SOUND EVENT DETECTION WITH RESNET AND SELF-MASK MODULE FOR DCASE 2019 TASK 4

Technical Report

Yu Kiyokawa, Sakiko Mishima, Takahiro Toizumi, Kazutoshi Sagi, Reishi Kondo and Toshiyuki Nomura

Data Science Research Laboratories, NEC Corporation,
1753 Shimonumabe, Nakahara-ku, Kawasaki 211-8666, Japan
{y-kiyokawa@cq, s-mishima@cb, t-toizumi@ct, ksagi@ah, kondoh@ct, t-nomura@da}.jp.nec.com

ABSTRACT

In this technical report, we propose a sound event detection system using a residual network (ResNet) with a self-mask module for a task 4 of detection and classification of acoustic scenes and events 2019 (DCASE 2019) challenge. Our system is constructed with a convolutional neural network based on a ResNet. We introduce a self-mask module as a region proposal network in order to detect event time boundaries. The self-mask module constrains time duration of silent and sound events by proposing candidates of the sound event region. These constraints improve detection accuracy of the sound event regions. Evaluation results show that our system obtains 36.09% of event-based F1-score for a sound event detection on a validation dataset of the task 4.

Index Terms— CNN, ResNet, SENet, SE Module, Self-mask, Region Proposal, Sound event detection, SED

1. INTRODUCTION

In this report, we challenge on a sound event detection (SED) [1, 2] problem defined by detection and classification of acoustic and scean event (DCASE) 2019 task 4 [3]. The goal of the task 4 is to evaluate systems for the detection of sound events using real data either weakly labeled or unlabeled, and simulated data that is strongly labeled (with time stamps). Our system predicts not only the event class but also the event time boundaries given that polyphonic events can be present in an audio recording. We employ a residual network (ResNet) [4] and a self-mask module for this task. The ResNet is selected as a base network for the sound detection, because ResNets have got state-of-the-art performance on the ILSVRC & MS COCO 2015 competitions in image classification, detection, localization and segmentation [4]. The self-mask module constrains time duration of silent and sound events for accurate detection of onset and offset in SED by proposing candidates of sound event regions. This combination has improved the estimation accuracy of the sound region.

2. PROPOSED METHOD

Figure 1 shows the overall neural network architecture of proposed method. The proposed system is constructed based on a ResNet with a self-mask module.

2.1. Feature extraction

Our system extracts an audio feature $\boldsymbol{x}$ from a raw signal before entering the network. A log mel spectrogram (2048 window length, 431 hop length) is extracted from an audio clip re-sampled to 44.1 kHz and padded or truncated to 10 seconds. Min-max feature scaling is applied to the log mel spectrogram in a row-wise manner, in a column-wise manner and in overall spectrogram. Each scaling generates a row-wise normalized matrix $\boldsymbol{A}^{\mathrm{row}}$, a column-wise normalized matrix $\boldsymbol{A}^{\mathrm{col}}$ and an overall normalized matrix $\boldsymbol{A}^{\mathrm{all}}$. The audio feature $\boldsymbol{x}$ is an element-wise product of the three normalized log mel spectrograms:

$$\boldsymbol{x} = \boldsymbol{A}^{\mathrm{row}} \circ \boldsymbol{A}^{\mathrm{col}} \circ \boldsymbol{A}^{\mathrm{all}}. \tag{1}$$

The audio feature $\boldsymbol{x}$ has 1024 frames by 128 mel frequency channels.

2.2. Network outputs

We denote an instance by the audio feature $\boldsymbol{x} \in [0,1]^{1024\times128}$, and its relevant class labels by a target vector (class target) $\boldsymbol{y}^{\mathrm{c}} \in \{0,1\}^{K}$, and its relevant event detection labels by a target matrix (strongly labeled target) $\boldsymbol{y}^{\mathrm{s}} \in \{0,1\}^{H\times K}$, where K is the number of labels, H is the size of time frames, c indicates a class target and s indicates a strongly labeled target in training data. Given a weakly labeled training dataset $\mathcal{D}^{\mathrm{w}} = \{\boldsymbol{x}_n, \boldsymbol{y}^{\mathrm{c}}_n\}_{n=1}^{N^{\mathrm{w}}}$, and a strongly labeled dataset $\mathcal{D}^{\mathrm{s}} = \{\boldsymbol{x}_n, (\boldsymbol{y}^{\mathrm{c}}_n, \boldsymbol{y}^{\mathrm{s}}_n)\}_{n=1}^{N^{\mathrm{s}}}$, where N^{w} and N^{s} are the numbers of samples in $\mathcal{D}^{\mathrm{w}}$ and $\mathcal{D}^{\mathrm{s}}$, respectively.

The network takes the audio feature $\boldsymbol{x}$ as an input, and has five output streams as shown in Fig. 1. The first stream, the second stream, the third stream, the fourth stream and the fifth stream outputs an event detection probability $f^{\mathrm{s}}(\boldsymbol{x}) \in [0,1]^{H\times K}$, a classification probability $f^{\mathrm{c}}(\boldsymbol{x}) \in [0,1]^{K}$, an event region probability $f^{\mathrm{e}}(\boldsymbol{x}) \in [0,1]^{H}$, a proposal region probability $f^{\mathrm{p}}(\boldsymbol{x}) \in [0,1]^{H}$ and a raw-event region probability $f^{\mathrm{r}}(\boldsymbol{x}) \in [0,1]^{H}$, respectively. This probability is used to calculate an event detection loss (in 2.6.2). The classification probability $f^{\mathrm{c}}(\boldsymbol{x})$ is classification score to compute a classification loss (in 2.6.1). The event region probability $f^{\mathrm{e}}(\boldsymbol{x})$, the raw-event region probability $f^{\mathrm{r}}(\boldsymbol{x})$ and the proposal region probability $f^{\mathrm{p}}(\boldsymbol{x})$ are probabilities about time duration of silent and sound events. These probabilities are used to calculate a region loss (in 2.6.3).

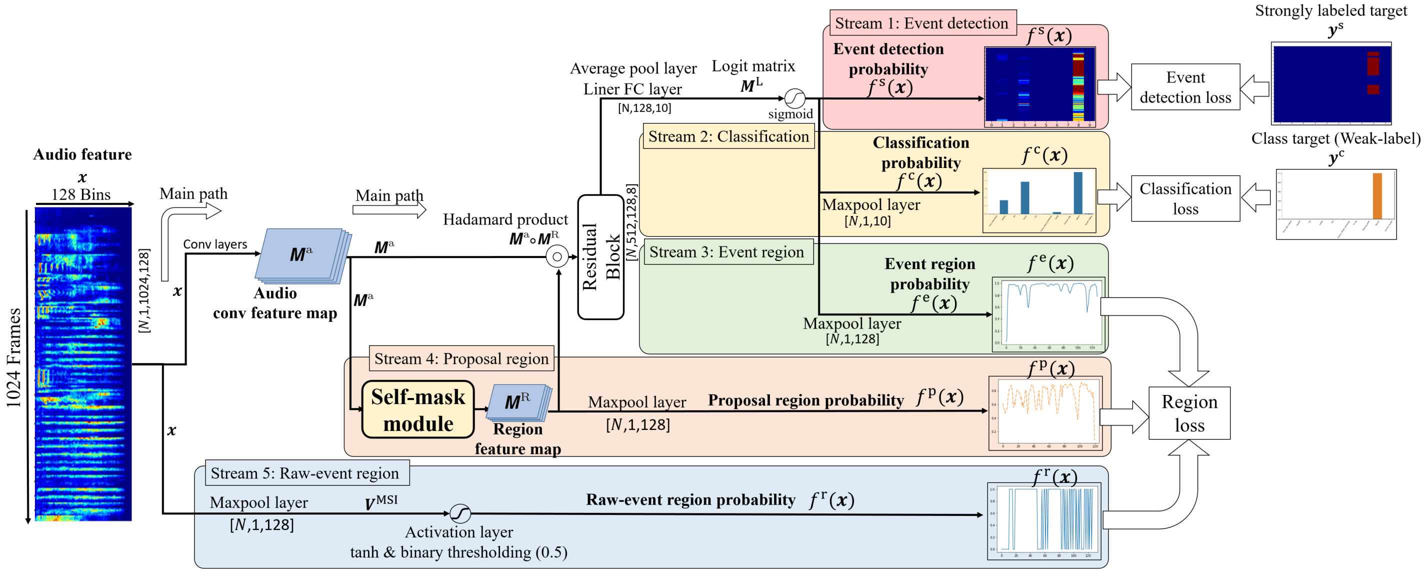

Figure 1: The overall neural network architecture of proposed method. There are five outputs: the event detection probability $f^{s}(\boldsymbol{x})$ for training class and location of sound event, the classification probability $f^{c}(\boldsymbol{x})$ for training only class, the others ($f^{e}(\boldsymbol{x})$,$f^{r}(\boldsymbol{x})$,$f^{p}(\boldsymbol{x})$) for training regions of sound events.

2.3. Network architecture

The network path branches into the main path and the raw-event region stream at the beginning. In the main path, the audio feature $\mathbf{x}$ is supplied to convolution layers (conv layers). The conv layers consist of a convolution layer (kernel size: 3×3), a batch normalization layer, a ReLU function layer and a maxpool layer (stride: 2). These layers increase a channel size of the audio feature $\boldsymbol{x}$ from 1 to 64, reduce the size of width and height by half, and convert the audio feature $\mathbf{x}$ to an audio conv feature map $\boldsymbol{M}^{a}$. The conv layers feed $\boldsymbol{M}^{a}$ to a residual block and a self-mask module. The audio conv feature map $\boldsymbol{M}^{a}$ before entering the residual block is multiplied with a region feature map $\boldsymbol{M}^{R}$ which is an output of the self-mask module. After the residual block, an average pool layer and a linear transformation layer convert an output of the residual block to a logit matrix M^{L}. The logit matrix M^{L} is activated by a sigmoid function, and becomes the event detection probability $f^{s}(\boldsymbol{x})$. The event detection probability $f^{s}(\boldsymbol{x})$ is sent to the event detection stream, the classification stream and the event region stream. In the event detection stream, the network directory outputs the event detection probability $f^{s}(\boldsymbol{x})$. In the classification stream, a maxpool layer extracts the classification probability $f^{c}(\boldsymbol{x})$ from $f^{s}(\boldsymbol{x})$. In the event region stream, another maxpool extracts the event region probability $f^{e}(\boldsymbol{x})$ from $f^{s}(\boldsymbol{x})$.

In the proposal region stream, the self-mask module takes the audio conv feature map $\boldsymbol{M}^{a}$, which generates a region feature map $\boldsymbol{M}^{R}$. The module sends the region feature map $\boldsymbol{M}^{R}$ to the main path and to a maxpool layer. As previously stated in the last paragraph, the returned $\boldsymbol{M}^{R}$ multiplied by the audio conv feature map $\boldsymbol{M}^{a}$ is supplied to the residual block. The maxpool layer converts the region feature map $\boldsymbol{M}^{R}$ to the proposal region probability $f^{p}(\boldsymbol{x})$.

In the raw-event region stream, the audio feature $\boldsymbol{x}$ is supplied to a maxpool layer. The maxpool layer extracts a maximum spectral intensity $\boldsymbol{V}^{MSI}$ from $\boldsymbol{x}$. An activation layer consists of a hyperbolic tangent function with a threshold values of 0.5. This layer converts the maximum spectral intensity $\boldsymbol{V}^{MSI}$ into the raw-event region probability $f^{r}(\boldsymbol{x})$.

2.4. Self-mask module

The self-mask module introduced to the shallow middle layer of the ResNet (Fig. 1) detects an event time boundaries. Figure 2 shows architecture of the self-mask module. The self-mask module consists of two feedforward neural networks with “shortcut connections” [4, 5]. The self-mask module takes the audio conv feature map $\boldsymbol{M}^{a}$ and generates a region probability map $\boldsymbol{M}^{R}$. The self-mask module is trained to make $f^{e}(\boldsymbol{x})$ and $f^{p}(\boldsymbol{x})$ a same probability of $f^{r}(\boldsymbol{x})$. These probabilities indicate the existence probability of sound events. Therefore, this training enables the self-mask module to constrain time duration of silent and sound events. These constraints improve detection accuracy of the sound event regions.

2.5. Residual block

Residual block is based on ResNet18 [4], and consists of 4 base blocks as shown in Fig. 3. We adopt pre-activation blocks [6] and SE modules [7] for each base block. These blocks and modules are known to improve ResNet performance [6, 7]. A base block consists of two pre-activation blocks, two SE modules, and two shortcut paths (Fig.3). A convolution layer and a batch normalization layer are installed on a first shortcut path in order to make a size of an input feature equal to that of a SE module output. The pre-activation block consists of two convolution layers, two batch normalization layers, and two ReLU layers as shown in the bottom of Fig.3. The SE module consists of a global average pooling layer, two fully connected (FC) layers, a ReLU layer and a shortcut path as shown in the top right in Fig.3. The base blocks double a channel size of a supplied feature, and reduce the feature width and height by half in the third and fourth blocks.

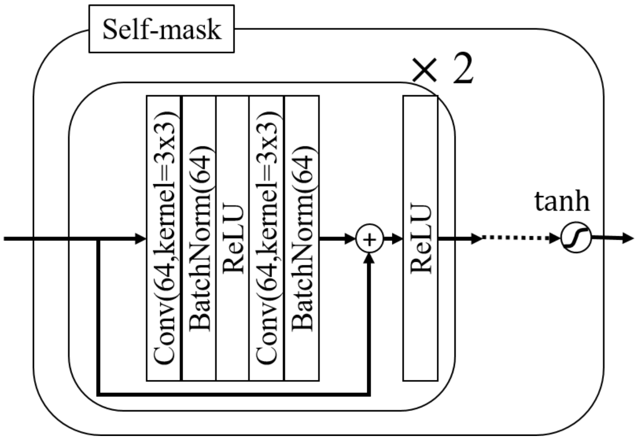

Figure 2: The architecture of the self-mask module.

2.6. Training

In the task 4, a real dataset labeled only class information $\mathcal{D}^{\mathrm{w}}$ and a simulated dataset labeled class with time stamps $\mathcal{D}^{\mathrm{s}}$ are given. Our system predicts both event classes and event time boundaries by checking the classification probability at each time frame of event detection probability $f^{\mathrm{s}}(\boldsymbol{x})$. A classification loss function $L^{\mathrm{class}}(f^{\mathrm{c}}(\boldsymbol{x}), \boldsymbol{y}^{\mathrm{c}})$ constrains networks to predict the event classes from class labeled dataset $\mathcal{D}^{\mathrm{w}}$ and $\mathcal{D}^{\mathrm{s}}$. An event detection loss function $L^{\mathrm{strong}}(f^{\mathrm{s}}(\boldsymbol{x}), \boldsymbol{y}^{\mathrm{s}})$ is required to predict the event time boundaries from strongly labeled dataset $\mathcal{D}^{\mathrm{s}}$. To improve detection accuracy of the sound event regions, we introduce a region loss function $L^{\mathrm{region}}(f^{e}(\boldsymbol{x}), f^{p}(\boldsymbol{x}), f^{r}(\boldsymbol{x}))$ for training the self-mask module. A total loss L becomes:

$$
\begin{aligned}
L = L^{\mathrm{class}}&(f^{\mathrm{c}}(\boldsymbol{x}), \boldsymbol{y}^{\mathrm{c}}) + L^{\mathrm{strong}}(f^{\mathrm{s}}(\boldsymbol{x}), \boldsymbol{y}^{\mathbf{s}}) \\
&+\alpha\, L^{\mathrm{region}}(f^{\mathrm{e}}(\boldsymbol{x}), f^{\mathrm{p}}(\boldsymbol{x}), f^{\mathrm{r}}(\boldsymbol{x})), \qquad (2)
\end{aligned}
$$

where α is a hyper parameter. We set α to 0.1 and 0.01 in the task 4 challenge.

2.6.1. *Classification loss*

The given dataset consists of 10 sound-event classes and they are imbalanced. In order to deal with the imbalanced data, we use weighted binary cross entropy (WBCE) loss [8]. A weight vector $\boldsymbol{w} = \{w_k\}_{k=1}^{K}$ of WBCE loss are calculated in

$$
w_k = \frac{e^{1/n_k}}{\sum_{k=1}^{K} e^{1/n_k}}, \qquad (3)
$$

where n_k is a total number of events for each class. The classification loss is calculated with a strongly labeled dataset $\mathcal{D}^{\mathrm{s}}$ and weakly labeled dataset $\mathcal{D}^{\mathrm{w}}$ respectively. The classification loss is defined as:

$$
\begin{aligned}
L_{\mathrm{WBCE}}(f^{\mathrm{c}}(\boldsymbol{x}), \boldsymbol{y}^{\mathrm{c}}) = \frac{1}{K}\sum_{i=1}^{K} &-w_i[y_i^{\mathrm{c}} \log(f_i^{\mathrm{c}}(\boldsymbol{x})) \\
&+ (1 - y_i^{\mathrm{c}}) \log(1 - f_i^{\mathrm{c}}(\boldsymbol{x}))], \quad (4)
\end{aligned}
$$

where w_i, y_i^c and $f_i^{\mathrm{c}}(\boldsymbol{x})$ are i-th element of $\boldsymbol{w}$, $\boldsymbol{y}^{\mathrm{c}}$ and $f^{\mathrm{c}}(\boldsymbol{x})$, respectively. The classification loss becomes:

$$
L^{\mathrm{class}} = L_{\mathrm{WBCE}}^{\mathrm{w}}(f^{\mathrm{c}}(\boldsymbol{x}), \boldsymbol{y}^{\mathrm{c}}) + L_{\mathrm{WBCE}}^{\mathrm{s}}(f^{\mathrm{c}}(\boldsymbol{x}), \boldsymbol{y}^{\mathrm{c}}), \qquad (5)
$$

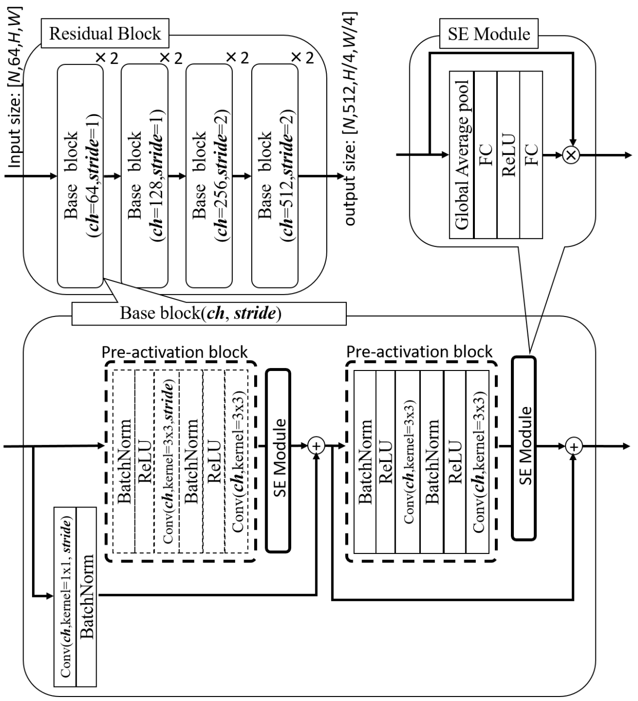

Figure 3: The architecture of the residual block, the base block, and SE Module.

where $L_{\mathrm{WBCE}}^{\mathrm{w}}$ is calculated with weakly labeled dataset $\mathcal{D}^{\mathrm{w}}$, and $L_{\mathrm{WBCE}}^{\mathrm{s}}$ is calculated with strongly labeled dataset $\mathcal{D}^{\mathrm{s}}$.

2.6.2. *Event detection loss*

An event detection loss of strongly labeled data is calculated using binary cross entropy (BCE) loss. The event detection loss becomes:

$$
\begin{aligned}
L^{\mathrm{strong}}(f^{\mathrm{s}}(\boldsymbol{x}), \boldsymbol{y}^{\mathrm{s}}) = \frac{1}{HK}\sum_{h}^{H}\sum_{k}^{K} &-[y_{hk}^{s} \log(f_{hk}^{s}(\mathbf{x})) \\
&+ (1 - y_{hk}^{s}) \log(1 - f_{hk}^{s}(\mathbf{x}))], \quad (6)
\end{aligned}
$$

where the target y_{hk}^{s} and f_{hk}^{s} are the h-th row and k-th column of the event labeled matrix $\boldsymbol{y}^{\mathrm{s}}$ and the event detection probability $f^{\mathrm{s}}(\boldsymbol{x})$, respectively.

2.6.3. *Region loss*

The proposed self-mask module is trained to minimize the loss between a event region probability $f^{\mathrm{e}}(\boldsymbol{x})$ and a proposal region probability $f^{\mathrm{p}}(\boldsymbol{x})$. The network is trained to minimize the loss between a raw event region probability $f^{\mathrm{r}}(\boldsymbol{x})$ and an event region probability $f^{\mathrm{e}}(\boldsymbol{x})$. The region loss becomes:

$$
\begin{aligned}
&L^{\mathrm{region}}(f^{\mathrm{e}}(\boldsymbol{x}), f^{\mathrm{p}}(\boldsymbol{x}), f^{\mathrm{r}}(\boldsymbol{x})) && (7)\\
&= \frac{1}{H}\sum_{h}^{H} -[f^{\mathrm{e}}_{h}(\boldsymbol{x})\log(f^{\mathrm{p}}_{h}(\boldsymbol{x}))\\
&\quad + (1 - f^{\mathrm{e}}_{h}(\boldsymbol{x}))\log(1 - f^{\mathrm{p}}_{h}(\boldsymbol{x}))]\\
&+ \frac{1}{H}\sum_{h}^{H} -[f^{\mathrm{r}}_{h}(\boldsymbol{x})\log(f^{\mathrm{e}}_{h}(\boldsymbol{x}))\\
&\quad + (1 - f^{\mathrm{r}}_{h}(\boldsymbol{x}))\log(1 - f^{\mathrm{e}}_{h}(\boldsymbol{x}))], && (8)
\end{aligned}
$$

where $f^{\mathrm{e}}_{h}(\boldsymbol{x})$, $f^{\mathrm{p}}_{h}(\boldsymbol{x})$ and $f^{\mathrm{r}}_{h}(\boldsymbol{x})$ are h-th element of $f^{\mathrm{e}}(\boldsymbol{x})$, $f^{\mathrm{p}}(\boldsymbol{x})$ and $f^{\mathrm{r}}(\boldsymbol{x})$.

3. ADDITIONAL PROCESSING SPECIFIC TO EVALUATION

Our system predicts both event classes and event time boundaries by checking the classification probability at each time frame of event detection probability $f^{\mathrm{s}}(\boldsymbol{x})$. In order to improve accuracy of event detection, we employ an ensemble method and an event connecting & cutting method.

The predictions of top 4 lowest loss for the validation dataset in the training iterations are ensembled by weighting with F1-score of each class.

In order to improve precision, columns of $f^{\mathrm{s}}(\boldsymbol{x})$ are replaced with zero column vectors when the corresponding frame is determined as a gap shorter than 0.2 second (3 frames), and replaced with one column vectors when a gap between the corresponding frames is less than 0.2 second.

4. EVALUATION RESULTS

Table 1 shows the results on validation set. We have submitted four models to the DCASE task 4. All the F1-score of the four models exceed the baseline event-based F1-score for SED on the validation set. The best model obtains 36.09% of event-based F1-score. It has increased by 12.39 percentage points from the baseline F1-score 23.70% [3].

Table 1: Overall metrics (event-based) on validation set

Models	F1 (%)
DCASE Baseline [3]	23.70
ResNet18 + self-mask ($\alpha = 0.1$)	31.65
ResNet18 + self-mask ($\alpha = 0.01$) + event connecting & cutting	32.87
ResNet18 + self-mask ($\alpha = 0.01$) + ensemble	34.51
ResNet18 + self-mask ($\alpha = 0.01$) + ensemble + event connecting & cutting	**36.09**

5. CONCLUSION

In this technical report, we have proposed a SED system using ResNet with self-mask module for a task 4 of DCASE 2019 challenge. The self-mask module in the proposed system is a region proposal network developed for a sound event detection, which constrains time duration of silent and sound events by proposing candidates of the sound event region. These constraints improve detection accuracy of the sound event regions. Our system has obtained an event-based F1-score of 36.09% on the validation dataset.

6. REFERENCES

[1] T. Komatsu, Y. Senda, and R. Kondo, "Acoustic event detection based on non-negative matrix factorization with mixtures of local dictionaries and activation aggregation," *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2259–2263, 2016.

[2] C.-C. Kao, W. Wang, M. Sun, and C. Wang, "R-crnn: Region-based convolutional recurrent neural network for audio event detection," arXiv:1808.06627 https://arxiv.org/abs/1808.06627, 2018.

[3] "DCASE2019 challenge," http://dcase.community/challenge2019/.

[4] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," arXiv:1512.03385 https://arxiv.org/abs/1512.03385, 2015.

[5] C. Bishop, *Neural networks for pattern recognition.* Oxford University Press, USA, 1995.

[6] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," arXiv:1603.05027 https://arxiv.org/abs/1603.05027, 2016.

[7] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-excitation networks," arXiv:1709.01507 https://arxiv.org/abs/1709.01507, 2017.

[8] C.-Y. Hsieh, Y.-A. Lin, and H.-T. Lin, "A deep model with local surrogate loss for general cost-sensitive multi-label learning," in *AAAI*, 2018.