THRESHOLD INDEPENDENT EVALUATION OF SOUND EVENT DETECTION SCORES

Janek Ebbers, Reinhold Haeb-Umbach

Paderborn University,
Department of Communications Engineering,
33098 Paderborn, Germany,
{ebbers,haeb}@nt.upb.de

Romain Serizel

Université de Lorraine, CNRS,
Inria, Loria,
F-54000 Nancy, France,
romain.serizel@loria.fr

ABSTRACT

Performing an adequate evaluation of sound event detection (SED) systems is far from trivial and is still subject to ongoing research. The recently proposed polyphonic sound detection (PSD)-receiver operating characteristic (ROC) and PSD score (PSDS) make an important step into the direction of an evaluation of SED systems which is independent from a certain decision threshold. This allows to obtain a more complete picture of the overall system behavior which is less biased by threshold tuning. Yet, the PSD-ROC is currently only approximated using a finite set of thresholds. The choice of the thresholds used in approximation, however, can have a severe impact on the resulting PSDS. In this paper we propose a method which allows for computing system performance on an evaluation set for all possible thresholds jointly, enabling accurate computation not only of the PSD-ROC and PSDS but also of other collar-based and intersection-based performance curves. It further allows to select the threshold which best fulfills the requirements of a given application. Source code is publicly available in our SED evaluation package *sed_scores_eval*[1].

Index Terms— sound event detection, polyphonic sound detection, evaluation, threshold independent, roc

1. INTRODUCTION

Recently, there is a rapid progress in Machine Listening aiming to imitate by machines the human ability to recognize, distinguish and interpret sounds [1]. The progress is driven by the annual Detection and Classification of Acoustic Scenes and Events (DCASE) challenges[2] and the releases of large-scale sound databases such as Google's AudioSet [2] and FSD50k [3].

For a successful development of such systems an adequate evaluation of the system's operating behavior is crucial, where, ideally, the evaluation metric correlates to the user satisfaction during system application [4].

In this paper we are concerned with the evaluation of sound event detection (SED) systems [5]. SED aims to recognize sound events in audio signals together with their onset and offset time. One particular challenge in SED is that labeling of ground truth event onset and offset times, referred to as strong labels, is expensive and time-consuming. Therefore, many systems aim to learn SED from weakly labeled data [6, 7], which only indicate the presence or absence of a sound event in an audio signal without providing its onset and offset times, and unlabeled data [8, 9]. Syntheticly generated soundscapes are another alternative to produce cheap strongly annotated data [10, 11]. Here, an insightful evaluation of systems is particularly important to be able to draw conclusions about the system's learning behavior w.r.t. the temporal localization of sounds.

Due to the temporal component of sound events, however, the adequate evaluation of SED performance is far from trivial. Traditional approaches perform segment-based and collar-based (event-based) evaluation [12] for only a single operating point (decision threshold). Further, segment-based evaluation does not sufficiently evaluate a system's capability of providing connected detections, whereas collar-based evaluation is sensitive to ambiguities in the definition of the ground truth event boundaries.

More recently, Bilen et al. [13] proposed the polyphonic sound detection (PSD)-receiver operating characteristic (ROC) curve and PSD score (PSDS), which is an important step towards an evaluation of SED systems which is independent of specific decision thresholds and therefore provides a more complete picture of the system's overall operating behavior and is less biased by a specific tuning of the decision thresholds.

However, PSD-ROC curves are only approximated so far due to the lack of a method which efficiently evaluates the system's performance for all possible decision thresholds. The approximation of the PSD-ROC curve can significantly underestimate the system's PSDS as we will show in Sec. 5.

In this paper, we therefore present such a method to efficiently compute the system's performance for all possible decision thresholds jointly, which allows us to accurately compute the PSD-ROC and PSDS. Further, it can also be used to compute other intersection-based and collar-based performance curves such as precision-recall (PR)-curves. The presented approach can be understood as a generalization of the method used for single instance evaluation[3] to more sophisticated evaluations such as collar-based or intersection-based evaluations. It is based on the definition of changes in the intermediate statistics that occur when the decision threshold falls below a certain score, which we refer to as deltas in the following. Then, absolute values can be obtained for all possible thresholds by performing a cumulative sum over the deltas.

The rest of the paper is structured as follows. Sec. 2 reviews current threshold-dependent approaches for SED evaluation. Sec. 3 describes commonly used threshold-independent evaluation methods for single instance evaluation[3] as well as the recently proposed PSD for the threshold-independent evaluation of SED. Then, we present our proposed approach for the accurate computation of PSD-ROC and other performance curves in Sec. 4. Finally we present experiments in Sec. 5 and draw conclusions in Sec. 6.

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) - 282835863.

[1] `https://github.com/fgnt/sed_scores_eval`

[2] `http://dcase.community/events#challenges`

[3] By single instance evaluation we refer to an evaluation where each classified instance is evaluated with its own target.

2. SOUND EVENT DETECTION EVALUATION

SED [1, 5] can be seen as a multi-label classification problem, where the system performs classifications at multiple points in time which usually happens in a frame-based manner. When a classification score y_t exceeds a certain decision threshold it is marked as positive. Connected positive classifications are merged into a detected event $(\hat{t}_{\text{on},i}, \hat{t}_{\text{off},i}, \hat{c}_i)$ with $\hat{t}_{\text{on},i}, \hat{t}_{\text{off},i}, \hat{c}_i$ being the onset time, offset time and class label, respectively, of the i-th detection.

As in other classification tasks the evaluation is based on true positive (TP), false positive (FP) and false negative (FN) counts. The TPs count N_{TP} represents the number of ground truth events that have been detected by the system. The FPs count N_{FP} sums up the number of detections which do not match a ground truth event. Hence, the total number of detected events is given as $N_{\text{DP}} = N_{\text{TP}} + N_{\text{FP}}$. The FNs count N_{FN}, which is the number of ground truth events missed by the system, is given as $N_{\text{FN}} = N_{\text{GP}} - N_{\text{TP}}$ with N_{GP} being the total number of ground truth events. From these intermediate statistics higher level measures can be derived such as the precision $P = N_{\text{TP}}/N_{\text{DP}}$, the recall (TP-Rates (TPRs)) $R = N_{\text{TP}}/N_{\text{GP}}$ and FP-Rate (FPR) $\text{FPR} = N_{\text{FP}}/N_{\text{GN}}$, where N_{GN} is the total number of ground truth negative instances in the evaluation data set.

Compared to single instance evaluation[3], it is less obvious in SED when to classify a ground truth event as detected, i.e. TP, and when to consider a detection as FP, due to the temporal extent of the target events over multiple classification scores/frames. Currently there exist three conceptually different ways for this, which are segment-based, collar-based (event-based) and intersection-based [12, 14, 13, 15].

2.1. Segment-based

In segment-based evaluation [12, 14], classifications and targets are defined in fixed length segments (1 s segments is a popular choice). Classifications and targets are considered positive if they are detected/labeled anywhere in the segment. This way evaluation can be treated as a single instance evaluation. However, segment-based evaluation overemphasizes the contribution of longer events which expand over multiple segments and it does not evaluate the system's capability of providing meaningful uninterrupted detections.

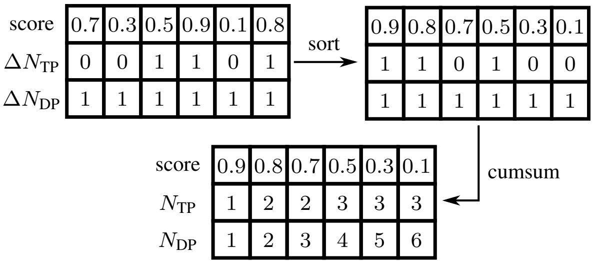
2.2. Collar-based

Collar-based, a.k.a. event-based, evaluation [12, 14] compares detections $(\hat{t}_{\text{on},i}, \hat{t}_{\text{off},i}, \hat{c}_i)$ with ground truth events $(t_{\text{on},j}, t_{\text{off},j}, c_j)$ directly. Only if there is a matching event pair (i, j) with $c_j = \hat{c}_i$, $|\hat{t}_{\text{on},i} - t_{\text{on},j}| \leq d$ and $|\hat{t}_{\text{off},i} - t_{\text{off},j}| \leq d_{\text{off},j}$, a TP is achieved. Other detections are counted as FPs. The offset collar $d_{\text{off},j} = \max(d, rT_j)$ usually depends on the length T_j of the ground truth event. Common choices are $d = 200\,\text{ms}$ and $r = 0.2$.

With collar-based evaluation, each ground truth event has equal contribution to the overall performance and systems can only achieve good performance if events are detected as single connected detections. This, however, introduces sensitivity to ambiguities in the annotation. If, e.g., an annotator labeled multiple dog barks as a single event but a system detects each bark as a separate event, this results in multiple FPs and one FN.

2.3. Intersection-based

Intersection-based evaluation [13, 15] determines the number of TPs and FPs based on intersections between detections and ground truth events. A detection tolerance criterion (DTC) classifies detections as FP if its intersection with ground truth events of the same event class, normalized by the length of the detected event, falls below a certain DTC ratio ρ_{DTC}. Else, it is considered relevant, which, however, does not necessarily mean TP. A ground truth event is only classified TP if its intersection with relevant same class detections, normalized by the length of the ground truth event, is greater or equal to a ground truth intersection criterion (GTC) ratio ρ_{GTC}.



score	0.7	0.3	0.5	0.9	0.1	0.8
ΔN_{TP}	0	0	1	1	0	1
ΔN_{DP}	1	1	1	1	1	1

0.9	0.8	0.7	0.5	0.3	0.1
1	1	0	1	0	0
1	1	1	1	1	1

score	0.9	0.8	0.7	0.5	0.3	0.1
N_{TP}	1	2	2	3	3	3
N_{DP}	1	2	3	4	5	6

Fig. 1. Illustration of the joint computation of intermediate statistics with single instance evaluation.

Bilen et al. [13] further introduced cross triggers (CTs) which are FP detections matching events from another event class and, thus, may impair user experience more than standalone FPs. Note that, although the concept of CTs has been proposed in conjunction with intersection-based evaluation, it is not restricted to it and could also be transferred to segment-based and collar-based evaluations. In intersection-based evaluation the cross trigger tolerance criterion (CTTC) counts a CT between a detected event class $\hat{c}_i$ and another event class c with $c \neq \hat{c}_i$ if the detection intersects with ground truth events of class c by at least ρ_{CTTC}.

3. THRESHOLD-INDEPENDENT EVALUATION

The computation of above intermediate statistics, such as the TP count, depend on the decision threshold that is applied to the classifier's output scores. Consequently, metrics such as F_1-scores and error-rates only evaluate a single threshold. A more complete picture of the classifier's performance, however, can be obtained when evaluating system performance for all possible thresholds.

3.1. Single Instance Evaluation

In single instance evaluation[3], the PR and ROC curves [16, 14] are frequently used to evaluate overall system behavior independently from a certain operating point. As the name suggests, the PR curve plots precisions over corresponding recall values which result from arbitrary decision thresholds. The ROC curve instead plots the recalls over corresponding FPRs. Frequently used metrics for system comparison are the area under the PR curve, a.k.a. average precision (AP), and the area under the ROC curve, which is often simply referred to as area under curve (AUC).

Rather than making decisions and evaluating performance seperately for a set of arbitrary thresholds, performance can be evaluated for all thresholds jointly by implementing a sorting of classification scores y together with some predefined deltas, as it is done, e.g., in the *scikit-learn toolkit* [17]. Here, deltas mean changes in the intermediate statistics, such as the number of TPs, when the decision threshold moves from above an instance's classification score to below of it, i.e., when the instance moves from being classified negative to being classified positive. Then absolute values can be obtained by simply performing a cumulative sum of the deltas.

This approach is illustrated in Fig. 1 for an exemplary data set with six instances. ΔN_{TP} means the change in the TP count which,

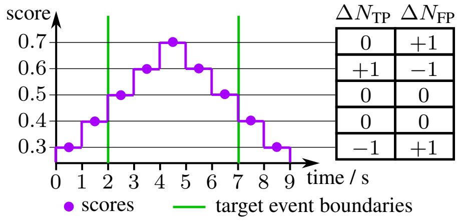


Fig. 2. Collar-based deltas example.

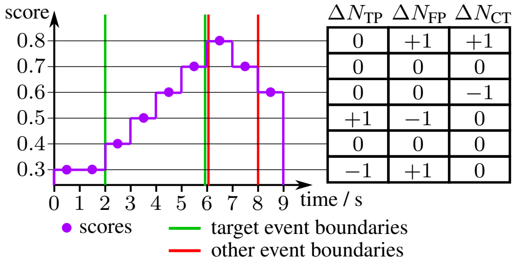


Fig. 3. Intersection-based deltas example.

for single instance evaluation, is simply the binary target of the instance. This is because, upon positive classification, the TP count only increases by one when the instance is labeled positive. ΔN_{DP} represents the change in the total number of system detections. Here ΔN_{DP} is always one as there is always one instance more being classified positive when the threshold falls below its classification score. The precisions $P = N_{\mathrm{TP}}/N_{\mathrm{DP}}$ can, e.g., now be read off for all decision thresholds in the third table containing the absolute values.

3.2. PSD-ROC

To the best of our knowledge, the PSD-ROC curve proposed in [13] is currently the only threshold-independent evaluation of SED systems. It first computes, for all event classes c, intersection-based ROC curves $\mathrm{ROC}_c(\mathrm{eFPR})$ which are monotonically increasing curves plotting TPR over effective FPR (eFPR), where the reader is referred to Bilen et al. [13] for further details about its computation. The final PSD-ROC summarizes the classwise ROC curves as

$$\text{PSD-ROC}(\mathrm{eFPR}) = \mu_{\mathrm{TPR}}(\mathrm{eFPR}) - \alpha_{\mathrm{ST}} \cdot \sigma_{\mathrm{TPR}}(\mathrm{eFPR})\,, \quad (1)$$

with $\mu_{\mathrm{TPR}}(\mathrm{eFPR})$ and $\sigma_{\mathrm{TPR}}(\mathrm{eFPR})$ being the mean and standard deviation over the classwise ROC curves at a certain eFPR, and where α_{ST} is a parameter penalizing instability across classes. The PSDS is the normalized area under the PSD-ROC curve up to a maximal $\mathrm{eFPR}_{\mathrm{max}}$.

Note that the number of thresholds, which may result in a different TPR-eFPR value pair, is as high as the number of classification scores in the data set. With a system outputting scores at a rate of 50 Hz and a rather small evaluation set of, e.g., only 1 h, this would be 180 k thresholds to be evaluated for each event class. Evaluating system performance for each of the thresholds separately is not feasible for obvious reasons. Therefore, due to a lack of an efficient joint computation of intersection-based TPR-eFPR value pairs for all thresholds, the PSD-ROC curve is commonly approximated with a reduced set of thresholds. For instance, the DCASE 2021 Challenge Task 4 [11] employed PSDSs using 50 linearly spaced thresholds. The approximation of PSD-ROC curves, however, can lead to a significant underestimation of the PSDS as we will demonstrate in Sec. 5. Non-linearly spaced thresholds could alleviate this to some extent, which, however, remains arbitrary and ad-hoc.

4. EFFICIENT COMPUTATION OF COLLAR- AND INTERSECTION-BASED CURVES

In this section we present how collar-based and intersection-based intermediate statistics can be efficiently computed jointly for all possible decision thresholds. For this we follow the same approach used for the computation of single instance evaluation curves which we described in Sec. 3.1. We aim to bring all classification scores into a sorted list together with the deltas of the intermediate statistics, which appear when the decision threshold falls below the classification score. Then we are able to obtain absolute values for all operating points by a simple cumulative sum over the deltas.

With collar-based and intersection-based evaluation, however, the computation of the deltas becomes more challenging compared to single instance evaluation, as here all scores of an audio signal have to be considered jointly and cannot be obtained instance-wise. The basic principle of the definition of the deltas is illustrated in Fig. 2 and Fig. 3.

In Fig. 2 collar-based evaluation is considered. For simplicity, we here assume scores/frames to have a width of 1 s, that target event boundaries lie exactly between two scores/frames and the on-/offset collars to be 1 s. Starting from a decision threshold above 0.7, no event would be detected as no score lies above the threshold. When the decision threshold falls below 0.7, a detection is spawned from second 4 to 5 as the 5th score lies above the threshold. However, the distances between the detected and the true onsets and offsets are 2 s for both, therefore not matching the collar. Hence, the newly spawned detection is a FP and we have $\Delta N_{\mathrm{FP}} = +1$. When the threshold falls below 0.6, however, the detection expands from second 3 to 6 and the FP disappears ($\Delta N_{\mathrm{FP}} = -1$) and becomes a TP detection ($\Delta N_{\mathrm{TP}} = +1$). When the decision threshold falls below 0.5 and below 0.4, nothing changes as the collars are still matched and the detection remains TP ($\Delta N_{\mathrm{TP}} = \Delta N_{\mathrm{FP}} = 0$). Finally, when the decision threshold falls below 0.3, the detection expands from 0 s to 9 s and the detected on-/offsets exceed the collar, and the TP disappears ($\Delta N_{\mathrm{TP}} = -1$) and becomes a FP again ($\Delta N_{\mathrm{FP}} = +1$).

A slightly more advanced example is shown in Fig. 3, where we consider intersection-based evaluation including CTs. We assume $\rho_{\mathrm{DTC}} = \rho_{\mathrm{GTC}} = \rho_{\mathrm{CTTC}} = 0.5$ and that again all event boundaries lie exactly between two scores/frames. When the decision threshold falls below 0.8 here, a detection is spawned from 6 s to 7 s which does not overlap with the target event at all, giving us $\Delta N_{\mathrm{FP}} = +1$. Further, the detected event completely lies within the ground truth event from another class (in red), giving us $\Delta N_{\mathrm{CT}} = +1$. When the threshold falls below 0.7, the detection's overlap with the target event is still only $1/3 < \rho_{\mathrm{DTC}}$. This is still a FP and therefore $\Delta N_{\mathrm{FP}} = 0$. The overlap with the other class event is $2/3 \geq \rho_{\mathrm{CTTC}}$. Therefore there is still a CT, with $\Delta N_{\mathrm{CT}} = 0$. When the threshold falls below 0.6, the detection's overlap with both the target event and the other class event is $2/5 < \rho_{\mathrm{DTC}} = \rho_{\mathrm{CTTC}}$. The detection is still FP ($\Delta N_{\mathrm{FP}} = 0$), but not a CT anymore ($\Delta N_{\mathrm{CT}} = -1$). When the threshold falls below 0.5 the overlap with the target event becomes $1/2 = \rho_{\mathrm{DTC}}$. The FP disappears ($\Delta N_{\mathrm{FP}} = -1$) and becomes a TP ($\Delta N_{\mathrm{TP}} = +1$). This remains unchanged until the decision threshold falls below 0.3, where the overlap with the ground truth event becomes only $4/9 < \rho_{\mathrm{DTC}}$. This is a FP again (but not a CT) with $\Delta N_{\mathrm{TP}} = -1$ and $\Delta N_{\mathrm{FP}} = +1$.

The proposed approach allows for efficient and accurate computation of collar-based and intersection-based PR and ROC curves, which not only enables us to compute threshold-independent metrics such as AP and PSDS precisely, but it also allows us to find the threshold which best suits specific application requirements.

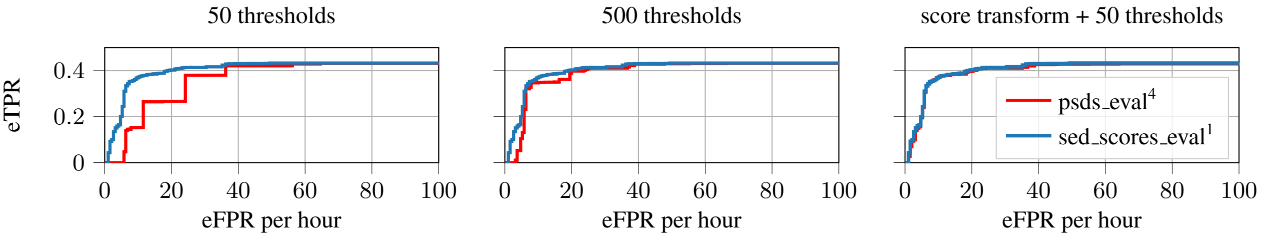


Fig. 4. PSD-ROC curves: The exact PSD-ROC curve being shown in blue, which becomes computable with our proposed methodology, and different approximations of the PSD-ROC curve shown in red.

Note that the proposed methodology is rather general and can be applied to arbitrary evaluations as long as one is able to determine the deltas in the intermediate statistics for each classification score in the evaluation data set.

5. EXPERIMENTS

In this section we demonstrate the usefulness of the proposed method for the accurate computation of threshold-independent curves and metrics as well as its potential for threshold tuning.

The presented curves and metrics are evaluated for one of our single model systems developed for DCASE 2021 Challenge Task 4, which employs a forward-backward convolutional recurrent neural network (FBCRNN) for audio tagging followed by a tag-conditioned CRNN (TCCRNN) for SED [18] outputting detection scores at a rate of 50 Hz. For more details about the system and its training, which are not relevant here, the reader is referred to Ebbers et al. [18].

In the challenge, systems have been evaluated by PSDSs which have been calculated using 50 thresholds linearly spaced from 0.01 to 0.99 for PSD-ROC curve approximation. In the following we consider the scenario 1 with $\rho_{DTC} = \rho_{GTC} = 0.7$, $\alpha_{\mathrm{CT}} = 0$, $\alpha_{\mathrm{ST}} = 1$ and $\mathrm{eFPR}_{\mathrm{max}} = 100/\mathrm{h}$ and report evaluations on the public evaluation set of the DESED database [19].

In Fig. 4 different PSD-ROC curves are shown. In the subplots we present different variants of PSD-ROC curve approximations (in red), which have been generated using the official *psds_eval* package[4], and compare them with the accurate PSD-ROC curve (in blue), which has been generated with our newly released package *sed_scores_eval*[1].

During our system development for the challenge, we recognized that our system mostly produces either very small or very high scores, which, without further measures, results in the PSD-ROC being approximated only very coarsely as shown in the left subplot of Fig. 4. Compared to the accurate computation proposed here, the approximated PSDS of 0.358 significantly underestimates the true PSDS of 0.400. Even if 500 linearly spaced thresholds from 0.001 to 0.999 are used, which is shown in the middle plot, this “step” artifact still appears on the PSD-ROC. The PSDS computed with these thresholds results to be 0.389 which still underestimates the true PSDS.

In order to obtain a smooth PSD-ROC in the challenge, we performed a non-linear transformation of our system’s classification scores, such that the classification scores of ground truth positive frames in the validation set are uniformly distributed between 0 and 1. Note, that a non-linear score transformation followed by linearly spaced thresholds results to be the same as non-linearly spaced thresholds. The resulting PSD-ROC approximation with 50 thresholds is shown in red in the right plot of Fig. 4, which then comes close to the true PSD-ROC. Note, that at this point a tuning of a score transformation function (or alternatively 50 thresholds) is required, which is highly undesired for a supposedly threshold-independent metric. However, with the proposed computation approach, the PSDS can be computed exactly and truly independently of a specific set of thresholds (with less computation time[5]).

Table 1. Collar-based F_1-score performance without and with optimal threshold tuning on validation set.

Thresholds	0.5	optimal (on val. set)
F_1-score	51.8 %	57.2 %

Next, we use the collar-based PR-curve to perform optimal threshold tuning for collar-based F_1-score evaluation, which has been an additional contrastive metric in the challenge. For each event class we choose the decision threshold, which achieves the highest F_1-score on the PR-curve of the validation set that was computed with the proposed approach. Table 1 shows collar-based F_1-score performance on the public evaluation set comparing the threshold, which is optimal on the validation set, with simply choosing a threshold of 0.5. Note that for a fair comparison, we performed a median filter size sweep for each threshold variant separately and chose for each threshold variant and event class the filter size that performed best on the validation set. At this point it may be worth noting that median filtering before and after a thresholding yields the same detection outputs, making it similarly applicable to SED scores before computing threshold-independent curves or metrics.

It can be observed that solely by tuning the decision threshold on the validation set, performance can be improved by 5.4 %. This demonstrates how threshold-dependent metrics can be biased by the tuning of an operating point. However, it also demonstrates the ability of our presented method to allow for searching the optimal operating point for a given target application.

6. CONCLUSIONS

In this paper we presented a methodology allowing for performing accurate computation of collar-based and intersection-based PR and ROC curves. Computing these metrics on a fixed set of thresholds could lead to biased estimation of the final metric. This can result in significant performance underestimation if an unfavorable set of thresholds is chosen. Our proposed method, however, enables truly threshold-independent collar-based and intersection-based SED metrics and provides a more accurate, system independent evaluation. Further, as the method allows to efficiently compute performances for arbitrary thresholds, it allows to determine the best operating point to fulfill the requirements of a specific application. We publicly released its implementation in a python package termed *sed_scores_eval*[1].

[4] `https://github.com/audioanalytic/psds_eval`

[5] See `https://github.com/fgnt/sed_scores_eval/blob/main/notebooks/psds.ipynb` for timings.

7. REFERENCES

[1] Tuomas Virtanen, Mark D Plumbley, and Dan Ellis, *Computational analysis of sound scenes and events*, Springer, 2018.

[2] Jort F Gemmeke, Daniel PW Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R Channing Moore, Manoj Plakal, and Marvin Ritter, "Audio set: An ontology and human-labeled dataset for audio events," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2017, pp. 776–780.

[3] Eduardo Fonseca, Xavier Favory, Jordi Pons, Frederic Font, and Xavier Serra, "Fsd50k: an open dataset of human-labeled sound events," *arXiv preprint arXiv:2010.00475*, 2020.

[4] Sacha Krstulović, "Audio event recognition in the smart home," *Computational Analysis of Sound Scenes and Events*, pp. 335–371, 2018.

[5] Annamaria Mesaros, Toni Heittola, Tuomas Virtanen, and Mark D Plumbley, "Sound event detection: A tutorial," *IEEE Signal Processing Magazine*, vol. 38, no. 5, pp. 67–83, 2021.

[6] Ankit Shah, Anurag Kumar, Alexander G Hauptmann, and Bhiksha Raj, "A closer look at weak label learning for audio events," *arXiv preprint arXiv:1804.09288*, 2018.

[7] Koichi Miyazaki, Tatsuya Komatsu, Tomoki Hayashi, Shinji Watanabe, Tomoki Toda, and Kazuya Takeda, "Weakly-supervised sound event detection with self-attention," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2020, pp. 66–70.

[8] Lu JiaKai, "Mean teacher convolution system for dcase 2018 task 4," Tech. Rep., Detection and Classification of Acoustic Scenes and Events Challenge, September 2018.

[9] Nicolas Turpault and Romain Serizel, "Training sound event detection on a heterogeneous dataset," in *Proc. Workshop on Detection and Classification of Acoustic Scenes and Events*, 2020.

[10] Nicolas Turpault, Romain Serizel, Scott Wisdom, Hakan Erdogan, John R Hershey, Eduardo Fonseca, Prem Seetharaman, and Justin Salamon, "Sound event detection and separation: a benchmark on desed synthetic soundscapes," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2021, pp. 840–844.

[11] Francesca Ronchini, Romain Serizel, Nicolas Turpault, and Samuele Cornell, "The impact of non-target events in synthetic soundscapes for sound event detection," *arXiv preprint arXiv:2109.14061*, 2021.

[12] Annamaria Mesaros, Toni Heittola, and Tuomas Virtanen, "Metrics for polyphonic sound event detection," *Applied Sciences*, vol. 6, no. 6, pp. 162, 2016.

[13] Çağdaş Bilen, Giacomo Ferroni, Francesco Tuveri, Juan Azcarreta, and Sacha Krstulović, "A framework for the robust evaluation of sound event detection," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2020, pp. 61–65.

[14] Annamaria Mesaros, Toni Heittola, and Dan Ellis, "Datasets and evaluation," in *Computational Analysis of Sound Scenes and Events*, pp. 147–179. Springer, 2018.

[15] Giacomo Ferroni, Nicolas Turpault, Juan Azcarreta, Francesco Tuveri, Romain Serizel, Çağdaş Bilen, and Sacha Krstulović, "Improving sound event detection metrics: insights from dcase 2020," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2021, pp. 631–635.

[16] Jesse Davis and Mark Goadrich, "The relationship between precision-recall and roc curves," in *Proc. 23rd international conference on Machine learning*. 2006, pp. 233–240, ACM Press.

[17] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[18] Janek Ebbers and Reinhold Haeb-Umbach, "Self-trained audio tagging and sound event detection in domestic environments," Tech. Rep., Detection and Classification of Acoustic Scenes and Events Challenge, June 2021.

[19] Nicolas Turpault, Romain Serizel, Ankit Parag Shah, and Justin Salamon, "Sound event detection in domestic environments with weakly labeled data and soundscape synthesis," in *Proc. Workshop on Detection and Classification of Acoustic Scenes and Events*, 2019.