IMPACT OF TEMPORAL RESOLUTION ON CONVOLUTIONAL RECURRENT NETWORKS FOR AUDIO TAGGING AND SOUND EVENT DETECTION

Wim Boes, Hugo Van hamme

ESAT, KU Leuven, Belgium

ABSTRACT

Many state-of-the-art systems for audio tagging and sound event detection employ convolutional recurrent neural architectures. Typically, they are trained in a mean teacher setting to deal with the heterogeneous annotation of the available data.

In this work, we present a thorough analysis of how changing the temporal resolution of these convolutional recurrent neural networks — which can be done by simply adapting their pooling operations — impacts their performance. By using a variety of evaluation metrics, we investigate the effects of adapting this design parameter under several sound recognition scenarios involving different needs in terms of temporal localization.

Index Terms— sound recognition, audio tagging, sound event detection, temporal resolution

1. INTRODUCTION

Recently, the popularity of research into audio-related tasks has surged because of, among other reasons, multiple versions of the Detection and Classification of Acoustic Scenes and Events (DCASE) challenge [1, 2, 3, 4, 5, 6]. The latest editions encompassed six categories, each dealing with a distinct sound recognition problem. Subtask number 4 [7] consistently covered sound event detection — joint classification and temporal localization of auditory events — in domestic environments. Submitted systems were ranked utilizing measures which represent scenarios involving variable requirements with regard to the estimation of time boundaries.

The baseline [7] supplied for the fourth problem of the DCASE 2021 and 2022 challenges, a convolutional recurrent neural network trained using the mean teacher principle [8], was based on the second-ranking system [9] of task 4 of DCASE 2019. Its output temporal resolution was predefined and fixed.

Our submission [10] for the fourth subtask of the DCASE 2021 challenge [7] demonstrated that simply adapting the amount of pooling in this network, which is directly linked to its temporal resolution, can significantly impact its performance.

In this work, we provide a more thorough and complete analysis of this interesting finding. We substantially supplement the discussion in multiple ways, as outlined in the two paragraphs below.

Firstly, we examine more than the specific evaluation scenarios prescribed in task 4 of the DCASE 2021 and 2022 challenges. Particularly, we also investigate what happens when the temporal resolutions of convolutional recurrent networks are adapted in the context of audio tagging. In this situation, the goal of the models is to perform clip-level auditory event classification. Unlike for sound event detection, temporal localization is not required in this case.

Secondly, for sound event detection, we inspect what happens using multiple types of measures. More specifically, in addition to the intersection-based scores employed in subtask 4 of the DCASE 2021 and 2022 challenges, we also utilize segment-based and event-based metrics, which are commonly used in this subdomain of machine learning as well. This is further elaborated upon in Section 4.

To this end, the following approach is taken: We start with a suitable baseline, which forms the basis for many state-of-the-art architectures for both audio tagging and sound event detection, such as [11], [12] and [13]. We change the temporal resolution of this model by adapting its pooling operations. We then analyze the results obtained by this modified system in a variety of experimental configurations, representing different evaluation scenarios.

In Section 2, we expand upon the baseline system. Afterwards, in Section 3, we describe how the pooling operations of the considered architecture can be adapted to modify the temporal resolution of the model. Then, in Section 4, we elaborate upon the experimental setup. Next, in Section 5, we analyze the results of the performed experiments, and finally, we draw a conclusion in Section 6.

2. BASELINE

In this section, the baseline system, which is nearly the same as in [14], is elaborated upon. It is a slightly adapted version of the baseline supplied for task 4 of the DCASE 2021 and 2022 challenges [7].

2.1. Architecture

A schematic visualization of the baseline model is given in Figure 1.

The input to the baseline is a spectral map of an audio recording. The amount of frequency bins is predefined and set to 128.

The first component of the architecture is a convolutional neural network (CNN) made up of seven blocks. Each of those consists of the following five layers: a convolutional layer, a batch normalization layer [15], a ReLU activation layer, a dropout layer [16] with a dropout rate of 33%, and lastly, an average pooling layer.

All convolutional layers use a square kernel of size 3 and utilize a stride of 1. For the first three blocks, the number of output channels of the convolutional operations is equal to 16, 32 and 64 respectively. For the last four blocks, this number is equal to 128.

The hyperparameters of the pooling operations are given per model block in Table 1. The first and second numbers of each tuple are connected to the time and frequency axes respectively.

After the last block, the frequency-related dimension of the spectral (auditory) input map has been brought down to one and the corresponding axis can therefore be squeezed, i.e., removed.

Afterwards, a bidirectional gated recurrent (BiGRU) unit, consisting of two layers with hidden sizes equal to 128, is used to model potential temporal relationships in the auditory data.

Spectral
features
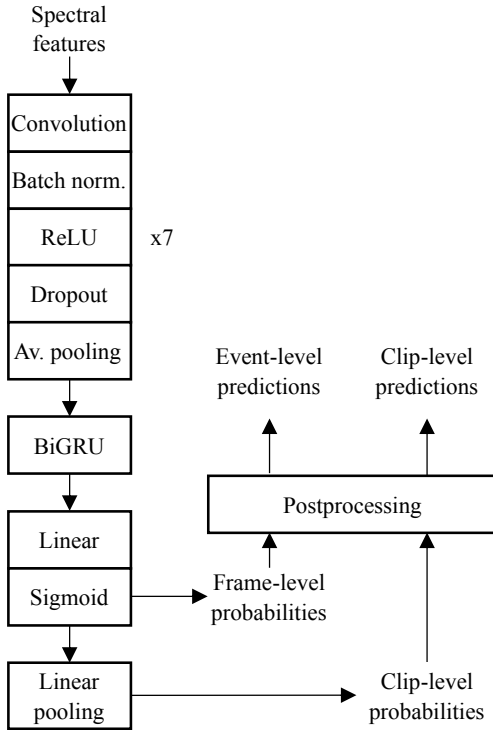
Figure 1: Schematic representation of the baseline

Table 1: Kernel sizes and strides of pooling layers in baseline CNN

Block	Kernel size = stride
0-1	(2, 2)
2-3-4-5-6	(1, 2)

The output of this recurrent layer is followed by a linear projection and application of the sigmoid function to obtain multi-label frame-level probabilities. These values indicate per temporal frame which sound categories are active, and can be postprocessed to obtain event-level predictions of sounds, which are relevant for sound event detection, as explained in the next subsection.

Finally, the frame-level sound probabilities are aggregated to get clip-level auditory event probabilities, which are relevant for the task of audio tagging, by performing linear (softmax) pooling [17].

2.2. Postprocessing

The clip- and frame-level probabilities are converted into binary values by enforcing a fixed threshold. The frame-level decisions are also passed through a smoothing median filter with a window of about 500 ms. Preliminary experiments showed that more complicated postprocessing (e.g., class-wise optimization on validation data) is unnecessary as it provides relatively insignificant benefits.

The binary clip-level decisions can readily be used to perform audio tagging. On the contrary, to produce outputs suitable for sound event detection, an extra step has to be taken: The frame-level decisions are converted into event-level predictions by performing a merging operation with a maximum gap tolerance of 200 ms.

2.3. Mean teacher training

As expanded upon in Section 4, the training data employed in this research project is heterogeneously annotated: Some of the samples include clip-level or weak labels, some come with event-level or strong labels, and the rest is unlabeled. To deal with this difficulty, the mean teacher training principle [8] is applied.

In the mean teacher training framework, two models called the student and the teacher are utilized. They share the same architecture, but their parameters are updated completely differently.

The student system is trained in a regular fashion: A differentiable loss function is optimized by an optimization algorithm such as Adam [18]. This is not the case for the teacher counterpart: The weights of this model are computed as the exponential moving average of the student parameters with a multiplicative decay factor of 0.999 per training iteration, also explaining the name of the method.

The loss used to train the student consists of four terms: The first two are clip-level and frame-level binary cross entropy functions, which are only calculated for the weakly and strongly labeled data samples respectively. The remaining two components are mean-squared error consistency costs between the clip-level and frame-level output probabilities of the student and teacher models, which can be computed for all examples, including the unlabeled ones. The classification and consistency terms are summed with weights equal to 1 and 2 respectively to obtain the final objective.

3. ADAPTATION OF TEMPORAL RESOLUTION

The goal of this project is to investigate how modifying the output temporal resolution of the considered convolutional recurrent neural network affects its sound recognition performance under different circumstances. In this section, we describe which parts of this model were changed to achieve these specific adaptations.

In the baseline model outlined in Section 2, the pooling layers in the first and second blocks of the CNN halve input feature maps along the temporal dimension, while the others do not change anything in this regard. This results in a total pooling factor of 4.

As explained in more detail in Section 4, most of the audio recordings used in this project have a duration of 10 seconds. They are fed to the baseline convolutional recurrent neural network as spectral feature maps consisting of 608 time frames. This model applies a temporal reduction factor of 4, and hence, the frame-level probabilities contain 152 values per sample. This comes down to a temporal resolution of about one class prediction vector per 65 ms.

We create adaptations of the baseline convolutional recurrent network in the following way: Instead of only allowing the first two pooling layers to apply a reduction in the number of temporal frames, we create model versions of which the first x pooling layers perform this halving operation, with x ranging from 1 up to 6. This results in systems with time reduction factors of 2, 4, 8, 16 and 32 respectively, which are equivalent to temporal resolutions of about one frame-level prediction bin per 32.5, 65, 130, 260 and 520 ms.

Important to note is that changing the amount of pooling in the considered convolutional recurrent neural network does not change the number of trainable parameters, and thus, its modeling capacity.

4. EXPERIMENTAL SETUP

In this section, we provide full details on the setup used during the experiments, of which the results are analyzed in Section 5.

4.1. Data

The data set for problem 4 of DCASE 2021 [7] is a multi-label collection consisting of audio recordings with a maximum length of 10 seconds. There are 10 possible sound event categories, related to domestic environments, which are not mutually exclusive.

The partition used to train models consists of three subsets:
- 1578 real samples with weak (clip-level) annotation
- 10000 synthetic samples with strong (event-level) annotation
- 14412 real samples without annotation

For evaluation purposes, two partitions were available: the so-called validation and public evaluation sets, holding 1168 and 692 real recordings respectively. Both subsets include strong or event-level labels of the active environmental auditory events.

4.2. Preprocessing

To get spectral maps of the available audio recordings, which are used as input features for the considered convolutional recurrent nets as explained in Section 2, we resampled all clips to 22050 Hz and performed peak amplitude normalization. Then, log mel spectrograms with 128 frequency bins were extracted using a Hamming window with a size of 2048 samples and a hop length of 363 samples. Lastly, per-frequency bin standardization was carried out.

As mentioned before, for a 10-second audio clip, these steps resulted in a spectral feature map consisting of 608 temporal frames.

4.3. Data augmentation

In order to avoid the risk of overfitting, we employed data augmentation during the training of the considered models. In particular, we used mixup [19], which comes down to creating extra learning examples (and associated labels) by linearly interpolating the original samples. We employed this method with a probability of 50% of applying it. The mixing ratios used in this algorithm were randomly sampled from a beta distribution with shape parameters set to 0.2.

Unlike for our related submission [10] for task 4 of the DCASE 2021 challenge [7], we did not employ time and frequency masking [20] in this project. This choice was based on two observations made during initial experiments: Firstly, the best hyperparameters for these techniques seemed to depend on the situation, e.g., used metric. Secondly, these methods only led to minimal improvements. As the focus of this work is on analysis rather than trying to get optimized performance, they were excluded for the sake of clarity.

4.4. Training and evaluation

All models were trained and evaluated using PyTorch [21].

4.4.1. Training

All models were trained for 200 epochs. Per epoch, 250 batches of 48 samples were given to the networks. Each batch contained 12 weakly labeled, 12 strongly labeled and 24 unlabeled examples.

Adam [18] was employed to train the weights of the student models. Learning rates were ramped up exponentially from 0 to 0.001 for the first 12500 optimization step. Thereafter, they decayed multiplicatively at a rate of 0.99995 per training iteration.

4.4.2. Evaluation

For audio tagging, we used the (micro-averaged) clip-based F1 measure [22], which was also utilized in task 4 of the DCASE 2017 challenge [23]. This score was computed for a single operating point, namely, the situation in which probabilities were converted into binary decisions by enforcing a 50% threshold.

For sound event detection, we employed multiple types of metrics. To start with, we utilized the intersection-based measures used for ranking in task 4 of the DCASE 2021 and 2022 challenges [7]. More specifically, we used two polyphonic sound event detection scores [24] representing distinct evaluation scenarios, denoted as PSDS 1 and 2. The former imposes strict requirements on the temporal localization accuracy, the latter is more lenient in this regard.

The hyperparameters utilized for calculating these PSDS measures are summarized in Table 2. These scores were computed using 50 operating points, in which thresholds linearly distributed from 0.01 to 0.99 were used to convert probabilities into binary decisions.

Details on the procedure for calculating PSDS scores and a discussion on the hyperparameters can be found in [24].

Table 2: PSDS hyperparameters

Hyperparameter	PSDS 1	PSDS 2
Detection tolerance criterion	0.7	0.1
Ground truth intersection criterion	0.7	0.1
Cross-trigger tolerance criterion	N/A	0.3
Cost of class instability	1	1
Cost of cross-triggers	0	0.5
Maximum false positive rate	100	100

Furthermore, we used the segment-based (micro-averaged) F1 score based on chunks of 1 s [22] to gauge the performance of the considered models. This metric was also employed in task 4 of the DCASE 2017 challenge [23]. Because of the long segment length, this measure quantifies systems in terms of their ability to perform more coarse-grained sound event detection, not unlike PSDS 2.

Lastly, we also employed the (macro-averaged) event-based F1 score with tolerances of 200 ms for onsets and 20% of the audio event lengths (up to a max of 200 ms) for offsets [22]. This measure was used in task 4 of the DCASE 2018, 2019 and 2020 challenges [7, 25]. This metric quantifies models in terms of their ability to perform fine-grained sound event detection, similar to PSDS 1.

These segment-based and event-based scores were computed for a single operating point, in which a threshold of 0.5 was enforced to convert frame-level probabilities into binary decisions.

All measures were computed using the clip-level or frame-level probabilities of the student models after the last training epoch.

5. EXPERIMENTAL RESULTS

In this section, we analyze the results obtained by the convolutional recurrent neural systems with varying output resolutions, as explained previously. We report the metrics for audio tagging and sound event detection elaborated upon in Section 4 after applying the following method to significantly diminish the variability of the results: For each experimental configuration, we train 20 models with independent initializations and average the scores they achieve on the public evaluation partition of the employed data set.
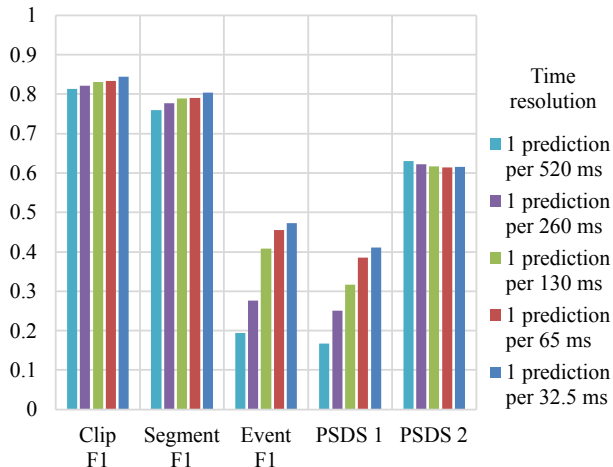
Figure 2: Results on public evaluation subset

The scores achieved on the public evaluation set are summarized in Figure 2. Precise numerical values are not essential to the interpretation below and are not disclosed due to a lack of space.

Only one of the metrics appears to display a (very slight) negative correlation with the temporal output resolution of the considered models, namely, PSDS 2. Intuitively, this behavior is not unexpected as this intersection-based score was designed to gauge relatively coarse-grained sound event detection performance.

All other measures correlate positively with respect to the time resolution. For the event-based F1 score and PSDS 1, the scores demanding precise estimations of the boundaries of sounds, this seems logical: Naturally, if the length associated with an output prediction bin is too long, this becomes more challenging or even impossible.

However, this positive relationship also appears to hold for the clip-based and segment-based F1 metrics, which inherently require much less accurate temporal differentiation — in the former case, this is not even needed at all. This counterintuitive result is all the more unanticipated given the trend of PSDS 2: Apparently, there is a certain discrepancy in the reaction of the examined types of measures for audio tagging and coarse-grained sound event detection to the temporal output resolution of the considered models.

The sensitivity of the performance of convolutional recurrent neural networks to their output resolutions depends on the criterion. In particular, scores designed for more fine-grained sound event detection show a high responsiveness in this regard: PSDS 1 and the event-based F1 measure drop sharply as the temporal resolution of the models at hand decreases. On the contrary, the clip- and segment-based F1 metrics as well as PSDS 2, used to gauge performance with regard to audio tagging and coarse-grained sound event detection, remain more stable as this hyperparameter varies.

It is also possible to study how the performance and sensitivity of the models differs across sound categories by inspecting class-wise scores. For F1-based measures, these can easily be obtained. For the PSDS metrics, they can also be computed, but unlike for their aggregated counterparts, the cost of class instability (see Section 4) must be disregarded. Exact numerical results are not crucial to the following analysis and are not included due to a lack of space.

In nearly all instances, the class-wise scores correlate to the temporal resolution in the same direction (positively/negatively) as
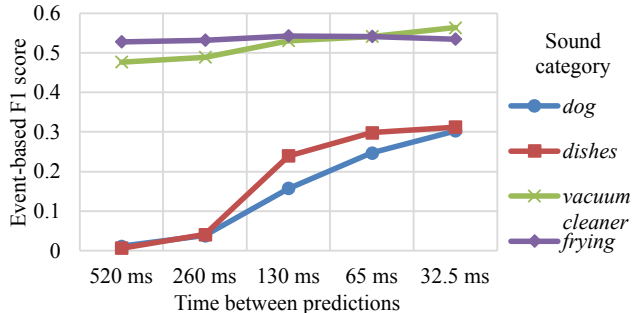

Figure 3: Subset of class-wise event-based F1 scores

their global versions. Nevertheless, the absolute performance and responsiveness strikingly vary across metrics and sound categories.

For all of the measures, there is a substantial link between the scores and the examined kind of sound. As an example, the best-performing category (*speech*) consistently outperforms the worst (*running water*) by a large margin. This behavior can be explained by multiple factors, e.g., qualitative differences between types of audio events and imbalance within the data used for training.

As is the case for the aggregated scores, the class-wise sensitivities are only strongly pronounced for fine-grained sound event detection measures. In these cases, they also greatly depend on the considered audio category. This is exemplified for the event-based F1 metric in Figure 3. For events which are short and localized (e.g., *dog*, *dishes*), reducing the temporal resolution leads to severe deterioration in terms of performance. For more long-lived sounds (e.g., *vacuum cleaner*), this dependency is much less outspoken. In the most extreme case of *frying*, there is even no correlation at all.

6. CONCLUSION

Numerous state-of-the-art sound recognition models are based on convolutional recurrent neural architectures. They are usually optimized in a mean teacher framework to deal with the heterogeneous labeling of the supplied data samples. In this work, we provided deeper insight into how changing the temporal resolution of such nets impacts their performance. We analyzed the effect of modifying pooling operations on a multitude of metrics, designed for distinct audio tagging and sound event detection scenarios.

The experiments showed that the performance of the considered models is highly susceptible to changes in terms of their pooling operations. More specifically, metrics designed for fine-grained sound event detection showed a strong, positive relation with respect to the temporal output resolutions of the systems at hand. For sound recognition measures involving less focus on temporal differentiation (among others, F1 score for audio tagging), the direction of the correlation was mixed and contingent on the scoring method.

The responsiveness appeared to depend on the considered evaluation scenario. Generally, measures devised for fine-grained sound event detection displayed a much higher reactivity than metrics built for audio tagging and coarse-grained sound event detection.

A class-wise study showed that for the latter, the sensitivity to temporal resolution was low across all sound types. Conversely, for fine-grained sound event detection scores, the reactivity was decidedly stronger for shorter than for longer-lasting events.

7. REFERENCES

[1] A. Mesaros, T. Heittola, E. Benetos, P. Foster, M. Lagrange, T. Virtanen, and M. D. Plumbley, "Detection and Classification of Acoustic Scenes and Events: Outcome of the DCASE 2016 Challenge," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 26, no. 2, pp. 379–393, 2018.

[2] T. Virtanen, A. Mesaros, T. Heittola, A. Diment, E. Vincent, E. Benetos, and B. M. Elizalde, *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2017 Workshop (DCASE2017)*, 2017.

[3] M. D. Plumbley, C. Kroos, J. P. Bello, G. Richard, D. P. Ellis, and A. Mesaros, *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018)*, 2018.

[4] M. Mandel, J. Salamon, and D. P. Ellis, *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2019 Workshop (DCASE2019)*, 2019.

[5] N. Ono, N. Harada, Y. Kawaguchi, A. Mesaros, K. Imoto, Y. Koizumi, and T. Komatsu, *Proceedings of the 5th Workshop on Detection and Classification of Acoustic Scenes and Events (DCASE 2020)*, 2020.

[6] F. Font, A. Mesaros, D. P. Ellis, E. Fonseca, M. Fuentes, and B. Elizalde, *Proceedings of the 6th Workshop on Detection and Classification of Acoustic Scenes and Events (DCASE 2021)*, 2021.

[7] N. Turpault, R. Serizel, A. Parag Shah, and J. Salamon, "Sound event detection in domestic environments with weakly labeled data and soundscape synthesis," in *Workshop on Detection and Classification of Acoustic Scenes and Events*, 2019.

[8] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," in *Advances in Neural Information Processing Systems*, 2017, pp. 1195–1204.

[9] L. Delphin-Poulat and C. Plapous, "Mean teacher with data augmentation for dcase 2019 task 4," Orange Labs Lannion, Tech. Rep., 2019.

[10] W. Boes and H. Van hamme, "Optimizing temporal resolution of convolutional recurrent neural networks for sound event detection," ESAT, Tech. Rep., 2021.

[11] N. K. Kim and H. K. Kim, "Self-training with noisy student model and semi-supervised loss function for DCASE 2021 challenge task 4," Gwangju Institute of Science and Technology, Tech. Rep., 2021.

[12] X. Zheng, H. Chen, and Y. Song, "Zheng USTC team's submission for DCASE2021 TASK4 - semi-supervised sound event detection," University of Science and Technology of China, Tech. Rep., 2021.

[13] J. Ebbers and R. Haeb-Umbach, "Pre-Training And Self-Training For Sound Event Detection In Domestic Environments," Paderborn University, Tech. Rep., 2022.

[14] W. Boes and H. Van hamme, "Audiovisual transfer learning for audio tagging and sound event detection," *arXiv preprint arXiv:2106.05408*, 2021.

[15] S. Ioffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," in *Proceedings of ICML*, 2015, pp. 448–456.

[16] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[17] Y. Wang, J. Li, and F. Metze, "A comparison of five multiple instance learning pooling functions for sound event detection with weak labeling," in *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 31–35.

[18] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[19] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," *arXiv preprint arXiv:1710.09412*, 2017.

[20] 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," *arXiv preprint arXiv:1904.08779*, 2019.

[21] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "PyTorch: An Imperative Style, High-Performance Deep Learning Library," in *Advances in Neural Information Processing Systems*, 2019, pp. 8024–8035.

[22] A. Mesaros, T. Heittola, and T. Virtanen, "Metrics for Polyphonic Sound Event Detection," *Applied Sciences*, vol. 6, no. 6, p. 162, 2016.

[23] A. Mesaros, T. Heittola, A. Diment, B. Elizalde, A. Shah, E. Vincent, B. Raj, and T. Virtanen, "DCASE 2017 challenge setup: Tasks, datasets and baseline system," in *DCASE 2017-Workshop on Detection and Classification of Acoustic Scenes and Events*, 2017.

[24] Ç. Bilen, G. Ferroni, F. Tuveri, J. Azcarreta, and S. Krstulović, "A framework for the robust evaluation of sound event detection," in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 61–65.

[25] Serizel, Romain and Turpault, Nicolas and Eghbal-Zadeh, Hamid and Shah, Ankit Parag, "Large-scale weakly labeled semi-supervised sound event detection in domestic environments," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018)*, 2018.