INTERSPEECH 2021
30 August – 3 September, 2021, Brno, Czechia

Acoustic Event Detection with Classifier Chains

Tatsuya Komatsu[1], *Shinji Watanabe*[2], *Koichi Miyazaki*[3], *Tomoki Hayashi*[3,4]

[1]LINE Corporation, Japan
[2]Carnegie Mellon University, USA
[3]Nagoya University, Japan
[4]Human Dataware Lab. Co., Ltd., Japan

komatsu.tatsuya@linecorp.com

Abstract

This paper proposes acoustic event detection (AED) with classifier chains, a new classifier based on the probabilistic chain rule. The proposed AED with classifier chains consists of a gated recurrent unit and performs iterative binary detection of each event one by one. In each iteration, the event's activity is estimated and used to condition the next output based on the probabilistic chain rule to form classifier chains. Therefore, the proposed method can handle the interdependence among events upon classification, while the conventional AED methods with multiple binary classifiers with a linear layer and sigmoid function have placed an assumption of conditional independence. In the experiments with a real-recording dataset, the proposed method demonstrates its superior AED performance to a relative 14.80% improvement compared to a convolutional recurrent neural network baseline system with the multiple binary classifiers.

Index Terms: acoustic event detection, multi-label classification, chain rule, classifier chains

1. Introduction

Acoustic event detection (AED) is a technology for automatically detecting and recognizing the wide variety of sounds around us and understanding the environment and situation where the sounds have been recorded. This technology can be employed for diverse applications, including monitoring and surveillance [1, 2, 3, 4].

AED is a task of labeling semantic events and marking their temporal location and duration in a given audio signal. Furthermore, it assumes that events can co-occur in time, i.e., AED is a multi-label classification problem. Hence, some approaches employ source separation techniques, for example, non-negative matrix factorization [5, 6]. With increasing data set sizes, neural network-based multi-label classification models have received much attention, e.g., using convolutional neural networks (CNNs) [7] and long short-term memory (LSTM) [8, 9]. Convolutional recurrent neural networks (CRNNs) [10, 11] have become a strong baseline for neural approaches. More recently, self-attention-based models including Transformer [12, 13, 14] and Conformer [15] have shown significant improvement in AED performances.

These conventional methods focus mainly on structures of the feature extractor for modeling the characteristics of acoustic events and less on the part of the classifier design. For the classifier design, almost all methods use multiple binary classifiers consisting of a linear layer and a sigmoid function. This combination maps input audio to the activities of multiple events. While this is a straightforward way to achieve multi-label classification, the assumption of conditional independence of each event activity is placed behind it. This classifier performs estimation independently of other events, and there is no explicit interaction among the events. However, sounds in the real world have dependencies on each other. For example, 'people speaking' is likely to occur with 'people walking,' and 'break squeaking' sounds should accompany the 'car.' Therefore, it is an inappropriate assumption that each event occurs independently.

Some AED methods focus on the classifier design and the co-occurrence of events. Imoto *et al.* [16] modeled the co-occurrence relationship of events as a graph and used it as a constraint for training the model parameters. They extracted co-occurrence relationships only as statistics in a specific length, such as within audio clips, and the constraint is used only during training. Drossos *et al.* [17] proposed an AED based on sequence-to-sequence modeling that takes into account the temporal context of acoustic events. However, this method only models the input's temporal dependence, and the classifier itself is still the multiple binary classifiers. Therefore, the assumption of conditional independence still remains in both cases.

In the machine learning field, the problem of label-dependencies regarding multi-label classification has been well studied [18]. For example, a classifier based on probabilistic chain rules [19], Bayesian approach to find label-dependencies has been proposed. A neural network based method has also been proposed and shown its effectiveness in image classification [20], text classification [21, 22], and the audio tagging task [23]. Also, in the speech field, multi-speaker speech separation, speech recognition, and speaker diarization with the chain rule [24, 25] have been proposed. It demonstrates better performance than the conventional method, which deals with multiple speakers as conditionally.

This paper proposes acoustic event detection with classifier chains based on the probabilistic chain rule to handle the dependencies among events. The proposed classifier chains consist of a gated recurrent unit and perform iterative binary detection of multiple events one by one. In each iteration, one event's activity is estimated, and the estimated activity is used to condition the classifier chains for the next iteration. The proposed method can handle the interdependence among events upon classification, while the conventional multiple binary classifiers has placed an assumption of conditional independence.

2. AED with classifier chains

2.1. Probabilistic formulation of AED and the conventional method

Let us consider AED with L event classes. AED is a multi-label classification problem on a set of labels $\mathcal{L}$ that identifies what acoustic events are active in the input audio. It can be

http://dx.doi.org/10.21437/Interspeech.2021-2218

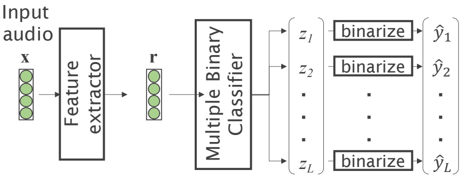

Figure 1: *General architecture of the conventional AED methods. The multiple binary classifiers estimate the activity for each event independently and there is no interaction among the events.*

formulated as a problem of estimating a subset of event labels $\hat{\mathcal{Y}} \subseteq \mathcal{L}$ from given input audio $\mathbf{X} \in \mathbb{R}^{T \times F}$:

$$\hat{\mathcal{Y}} = \underset{\mathcal{Y} \subseteq \mathcal{L}}{\operatorname{argmax}} P(\mathcal{Y} \mid \mathbf{X}), \tag{1}$$

where $\mathbf{X} = [\mathbf{x}_1, ..., \mathbf{x}_T]$ is a F dimensional audio feature sequence of the input audio with T temporal frames. The label-subset $\mathcal{Y}$ is typically represented by an L-dimensional multi-hot vector $\mathbf{y} \in \{0, 1\}^L$ that $\{0, 1\}$ indicate activity {inactive, active} of each event class. Using this multi-hot activity representation, Eq. (1) can be rewritten as the following joint probability:

$$\begin{aligned} \hat{\mathbf{y}} &= \underset{\mathbf{y} \in \{0,1\}^L}{\operatorname{argmax}} P(\mathbf{y} \mid \mathbf{X}) \qquad (2) \\ &= \underset{y_1, \ldots, y_L}{\operatorname{argmax}} P(y_1, ..., y_L \mid \mathbf{X}). \qquad (3) \end{aligned}$$

Figure 1 illustrates a general architecture of AED methods. The audio feature sequence $\mathbf{X}$ is transformed to a latent representation $\mathbf{R} = [\mathbf{r}_1, ..., \mathbf{r}_T]$ by the feature extractor, where $\mathbf{r} \in \mathbb{R}^D$ is a D-dimensional latent representation. The structure of the feature extractor is generally designed with neural networks, such as LSTMs [8, 9], CRNNs [10, 11], and self-attention [12, 15]. The latent representation $\mathbf{r}_t$ at each temporal frame t is fed into any classifier and obtain score vector $\mathbf{z}_t \in (0, 1)^L$ whose elements represent the activity-score of the corresponding event class. Then, each element of $\mathbf{z}_t$ is binarized with an appropriate threshold and resulting in an estimated activity of each event set $\hat{\mathbf{y}}_t$.

Almost all conventional methods employ multiple binary classifiers that consist of a linear layer and a sigmoid function. Here, let $\mathbf{W} \in \mathbb{R}^{L \times D}$ and $\mathbf{b} \in \mathbb{R}^L$ denote the weight and bias parameters of the linear layer, respectively. The multiple binary classifiers are written as follows:

$$\begin{aligned} \mathbf{z} &= \sigma(\mathbf{W}\mathbf{r} + \mathbf{b}), \qquad (4) \\ z_i &= \sigma(\mathbf{w}_i \mathbf{r} + b_i), \qquad (5) \end{aligned}$$

where, $\mathbf{w}_i$ is i-th row vector in $\mathbf{W}$, b_i is i-th elements of $\mathbf{b}$ and $\sigma(\cdot)$ represents the sigmoid function. As can be seen from Eq. (5), the estimation of the i-th class score z_i depending only on the latent representation $\mathbf{r}$ and the i-th weight $\mathbf{w}_i$, and no information on the other event classes are involved. Each event score and activity are estimated class-independently, depending only on the latent representation of the input audio. In other words, there is the conditional independence assumption for each event in the conventional method. From the view point, conventional methods approximate the joint probability of event classes in Eq. (2) by the product of probabilities with class-independent:

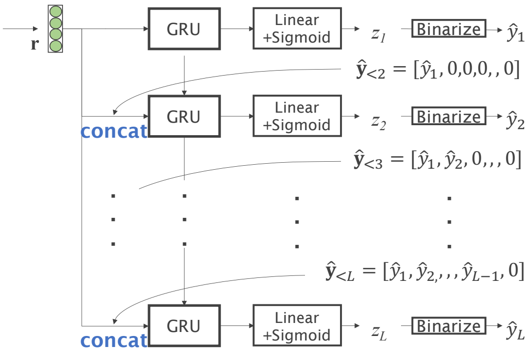
r
GRU
Linear
+Sigmoid
z_1
Binarize
$\hat{y}_1$
$\hat{\mathbf{y}}_{<2} = [\hat{y}_1, 0,0,0,,0]$
concat
GRU
Linear
+Sigmoid
z_2
Binarize
$\hat{y}_2$
$\hat{\mathbf{y}}_{<3} = [\hat{y}_1, \hat{y}_2, 0,,,0]$
$\hat{\mathbf{y}}_{<L} = [\hat{y}_1, \hat{y}_2,,,\hat{y}_{L-1}, 0]$
concat
GRU
Linear
+Sigmoid
z_L
Binarize
$\hat{y}_L$

Figure 2: *The proposed classifier chains for AED. Each event activity is estimated iteratively and the estimation result in the previous iteration is used for conditioning next estimation.*

$$\begin{aligned} P(\mathbf{y} \mid \mathbf{X}) &= P(y_1, ..., y_L \mid \mathbf{X}) \qquad (6) \\ &= \Pi_{i=1}^{L} P(y_i \mid \cancel{y_{i-1}, ..., y_1}, \mathbf{X}). \qquad (7) \end{aligned}$$

2.2. Proposed AED with classifier chains

The proposed method introduces classifier chains for AED, designed without conditional independence assumption based on the probabilistic chain rule. The classifier of the proposed method iteratively estimates y_i not only using the latent representation $\mathbf{r}$ but also conditioning by the estimated active events $\{\hat{y}_1, ..., \hat{y}_{i-1}\}$ in the previous iterations. Therefore, the classification of the proposed method is performed assuming the following joint probability:

$$P(\mathbf{y} \mid \mathbf{X}) = \Pi_{i=1}^{L} P(y_i \mid y_1, ..., y_{i-1}, \mathbf{X}). \tag{8}$$

For the multi-label classification, it is important to approximate the joint probability of multi-class in Eq. (3). The conventional methods make the assumption of conditional independence on each class and approximate Eq. (3) with the product of the probability of each class in Eq. (7). In contrast to the conventional method, the proposed classifier does not make the conditional independence assumption, and is equivalent to Eq. (3) based on the probabilistic chain rule as Eq. (8) so that the dependency among events can be modeled. Here, the order of the classes is an essential key in the chain rule. The order used in this paper and its impact on performance will be described in detail in the following sections.

Figure 2 illustrates the proposed classifier chains. The proposed method constructs classifier chains by iteratively estimating each event's activity using gated recurrent units (GRU). First, the latent representation $\mathbf{r}$ is extracted from input audio $\mathbf{x}$ using a feature extractor. For the feature extractor, any structure can be used. In this paper, we employ the CRNN-based feature extractor, which is commonly used as a popular and strong baseline [10, 11, 26]:

$$\mathbf{R} = \text{CRNN}^{(F \to D)}(\mathbf{X}) \in \mathbb{R}^D. \tag{9}$$

The extracted latent representation $\mathbf{R} = [\mathbf{r}_1, ..., \mathbf{r}_T]$ is fed into the classifier chain and event activities $\{\hat{y}_1, ..., \hat{y}_L\}$ are estimated iteratively. When estimating $\hat{y}_i$, the proposed method uses the latent representationn $\mathbf{r}$ of input audio and the estimated active event information $\hat{\mathbf{y}}_{<i} \in \{0,1\}^L$ in the previous iterations. The active event information $\hat{\mathbf{y}}_{<i}$ is represented as a multi-hot vector with elements which correspond to detected events are one, and the others are 0. Note that $\hat{\mathbf{y}}_{<i}$ has the fixed dimention L for each iteration i and elements with indices greater than i are padded with zeros. For example, suppose the third and fourth event classes have already been estimated as active in the previous iterations. In that case, the third and fourth elements of $\hat{\mathbf{y}}_{<i}$ have 1, and others are 0, and it is used at the next iteration. Using $\mathbf{r}$ and $\hat{\mathbf{y}}_{<i}$, the estimation of $\hat{y}_i$ is as follows:

$$\mathbf{r}'_i = \text{Concatenate}\left(\mathbf{r}, \hat{\mathbf{y}}_{<i}\right) \in \mathbb{R}^{(D+L)} \tag{10}$$

$$z_i = \sigma\left(\text{Linear}^{(D+L \rightarrow 1)}(\mathbf{r}'_i)\right) \tag{11}$$

$$\hat{y}_i = \begin{cases} 1 & z_i > \epsilon_i \\ 0 & \text{otherwise}, \end{cases} \tag{12}$$

where ϵ_i is a threshold for binarizing z_i. For the initial conditioning $\hat{\mathbf{y}}_{<0}$ is set to zero.

2.3. The order of the classes for classifier chains

The order of the classes is critical for classifier chains that iteratively identify each class. In the chain rule-based speaker diarization [24], the speaker's order for the chain rule is determined beforehand, verifying every speaker's permutation, and then the classifier chains are constructed. However, the permutation of classes is a factorial of the number of classes. Even if 10-class AED, as one of the typical AED task settings, there are 3,628,800 permutation patterns, and it is unrealistic to evaluate them all. The proposed method, therefore, takes three approaches to set up the order of the classes.

Higher/Lower F1 order

Orders based on event-wise f1-scores obtained preliminary experiment by the baseline method. In this paper, the order sets up two patterns: the order of the highest performance (Higher F1) and the lowest performance (Lower F1).

Higher/Lower Frequency order

Orders of the frequency of occurrence of each class in the training data set.

Random order

Orders determined by random permutations of the class index. In this paper, we investigated multiple random values.

3. Experiments

3.1. Datasets

We conducted the experiments using two datasets; a synthetic dataset, URBAN-SED[27], and a real-recording dataset, TUT Sound Events 2017 [28]. The URBAN-SED dataset is a synthetic dataset consisting of 10 acoustic event classes with 10,000 audio clips and annotations generated using the scaper library[27]. The specification of each audio clip is a single channel, 44,100Hz, and 16-bit WAV format. The files were split into the training set of 6,000, the validation set of 2,000, and the test set of 2,000.

TUT Sound Events 2017 dataset [28] is a real-recording dataset consisting of 6 event classes with manually annotated labels. Each audio clip is recorded in a single acoustic scene, street, with a 44.1 kHz sampling rate and 24-bit resolution.

Table 1: *Network architecture of the proposed method.*

Feature Extractor
Input: (T=512 frames, F=64 bins)
2D CNN filter=(3, 3, 64) Batch normalization, Pooling = (1, 4)
2D CNN filter=(3, 3, 64) Batch normalization, Pooling = (1, 4)
2D CNN filter=(3, 3, 64) Batch normalization, Pooling = (1, 4)
Bi-GRU, Unit size= 62
Classifier Chains
GRU, Unit size= 124 Linear layer + Sigmoid
Output shape = (T=512 frames, L= # of classes*)

* 10 for URBAN-SED and 6 for TUT Sound Events 2017

Each event in the URBAN-SED dataset is randomly synthesized to the background sound, and the frequency of occurrence and overlap of each event is artificially set. On the other hand, TUT Sound Events 2017 dataset reflects the dependency among events in the real world. Using these two different types of datasets, we can measure the effectiveness of the proposed chain-classifier in capturing event dependencies.

3.2. Experimental conditions

For the neural network input, every audio clip was transformed into a log-mel spectrogram of dimension $F = 64$, a chunk of temporal frames $T = 512$ with 20 ms window length and 10 ms hop length. The network architecture of the proposed method is shown in Table 1. 2D CNN filter $= (A, B, C)$ denotes a 2D CNN layer with (A, B) filter size and C channels. Pooling= (a, b) denotes the max pooling operation on a temporal frames and b bins on the frequency axis. The mini-batch size was 32, and the number of epochs was 100. Adam [29] with learning rate 0.001 was used as the stochastic optimization method. The activity-score threshold ϵ_i in Eq. (12) was optimized for the validation set.

The evaluation metric is frame-based and segment-based (1 sec.) macro f1-score [30], which is the harmonic mean of recalls and precisions of frame/segment-wise classification results. The f1-scores were calculated for each event class, and the average of those f1-scores has been used.

For the comparison, we evaluated the CRNN with the multiple binary classifiers as "baseline", which is the strong and widely used baseline [10, 11, 26]. The difference with the proposed method was whether the classifier design is the classifier chains or the multiple binary classifier. All other conditions were set up to be common. In addition, for a more valid evaluation, we also evaluated a method with a CRNN-based feature extractor and a GRU classifier. It is exactly the same architecture as the proposed method, only without the chain rule, i.e., without conditioning path in Figure 2.

As a validation of the proposed method, we trained the classifier chains using five class orders, as described in Section 2.3. The five kinds of orders were, *Higher/Lower F1 order*,

Table 2: *AED performances of each classifier. All classifiers use CRNN for feature extractor.*

URBAN-SED		
	Frame-based	Seg.-based
Baseline	0.612	0.625
+GRU (w/o Chain)	0.624	0.635
+Chain (proposed)	**0.631**	**0.647**
TUT Sound Event 2017		
	Frame-based	Seg.-based
Baseline	0.358	0.375
+GRU (w/o Chain)	0.375	0.395
+Chain (proposed)	**0.411**	**0.426**

Table 3: *Effect of the chain-order on the frame-based F1-score. For both datasets, the higher the F1 order shows the best performance. The averages for all orders are shown with their standard deviations.*

	TUT Sound Events 2017	URBAN-SED
Higher F1	**0.411**	**0.631**
Lower F1	0.358	0.624
Higher Freq.	0.367	0.630
Lower Freq.	0.368	0.630
Random 1	0.364	0.627
Random 2	0.382	0.624
Random 3	0.355	0.628
Random 4	0.380	0.629
Random 5	0.385	0.624
Average	0.374±0.017	0.628±0.003

Higher/Lower Freq. order and *Random order*.

3.3. Evaluation results

Table 2 shows experimental results of comparison among the baseline CRNN with the multiple binary classifiers (baseline), baseline with a GRU classifier (+GRU), and the proposed method using the classifier chains (+Chain). The chain order in these results is the Higher F1 order. In both datasets, the proposed method showed the highest performance. Comparing the GRU with the baseline, we can see the improvement due to the classifier architecture, and the proposed method further improves the performance. On TUT Sound Events 2017, the proposed method showed an improvement of 3.1% on URBAN-SED and 14.8% compared to the baseline. The effect of the proposed method on URBAN-SED was limited because URBAN-SED is an artificially generated dataset with almost uniform dependency among events. On the other hand, the impact of the proposed method on TUT Sound Events, which is real-recording data, was remarkable. This indicates that the proposed method can adequately capture the real-world dependencies among events and contributes to the classification.

3.4. Impact of the chain order

Table 3 shows f1-scores of each class-order for both dataset. Among all class order settings, the Higher F1 order showed the best performance. This is a convincing result, as the easiest class is classified first and then used to condition the subsequent classes. However, the performance was slightly worse than the baseline in some orders, such as Random 3 of TUT. Moreover, the average performance of all orders' results was almost equal to that of the GRU classifier. Therefore, the (inappropriate) choice of the order may harm the performance. Particularly, for the real-recording dataset TUT Sound Events 2017, the effect of the order on performance was significant. The performance with Random order 3 is 13.6% lower than the best performance with the Higher F1 order. For the synthetic dataset, URBAN-SED, there was little difference in performance by order, and the standard deviation of performance for each order was minimal compared to TUT Sound Events 2017. This result also indicates that the effect of the proposed method is not significant for synthetic data sets where the dependency among classes is little or no class dependency.

Table 4: *Frame-based F1-scores for each event on the TUT Sound Events 2017 dataset. The event classes are listed in the Higher F1 order and the arrows indicate the order of the chain.*

	Baseline	Chain Higher F1	Chain Lower F1
brakes squeaking	**0.611**	0.573	0.532
car	0.538	**0.583**	0.568
people walking	0.487	**0.531**	0.527
large vehicle	**0.336**	0.324	0.319
people speaking	0.158	**0.380**	0.203
children	0.017	**0.073**	0.000
average	0.358	**0.411**	0.358

3.5. Event-wise AED performance

Table 4 shows event-wise AED performance. While some classes show degradation, the latter classes showed significant improvement. For example, in 'brakes squeaking,' the degradation was 6.3%, and in 'large vehicle,' the degradation was 3.5%. The class 'brakes squeaking' was the first class of the chain and classified without any conditioning, so its performance is considered to have worsened. On the other hand, there were remarkable improvements in the class 'people speaking' and class 'children.' This is due to the impact of conditioning became apparent as the chain progressed, leading to the remarkable improvement. The Lower F1 order did not classify the 'children' that were difficult to classify by the baseline. Furthermore, these results also show that when a chain is started with a class that is difficult to classify, the overall performance shows degradation. Since there are large errors in the estimated activity of the challenging class, the errors accumulate with each iteration leading to degradation of the performance in subsequent iterations .

These results show that the proposed method is very powerful, but the performance is highly dependent on the choice of class order, which is consistent with the findings of speaker diarization [24, 25].

4. Conclusion

This paper proposed acoustic event detection with the classifier chain. The proposed classifier chains consist of GRU and iteratively detect multiple events. In the experiments, the proposed method demonstrated its powerful performance comparing to the multiple binary classifiers. The experimental results also showed the importance of the class order of the chain, which has a significant impact on the performance of the proposed method. Future directions include an extension of the proposed method to weakly supervised training and investigating the effect of an order for datasets with a large number of classes, such as AudioSet [31].

5. References

[1] Johannes A Stork, Luciano Spinello, Jens Silva, and Kai O Arras, "Audio-based human activity recognition using non-markovian ensemble voting," in *IEEE RO-MAN*, 2012, pp. 509–514.

[2] Keisuke Imoto, Suehiro Shimauchi, Hisashi Uematsu, and Hitoshi Ohmuro, "User activity estimation method based on probabilistic generative model of acoustic event sequence with user activity and its subordinate categories.," in *Proc. INTERSPEECH*, 2013, pp. 2609–2613.

[3] Regunathan Radhakrishnan, Ajay Divakaran, and A Smaragdis, "Audio analysis for surveillance applications," in *Proc. WASPAA*, 2005, pp. 158–161.

[4] Chloé Clavel, Thibaut Ehrette, and Gaël Richard, "Events detection for an audio-based surveillance system," in *Proc. ICME*, 2005, pp. 1306–1309.

[5] Onur Dikmen and Annamaria Mesaros, "Sound event detection using non-negative dictionaries learned from annotated overlapping events," in *Proc. WASPAA*. IEEE, 2013, pp. 1–4.

[6] Tatsuya Komatsu, Takahiro Toizumi, Reishi Kondo, and Yuzo Senda, "Acoustic event detection method using semi-supervised non-negative matrix factorization with a mixture of local dictionaries," in *IEEE AASP Challenge: DCASE2016*, 2016, pp. 45–49.

[7] Arseniy Gorin, Nurtas Makhazhanov, and Nickolay Shmyrev, "DCASE 2016 sound event detection system based on convolutional neural network," in *IEEE AASP Challenge: DCASE2016*, 2016.

[8] Giambattista Parascandolo, Heikki Huttunen, and Tuomas Virtanen, "Recurrent neural networks for polyphonic sound event detection in real life recordings," in *Proc. ICASSP*. IEEE, 2016, pp. 6440–6444.

[9] Tomoki Hayashi, Shinji Watanabe, Tomoki Toda, Takaaki Hori, Jonathan Le Roux, and Kazuya Takeda, "Duration-controlled LSTM for polyphonic sound event detection," *IEEE TASLP*, vol. 25, no. 11, pp. 2059–2070, 2017.

[10] Emre Cakır, Giambattista Parascandolo, Toni Heittola, Heikki Huttunen, and Tuomas Virtanen, "Convolutional recurrent neural networks for polyphonic sound event detection," *IEEE TASLP*, vol. 25, no. 6, pp. 1291–1303, 2017.

[11] Sharath Adavanne, Pasi Pertilä, and Tuomas Virtanen, "Sound event detection using spatial features and convolutional recurrent neural network," in *Proc. ICASSP*, 2017, pp. 771–775.

[12] Koichi Miyazaki, Tatsuya Komatsu, Tomoki Hayashi, Shinji Watanabe, Tomoki Toda, and Kazuya Takeda, "Weakly-supervised sound event detection with self-attention," in *Proc. ICASSP*. IEEE, 2020, pp. 66–70.

[13] Qiuqiang Kong, Yong Xu, Wenwu Wang, and Mark D Plumbley, "Sound event detection of weakly labelled data with cnn-transformer and automatic threshold optimization," *IEEE TASLP*, vol. 28, pp. 2450–2460, 2020.

[14] Niko Moritz, Gordon Wichern, Takaaki Hori, and Jonathan Le Roux, "All-in-one transformer: Unifying speech recognition, audio tagging, and event detection," *Proc. Interspeech 2020*, pp. 3112–3116, 2020.

[15] Koichi Miyazaki, Tatsuya Komatsu, Tomoki Hayashi, Shinji Watanabe, Tomoki Toda, and Kazuya Takeda, "Conformer-based sound event detection with semi-supervised learning and data augmentation," in *Proc. DCASE2020 Workshop*. DCASE, 2020.

[16] K. Imoto and S. Kyochi, "Sound event detection using graph laplacian regularization based on event co-occurrence," in *Proc. ICASSP*, 2019, pp. 1–5.

[17] Konstantinos Drossos, Shayan Gharib, Paul Magron, and Tuomas Virtanen, "Language modelling for sound event detection with teacher forcing and scheduled sampling," in *Proc. DCASE2019 Workshop*, 2019, p. 59.

[18] Min-Ling Zhang and Zhi-Hua Zhou, "A review on multi-label learning algorithms," *IEEE TKDE*, vol. 26, no. 8, pp. 1819–1837, 2013.

[19] Jesse Read, Bernhard Pfahringer, Geoff Holmes, and Eibe Frank, "Classifier chains for multi-label classification," *Machine learning*, vol. 85, no. 3, pp. 333, 2011.

[20] Zhouxia Wang, Tianshui Chen, Guanbin Li, Ruijia Xu, and Liang Lin, "Multi-label image recognition by recurrently discovering attentional regions," in *Proc. ICCV*. 2017, pp. 464–472, IEEE.

[21] Wei-Cheng Chang, Hsiang-Fu Yu, Kai Zhong, Yiming Yang, and Inderjit S Dhillon, "Taming pretrained transformers for extreme multi-label text classification," in *Proc. SIGKDD*, 2020, pp. 3163–3171.

[22] Jibing Gong, Zhiyong Teng, Qi Teng, Hekai Zhang, Linfeng Du, Shuai Chen, Md Zakirul Alam Bhuiyan, Jianhua Li, Mingsheng Liu, and Hongyuan Ma, "Hierarchical graph transformer-based deep learning model for large-scale multi-label text classification," *IEEE Access*, vol. 8, pp. 30885–30896, 2020.

[23] Weiwei Cheng, Eyke Hüllermeier, and Krzysztof J Dembczynski, "Bayes optimal multilabel classification via probabilistic classifier chains," in *Proc. ICML*, 2010, pp. 279–286.

[24] Yusuke Fujita, Shinji Watanabe, Shota Horiguchi, Yawen Xue, Jing Shi, and Kenji Nagamatsu, "Neural speaker diarization with speaker-wise chain rule," *arXiv preprint arXiv:2006.01796*, 2020.

[25] Jing Shi, Xuankai Chang, Pengcheng Guo, Shinji Watanabe, Yusuke Fujita, Jiaming Xu, Bo Xu, and Lei Xie, "Sequence to multi-sequence learning via conditional chain mapping for mixture signals," in *Proc. NeurIPS*, 06 2020.

[26] Nicolas Turpault and Romain Serizel, "Training sound event detection on a heterogeneous dataset," in *Proc DCASE2020 Workshop*, 2020.

[27] Justin Salamon, Duncan MacConnell, Mark Cartwright, Peter Li, and Juan Pablo Bello, "Scaper: A library for soundscape synthesis and augmentation," in *Proc WASPAA*. IEEE, 2017, pp. 344–348.

[28] Annamaria Mesaros, Toni Heittola, and Tuomas Virtanen, "TUT database for acoustic scene classification and sound event detection," in *Proc. EUSIPCO*, Budapest, Hungary, 2016.

[29] Diederik P Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," in *Proc. ICLR*, 2015.

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

[31] Jort F. Gemmeke, Daniel P. W. 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. ICASSP*, New Orleans, LA, 2017.