V. Morfi, D. Stowell, V. Lostanlen, A. Strandburg-Peshkin, L. Gill, H. Pamula, D. Benvent, I. Nolasco, S. Singh, S. Sridhar, M. Duteil, and A. Farnsworth, "DCASE 2021 Task 5: Few-shot Bioacoustic Event Detection Development Set," Feb. 2021. [Online]. Available: https://doi.org/10.5281/zenodo.4543504

[5] J. LeBien, M. Zhong, M. Campos-Cerqueira, J. P. Velev, R. Dodhia, J. L. Ferres, and T. M. Aide, "A pipeline for identification of bird and frog species in tropical

soundscape recordings using a convolutional neural network," *Ecological Informatics*, vol. 59, p. 101113, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1574954120300637

[6] Y. Wang, J. Salamon, N. J. Bryan, and J. Pablo Bello, "Few-shot sound event detection," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 81–85.

[7] V. Lostanlen, J. Salamon, M. Cartwright, B. McFee, A. Farnsworth, S. Kelling, and J. P. Bello, "Per-channel energy normalization: Why and how," *IEEE Signal Processing Letters*, vol. 26, no. 1, pp. 39–43, 2019.

[8] 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," *Interspeech 2019*, Sep 2019. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2019-2680

[9] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011.

[10] J. Snell, K. Swersky, and R. S. Zemel, "Prototypical networks for few-shot learning," *CoRR*, vol. abs/1703.05175, 2017. [Online]. Available: http://arxiv.org/abs/1703.05175