USING SEQUENTIAL INFORMATION IN POLYPHONIC SOUND EVENT DETECTION

Guangpu Huang, Toni Heittola, Tuomas Virtanen

Audio Research Group, Tampere University of Technology, Finland

ABSTRACT

To detect the class, and start and end times of sound events in real world recordings is a challenging task. Current computer systems often show relatively high frame-wise accuracy but low event-wise accuracy. In this paper, we attempted to merge the gap by explicitly including sequential information to improve the performance of a state-of-the-art polyphonic sound event detection system. We propose to 1) use delayed predictions of event activities as additional input features that are fed back to the neural network; 2) build N-grams to model the co-occurrence probabilities of different events; 3) use sequential loss to train neural networks. Our experiments on a corpus of real world recordings show that the N-grams could smooth the spiky output of a state-of-the-art neural network system, and improve both the frame-wise and the event-wise metrics.

Index Terms— Polyphonic sound event detection, language modelling, sequential information

1 Introduction

Environmental audio contain many overlapping or polyphonic sound events, e.g., foot steps and car passing. Computer systems that automatically detect the class, start, and end of these sound events in the audio stream are called sound event detection (SED) [1]. SED poses a challenging research topic. Not only are the sound events often overlapping with each other, i.e., polyphonic, the number of simultaneously occurring sound events at certain time is also not set. Recent advances in polyphonic SED are largely attributed to the use of deep neural networks (DNNs) [2, 3]. In particular, the use of recurrent neural networks (RNNs) have significantly improved SED performance in the past few years. RNNs are able to model the contextual correlation or relationships between sound events [4, 5, 6]. Cakir et al. have build a convolution recurrent neural network which reaches state-of-the-art performance on polyphonic SED [7]. Majority of DNN systems use a frame-wise cost function for training the network and make prediction on a frame-by-frame basis. The frame-wise $F1$ score is often considerably higher, more than 5 times, than the event-wise accuracy. For example, in DCASE2016 challenge the $F1$ score is around 30% at frame level (frame length 1 $second$), but only around 5% at event level[1] [8, 9]. Frame-based information alone is not enough to produce high quality system output that are similar to human annotations, and we want to build accurate systems that obtain high accuracy at both frame- and event-level.

Many recent works in the field of environmental audio research are borrowing techniques from automatic speech recognition (ASR) to model the sequential structure of polyphonic sound events. These techniques step away from frame-wise networks to improve event-level accuracy. For example, Hayashi et al. used hidden semi-Markov model to separately model the duration of sound events on top of DNNs [10]. Wang et al. used connectionist temporal classification (CTC) cost function in a sequence-to-sequence model for the SED task [4, 6, 11]. Sigtia et al. use language models (LMs) and decoding methods which are analogous to ASR systems, to deal with polyphony in automatic piano music transcription [12]. Inspired by these advances, we propose to explicitly use sequential information to improve the event-level accuracy of polyphonic SED system, and aim to bridge the gap between automatic systems and human annotator at the frame and event-level.

Methods that explicitly model the sequential information are not common in SED systems. Unlike speech and music, environmental sound events are much more diverse and sparse, making it the most challenging. Therefore, we proposed to incorporate sequential information to a state-of-the-art polyphonic SED system via:

1. using delayed predictions of event activities as additional input features to the neural network;
2. building N-grams from annotations and using them for decoding with acoustic models;
3. using sequential loss function, i.e., a CTC decoder to find the most probable event sequences. The proposed CTC model is able to give the precise event start and end time which has not been done in previous research.

We argue that the state-of-the-art polyphony SED systems could benefit from the explicit use of sequential information.

2 Method

A state-of-the-art convolutional recurrent neural network (CRNN) model is used as our baseline, as was described in [7]. Figure 1 shows the system structure. The mel-frequency cepstral coefficients (MFCCs) of the audio recordings are used at the input. The output layer has logistic activation functions and one neuron for each class. The model generates the posterior probabilities of each sound class being active at each time step. The probability is compared against a threshold value 0.5, and converted to a binary event vector where $0/1$ denotes the presence/absence of the event. In this

This work received funding from the European Research Council under the ERC Grant Agreement 637422 EVERYSOUND.

[1]DCASE2016 Challenge: http://www.cs.tut.fi/sgn/arg/dcase2016/index

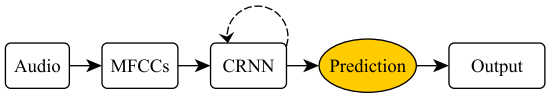

Fig. 1: An overview of the CRNN system for polyphonic sound event detection.

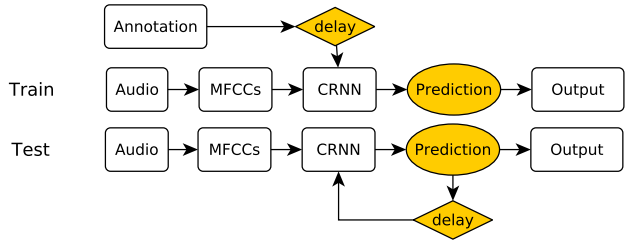

Fig. 2: Using annotations with a time-delay at the CRNN system input.

section, we describe the three methods to compensate for the sequential information in the CRNN SED system.

2.1 Using delayed prediction of events activities as additional input features

In real-world environments, some sound events exhibit intrinsic relationships, where one event happening is likely to trigger another. For example, *water running* is likely to be associated with *dish washing* in the *home* environment. We propose to use event labels from previous frames as additional input features to give the neural networks more information during training. Though the RNN layers already include the feedback loop in the CRNN system, the loops are only operating within one layer. Feeding back the predictions explicitly will show whether more information could be gained. This method will reveal whether the bottom layers of CRNN would provide anything significantly new. The ground-truth event annotations are converted to a N *by* T binary matrix, where N is the total number of event classes, T is the total number of time frames ($40\ ms$ long), and $0/1$ in each cell denotes the presence/absence of the event. As shown in Figure 2, during training, we feed the event vector, with one time frame delay, as additional input features to the CRNN. Note that the event vector without the time frame delay is also the target matrix for training the network. During testing, the prediction at the CRNN output is firstly converted into a event vector, then added a time delay, and feed back as additional input. Since this will create a mismatch between training and testing conditions, we also train the system by feeding the delayed predicted event vector, in binary discrete form, back to CRNN and carried out a separate experiment.

2.2 Using N-grams

Language model priors can significantly reduce the ambiguities from using the acoustic model alone [13, 14]. Current polyphonic SED systems are not doing any explicit language modelling, i.e., use the *grammar* in the annotations to assist system prediction. We propose to build N-gram LMs to estimate the statistic in the event annotations. For each audio signal, the event annotations are projected onto the time axis to make a single sequence, using a moving window empirically set to be $100\ ms$ long. We tried different window sizes: $20\ ms, 40\ ms, 80\ ms, 100\ ms, 120\ ms$ and choose the one that gave the best results. As shown in Figure 3, we first transcribe the beginning and the end of a event class e with $\langle e\ start \rangle$ and $\langle e\ end \rangle$, according to its time. When the event is active, we transcribe e every $100\ ms$, if the event spans more than $100\ ms$. These transcriptions are ordered by their time markers. As the frame resolution during feature extraction in the CRNN system is $40\ ms$, if the event is shorter than $100\ ms$ and longer than $40\ ms$, we mark three points, the event boundaries and the event itself in the middle point. If the event span is shorter than $40\ ms$, only the event label itself is marked. This is done for all events to produce a text transcription for training statistical N-gram LMs.

Given a sequence $E = \{e_t^i\}$, where e_t^i is a projected sound event e^i at time step t. The N-gram method counts the number of occurrences of event, event-pairs (2-grams), and event trios (3-grams), and defines a prior probability distribution P_{LM}, where

$$P_{LM} = 0.2 * P_{unigram} + 0.2 * P_{2-gram} + 0.6 * P_{3-gram} \quad (1)$$

and

$$P_{unigram} = P(e^i)) \quad (2)$$

$$P_{2-gram} = \sum_{k=1,k\neq i}^{K} P(e_t^i|(e_{t-1}^k)) \quad (3)$$

$$P_{3-gram} = \sum_{k=1,k\neq i}^{K} P(e_t^i|(e_{t-1}^k, e_{t-2}^k)) \quad (4)$$

It accounts for the co-occurrence probabilities of different events: which event classes are likely to happen simultaneously and the temporal sequences of sound events: which events are likely to follow each other. The N-grams are combined with the CRNN acoustic models via a decoder to find the most probably event sequences, as shown in Figure 4. Analogous to decoding in ASR [13], the acoustic model and N-gram model are combined via

$$P(E) = log(P_{AM}) + LMSF * log(P_{LM}) \quad (5)$$

where P_{AM} is the acoustic model probability (continuous) at the CRNN output layer, LMSF is the language model scaling factor (set to be 0.8 in our experiments, which shows better SED performance on the test data). However, searching for the best sequence is intractable as the number of possible paths through the event activation matrix grows exponentially with time 2^T, where T is the number of time frames. We performed beam search decoding, where the number of candidate solution at each step is pruned and limited to top k. The likelihood of the top paths are calculated in the log domain to avoid underflow. The N-gram models are built using KenLM[2]

[2]https://kheafield.com/code/kenlm/

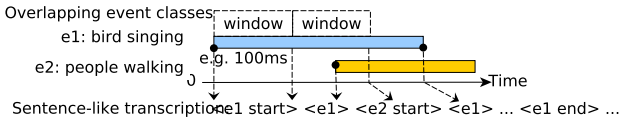

Fig. 3: Projecting annotations of polyphonic sound events to sentence-like transcription. Two event classes: *bird singing end* and *people walking* are used to illustrate the process.

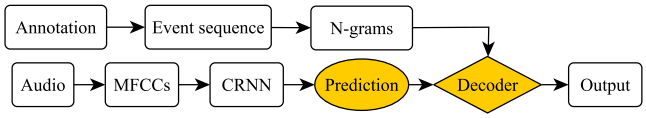

Fig. 4: Using N-grams to decode polyphonic sound events.

and applied to the CRNN system via a python interface[3].

2.3 Using CTC sequential loss function

To move away from a frame-level cost function, it is desirable to make a predictions based on the overall sequential loss for polyphonic sounds, as shown by [12]. We implemented a CTC decoder for the polyphonic SED task. CTC loss is based on the probability of the entire even sequence over all possible alignments, and generates a probability distribution of all the event tokens in the output [11, 15]. This could reduce the gap of high frame-level accuracy and low event-level accuracy. Our approach is similar to [4], but they use only event boundaries to create a single sequence. In this paper, we use the sequential transcriptions generated by the projection scheme (Section 2.2) in parallel with the audio to train the CTC networks. This enables the CTC model to locate the precise event start and end time, i.e., e with $\langle e\ start \rangle$ and $\langle e\ end \rangle$, during training, which are not available in the previous CTC approach [4].

3 Experiments

3.1 Data

The dataset is the same as used in DCASE 2016 SED task, the *TUT SED 2016 development set* (TUT-SED2016)[4]. The corpus consists of 22 audio recordings from two real world environments: indoor $home$ with 10 recordings from different homes (total audio length is about 36 minutes) and outdoor $residential\ area$ with 12 recordings from different streets in residential area (total audio length is about 42 minutes). The recordings are annotated with a list of 17 sound event classes, with onset and offset time marker for each event class. The average length of sound event is about 1.94 $seconds$ at $home$, and about 6.29 $seconds$ at the $residential\ area$. Most of the event classes are scene-specific. The average number of events active in the same time frame, 40 ms, is about 2. The percentages of overlapping or polyphony level in the dataset are shown in Table 1. At polyphony level N, there are N overlapping event classes.

[3]https://github.com/kpu/kenlm

[4]TUT-SED2016 corpus [online] https://zenodo.org/record/45759

Table 1: Polyphony Statistics of Environmental Recordings.

	home (%)	residential area (%)
polyphony 1	50.70	26.42
polyphony 2	41.15	58.31
polyphony 3	5.65	12.50
polyphony 4	0.25	0.96
polyphony 5	0.00	0.34

3.2 Setup

We used a state-of-the-art scene independent polyphonic CRNN SED system as a baseline [7]. The system has reported the highest frame-wise accuracy on the TUT-SED2016 dataset used here. The classifier is implemented with Keras[5] (version 1.1.0) with Theano (version 0.8.2) backend [16]. The networks are trained on NVIDIA Tesla K80 GPUs. Training takes about 4 hours and converges after about 120 epochs. The acoustic features are 40 MFCCs (13 cepstral, 13 delta, 13 delta-delt, 1 log energy). The window size is $40ms$ and the hop length is 20 ms. The CRNN output is a probabilistic activity matrix which gives a probability of each sound event in every time-frame 40 ms for each audio recording. During evaluation, the matrix is compared to a threshold value, 0.5, to obtain the binary matrix, where presence/absence of a sound event in each time frame is indicated by $0/1$ in each cell. The predicted matrix is compared with the ground-truth annotation to calculate frame-wise and event-wise metrics [17].

3.3 Evaluation Metrics

We report the frame-wise and the event-wise (onset-only) performances on TUT-SED2016. The frame-wise metrics are calculated in non-overlapping 1-second segments. In other words, it is done in a fixed time grid, using segments of one second length to compare the ground truth and the system output. For each segment in the test set, the number of true positive (TP), false positive (FP) and false negative (FN) are calculated. The event-wise metrics are calculated with respect to event instances. All the numbers are accumulated over the test data to output the $F1$ score and the error rate (ER) [17],

$$F1 = \frac{\sum_{t=1}^{N} 2 \cdot TP_t}{\sum_{t=1}^{N} (2 \cdot TP_t + FN_t + FP_t)} \tag{6}$$

$$ER = \frac{\sum_{t=1}^{N} (S_t + I_t + D_t)}{\sum_{t=1}^{N} A_t} \tag{7}$$

where S is the number of substitutions, I insertions, D deletions, t the segment index, and A the total number of active sound events in the reference segments.

3.4 Results and Discussion

Table 2 summarizes the average $F1$ scores and error rates (ER) of the proposed methods on the TUT-SED2016 corpus. The results are calculated over all the event segments on the whole test set for each scene and then averaged to get an overall metric. The baseline CRNN system achieves 27.1%

[5]Online: https://github.com/fchollet/keras

Table 2: Experimental results on polyphonic sound event corpus TUT-SED2016: 2 scenes 17 events.

		F1 (%)		Error Rate	
		Frame	Event	Frame	Event
CRNN (Baseline)		27.1	3.0	0.95	2.45
+event activity	prediction	27.3	3.7	0.96	1.55
	ground-truth	1.1	0.3	1.04	1.54
+Ngram LM	top $k = 5$	**29.1**	**5.4**	0.94	1.56
+CTC decoder		25.0	4.2	1.09	2.50
+CTC +Ngram		27.5	1.8	1.35	4.78

frame-wise $F1$ score, and 3.0% event-wise $F1$. The frame-wise ER is 0.95 and the event-wise ER is 2.45.

Using N-grams gives some gain over the baseline system. In SED, the number of overlapping events is unknown, and the audio recording is usually longer than 10 events. It is estimated that the LM posteriors of adjacent events smooths the acoustic scores at the neural network output. At the output of the beam search decoder, we found that the top paths are usually very similar and contain lots of silent labels. As the length of the sequence grows, the posterior drops significantly. We restrained the decoder search the top $k = 5$ path in every 100 frames, 4 seconds, and then combine these path with logic 'OR' function. The N-gram method improves event-wise $F1$ by 2.4% absolute, and frame-wise $F1$ by 2.0%. The method also reduces the ER by 0.89 absolute, as shown in Table 2. The gains are mostly from events like *dishes* and *object_impact* in the $home$ scene. For *dishes*, N-gram method improves event-wise $F1$ by 4.3% absolute, and frame-wise $F1$ by 6.0%. N-gram statistics show that *dishes* has a relatively higher log-likelihood than others. In fact, out of 586 event labels in the annotation of the $home$ scene, 156 are *object_impact* and 94 are *dishes*. For other events in either the $home$ or the $residential\ area$ scene, there is not much gain, and using the proposed methods do not worsen the results either. Varying the number of top paths, k, does not give further improvements.

Using the delayed predicted event activities as additional features which are fed back to CRNN during training gives 0.7% gain on event-wise F1. In contrast, using delayed ground truth event activities worsens the performance significantly. It is probably that the system output is significantly different from ground truth at event-level, and feeding back the true annotations only corrupts the sequential regularities. Using the CTC cost function in CRNN training improves the event $F1$ by 1.2%, but decreases 2.1% on frame-wise F1. Furthermore, applying N-grams in CTC decoding does not improve event F1. It performs better on harmonic sound events that have plenty of labels in the training data, e.g., bird singing.

The results confirms that using grammar or language models could benefit the current state-of-the-art SED system. Yet the problem of overlapping is more challenging than we estimated. The overall gain is small. The largest gain is for the event *dishes* and *object_impact* in the $home$ scene, which have more occurrences than the other events. It is probable that the used dataset contains sparsely distributed sound events. For instance, a large portion of the annotations are silence. The consequence is that the activity matrix at the CRNN output contains many short bursts of events, which introduces a lot of insertion error, especially in the event-wise metric. The N-grams could give a smoothing effect to this end, but the paths from beam search decoder are too similar to each other. In experiments on a much larger corpus with 11 scenes and 63 sound classes [7], using the N-grams improves the CRNN system frame-wise $F1$ by 1.3% and event-wise $F1$ by 1.5%. The method also reduces the ER by 0.7 absolute, as shown in Table 3.

Table 3: Experimental results on a larger polyphonic sound event corpus: 11 scenes 63 events.

	F1 (%)		Error Rate	
	Frame	Event	Frame	Event
CRNN (Baseline)	69.1	8.9	0.46	2.42
+Ngram LM	**70.4**	**10.4**	0.46	1.72

4 Conclusion

Environmental sound events have temporal structures and sequential dependencies. Human use the contextual relationships between sound events to to identify and isolate environmental sound events of interests with ease. In most automatic polyphonic SED systems, there is not an explicit LM component, and sequential information is assumed to be implicitly modelled by the recurrent neural networks. We exploit using sequential information: calculating N-gram probability of events, using delayed event annotations, and using sequential CTC loss function to train neural networks, to improve a state-of-the-art polyphonic SED system. Using a corpus with two real environmental scenes, indoor home and outdoor residential area, our results show that LM is able to improve both the frame-wise and event-wise accuracy, though with small gains.

The proposed methods have several weaknesses. The statistical language models could only manage short context, and fails to account for longer contexts. One solution is to train separate language models to learn longer-term dependencies and to rescore the decoded paths as done in speech and music recognition systems [12, 18, 19, 20]. The system needs to deal with multiple overlapping sources in this scenario. Moreover, LMs is mostly useful when learned separately using an external large collection of text resources. Thus it will be desirable to combine information from multiple databases.

Acknowledgement

The authors wish to acknowledge CSC-IT Center for Science, Finland, for providing the computational resources. This work would not have been possible without the assistance from Annamaria Mesaros and Emre Cakir, and discussion from members at the TUT Audio Research Lab.

5 References

[1] Tuomas Virtanen, Mark Plumbley, and Dan Ellis, *Computational Analysis of Sound Scenes and Events*, Springer, 2017.

[2] Geoffrey Hinton, Li Deng, Dong Yu, George E Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N Sainath, et al., "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, 2012.

[3] Emmanouil Benetos, Dan Stowell, and Mark Plumbley, "Approaches to complex sound scene analysis," in *Computational Analysis of Sound Scenes and Events*, pp. 215–242. Springer, 2018.

[4] Yun Wang and Florian Metze, "A first attempt at polyphonic sound event detection using connectionist temporal classification," in *ICASSP*, 2017, pp. 2986–2990.

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

[6] Yun Wang, Leonardo Neves, and Florian Metze, "Audio-based multimedia event detection using deep recurrent neural networks," in *ICASSP*, 2016, pp. 2742–2746.

[7] E. Cakir, G. Parascandolo, T. Heittola, H. Huttunen, and T. Virtanen, "Convolutional recurrent neural networks for polyphonic sound event detection," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 25, no. 6, pp. 1291–1303, 2017.

[8] Tuomas Virtanen, Annamaria Mesaros, Toni Heittola, Mark D. Plumbley, Peter Foster, Emmanouil Benetos, and Mathieu Lagrange, *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2016 Workshop (DCASE2016)*, Tampere University of Technology, 2016.

[9] Annamaria Mesaros, Toni Heittola, and Tuomas Virtanen, "Assessment of human and machine performance in acoustic scene classification: DCASE 2016 case study," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2017, pp. 319–323.

[10] Tomoki Hayashi, Shinji Watanabe, Tomoki Toda, Takaaki Hori, Jonathan Le Roux, and Kazuya Takeda, "Duration-Controlled LSTM for Polyphonic Sound Event Detection," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 25, no. 11, pp. 2059–2070, 2017.

[11] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *International conference on Machine learning*, 2006, pp. 369–376.

[12] Siddharth Sigtia, Emmanouil Benetos, and Simon Dixon, "An end-to-end neural network for polyphonic piano music transcription," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 24, no. 5, pp. 927–939, 2016.

[13] Lawrence Rabiner and Biing-Hwang Juang, *Fundamentals of Speech Recognition*, Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1993.

[14] Elizabeth Shriberg, Andreas Stolcke, and Don Baron, "Observations on overlap: Findings and implications for automatic processing of multi-party conversation," in *Seventh European Conference on Speech Communication and Technology*, 2001.

[15] Alex Graves and Navdeep Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *International Conference on Machine Learning*, 2014, pp. 1764–1772.

[16] The Theano Development Team, Rami Al-Rfou, Guillaume Alain, Amjad Almahairi, Christof Angermueller, Dzmitry Bahdanau, Nicolas Ballas, Frédéric Bastien, Justin Bayer, Anatoly Belikov, et al., "Theano: A python framework for fast computation of mathematical expressions," *arXiv preprint arXiv:1605.02688*, 2016.

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

[18] Guangpu Huang, Arseniy Gorin, Jean-Luc Gauvain, and Lori Lamel, "Machine translation based data augmentation for Cantonese keyword spotting," in *ICASSP*, 2016, pp. 6020–6024.

[19] Guangpu Huang, Thiago Fraga da Silva, Lori Lamel, Jean-Luc Gauvain, Arseniy Gorin, Antoine Laurent, Rasa Lileikyte, and Abdel Messouadi, "An investigation into language model data augmentation for low-resourced STT and KWS," in *ICASSP*, 2017, pp. 5790–5794.

[20] Jen-Tzung Chien and Chuang-Hua Chueh, "Joint acoustic and language modeling for speech recognition," *Speech Communication*, vol. 52, no. 3, pp. 223–235, 2010.