A simple model for detection of rare sound events

Weiran Wang, Chieh-chi Kao, Chao Wang

Amazon Alexa
101 Main St, Cambridge, MA 02142, USA
{weiranw,chiehchi,wngcha}@amazon.com

Abstract

We propose a simple recurrent model for detecting rare sound events, when the time boundaries of events are available for training. Our model optimizes the combination of an utterance-level loss, which classifies whether an event occurs in an utterance, and a frame-level loss, which classifies whether each frame corresponds to the event when it does occur. The two losses make use of a shared vectorial representation the event, and are connected by an attention mechanism. We demonstrate our model on Task 2 of the DCASE 2017 challenge, and achieve competitive performance.

1. Introduction

The task of detecting rare sound events from audio has drawn much recent attention, due to its wide applicability for acoustic scene understanding and audio security surveillance. The goal of this task is to classify if certain type of event occurs in an audio segment, and when it does occur, detect also the time boundaries (onset and offset) of the event instance.

The task 2 of DCASE 2017 challenge provides an ideal testbed for detection algorithms [1]. The data set consists of isolated sound events for three target classes (baby crying, glass breaking, and gun shot) embedded in various everyday acoustic scenes as background. Each utterance contains at most one instance of the event type, and the data generation process provides temporal position of the event which can be used for modeling.

The most direct solution to this problem is perhaps to model the hypothesis space of segments, and to predict if each segment corresponds to the time span of the event of interest. This approach was adopted by [2] and [3], whose model architecture heavily drew inspirations from the region proposal networks [4] developed in the computer vision community. There are a large number of hyper-parameters in such models, which requires much human guidance in tuning. More importantly, this approach is generally slow to train and test, due to the large number of segments to be tested.

Another straight-forward approach to this task is to generate reference labels for each frame indicating if the frame correspond to the event, and then train a classifier to predict the binary frame label. This was indeed the approach taken by many participants of the challenge (e.g., [5, 6]). The disadvantage of this approach is that it does not directly provide an utterance-level prediction (if an event occurs at all), and thus requires heuristics to aggregate frame-level evidence for that. It is the motivation of our work to solve this issue.

We propose a simple model for detecting rare sound events without aggregation heuristics for utterance-level prediction. Our learning objective combines a frame-level loss similar to the abovementioned approach, with an utterance-level loss that automatically collects the frame-level evidence. The two losses share a single classifier which can be seen as the vectorial representation of the event, and they are connected by an attention mechanism. Additionally, we use multiple layers of recurrent neural networks (RNNs) for feature extraction from the raw features, and we propose an RNN-based multi-resolution architecture that consistently improve over the standard multi-layer bi-directional RNNs architectures for our task. In the rest of this paper, we discuss our learning objective in Section 2, introduce the multi-resolution architecture in Section 3, demonstrate them on the DCASE challenge in Section 4, and provide concluding remarks in Section 5.

2. Our model

Denote an input utterance by $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_T]$ where $\mathbf{x}_i \in \mathbb{R}^d$ contains the audio features for the i-th frame. For our task (detecting a single event at a time), we are given the binary utterance label y which indicates if an event occurs ($y = 1$) or not ($y = 0$). If $y = 1$, we have additionally the onset and offset time of the event, or equivalently frame label $\mathbf{y} = [y_1, \dots, y_T]$, where $y_t = 1$ if the event is on at frame t and $y_t = 0$ otherwise. Our goal is to make accurate predictions at both the utterance level and the frame level.

Our model uses a multi-layer RNN architecture f to extract nonlinear features from $\mathbf{X}$, which yields a new representation

$$f(\mathbf{X}) = [\mathbf{h}_1, \dots, \mathbf{h}_T] \in \mathbb{R}^{h \times T},$$

containing temporal information. We also learn a vectorial representation of the acoustic event by $\mathbf{w} \in \mathbb{R}^h$, which serves the purpose of a classifier and will be used in predictions at two levels.

With the standard logistic regression model, we perform per-frame classfication based on the frame-level representation and the classifier $\mathbf{w}$: for $t = 1, \dots, T$,

$$p_t := P(y_t = 1|\mathbf{X}) = \frac{1}{1 + \exp\left(-\mathbf{w}^\top \mathbf{h}_t\right)} \in [0, 1],$$

and we measure the frame-level loss if the event occurs:

$$\mathcal{L}_{frame}(\mathbf{X}, \mathbf{y}) = \begin{cases} \frac{1}{T} \sum_{t=1}^{T} y_t \log p_t + (1 - y_t) \log(1 - p_t) & : y = 1 \\ 0 & : y = 0 \end{cases}.$$

Note that we do not calculate the frame loss if no event occurs, even though one can consider the frame label to be all 0's in this case. This design choice is consistent with the evaluation metric for rare events, since if we believe no event occurs in an utterance, the onset/offset or the frame labels are meaningless.

On the other hand, we make the utterance-level prediction by collecting evidence at the frame level. Since the above p_t's provide the alignment between each frame and the target event,

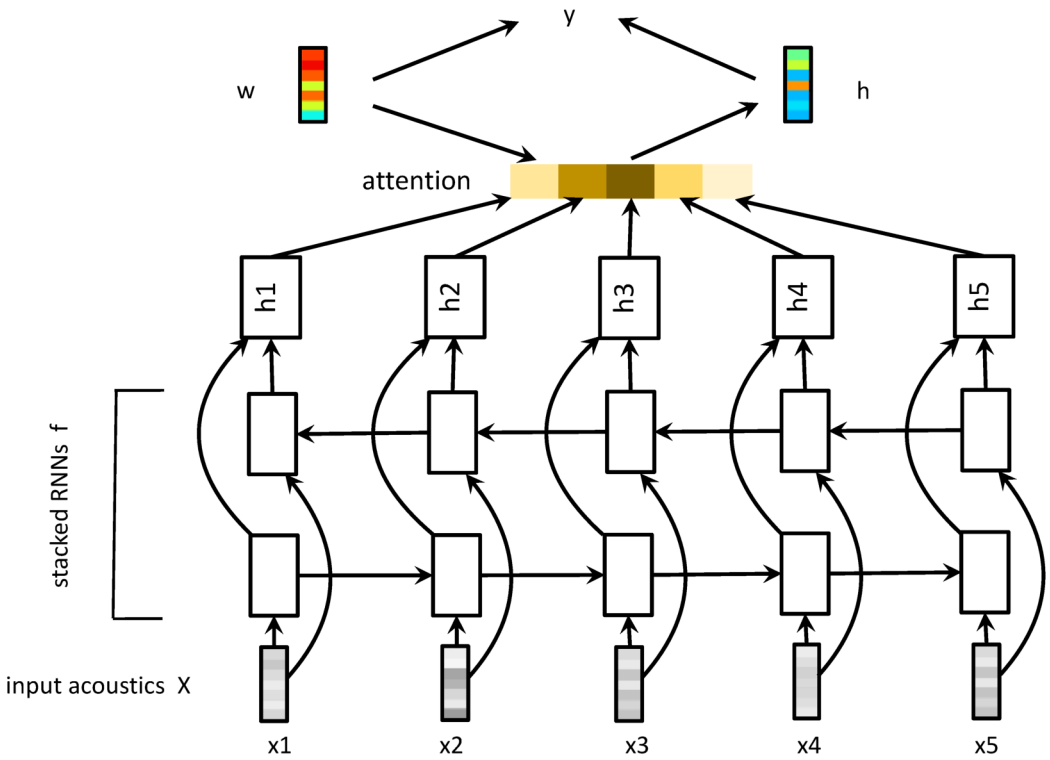

Figure 1: *Illustration of our RNN-based attention mechanism for rare sound events detection.*

we normalize them over the entire utterance to give the "attention" [7, 8]:

$$a_t = \frac{p_t}{\sum_{t=1}^{T} p_t}, \qquad t = 1, \ldots, T,$$

and use these attention weights to combine the frame representations to form the utterance representation as

$$\mathbf{h} = \sum_{t=1}^{T} a_t \mathbf{h}_t.$$

We make utterance-level prediction by classifying $\mathbf{h}$ using $\mathbf{w}$:

$$p := P(y = 1|\mathbf{X}) = \frac{1}{1 + \exp\left(-\mathbf{w}^\top \mathbf{h}\right)} \in [0, 1],$$

and define the utterance-level loss based on it:

$$\mathcal{L}_{utt}(\mathbf{X}, y) = y \log p + (1 - y) \log(1 - p).$$

This loss naturally encourages the attention to be peaked at the event frames (since they are better aligned with $\mathbf{w}$), and low at the non-event frames.

Our final objective function is a weighted combination of the two above losses:

$$\mathcal{L}(\mathbf{X}, y, \mathbf{y}) = \mathcal{L}_{utt}(\mathbf{X}, y) + \alpha \cdot \mathcal{L}_{frame}(\mathbf{X}, \mathbf{y}),$$

where $\alpha > 0$ is a trade-off parameter. During training, we optimize $\mathcal{L}(\mathbf{X}, y, \mathbf{y})$ jointly over the parameters of RNNs $\mathbf{f}$ and the event representation $\mathbf{w}$. An illustration of our model is given in Figure 1.

2.1. Inference

For a test utterance, we first calculate p and predict that no event occurs if $p \leq thres_0$, and in the case of $p > thres_0$ which indicates that an event occurs, we threshold $[p_1, \ldots, p_T]$ by $thres_1$ to predict if the event occurs at each frame. For the DCASE challenge task 2, where we need to output the time boundary for a predicted event (and there is at most one event in each utterance), we simply return the boundary of the longest connected component of 1's in the thresholded frame prediction. We have simply used $thres_0 = thres_1 = 0.5$ in our experiments.

3. Multi-resolution feature extraction

Different instances of the same event type may occur with somewhat different speeds and durations. To be robust to variations in the time axis, we propose a multi-resolution feature extraction architecture based on RNNs, as depicted in Figure 2, which will be used as the $f(\mathbf{X})$ mapping in our model.

This architecture works as follows. After running each recurrent layer, we perform subsampling in the time axis with a rate of 2, i.e., the outputs of the RNN cell for two neighboring frames are averaged, and the resulting sequence, whose length is half of the input length of this layer, is then used as input to the next recurrent layer. In such a way, the higher recurrent layers effectively view the original utterance at coarser resolutions (larger time scales), and extract information from increasingly larger context of the input.

After the last recurrent layer, we would like to obtain a representation for each of the input frames. This is achieved by upsampling (replicating) the subsampled output sequences from each recurrent layer, and summing them for corresponding frames. Therefore, the final frame representation produced by this architecture takes into account information at different

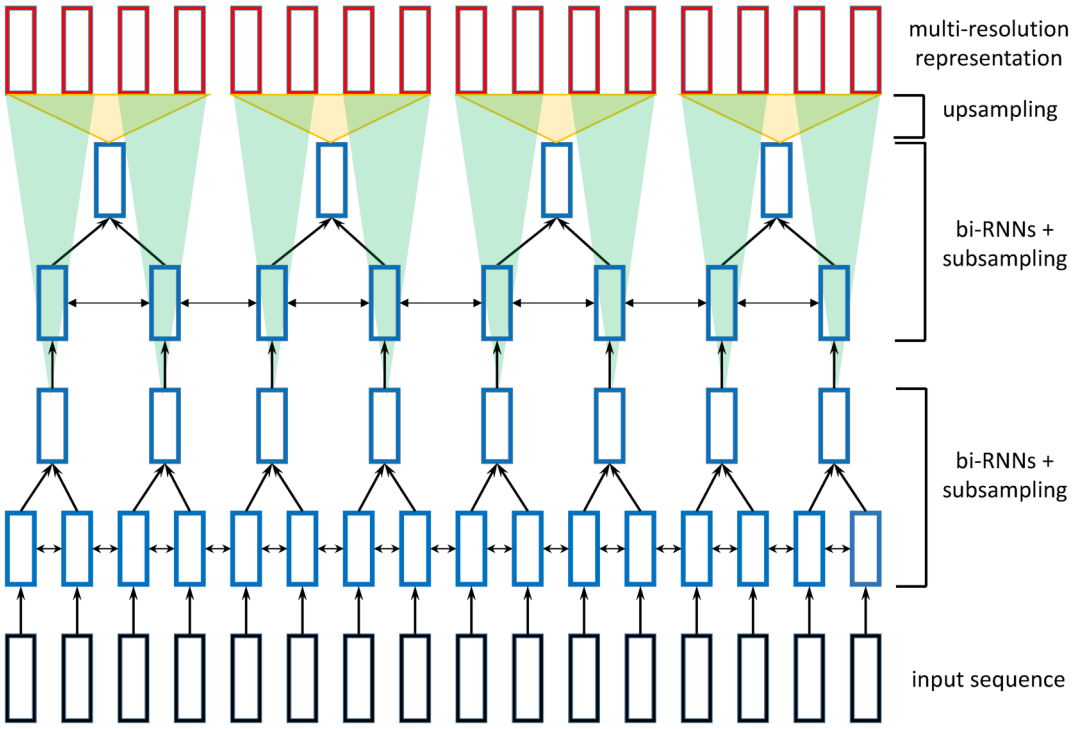

Figure 2: *RNN-based multi-resolution modeling.*

resolutions. We note that the idea of subsampling in deep RNNs architecture is motivated by that used in speech recognition [9], and the idea of connecting lower level features to higher layers is similar to that of resnet [10]. We have implemented our model in the tensorflow framework [11].

4. Experimental results

Data generation We demonstrate our rare event detection model on the task 2 of DCASE 2017 challenge [12] . The task data consist of isolated sound events for three target events (babycry, glassbreak, gunshot), and recordings of 15 different audio scenes (bus, cafe, car, etc.) used as background sounds from TUT Acoustic Scenes 2016 dataset [13]. The synthesizer provided as a part of the DCASE challenge is used to generate the training set, and the mixing event-to-background ratios (EBR) are -6, 0 and 6 dB. The generated training set has 5000 or 15000 utterances for each target class, and each utterance contains either one target class event or no events. We use the same development and evaluation set (both of about 500 utterances) provided by the DCASE challenge.

Feature extraction The acoustic features used in this work are log filter bank energies (LFBEs). The feature extraction operates on mono audio signals sampled at 44.1 kHz. For each 30 seconds audio clip, we extract 64 dimensional LFBEs from frames of 46 ms duration with shifts of 23 ms.

Evaluation metrics The evaluation metrics used for audio event detection in DCASE 2017 are event-based error rate (ER) and F1-score. These metrics are calculated using onset-only condition with a collar of 500 ms, taking into account insertions, deletions, and substitutions of events. Details of these metrics can be found in [12].

Table 1: *ER results of our model on the development set for different RNN architectures. Here the training set size is* 5000*, and we fix the number of GRU layers to be* 3*.*

	babycry	glassbreak	gunshot
uni-directional	0.24	0.06	0.31
bi-directional	0.18	0.07	0.26
multi-resolution	0.13	0.04	0.20

4.1. Training with 5K samples

For each type of event, we first explore different architectures and hyperparameters on training sets of 5000 utterances, 2500 of which contain the event. This training setup is similar to that of several participants of the DCASE challenge.

For the frame-level loss $\mathcal{L}_{frame}$, instead of summing the cross-entropy over all frames in a positive utterance, we only consider frames near the event and in particular, from 50 frames before the onset to 50 frames after the offset. In this way, we obtain a balanced set of frames (100 negative frames and a similar amount of positive frames per positive utterance) for $\mathcal{L}_{frame}$.

Our models are trained with the ADAM algorithm [14] with a minibatch size of 10 utterances, an initial stepsize of 0.0001, for 15 epochs. We tune the hyperparameter α over the grid $\{0.1, 0.5, 1, 5, 10\}$ on the development set. For each α, we monitor the model's performance on the development set, and select the epoch that gives the lowest ER.

4.1.1. Effect of RNN architectures

We explore the effect of RNN architectures for the frame feature transformation. We test 3 layers of uni-directional, bi-

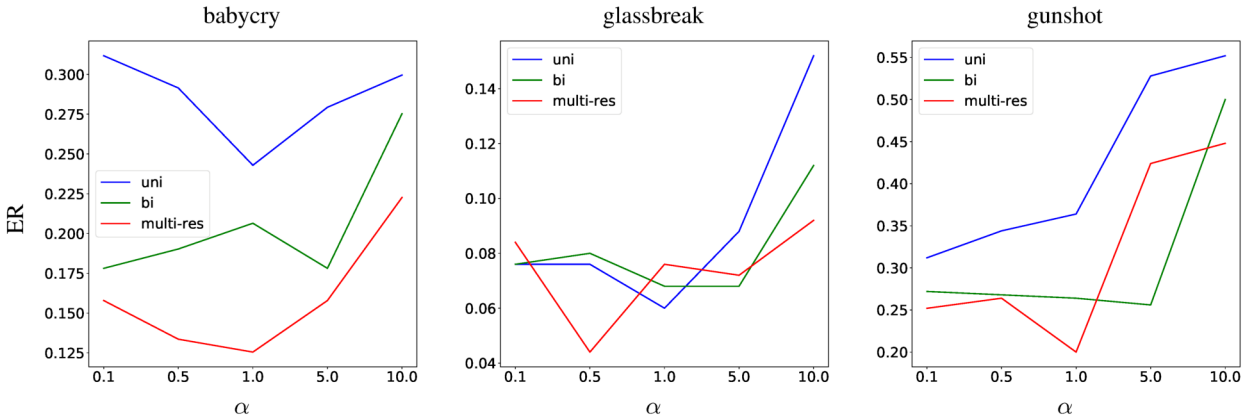

Figure 3: *Performance of different RNN architectures for a range of* α*. Here the training set size is* 5000*.*

Table 2: *Performance of our model with 15000 training samples and* 4 *GRU layers.*

	Methods	babycry		glassbreak		gunshot		average	
		ER	F1 (%)	ER	F1 (%)	ER	F1 (%)	ER	F1 (%)
Development set	Ours	0.11	94.3	0.04	97.8	0.18	90.6	0.11	94.2
	DCASE Baseline							0.53	72.7
	DCASE 1st place [5]							0.07	96.3
	DCASE 2nd place [6]							0.14	92.9
Evaluation set	Ours	0.26	86.5	0.16	92.1	0.18	91.1	0.20	89.9
	DCASE Baseline							0.64	64.1
	DCASE 1st place							0.13	93.1
	DCASE 2nd place							0.17	91.0

directional, and multi-resolution RNNs described in Section 3 for $f(\mathbf{X})$. The specific RNN cell we use is the standard gated recurrent units [15], with 256 units in each direction. We observe that bi-directional RNNs tend to outperform uni-directional RNNs, and on top of that, the multi-resolution architecture brings further improvements on all events types.

4.1.2. *Effect of the* α *parameter*

In Figure 3, we plot the performance of different RNN architectures at different values of trade-off parameter α. We observe that there exists a wide range of α for which the model achieves good performance. And for all three events, the optimal α is close to 1, placing equal weight on the utterance loss and frame loss.

4.2. Training with 15K samples

For each type of event, we then increase the training set to 15000 utterances, 7500 of which contain the event. We use 4 GRU layers in our multi-resolution architecture, and set $\alpha = 1.0$. Training stops after 10 epochs and we perform early stopping on the development set as before.

The results of our method, in terms of both ER and F1-score, are given in Table 2. With the larger training set and deeper architecture, our development set ER performance is further improved on babycry and gunshort; the averge ER of 0.11 is only worse than the first place's result of 0.07 among all challenge participants.

5. Conclusion

We have proposed a new recurrent model for rare sound events detection, which achieves competitive performance on Task 2 of the DCASE 2017 challenge. The model is simple in that instead of heuristically aggregating frame-level predictions, it is trained to directly make the utterance-level prediction, with an objective that combines losses at both levels through an attention mechanism. To be robust to the variations in the time axis, we also propose a multi-resolution feature extraction architecture that improves over standard bi-directional RNNs. Our model can be trained efficiently in an end-to-end fashion, and thus can scale up to larger datasets and potentially to the simultaneous detection of multiple events.

6. Acknowledgements

The authors would like to thank Ming Sun and Hao Tang for useful discussions, and the anonymous reviewers for constructive feedback.

7. References

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

[2] K. Wang, L. Yang, and B. Yang, "Audio events detection and classification using extended R-FCN approach," DCASE2017 Challenge, Tech. Rep., 2017.

[3] C. Kao, W. Wang, M. Sun, and C. Wang, "R-CRNN: Region-based convolutional recurrent neural network for audio event de-

tection," in *Proc. of Interspeech'18*, Hyderabad, India, Sep. 2–6 2018, to appear.

[4] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems (NIPS)*, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds., vol. 28. MIT Press, Cambridge, MA, 2015, pp. 91–99.

[5] H. Lim, J. Park, and Y. Han, "Rare sound event detection using 1D convolutional recurrent neural networks," DCASE2017 Challenge, Tech. Rep., 2017.

[6] E. Cakir and T. Virtanen, "Convolutional recurrent neural networks for rare sound event detection," DCASE2017 Challenge, Tech. Rep., 2017.

[7] D. Bahdanau, J. Chorowski, D. Serdyuk, P. Brakel, and Y. Bengio, "End-to-end attention-based large vocabulary speech recognition," Aug. 18 2015, arXiv:1508.04395 [cs.CL].

[8] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proc. of the 3rd Int. Conf. Learning Representations (ICLR 2015)*, San Diego, CA, May 7–9 2015.

[9] Y. Miao, J. Li, Y. Wang, S.-X. Zhang, and Y. Gong, "Simplifying long short-term memory acoustic models for fast training and decoding," in *Proc. of the IEEE Int. Conf. Acoustics, Speech and Sig. Proc. (ICASSP'16)*, Shanghai, China, Mar. 20–25 2016.

[10] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. of the 2016 IEEE Computer Society Conf. Computer Vision and Pattern Recognition (CVPR'16)*, Las Vegas, NV, Jun. 26–Jul. 1 2016.

[11] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015. [Online]. Available: https://www.tensorflow.org

[12] A. Mesaros, T. Heittola, and T. Virtanen, "Metrics for polyphonic sound event detection," *Applied Sciences*, vol. 6, no. 6, p. 162, 2016.

[13] ——, "Tut database for acoustic scene classification and sound event detection," in *Proc. EUSIPCO*, 2016.

[14] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. of the 3rd Int. Conf. Learning Representations (ICLR 2015)*, San Diego, CA, May 7–9 2015.

[15] K. Cho, B. van Merrienboer, C. Gulcehre, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," in *Proc. 2014 Conference on Empirical Methods in Natural Language Processing*, Doha, Qatar, Oct. 25–29 2014.