CONNECTIONIST TEMPORAL LOCALIZATION FOR SOUND EVENT DETECTION WITH SEQUENTIAL LABELING

Yun Wang and Florian Metze

Language Technologies Institute, Carnegie Mellon University, Pittsburgh, PA, U.S.A.
`maigoakisame@gmail.com, fmetze@cs.cmu.edu`

ABSTRACT

Research on sound event detection (SED) with weak labeling has mostly focused on presence/absence labeling, which provides no temporal information at all about the event occurrences. In this paper, we consider SED with sequential labeling, which specifies the temporal order of the event boundaries. The conventional connectionist temporal classification (CTC) framework, when applied to SED with sequential labeling, does not localize long events well due to a "peak clustering" problem. We adapt the CTC framework and propose connectionist temporal localization (CTL), which successfully solves the problem. Evaluation on a subset of Audio Set shows that CTL closes a third of the gap between presence/absence labeling and strong labeling, demonstrating the usefulness of the extra temporal information in sequential labeling. CTL also makes it easy to combine sequential labeling with presence/absence labeling and strong labeling.

Index Terms— Sound event detection (SED), weak labeling, sequential labeling, connectionist temporal classification (CTC)

1. INTRODUCTION

Sound event detection (SED) is the task of classifying and localizing occurrences of sound events in audio streams. The training of SED models used to rely upon *strong labeling*, which specifies the type, onset time and offset time of each sound event occurrence. Such labeling, however, is very tedious to obtain by hand. In order to scale SED up, many successful attempts have been made to train SED systems with *weak labeling*, such as [2, 3, 4] and our TALNet [5]. Even though trained with weak labeling, some of these systems are able to temporally localize events in their output.

When the term *weak labeling* is used in the literature, it often specifically refers to *presence/absence labeling*, which only specifies the types of sound events present in a recording but does not provide any temporal information. Presence/absence labeling is popular because it takes the least effort to produce; as such, Audio Set [6], the currently largest corpus for SED, is also labeled this way. In this paper, however, we study SED with sequential labeling, which specifies the order of the boundaries of events occurring in each recording. We demonstrate that the extra temporal information in sequential labeling, though incomplete, can still improve the localization of sound events.

Connectionist temporal classification (CTC) [7] is a popular framework used for speech recognition when the supervision is sequential, *e.g.* phoneme sequences without temporal alignment [8]. CTC has been directly applied to SED with sequential labeling in [9] and a previous work of ours [10]; the latter found that a "peak

clustering" problem impeded the accurate localization of long sound events. In this paper, we make three major modifications to CTC and propose a connectionist temporal localization (CTL) framework, which successfully solves the peak clustering problem. Evaluation on a subset of Audio Set shows that CTL closes a third of the gap between presence/absence labeling and strong labeling.

Our CTL framework also provides a way to easily combine multiple types of labeling, such as presence/absence labeling, sequential labeling, and strong labeling. When we have stronger labeling available in a smaller amount and weaker labeling available in a larger amount, such a combination makes it possible to fully exploit the information in all the data.

2. CTL: MOTIVATION AND ALGORITHM

2.1. Sequential Labeling

In speech recognition, a typical form of supervision is a phoneme sequence for each utterance without temporal alignment. A direct analogy for SED would be a sequence of sound events for each recording, but the order of sound events can be hard to define when they overlap. To avoid this problem, we define sequential labeling to be a sequence of event boundaries. For example, if the content of a recording can be described as "a dog barks while a car passes by", the sequence of event boundaries will be: car onset, dog onset, dog offset, car offset. We denote this by $\underline{\acute{C}\acute{D}\grave{D}\grave{C}}$: letters with the rising accent $\acute{C}$, $\acute{D}$ stand for the onsets of the "car" and "dog" events, while letters with the falling accent $\grave{C}$ and $\grave{D}$ stands for their offsets; the underline means this is a sequence without temporal alignment.

For annotators, sequential labeling is not too much harder to produce than presence/absence labeling; the difficulty mainly arises when sound events occur densely or overlap. In any case, it is still easier to produce than strong labeling, because it is not necessary to mark the precise onset and offset times of each sound event occurrence. Also, sequential labeling may be automatically mined from textual descriptions of audio recordings, such as "a dog barks while a car passes by".

2.2. The Peak Clustering Problem of CTC

CTC can be applied to SED with sequential labeling as follows. First, we define the vocabulary of CTC output to include the onset and offset labels of each event type, plus a "blank" label (denoted by $-$). For an SED system that deals with n types of events, the vocabulary size is $2n + 1$. A neural network (often with a recurrent layer) predicts the frame-wise probability of each label in the vocabulary; these probabilities sum to 1 at each frame. The probabilities of specific temporal alignments (*e.g.* $-\acute{C}\acute{D}\grave{D}\grave{C}-$, $\acute{C}-\acute{D}\grave{D}\grave{C}\grave{C}$) can be calculated by multiplying the probabilities of individual labels at each frame. The total probability of the ground truth sequence (*e.g.* $\underline{\acute{C}\acute{D}\grave{D}\grave{C}}$) is defined as the sum of the probabilities of all alignments

that can be reduced to the ground truth sequence by a many-to-one mapping $\mathcal{B}$; this mapping first collapses all consecutive repeating labels into a single one, then removes all blank labels. For example, both the alignments $-\acute{C}\acute{D}\grave{D}\grave{C}-$ and $\acute{C}-\acute{D}\grave{D}\grave{C}\grave{C}$ can be reduced to the unaligned sequence $\underline{\acute{C}\acute{D}\grave{D}\grave{C}}$, therefore $P(\underline{\acute{C}\acute{D}\grave{D}\grave{C}}) = P(-\acute{C}\acute{D}\grave{D}\grave{C}-) + P(\acute{C}-\acute{D}\grave{D}\grave{C}\grave{C})$ plus the probabilities of many other alignments. A systematic forward algorithm is proposed in [7] to compute this total probability efficiently. The loss function for this recording is defined as $-\log P(\underline{\acute{C}\acute{D}\grave{D}\grave{C}})$; this can be minimized with any neural network training algorithm, such as gradient descent.

When CTC is directly applied to SED with sequential labeling, it has been found in [10] to detect short events well: a peak appears in the frame-wise probabilities of the onset and offset labels around the actual occurrence of the event. For long events, however, CTC tends to predict peaks for the onset and the offset next to each other, which means the event is not well localized (see Sec. 3.3 for an example).

This "peak clustering" problem occurs for several reasons. First, because sound events do not overlap too often, adjacent onset and offset labels are an extremely common pattern in the training label sequences. As a result, CTC may misunderstand a pair of onset and offset labels as collectively indicating the existence of an event, instead of understanding them as separately indicating the event boundaries. Second, the CTC loss function only mandates the order of the predicted labels, without imposing any temporal constraints. In this case, the recurrent layer of the network will prefer to emit onset and offset labels next to each other, because this minimizes the effort of memory. The root cause of the "peak clustering" problem is that the output layer of the network is only trained to detect event *boundaries*; it is expected to keep "silent" both when an event is inactive and when an event is continuing, despite the potentially huge differences in the acoustic features. When the network predicts the onset and offset labels of a long event occurrence next to each other, it actually does not violate this expectation on too many frames, and does not have enough incentive to correct this behavior.

2.3. Connectionist Temporal Localization

In this section we make three major modifications to the CTC framework, and present a connectionist temporal localization (CTL) framework suitable for localizing sound events. We also describe the corresponding forward algorithm for calculating the total probability of an event boundary sequence.

The first modification addresses the root cause of the "peak clustering" problem: *the output layer of the network should predict the frame-wise probabilities of the events themselves instead of those of the event boundaries*. In this way, the network can learn to make different predictions with different acoustic features. The boundary probabilities are then derived from the event probabilities using a "rectified delta" operator. More formally, let $y_t(\text{E})$ be the probability of the event E being active at frame t. Here $1 \leq t \leq T$, where T is the number of frames in the recording in question. Let $z_t(\acute{\text{E}})$ and $z_t(\grave{\text{E}})$ be the probabilities of the onset and offset labels of the event E at frame t. We calculate them using the following equations:

$$z_t(\acute{\text{E}}) = \max[0, y_t(\text{E}) - y_{t-1}(\text{E})]$$
$$z_t(\grave{\text{E}}) = \max[0, y_{t-1}(\text{E}) - y_t(\text{E})] \tag{1}$$

In these equations we allow t to range from 1 to $T + 1$, in order to accommodate events that start at the first frame or end at the last frame. When $y_0(\text{E})$ or $y_{T+1}(\text{E})$ is referenced, we assume it to be 0.

Now we have the frame-wise probabilities of all event boundaries, we only need to define the frame-wise probability of the blank. However, a difficulty arises because the sum of the boundary probabilities at a given frame may exceed 1. To solve this problem, we make the second modification to CTC: *we treat the probabilities of different event boundaries at the same frame as mutually independent, instead of mutually exclusive*. In this way, the probability of no event boundaries occurring at frame t can be calculated by:

$$\epsilon_t = \prod_l [1 - z_t(l)] \tag{2}$$

where l goes over all event boundaries. The probability of emitting a single event boundary l at frame t is then:

$$p_t(l) = z_t(l) \cdot \prod_{l' \neq l} [1 - z_t(l')] \tag{3}$$

If we define

$$\delta_t(l) = \frac{z_t(l)}{1 - z_t(l)} \tag{4}$$

Then Eq. 3 reduces to

$$p_t(l) = \epsilon_t \cdot \delta_t(l) \tag{5}$$

The assumption that boundary labels at the same frame are mutually independent seems to eliminate the need for the blank label. Indeed, the blank label in CTC serves two purposes: (1) to allow emitting nothing at a frame, and (2) to separate consecutive repetitions of the same label. With the independence assumption, the first purpose is naturally achieved. Here we make the third modification to CTC: *the mapping $\mathcal{B}$ no longer collapses consecutive repeating labels into a single one*. With this simplification, the blank label can be removed altogether.

The independence assumption also allows us to assess the probability of emitting multiple labels at the same frame, which is not possible with the standard CTC. The probability of emitting multiple labels $l_1, \ldots, l_k$ together at frame t can be calculated as

$$p_t(l_1, \ldots, l_k) = \prod_{i=1}^k z_t(l_i) \cdot \prod_{l \notin \{l_1, \ldots, l_k\}} [1 - z_t(l)]$$
$$= \epsilon_t \cdot \prod_{i=1}^k \delta_t(l_i) \tag{6}$$

Now we can formulate our CTL forward algorithm. What we want to find is the total probability of emitting the ground truth label sequence $L = l_1, \ldots, l_{|L|}$, regardless of the temporal alignment. What we are given is the frame-level probabilities of events $y_t(\text{E})$, from which we can derive the probability $p_t(\cdot)$ of emitting zero, one or more labels at each frame by Eq. 6. Let $\alpha_t(i)$ be the probability of having emitted exactly the first i labels of L after t frames. The α's can be computed with the following recurrence formula:

$$\alpha_t(i) = \sum_{j=0}^i \alpha_{t-1}(i-j) \cdot p_t(l_{i-j+1}, \ldots, l_i)$$
$$= \sum_{j=0}^i \alpha_{t-1}(i-j) \cdot \epsilon_t \cdot \prod_{k=i-j+1}^i \delta_t(l_k) \tag{7}$$

In the summation, the index j stands for the number of labels emitted at frame t. The initial values are:

$$\alpha_0(i) = \begin{cases} 1, & \text{if } i = 0 \\ 0, & \text{if } i > 0 \end{cases} \tag{8}$$

The final value, $\alpha_{T+1}(|L|)$, is the total probability of emitting the label sequence L, and its negative logarithm is the contribution of the recording in question to the loss function.

Eq. 7 allows emitting arbitrarily many labels at the same frame. When the ground truth label sequence is long, this can pose a problem of time complexity. In practice, it is rare for multiple labels to be emitted at the same frame. Therefore, it can be desirable to limit the maximum number of concurrent labels, *i.e.* the maximum value of j in Eq. 7. We call this maximum value the *max concurrence*.
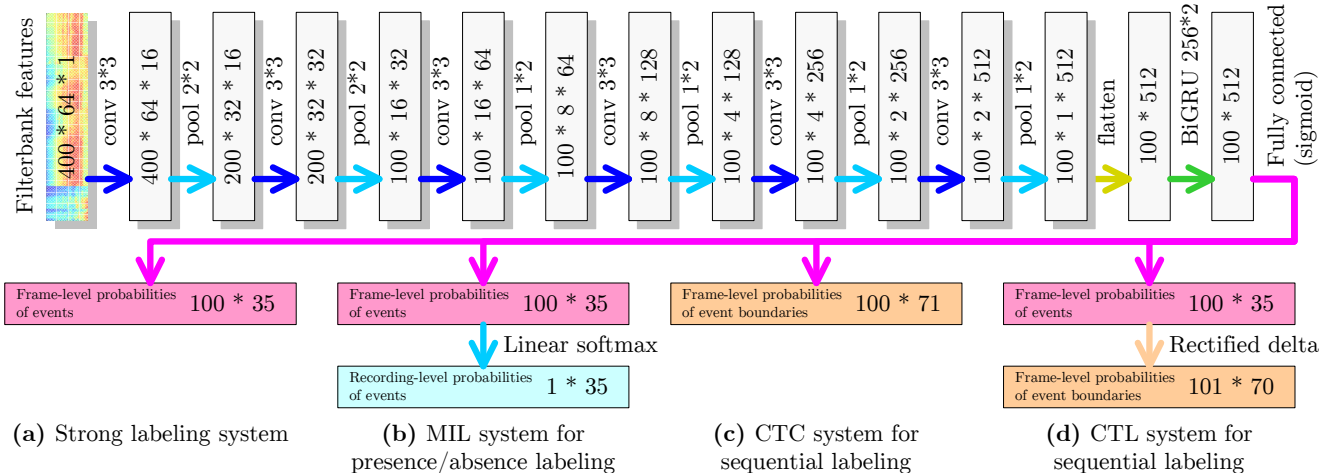
(a) Strong labeling system

(b) MIL system for presence/absence labeling

(c) CTC system for sequential labeling

(d) CTL system for sequential labeling

Fig. 1. Structures of the four networks trained in Sec. 3.2. The shape is specified as "frames * frequency bins * feature maps" for 3-D tensors (shaded), and "frames * feature maps" for 2-D tensors. "conv $n*m$" stands for a convolutional layer with the specified kernel size and ReLU activation; batch normalization is applied before the ReLU activation. "pool $n*m$" stands for a max pooling layer with the specified stride.

3. EXPERIMENTS

3.1. Data Preparation

We carried out experiments on a subset of Audio Set [6]. Audio Set consists of over 2 million 10-second excerpts of YouTube videos, labeled with the presence/absence of 527 types of sound events. Because we would need sequential labeling for training and strong labeling for evaluation, we generated sequential and strong labeling for all the recordings using TALNet [5] – a state-of-the-art network trained with presence/absence labeling that is good at localizing sound events. We used a frame length of 0.1 s, so each recording consisted of 100 frames.

Not all of the 527 sound events types of Audio Set were labeled with high quality, and the labels generated by TALNet would be even noisier. To reduce the effect of such label noise, we selected 35 sound event types that had relatively reliable labels (see Table 4.1 of [11] for a complete list). Four of these event types (`speech`, `sing`, `music` and `crowd`) were overwhelmingly frequent; we filtered the recordings of Audio Set to retain only those that contained at least one of the remaining 31 types of sound events. This left us with 359,741 training recordings, 4,879 validation recordings and 5,301 evaluation recordings. The total duration of these recordings is around 1,000 hours, or 18% of entire corpus.

3.2. Network Structures and Training

We trained four networks whose structures are illustrated in Fig. 1. All the layers up to the GRU layer are shared across the four networks; these layers highly resemble the hidden layers of TALNet [5], but are shallower and narrower. The four systems have different output ends. The first system predicts the probabilities of the 35 types of sound events, and directly receives strong labeling as supervision. The second system is a multiple instance learning (MIL) system for presence/absence labeling: it first predicts frame-wise probabilities, then aggregates them into recording-level probabilities with a linear softmax pooling function just like TALNet. These two systems serve as the topline and the baseline for the CTC and CTL systems. The CTC system directly predicts the frame-wise probabilities of event boundaries and the blank label; the output layer has $35*2+1=71$

System		Loc. F_1 (%)
Strong labeling (topline)		67.38
MIL (baseline)		55.83
CTC		31.91
CTL	Max concurrence = 1	59.92
	Max concurrence = 2	57.49
	Max concurrence = 3	53.63

Table 1. Localization performance of the four systems.

units. The CTL system predicts the frame-wise probabilities of the events and then derives the boundary probabilities with the "rectified delta" operator. We tried max concurrence values of 1, 2 and 3.

The systems were trained using the Adam optimizer [12] with a constant learning rate of 10^{-3}. The batch size was 500 recordings. We applied data balancing to ensure that each minibatch contained roughly equal numbers of recordings of each event type. After every 200 minibatches (called a *checkpoint*), we evaluated the network's localization performance using the frame-level F_1 macro-averaged across the 35 event types. For the strong labeling, MIL and CTL systems, we first tuned class-specific thresholds to optimize the frame-level F_1 of each event type on the validation data, then applied them directly to the evaluation data. For the CTC system, we picked the most probable label at each frame, and marked each event as active between innermost matching pairs of onset and offset labels.

3.3. Performance of CTL for Sequential Labeling

Table 1 lists the highest evaluation F_1 obtained by the various systems within 100 checkpoints. The CTC system falls long behind the baseline; as we shall see, this is due to the "peak clustering" problem. The CTL system (with a max concurrence of 1) successfully outperforms the baseline, and closes a third of the gap between the baseline of MIL with presence/absence labeling and the topline of strong labeling. A class-wise error analysis shows that the CTL system exhibits a uniform improvement across classes, outperforming the MIL baseline for 28 of the 35 event types. In addition, it appears unnecessary to allow multiple labels to occur at the same frame.

Fig. 2 presents the output of the four systems on an evaluation recording, which contains the whining of a dog intermingled with
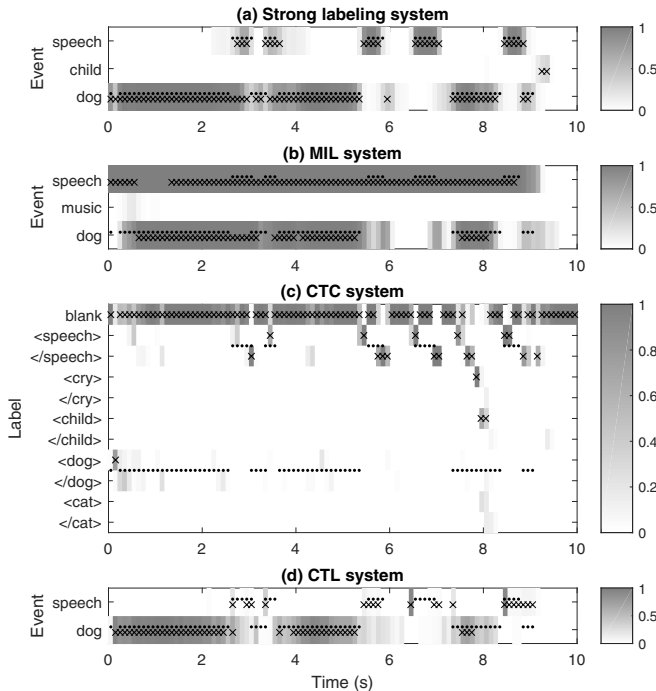
Fig. 2. The frame-level predictions of the four systems on the evaluation recording `0F04c_rY4aw`. Dots stand for the ground truth; shades of gray indicate the frame-level probabilities of events, event boundaries or the blank label. Crosses indicate the most probable label at each frame (for the CTC system), or events with probabilities higher than the class-specific thresholds (for the other systems). `<E>` and `</E>` stand for the onset and offset labels of the event E. Unimportant events are omitted.

speech. The topline strong labeling system localizes both events well; the baseline MIL system fails to localize the speech event. The CTC system can localize the occurrences of speech (although with a few spurious detections); for the dog event, however, it exhibits the "peak clustering" problem: it predicts (with low confidence) many pairs of onset and offset labels of dog next to each other. The CTL system avoids the "peak clustering" problem, and also localizes the speech occurrences better than the MIL system.

3.4. Combining Sequential Labeling with Presence/Absence Labeling

When sequential labeling is available for training a SED system, presence/absence labeling is automatically also available. This prompts us to think about combining a CTL system trained with sequential labeling and an MIL system trained with presence/absence labeling. Because the two systems share all layers up to the frame-wise probabilities of events, this combination turns out to be surprisingly easy: it suffices to combine the loss functions of the two systems using a weighted average. At test time, the localization output can be directly taken from the shared layer of frame-wise event probabilities. In contrast, it is more difficult to combine a CTC system with an MIL system because they have different output ends.

We combined an MIL system with CTL systems trained with different values of max concurrence: 1, 2 and 3. When we trained the systems alone, we found that the loss of the CTL systems usually stabilized around 0.2, while the loss of the MIL system usually stabilized around 0.02. For the combination experiments,
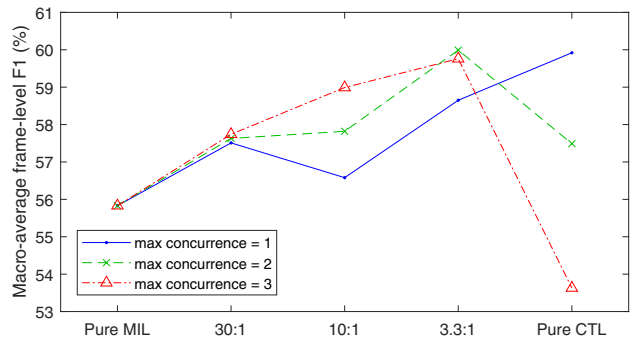

Fig. 3. The localization performance obtained by combining CTL and MIL with different weights.

we fixed the weight of the CTL loss to 1, and tried out the following weights for the MIL loss: 30 (emphasizing the MIL loss more), 10 (weighting both losses equally), and 3.3 (emphasizing the CTL loss more). The resulting localization performances are plotted in Fig. 3. A mixing weight of 3.3:1 appears to be generally a good choice, and gives a marginal improvement on top of pure CTL.

The potential use of combining a CTL system with other systems is not limited to the experiments above. Because sequential labeling takes more effort to produce than presence/absence labeling after all, it can be well imagined that there will be less data with sequential labeling available than data with presence/absence labeling. System combination allows us to exploit the information in both types of labeling: we can compute the MIL loss on all the data and the CTL loss on the part of the data with sequential labeling, and train a system to minimize an appropriate weighted average of the two loss functions. If we also have data with strong labeling, then the frame-wise cross-entropy loss of a strong labeling system can be added to the weighted average, too. A CTL system can be combined with an MIL system and a strong labeling system with no effort, thanks to the fact that it computes frame-wise probabilities of events in the same way as the other two systems.

4. CONCLUSION AND DISCUSSION

We made three modifications to the connectionist temporal classification (CTC) framework: (1) instead of predicting frame-wise boundary probabilities directly, the network predicts event probabilities and then derives boundary probabilities using a "rectified delta" operator; (2) the boundary probabilities at the same frame are regarded as mutually independent instead of mutually exclusive; (3) the mapping $\mathcal{B}$ from alignments to unaligned label sequences no longer collapses consecutive repeating labels. The resulting framework, which we name "connectionist temporal localization" (CTL), successfully solves the "peak clustering" problem of CTC, and closes a third of the gap between the baseline of presence/absence labeling and the topline of strong labeling.

Because a CTL system predicts frame-wise event probabilities in the same way as an MIL system for presence/absence labeling and a strong labeling system, the combination of the three systems is as easy as a weighted average of the loss functions. This makes it possible to exploit the information in all three types of labeling when we have different data labeled at different granularities.

For more details about the CTL algorithm and the experiments, please refer to Chapter 4 of the first author's PhD thesis [11]. The code and acoustic features for the experiments are available at https://github.com/MaigoAkisame/cmu-thesis.

5. REFERENCES

[1] J. Towns, T. Cockerill, M. Dahan, I. Foster, K. Gaither, A. Grimshaw, V. Hazlewood, S. Lathrop, D. Lifka, G. D. Peterson, R. Roskies, J. R. Scott, and N. Wilkins-Diehr, "XSEDE: Accelerating scientific discovery," *Computing in Science & Engineering*, vol. 16, no. 5, pp. 62–74, 2014.

[2] S. Hershey *et al.*, "CNN architectures for large-scale audio classification," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, IEEE, 2017, pp. 131–135.

[3] Q. Kong, Y. Xu, W. Wang, and M. D. Plumbley, "Audio set classification with attention model: A probabilistic perspective," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, IEEE, 2018, pp. 316–320.

[4] C. Yu, K. S. Barsim, Q. Kong, and B. Yang, "Multi-level attention model for weakly supervised audio classification," *arXiv e-prints*, 2018. [Online]. Available: http://arxiv.org/abs/1803.02353.

[5] Y. Wang, J. Li, and F. Metze, "A comparison of five multiple instance learning pooling functions for sound event detection with weak labeling," *arXiv e-prints*, 2018. [Online]. Available: http://arxiv.org/abs/1810.09050.

[6] J. F. Gemmeke, D. P. W. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, "Audio Set: An ontology and human-labeled dataset for audio events," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, IEEE, 2017, pp. 776–780.

[7] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks," in *International Conference on Machine Learning (ICML)*, ACM, 2006, pp. 369–376.

[8] A. Graves and N. Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *International Conference on Machine Learning (ICML)*, ACM, 2014, pp. 1764–1772.

[9] Y. Hou, Q. Kong, J. Wang, and S. Li, "Polyphonic audio tagging with sequentially labelled data using crnn with learnable gated linear units," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2017 Workshop (DCASE2017)*, 2018, pp. 78–82.

[10] Y. Wang and F. Metze, "A first attempt at polyphonic sound event detection using connectionist temporal classification," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, IEEE, 2017, pp. 2986–2990.

[11] Y. Wang, "Polyphonic sound event detection with weak labeling," PhD thesis, Carnegie Mellon University, 2018.

[12] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv e-prints*, 2014. [Online]. Available: http://arxiv.org/abs/1412.6980.