FEW-SHOT BIOACOUSTIC EVENT DETECTION USING AN EVENT-LENGTH ADAPTED ENSEMBLE OF PROTOTYPICAL NETWORKS

John Martinsson[1,2*], *Martin Willbo*[1], *Aleksis Pirinen*[1],
Olof Mogren[1], *Maria Sandsten*[2]

[1]Computer Science, RISE Research Institutes of Sweden, Sweden
{john.martinsson, martin.willbo, aleksis.pirinen, olof.mogren }@ri.se
[2] Centre for Mathematical Sciences, Lund University, Sweden
maria.sandsten@matstat.lu.se

ABSTRACT

In this paper we study two major challenges in few-shot bioacoustic event detection: variable event lengths and false-positives. We use prototypical networks where the embedding function is trained using a multi-label sound event detection model instead of using episodic training as the proxy task on the provided training dataset. This is motivated by polyphonic sound events being present in the base training data. We propose a method to choose the embedding function based on the average event length of the few-shot examples and show that this makes the method more robust towards variable event lengths. Further, we show that an ensemble of prototypical neural networks trained on different training and validation splits of time-frequency images with different loudness normalizations reduces false-positives. In addition, we present an analysis on the effect that the studied loudness normalization techniques have on the performance of the prototypical network ensemble. Overall, per-channel energy normalization (PCEN) outperforms the standard log transform for this task. The method uses no data augmentation and no external data. The proposed approach achieves a F-score of 48.0% when evaluated on the hidden test set of the Detection and Classification of Acoustic Scenes and Events (DCASE) task 5.

Index Terms— Machine listening, bioacoustics, few-shot learning, ensemble

1. INTRODUCTION

The human-induced accelerated loss in biodiversity [1] has led to a need for automated and low-cost wildlife monitoring where machine learning is a promising way forward [2]. Passive acoustic monitoring (PAM) is becoming an important tool in ecology for monitoring animal populations through their vocalizations [3]. Annotating PAM data is costly and requires specific domain expertise which motivates research on few-shot learning for bioacoustic event detection [4]. The goal of few-shot bioacoustic event detection is to detect the onset and offset of animal vocalizations in sound recordings using only a few annotated examples.

Recent work has demonstrated that prototypical networks are a promising approach for few-shot sound event detection [5, 6, 7], but a remaining challenge is high variance in classification accuracy between models because of the small amount of training data. Recent work on audio classification and sound event detection has demonstrated promising results using ensembles [8, 9, 10]. Ensembles may be especially useful for the few-shot task due to the high variance in classification accuracy between models [11]. To the best of our knowledge, prior work on ensemble methods for few-shot sound event detection remains understudied and motivated by this we study the effect of using an ensemble of prototypical networks for few-shot bioacoustic event detection.

Another challenge in few-shot sound event detection is the high variability in event lengths for the different event classes [6]. The event lengths can range from milliseconds to multiple seconds, which necessitates methods capable of adapting to the task specific event lengths. Wang et al. [6] suggest that future work should look into adapting the context window to the few-shot task. A common approach is to use a model which can handle variable context windows, train using a fixed context window, and at test time adapt the context window to the few-shot task. In this work we propose choosing the embedding function as well as the context window based on the few-shot examples. The embedding function is chosen from a set of embedding functions trained on different context windows, thus acting as experts on certain event lengths. Another way to approach this problem is by using a proposal based method [12].

2. METHOD

In this section we present our method which is based on prototypical networks [13] and extended with an event-length adapted ensemble. We describe how each embedding function for the prototypical networks is trained and how the embedding functions are selected based on the few-shot examples to produce an ensemble prediction at test time. The full source code and instructions on how to reproduce the results can be found at: `https://github.com/johnmartinsson/few-shot-learning-bioacoustics`.

2.1. Training the embedding function

The goal is to learn an embedding function from the base training data, acting as a proxy task for the few-shot task. The base training data set consists of annotated sound recordings for 47 known event classes and one "unknown" event class. The set of sound event classes are disjoint between the base training data and the few-shot task. We are given the start and end times $\mathcal{A}_k = \{(s_i^k, e_i^k)\}_{i=1}^N$ of these classes, where (s_i^k, e_i^k) denotes the start and end time of sound event class k for annotation i. There is overlap in the annotations, i.e. two different sound events can occur (partially) simultaneously, and we therefore treat this as a multi-label problem. We model the

[*] Thanks to the Swedish foundation for strategic research for funding.

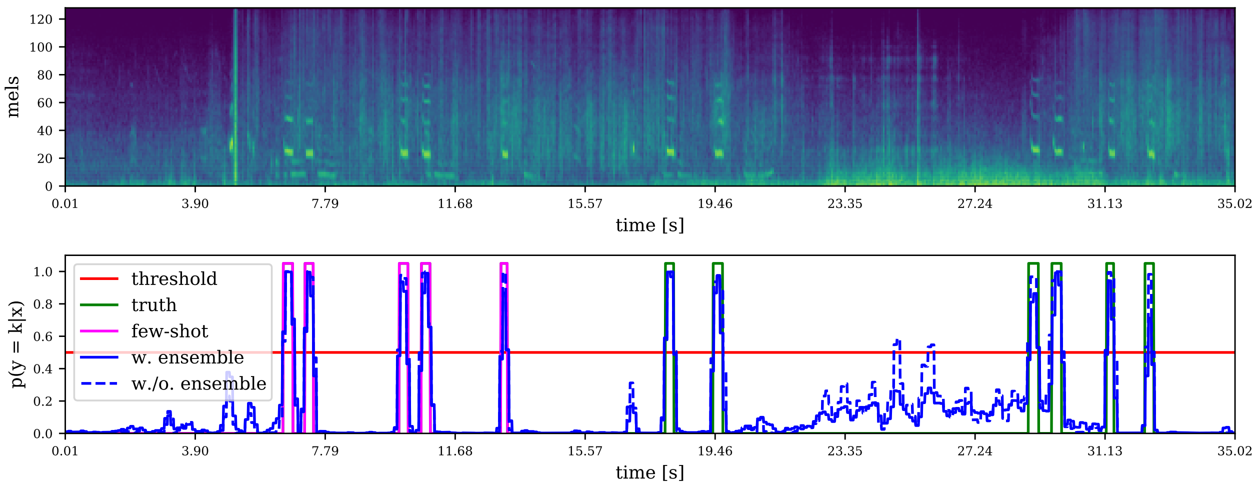


Figure 1: A log Mel spectrogram of part of a sound recording (top) and examples of predictions (bottom) from an ensemble prototypical network (solid blue line) and a prototypical network (dashed blue line) as well as the given few-shot examples (purple line) and remaining ground truth events (green line). The decision threshold τ is 0.5 (red line).

47 known sound event classes and the "unknown" sound event class identically, yielding a total of $K = 48$ classes.

We assume that a fixed length audio segment $x \in \mathbb{R}^T$, consisting of T consecutive audio samples, is fed to the embedding function $f_\theta^T : \mathbb{R}^T \rightarrow \mathbb{R}^M$ (see section 2.4 for further details), where $M \ll T$. We split the audio recordings into audio segments $x_i \in \mathbb{R}^T$ by sliding a window of size T with a hop size of $T/2$ over each recording. For each audio segment x_i, a target vector $y_i \in \{0,1\}^{K \times n}$ is derived. If $n = T$ it means that the target contains one label per audio sample. Choosing $n < T$ means that the temporal resolution for the target is reduced. The resulting dataset $\mathcal{D}_b = \{(x_i, y_i)\}_{i=1}^N$ defines the sound event detection task used to train the embedding function.

A prediction of the target classes for a given audio segment x_i is derived by $\hat{y}_i = h_\phi(f_\theta^T(x_i))$, where $h_\phi(\cdot)$ is a linear layer followed by an element-wise sigmoid activation function, and $f_\theta^T(\cdot)$ is a convolutional neural network where the first layer is a (non-learnable) time-frequency transform.

The loss function is the mean element-wise binary cross-entropy between the target y_i and the prediction $\hat{y}_i$, where the mean is taken over the class dimension K and the temporal dimension n.

For a fixed T, we train a set of C different embedding functions, parameterized as $\Theta = \{\theta_1, \dots, \theta_C\}$, each with different randomly initialized weights of the neural network, different training and validation splits of the base training data, and different time-frequency transforms in the first layer of the embedding function.

2.2. Prototypical network at test time

At test time we are given a sound recording and the $M = 5$ first event examples of the class of interest. We denote these $A_p = \{(s_i, e_i)\}_{i=1}^M$ and call them the *positive* sound events. We assume that the gaps between the positive event annotations are background noise and let $A_n = \{(e_i, s_{i+1})\}_{i=1}^{M-1}$ denote the start and end time of the $M - 1$ first *negative* sound events. We assume the likelihood of an annotator missing events to be low.

Let $l_i = e_i - s_i$ be the length of annotation i. If $l_i < T$ we "expand" the annotation with the $(T - l_i)/2$ preceding and subsequent audio samples to get an audio segment of length T, and if $l_i \geq T$ we do not expand. We then split this into segments of length T by sliding a window of size T over the signal with a hop size of $T/16$ (if expanded this will only result in one segment). Let S_p denote the set of positive audio segments derived from these annotated start and end times, and let S_n denote the set of negative audio segments. We use the embedding function f_θ^T and define the prototypes as

$$c_k = \frac{1}{|S_k|} \sum_{x \in S_k} f_\theta^T(x) \qquad (1)$$

and derive a pseudo-probability of audio segment x belonging to sound class k from the prototypical network by

$$p_\theta(y = k|x) = \frac{\exp(-d(f_\theta^T(x), c_k))}{\sum_{k'} \exp(-d(f_\theta^T(x), c_{k'}))}, \qquad (2)$$

where $k \in \{n, p\}$ and $d(f_\theta^T(x), c_k)$ denotes the Euclidean distance between the query $f_\theta^T(x)$ and the prototype c_k.

The query set S_q is derived by sliding a window of size T over the signal with a hop size of $T/2$. The reason for setting the hop size relative to T is that this means that we do equally many predictions for each audio sample in the validation recordings.

2.3. Our contributions

We now present the two main contributions of this paper: i) an event-length adapted embedding function for the few-shot task, and ii) using an ensemble of predictions.

Adapting the embedding function. We use the annotated positive events $A_p = \{(s_i, e_i)\}_{i=1}^M$ and compute the set of event lengths $L = \{e_i - s_i\}_{i=1}^M$. We choose $T^* \in \{T_1, 2^1T_1, 2^2T_1, 2^3T_1\}$ such that $\sqrt{(T - l_{\text{min}}/2)^2}$ is minimized, where l_{min} is the shortest event length in L.

We choose $T_1 = 2048$ which is 0.09 seconds at a sampling rate of 22050 Hz so that we can plausibly detect the shortest events in the few-shot validation set. We limit the amount of extra computation needed during training and the extra memory needed during inference by setting the maximum T to $2^3 T_1$.

Ensemble. Let $\Theta = \{\theta_i^{T^*}\}_{i=1}^{C}$ denote the set of parameters of C different prototypical network models adapted to the average event length of the few-shot task. Then we define

$$p_\Theta(y = k|x) = \frac{1}{C}\sum_{\theta\in\Theta} p_\theta(y = k|x) \tag{3}$$

as in [14], which can be viewed as a uniformly-weighted mixture of experts. We say that x belongs to the positive event class if $p_\Theta(y = p|x) > \tau$ and otherwise x belongs to the negative event class. This is done for every $x \in S_q$. Finally, if the query is classified as a positive event then the start and end time associated with that query is used as the predicted positive event timings.

2.4. Details of the embedding function

The embedding function consists of a time-frequency transform followed by a convolutional neural network, both of which are briefly described below.

Time-frequency transform. The first layer of the embedding function is a time-frequency transform. We use the Mel transform where the number of Mel bins is 128, the window size is roughly 25ms, and the hop size is half the window size. We either use the log transform as a loudness normalization or we use PCEN [15] with fixed parameters developed for speech audio or for bioacoustics as suggested in [16].

Convolutional neural network. The convolutional neural network used is an adapted version of the 10-layer residual neural network [17] used in the baseline for the challenge. Specifically, we i) add the classification head $h_\phi(\cdot)$ so that we can model the defined multi-label task, ii) use the same number of filters in every convolutional layer, and iii) reduce the max pooling along the time-dimension when audio segments are too short.

2.5. Evaluation metric

The method is evaluated by taking the harmonic mean over the F-scores for the different subsets in the evaluation sets. The F-score is computed by a bi-partite matching between the predicted and ground truth events, where the requirement for a match is an intersection-over-union (IoU) of at least 0.3 [4].

2.6. Post-processing

Since we get one prediction for each query audio segment, this limits the possible length of the prediction with this model. To solve this, we simply merge all overlapping predicted positive events into one detected event with a single start and end time.

A predicted positive event will only be considered to be a match with a true positive event during evaluation if they have an intersection-over-union (IoU) of at least 0.3. We therefore remove predictions which are shorter than $0.3 * l_{\text{avg}}$ or longer than $(1/0.3) * l_{\text{avg}}$, where l_{avg} is the average event length of the given five annotations. Since predictions of these lengths can on average not be matched with true events as the evaluation is defined.

Subset	Mean event length	Mean gap length	Mean density
HB	11.25 ± 3.11	6.12 ± 5.39	0.73 ± 0.12
ME	0.22 ± 0.03	1.40 ± 0.04	0.17 ± 0.02
PB	0.12 ± 0.08	59.89 ± 55.55	0.01 ± 0.02

Table 1: Few-shot validation data statistics.

3. DATA

We use the few-shot examples to compute the mean event length, the mean gap length, and the density of annotated sound events – see table 1. The few-shot validation set consists of three different subsets: HB, ME, and PB. The HB subset contains long events with low noise. The ME subset contains short events with low noise. The PB subset contains very short events with very high noise. The mean event length is defined as the mean length of the five annotated events; the mean gap length is defined as the mean length of the *unannotated* gaps between the five annotated events; and the density is the sum of the time of the five annotated events divided by the total time. A full description of the dataset can be found in [18].

4. EXPERIMENTS AND RESULTS

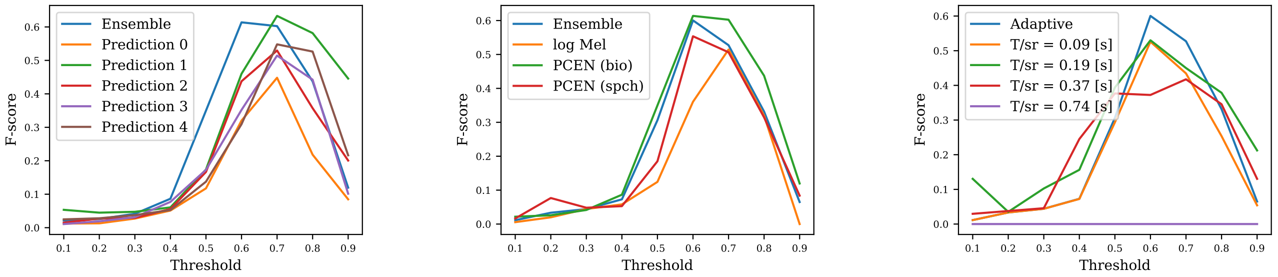
We have trained each embedding function on the described multi-label task on the base training data using the Adam [19] optimizer with a learning rate of 1e−3. The network is trained on a random split with 80% training data and validated on the remaining 20%. Each network in the ensemble is trained on a different random split. The training proceeds until we have observed no reduction in validation loss for the last 10 epochs and the model with the lowest validation loss is chosen as the final model. The temporal resolution of the targets have been fixed to $n = 16$, meaning that we have 16 targets for any given audio segment.

In figure 2a we compare the F-score achieved on the few-shot validation set when using an ensemble of five predictions with using each of these predictions by themselves. The time-frequency transform used is PCEN (bioacoustics). The achieved F-score by the ensemble is higher than the best of these individual predictions for $0.4 \leq \tau \leq 0.6$, and outperforms or matches the mean of them for other τ. We also note that the optimal τ is around 0.7 for the single predictions, and moves to 0.6 for the ensemble.

In figure 2b we compare the F-score of a five prediction ensemble for each time-frequency transform and compare this to an ensemble over all three time-frequency transform ensembles. We do not observe a significant increase in F-score when comparing the time-frequency ensemble to the ensemble using the PCEN (bioacoustic) time-frequency transform, but the time-frequency ensemble outperforms the ensemble using PCEN (speech) and log Mel transform. The optimal threshold τ varies around 0.6 to 0.7 for the ensembles using a single transform, and is at 0.6 for the time-frequency ensemble.

In figure 2c we compare the F-score achieved on the few-shot validation set when using the event-length adapted embedding functions in the ensemble with using any of the fixed $T \in \{T_1, 2^1T_1, 2^2T_1, 2^3T_1\}$. Adapting the embedding function increases performance from 53.0% (using best $T = 4096$) to 60.0% F-score for $\tau = 0.6$.

In table 2 we show an ablation study. Adapting the embedding function increases the F-score on average with 8.3 percentage points, and adding the ensemble increases the F-score an additional



(a) Ensemble over seeds, and training splits.

(b) Ensemble over time-frequency transforms.

(c) Adapted embedding function.

Figure 2: A comparison between: (a) an ensemble of five predictions using embedding functions trained on PCEN (bioacoustics) features with each of the individual predictions, (b) an ensemble of embedding functions trained and tested on log Mel, PCEN (bioacoustics), or PCEN (speech), with an ensemble of predictions over all three, and (c) the adaptive embedding function with using each of the fixed size embedding functions respectively (sr denotes the sample rate). All results in the figure are derived on the few-shot validation set.

Method	Ensemble	Adaptive	F-score
Ours	No	No	41.3 ± 3.8
Ours	No	Yes	49.6 ± 5.3
Ours	Yes	Yes	60.0

Table 2: An ablation study of our system on the few-shot validation set where we add adaptive embedding functions and ensemble.

System	External data	Augmentation	F-score
Baseline (TM)	No	No	12.3
Baseline (PN)	No	No	5.3
Ours [20]	No	No	48.0
Tang et al., [21]	No	No	62.1
Liu et al., [22]	Yes	Yes	48.2
Hertkorn [23]	No	No	44.4
Liu et al., [24]	Yes	Yes	44.3

Table 3: The final F-score evaluation on the hidden test set for the baselines provided by the challenge organizers: template matching (TM) and prototypical networks (PN), and the top five submissions for the challenge.

11.4 percentage points. We compare against a prototypical network using an embedding function (no ensemble) which has been trained on PCEN (speech) and a best performing fixed segment length of 4096. The F-score when no ensemble is performed is the average (and standard deviation) over each single network in the ensemble.

In table 3 we present the F-score from the final evaluation on the hidden test set from the challenge. We include information on whether or not the system uses data augmentation techniques or external datasets during training.

5. DISCUSSION AND CONCLUSIONS

In this section we will discuss our results and relate them to the baselines which were all developed concurrently to our work.

During development of this method we observed that random sampling of S_n, the set of negative examples does not work well for validation files with high event densities, which is why we chose to use the gaps between the first five annotated events instead. This observation was also made in concurrent work submitted to the challenge [21, 22, 24]

We further observed that a fixed audio segment size T resulted in poor predictive performance on the few-shot validation set in cases where event-lengths deviated from size T. Indicating the importance of adapting the embedding function.

We observed that the optimal threshold was different for the few-shot validation tasks and choosing a default value of $\tau = 0.5$ to be detrimental. However, finding an optimal threshold for the few-shot tasks is a difficult problem. Using an ensemble alleviates this issue by moving the optimal threshold closer to the default value.

The ensemble improves performance by correctly predicting most true positives, while reducing the number of false positives. This could intuitively be thought of as the ensemble being in agreement for true positive predictions, the average of which still yields a high pseudo-probability, while being in disagreement when predicting false positives, the average of which would be closer to 0.5. This effect can be seen in figure 1, where some of the false positives predicted when not using an ensemble (dashed blue line) are removed by using an ensemble of the predictions (solid blue line), leading to a reduction in false-positives.

The baselines in this study were all developed concurrently to our work. Tang et al., [21] propose using a frame-level cross-entropy loss function for training instead of episodic training as the proxy task. Our approach is similar when setting the temporal resolution n of the target vector to the number of frames in the time-frequency image. The effect of varying temporal resolutions n for the proxy task would be interesting to study in future work. Tang et al. [21] further propose an iterative training scheme to adapt their method to the few-shot task [21] where the unlabeled audio in the test files is iteratively classified and then used for training. Liu et al. [22] and Liu et al. [24] use transductive inference to better adapt to the evaluation set, and Hertkorn [23] studies the importance of choosing appropriate parameters for the used time-frequency transform.

In conclusion, we have shown that choosing the embedding function based on the event lengths will increase performance, and that false-positives can be reduced by an ensemble of predictions. We have also shown that out of the three time-frequency transforms we have studied, PCEN (bioacoustics) performs best, followed by PCEN (speech) and log Mel.

6. REFERENCES

[1] S. Díaz, J. Settele, E. S. Brondízio, H. T. Ngo, J. Agard, A. Arneth, P. Balvanera, K. A. Brauman, S. H. Butchart, K. M. Chan, A. G. Lucas, K. Ichii, J. Liu, S. M. Subramanian, G. F. Midgley, P. Miloslavich, Z. Molnár, D. Obura, A. Pfaff, S. Polasky, A. Purvis, J. Razzaque, B. Reyers, R. R. Chowdhury, Y. J. Shin, I. Visseren-Hamakers, K. J. Willis, and C. N. Zayas, "Pervasive human-driven decline of life on Earth points to the need for transformative change," *Science*, vol. 366, no. 6471, 2019.

[2] D. Tuia, B. Kellenberger, S. Beery, B. R. Costelloe, S. Zuffi, B. Risse, A. Mathis, M. W. Mathis, F. van Langevelde, T. Burghardt, R. Kays, H. Klinck, M. Wikelski, I. D. Couzin, G. van Horn, M. C. Crofoot, C. V. Stewart, and T. Berger-Wolf, "Perspectives in machine learning for wildlife conservation," *Nature Communications*, vol. 13, no. 1, pp. 1–15, 2022.

[3] R. Gibb, E. Browning, P. Glover-Kapfer, and K. E. Jones, "Emerging opportunities and challenges for passive acoustics in ecological assessment and monitoring," *Methods in Ecology and Evolution*, vol. 10, no. 2, pp. 169–185, 2019.

[4] V. Morfi, I. Nolasco, V. Lostanlen, S. Singh, A. Strandburg-Peshkin, L. Gill, H. Pamuła, D. Benvent, and D. Stowell, "Few-Shot Bioacoustic Event Detection: A New Task at the DCASE 2021 Challenge," *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2021 Workshop (DCASE2021)*, no. November, pp. 145–149, 2021.

[5] B. Shi, M. Sun, K. C. Puvvada, C. C. Kao, S. Matsoukas, and C. Wang, "Few-Shot Acoustic Event Detection Via Meta Learning," *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, vol. 2020-May, pp. 76–80, 2020.

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

[7] J. Pons, J. Serra, and X. Serra, "Training Neural Audio Classifiers with Few Data," *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, vol. 2019-May, pp. 16–20, 2019.

[8] D. Lee, S. Lee, Y. Han, and K. Lee, "Ensemble of Convolutional Neural Networks for Weakly-Supervised Sound Event Detection Using Multiple Scale Input," *DCASE 2017*, vol. 1, no. November, pp. 14–18, 2017. [Online]. Available: http://www.cs.tut.fi/sgn/arg/dcase2017/documents/challenge_technical_reports/DCASE2017_Lee_199.pdf

[9] L. Nanni, Y. M. Costa, R. L. Aguiar, R. B. Mangolin, S. Brahnam, and C. N. Silla, "Ensemble of convolutional neural networks to improve animal audio classification," *Eurasip Journal on Audio, Speech, and Music Processing*, vol. 2020, no. 1, 2020.

[10] L. Nanni, G. Maguolo, S. Brahnam, and M. Paci, "An ensemble of convolutional neural networks for audio classification," *Applied Sciences (Switzerland)*, vol. 11, no. 13, pp. 1–27, 2021.

[11] N. Dvornik, J. Mairal, and C. Schmid, "Diversity with cooperation: Ensemble methods for few-shot classification," *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2019-October, pp. 3722–3730, 2019.

[12] P. Wolters, C. Daw, B. Hutchinson, and L. Phillips, "Proposal-based Few-shot Sound Event Detection for Speech and Environmental Sounds with Perceivers," pp. 1–7, 2021. [Online]. Available: http://arxiv.org/abs/2107.13616

[13] J. Snell, K. Swersky, and R. Zemel, "Prototypical networks for few-shot learning," *Advances in Neural Information Processing Systems*, vol. 2017-Decem, pp. 4078–4088, 2017.

[14] B. Lakshminarayanan, A. Pritzel, and C. Blundell, "Simple and scalable predictive uncertainty estimation using deep ensembles," *Advances in Neural Information Processing Systems*, vol. 2017-December, pp. 6403–6414, 2017.

[15] Y. Wang, P. Getreuer, T. Hughes, R. F. Lyon, and R. A. Saurous, "Trainable frontend for robust and far-field keyword spotting," *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, no. 1, pp. 5670–5674, 2017.

[16] 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*, no. September, pp. 1–6, 2018. [Online]. Available: http://www.justinsalamon.com/uploads/4/3/9/4/4394963/lostanlen_pcen_spl2018.pdf

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," *Arxiv.Org*, vol. 7, no. 3, pp. 171–180, 2015. [Online]. Available: http://arxiv.org/pdf/1512.03385v1.pdf

[18] I. Nolasco, S. Singh, E. Vidana-Villa, E. Grout, J. Morford, M. Emmerson, F. Jensens, H. Whitehead, I. Kiskin, A. Strandburg-Peshkin, L. Gill, H. Pamula, V. Lostanlen, V. Morfi, and D. Stowell, "Few-shot bioacoustic event detection at the dcase 2022 challenge," 2022. [Online]. Available: https://arxiv.org/abs/2207.07911

[19] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," pp. 1–15, 2014. [Online]. Available: http://arxiv.org/abs/1412.6980

[20] J. Martinsson, M. Willbo, A. Pirinen, O. Mogren, and M. Sandsten, "Few-shot bioacoustic event detection using a prototypical network ensemble with adaptive embedding functions," Tech. Rep., 2022.

[21] 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," iFLYTEK Research Institute, Hefei, China, Tech. Rep., 2022.

[22] H. Liu, X. Liu, X. Mei, Q. Kong, W. Wang, and M. D. Plumbley, "Surrey system for DCASE 2022 task 5 : few-shot bioacosutic event detection with segment-level metric learning," University of Surrey, Surrey, United Kingdom, Tech. Rep., 2022.

[23] M. Hertkorn, "Few-shot bioacoustic event detection : don't waste information," ZF Friedrichshafen AG, Friedrichshafen, Germany, Tech. Rep., 2022.

[24] M. Liu, J. Zhang, L. Wang, J. Peng, and C. Hu, "Bit SRCB teams 's submission for DCASE2022 task5 - few-shot bioacoustic event detection," Beijing Institute of Technology, Beijing, China, Tech. Rep., 2022.