Final Project: Few-Shot Detection of Bioacoustic Events

Vicente José Escarigo Miranda
UAM, Universidad Autonoma de Madrid

***Abstract*—Few-shot learning is a promising paradigm for bioacoustic sound event detection, where only a small number of labeled examples are available for a target species or call type.**

This report proposes a lightweight model for DCASE 2023 Task 5 (Few-shot Bioacoustic Event Detection). The method follows a general pipeline that converts raw audio into time–frequency representations, extracts frame-level embeddings with a shared encoder, performs support-guided scoring of query audio, applies temporal post-processing to form event predictions, and evaluates performance using the official event-based F-measure based on IoU and bipartite matching.

The goal of this project is to understand the task constraints in depth and deliver a reproducible baseline system that can be extended with alternative modeling choices.

***Index Terms*—few-shot learning, bioacoustics, sound event detection, embeddings, prototypical methods, event-based evaluation**

I. Introduction

Few-Shot Learning (FSL) is a machine learning approach where a model is required to adapt to new, previously unseen categories using only a small number of labeled examples. This technique is especially valuable in scenarios where data collection or annotation is expensive or very time consuming. One application of this is Sound Event Detection (SED), which involves identifying the start and end points of specific sounds within an audio recording. In fields like bioacoustics, recordings often span long durations with only a few relevant sound events, making manual labeling especially challenging. FSL offers a solution by detecting these events using minimal annotated data. The DCASE Task 5 challenge exemplifies this problem under strict constraints: each file provides only five positive events as support, evaluation ignores everything before the fifth event, and each file must be treated independently [1]. These rules make the task substantially harder than standard supervised SED and emphasize issues such as background noise, variable event durations, and domain mismatch across recording sources.

This project builds on the official DCASE 2023 prototypical network baseline as a reference point [2] and experiments on how far it can be improved without heavy architectures or external data. The focus is therefore on feasible, secondary improvements such as feature representation, adaptive windowing, negative sampling strategy, transductive refinement, and post-processing.

Although the 2024 edition expands the validation set and updates the official baseline, the 2023 edition uses the same development data and baseline definition used across the 2022–2023 technical reports. I therefore use the 2023 edition to keep the reference baseline fixed and to attribute performance changes to the proposed modifications rather than to a shifted validation split or baseline update.

II. State of the Art

Prototypical Networks introduced an important development approach to few-shot classification through metric learning, where the model is trained to map inputs into an embedding space. In this space, classification is done by measuring the distance to prototype representations of each class, resulting in effective generalization from just a few labeled examples [2]. In 2022, most Task 5 systems stayed within this template, improving the encoder, segmentation strategy, and post-processing [3], [4]. By 2023, the best results came from frame-level fine-tuning and auxiliary objectives (e.g., multi-task branches) and from contrastive pretraining on the official data [5], [6]. In 2024, the trend continued toward stronger backbones and pretraining (e.g., U-Net variants and AAPM-style encoders), while prototypical methods were still strongly present in modified form through better negatives, attention mechanisms, or hybrid front-ends [7].

III. Description of the Proposed Method

A. Overview

Following a lightweight prototypical-network setup [2], audio is converted to PCEN features with delta-MFCCs [8], [9] and embedded by a shallow ResNet encoder [10]. The five positives define a positive prototype, while negatives are taken from the gaps between positives as suggested by Tang et al. [4]. Query segments are scored by distance to the positive and negative prototypes and then converted into events based on sigmoid activation, followed by post-processing.

B. Feature extraction

All audio is resampled to 22.05 kHz and converted to 128-bin mel spectrograms ($n\_fft = 512$, hop=256). Per-Channel Energy Normalization (PCEN) is applied, an automatic gain control followed by compression that improves robustness to channel variability and background noise [8]. In parallel, I compute MFCCs and their temporal derivatives (delta-MFCC) to capture short-term spectral dynamics [9]. Liu et al. reported that PCEN concatenated with delta-MFCC gave the best average validation performance in DCASE2022 Task 5, so I adopt the same input representation [3].

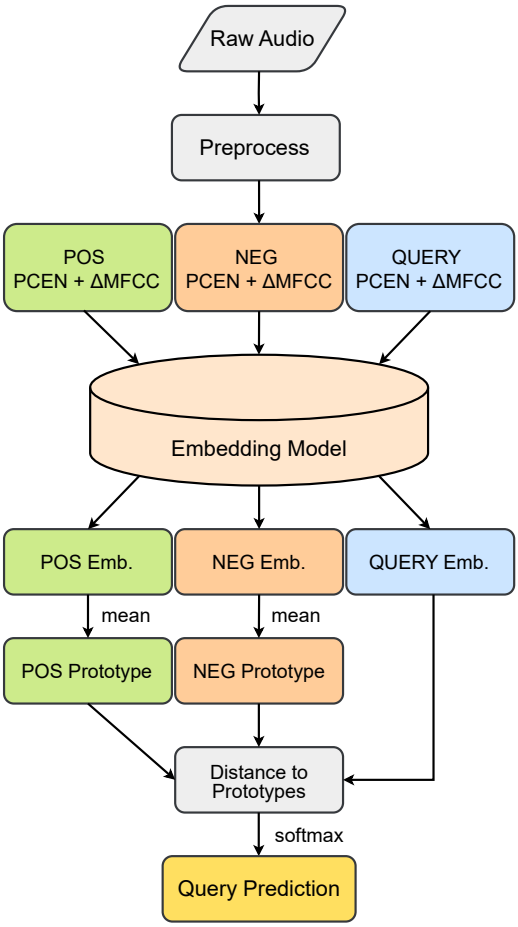

Fig. 1. Pipeline overview.

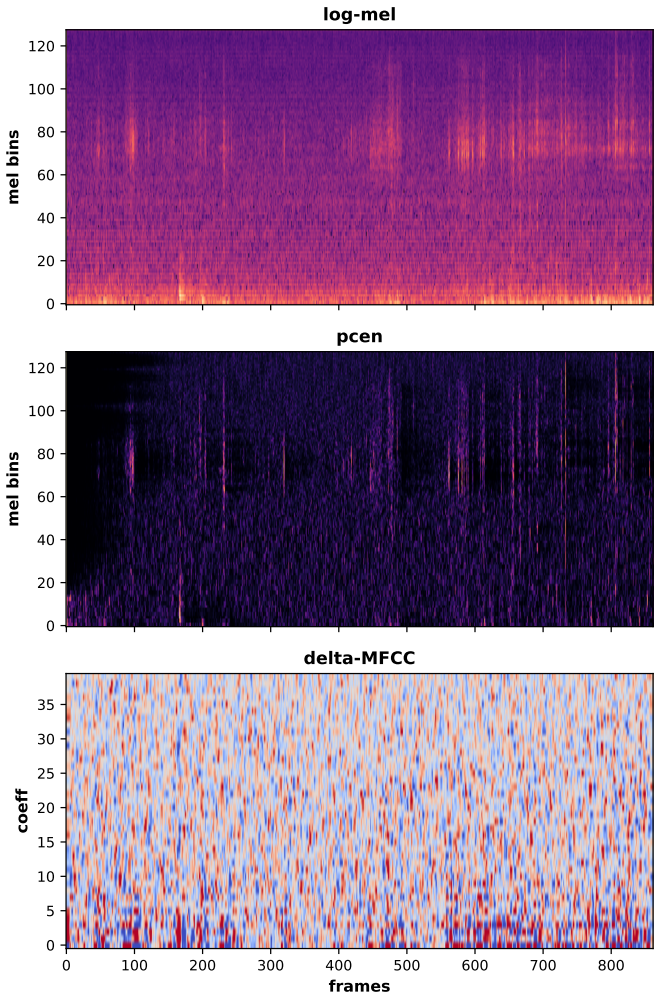

Fig. 2. Example input representations: log-mel, PCEN, and delta-MFCC.

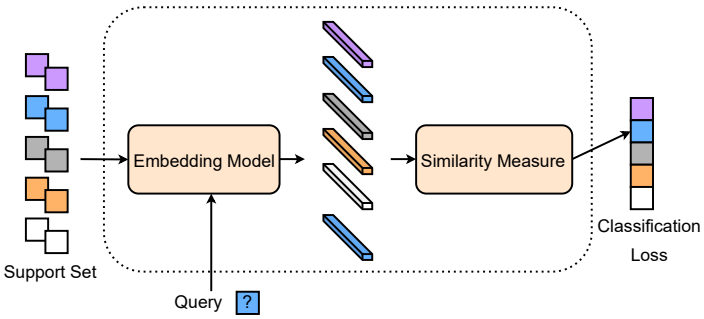

Fig. 3. Episodic training schematic (C-way (5), K-shot (2)).

C. Segmentation

For training, fixed-length segments of 0.2 s are ext with 0.1 s hop. At evaluation, segment length is adap each file using the mean duration of the five positive while the hop length is set as a fraction of that window et al. showed that adapting window length to the positive durations improves resolution for variable-length calls while keeping enough context for matching [4]. I follow that strategy.

D. Encoder and episodic training

I use a shallow ResNet encoder with residual blocks [10] and train it episodically in a 5-way, 5-shot setting, as in the prototypical learning paradigm [2]. Each episode samples positive and query segments, and the encoder is optimized with Adam ($\mathrm{lr} = 10^{-3}$, 2000 episodes per epoch).

E. Prototype scoring and transductive refinement

For each evaluation file, the positive prototype is the mean embedding of the five supports. The negative prototype is estimated from segments sampled in the gaps between positives; Tang et al. reported this reduces positive contamination

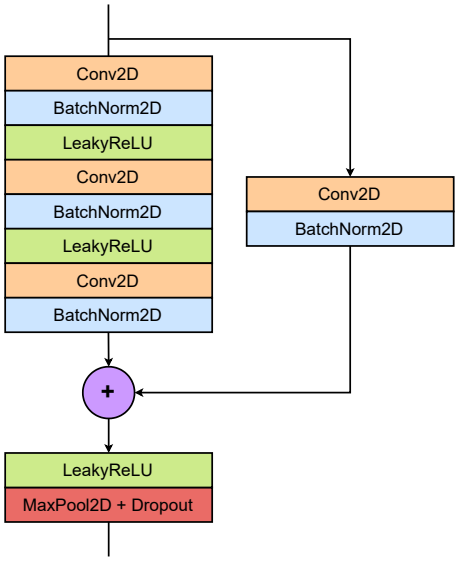

Fig. 4. Residual block used in the ResNet encoder.

TABLE I
ENCODER LAYER OVERVIEW. EACH RESIDUAL BLOCK CONTAINS THREE 3×3 CONVOLUTIONS.

Layers	Channels	Kernel size
ResidualBlock	64	3×3
ResidualBlock	128	3×3
ResidualBlock	64	3×3
AdaptiveAvgPooling	–	4×2

versus random negatives in dense-event files [4]. To reduce randomness, I sample negatives multiple times and average the resulting scores. I optionally perform a transductive refinement step that re-weights prototypes using query embeddings, similar in spirit to transductive information maximization in few-shot learning [11].

F. Post-processing

Frame-level probabilities are thresholded, then short detections are filtered using a minimum duration derived from the support events. Neighboring detections separated by very short gaps are merged. Hertkorn et al. showed that duration-based filtering and gap-merging can suppress bursts of false positives in Task 5 evaluation, so I use similar rules here [12].

IV. Experimental Setup

A. Dataset overview

Experiments use the DCASE 2023 Task 5 development set [1]. The Training_Set aggregates five sources (BV, HT, JD, MT, WMW) with multi-class annotations; the official totals are 174 recordings, 21 h, 47 classes, and 14 229 positive events. The Validation_Set in the dev package used here contains

TABLE II
TRAINING SET OVERVIEW (OFFICIAL STATISTICS).

Subset	Recordings	Duration	Classes	Events
BV	5	10 h	11	9 026
HT	5	5 h	3	611
JD	1	10 m	1	357
MT	2	1 h 10 m	4	1 294
WMW	161	4 h 40 m	26	2 941
Total	**174**	**21 h**	**47**	**14 229**

TABLE III
VALIDATION SET OVERVIEW (LOCAL DEV PACKAGE).

Subset	Recordings	Duration	Classes	Events
HB	10	2 h 38 m	1	712
PB	6	3 h	2	260
ME	2	20 m	2	62
Total	**18**	**5 h 57 m**	**5**	**1 034**

three subsets (HB, PB, ME) totaling 18 recordings and 1 034 positive events. Sampling rates range from 6 kHz to 48 kHz, and there is no class overlap between training and validation.

B. Protocol and annotations

Validation follows the 5-shot protocol: only the first five POS events per file are used as support, and all time before the end of the fifth event is ignored in evaluation. Each file is treated independently. UNK segments are excluded from metric computation.

C. Feature extraction

All audio is resampled to 22.05 kHz. I compute 128-bin mel spectrograms with $n\_fft = 512$ and hop=256, apply PCEN, and concatenate delta-MFCC features ($n\_mfcc = 40$). Training segments are 0.2 s with 0.1 s hop.

D. Model training

The encoder is a shallow ResNet with time-only pooling. I train episodically with 5-way, 5-shot batches and 2000 episodes per epoch for 15 epochs. The optimizer is Adam with $\mathrm{lr} = 10^{-3}$ and a StepLR schedule (step size 10, gamma 0.5). The best checkpoint is selected by validation accuracy.

E. Evaluation configuration

At evaluation, segment length is adapted per file using the maximum positive duration (in frames) with a cap of 20 frames; the hop is set to half the segment length. Negative prototypes are built from gap segments between positives. We sample 30 negatives and average over 6 iterations. Transductive refinement uses 2 steps (temperature 1.0, query weight 0.1). Post-processing applies a fixed threshold of 0.6, removes events shorter than 0.2 of the average positive length, and merges gaps shorter than 0.05 of that length.

F. Evaluation metric

I use the official event-based, macro-averaged F-measure with IoU and bipartite matching. The metric ignores the support region (up to the fifth POS event).

G. Reproducibility

All runs fix the train and eval random seeds to 0. Each run writes a ‘config_snapshot.yaml‘ to its output folder to preserve the exact settings.

V. Results

Results will be reported using the official event-based F-measure, alongside precision and recall for diagnostic analysis.

VI. Discussion and Conclusion

This project proposes a reproducible baseline framework for DCASE 2023 Task 5 that emphasizes correctness, modularity, and computational efficiency.

By implementing the full pipeline from audio to event predictions and official evaluation, the project will establish a strong foundation for exploring alternative encoders and few-shot scoring strategies while remaining aligned with the task constraints.

References

[1] D. Challenge, “DCASE 2023 Task 5,” https://dcase.community/challenge2023/task-few-shot-bioacoustic-event-detection#development-set.

[2] J. Snell, K. Swersky, and R. S. Zemel, “Prototypical Networks for Few-shot Learning,” Jun. 2017.

[3] H. Liu, X. Liu, X. Mei, Q. Kong, W. Wang, and M. D. Plumbley, “SURREY SYSTEM FOR DCASE 2022 TASK 5: FEW-SHOT BIOACOUSTIC EVENT DETECTION WITH SEGMENT-LEVEL METRIC LEARNING,” 2022.

[4] J. Tang, X. Zhang, T. Gao, D. Liu, X. Fang, J. Pan, Q. Wang, J. Du, K. Xu, and Q. Pan, “FEW-SHOT EMBEDDING LEARNING AND EVENT FILTERING FOR BIOACOUSTIC EVENT DETECTION,” 2022.

[5] G. Yan, R. Wang, L. Zou, J. Du, Q. Wang, T. Gao, and X. Fang, “MULTI-TASK FRAME LEVEL SYSTEM FOR FEW-SHOT BIOACOUSTIC EVENT DETECTION,” 2023.

[6] I. Moummad, R. Serizel, and N. Farrugia, “SUPERVISED CONTRASTIVE LEARNING FOR PRE-TRAINING BIOACOUSTIC FEW-SHOT SYSTEMS,” 2023.

[7] X. Deng, Y. Sun, K. Xu, and Y. Dou, “Wei Liu1, Hy Liu1, Fl Lin1, Hs Liu1, Tian Gao1, Xin Fang1, Jh Liu,” 2024.

[8] Y. Wang, P. Getreuer, T. Hughes, R. F. Lyon, and R. A. Saurous, “Trainable Frontend For Robust and Far-Field Keyword Spotting,” Jul. 2016.

[9] M. Hossan, S. Memon, and M. Gregory, “A novel approach for MFCC feature extraction,” Jan. 2011, pp. 1–5.

[10] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Las Vegas, NV, USA: IEEE, Jun. 2016, pp. 770–778.

[11] M. Boudiaf, H. Kervadec, Z. I. Masud, P. Piantanida, I. B. Ayed, and J. Dolz, “Few-Shot Segmentation Without Meta-Learning: A Good Transductive Inference Is All You Need?” Mar. 2021.

[12] M. Hertkorn, “FEW-SHOT BIOACOUSTIC EVENT DETECTION: DON’T WASTE INFORMATION,” 2022.