SOUND EVENT DETECTION FROM WEAK ANNOTATIONS: WEIGHTED GRU VERSUS MULTI-INSTANCE LEARNING

Léo Cances, Thomas Pellegrini, Patrice Guyot

IRIT, Université de Toulouse, CNRS, Toulouse, France
{leo.cances,thomas.pellegrini,patrice.guyot}@irit.fr

ABSTRACT

In this paper, we address the detection of audio events in domestic environments in the case where a weakly annotated dataset is available for training. The weak annotations provide tags from audio events but do not provide temporal boundaries. We report experiments in the framework of the task four of the DCASE 2018 challenge. The objective is twofold: detect audio events (multi-categorical classification at recording level), localize the events precisely within the recordings. We explored two approaches: 1) a ”weighted-GRU” (WGRU), in which we train a Convolutional Recurrent Neural Network (CRNN) for classification and then exploit its frame-based predictions at the output of the time-distributed dense layer to perform localization. We propose to lower the influence of the hidden states to avoid predicting a same score throughout a recording. 2) An approach inspired by Multi-Instance Learning (MIL), in which we train a CRNN to give predictions at frame-level, using a custom loss function based on the weak label and statistics of the frame-based predictions. Both approaches outperform the baseline of 14.06% in F-measure by a large margin, with values of respectively 16.77% and 24.58% for combined WGRUs and MIL, on a test set comprised of 288 recordings.

***Index Terms*—** Sound event detection, weakly supervised learning, multi-instance learning, convolutional neural networks, weighted gate recurrent unit

1. INTRODUCTION

The coming of Deep Learning [1] has opened a new era in the domain of artificial intelligence. Deep neural networks in particular became the state of the art in many application domains involving classification and detection tasks. Most often, these improvements rely on the availability of ever-growing annotated datasets to train the models. While many previous works that heavily rely on supervised training based on a precise manual annotation, new challenges arise from the use of large datasets without supervision.

Recently, different databases of Terabytes of data have been released by Google. The Audioset database provides a large set of audio data extracts from video [2]. Annotations for audio events are mainly based on tags by YouTube users and do not contain temporal information.

In that scope, the DCASE challenge includes a task on sound event detection in domestic environment [3]. This task proposes a framework to build a system that aims at detecting audio events from a set of 10 classes of sound events forming a subset of Audioset. More precisely, the aim is to provide starting and ending boundaries of the audio events (strong labels) while the training set relies only on global tags (weak labels). As mentioned in [3], the duration of the targeted sounds depends heavily on their class. For example, the class *vacuum cleaner* contains mostly audio events of 10 seconds, while the class *dog* is mainly composed of sounds shorter than half-second.

Sound event detection (SED) has been deeply investigated [4]. In real life, sound events overlap to produce a mixture. In the same way, the current challenge aims at detecting overlapping sound events, referred as polyphonic SED. Polyphonic SED covers a wide set of applications including ecology [5] and surveillance [6]. The detection of domestic sounds provides interesting clues for health applications [7] and Intelligent Virtual Assistants such as Google Home or Amazon Echo.

Ongoing research works on SED are mostly based on a Deep Neural Networks (DNNs). They include fully-connected DNNs [8], Convolutional Neural Networks (CNNs) [9] and Recurrent Neural Networks (RNNs) [10]. Most of recent approaches are based on a combination of layers including these different elements [11]. In particular the issue of audio event detection using weakly labeled data was addressed in [12, 13] and formulated into a Multi-Instance Learning (MIL) problem.

The baseline method provided by the challenge organizers relied on two Convolutional Recurrent Neural Networks (CRNNs): the first one for file-level audio classification (weak labels), the second one for the localization of the previously detected events within the recordings (strong labels). We explored two separate approaches that both outperformed the baseline.

Firstly, we modify the recurrent layer of a CRNN to be able to weight the influence of the hidden state of the recurrent cells. Secondly, we generalize to our multiclass classification problem a new loss function inspired by Multi-Instance Learning (MIL), very recently proposed for singing bird localization in [14].

Section 2 describes the first approach, that we will refer to as “weighted Gated Recurrent Unit” (WGRU), followed by a section describing the second approach (MIL). We report the experimental setup in Section 4, and finally analyze the results and limitations of the two approaches.

2. WEIGHTED GATED RECURRENT UNIT (WGRU)

2.1. Temporal detection from an adapted baseline method

As mentioned above, the baseline system is based on two convolutional recurrent neural networks (CRNN). The first CRNN detects the presence/absence of the ten sound events of interest at file-level. Then, a second CRNN is used for localization. Still, we may assume that the temporal information required for localization is reachable from the first CRNN. In this way, after the classification training of the first CRNN on weak labels, we propose to simply remove its final global average pooling layer in order to get frame-based predictions used for detection. This modified model produces frame-level predictions thank to its time-distributed dense comprised of ten sigmoid units corresponding to the ten classes of interest.

In the following, we will focus on the recurrent layer of this CRNN, since modifications of its internal functioning had to be made to make localization possible.

2.2. Recurrent Neural Networks

Recurrent Neural Networks capture temporal behavior in sequential data [15]. The hidden state of a RNN cell depends on x_t, the incoming output of the previous layer at time t, and h_{t-1}, its hidden state at time $t-1$, as defined in Equation 1:

$$h_t = g(Wx_t + Uh_{t-1} + b) \tag{1}$$

where g is a point-wise activation function (hyperbolic tangent function in our case); and W and U are weight matrices to be learned together with the bias b.

2.3. Weighted RNN

RNNs have proven to model sound events efficiently, since these often have an underlying sequential structure [10].However, some audio event classes have different typical durations, as described in [3]. Long-duration sounds (*vacuum cleaner*, *running water*) are expected to be easier to model by an RNN than short sounds (*dog*, *cat)*. In order to adjust our models to different kinds of sounds, we propose a new adaptation of RNNs. This approach aims at configuring different kinds of temporal behavior according to the class of a sound event. As a first attempt in that direction, we weight the influence of the hidden state of the recurrent cells by a factor $\omega \leq 1$, which we set globally for the classes on a development subset. This modifies the impact of sequentiality in the RNN, as formulated in equation 2. Figure 1 shows the impact of the weight on a cell at time t, colored in blue. The rational behind lowering the impact of the hidden states lies in the fact that the CRNN is trained to detect a sound event at file level. Thus, if an event is detected at the beginning of the file, the hidden states are expected to keep that information throughout the file even if the detected event is not present in the whole file, and the localization afterwards will fail.

The baseline method is based on Gated Recurrent Units (GRU) [16]. In the following, we will use weighted RNN models using a GRU layer that we will be referred to as Weighted GRU (WGRU).

$$h_t = g(Wx_t + U\boldsymbol{\omega} h_{t-1} + b) \tag{2}$$

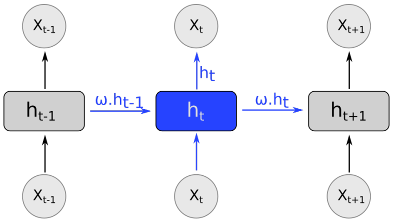

Figure 1: Configuration of the recurrent links between RNN cells according to the weight ω.

The next question that arises is how to set ω. We decided to set a single value for all the sound types based on the localization performance measured on a held-out validation subset. A weight of $\omega = 1$ corresponds to a standard GRU. A lower weight is expected to be more adapted for events of short duration.

The use of different weights does not require to retrain a model. We simply replace the GRU layer by a WGRU at inference time. Thus, we perform two forward passes with a single model: one with GRU for classification, one with WGRU for localization. Finally, we are able to produce temporal predictions with different weights, and eventually combine them to improve the localization.

3. MULTI-INSTANCE LEARNING

Another approach is related to the Multi-Instance Learning (MIL) paradigm [17], which handles cases with weak labels. In our case, we need to make predictions at frame-level, whereas the reference tags are at file-level. When a recording is labeled with the *Cat* class, for instance, not all the acoustic frames are positive with respect to that class. Thus, we are in presence of both positive and negative instances at frame-level. A straightforward but suboptimal solution is the so-called ”false strong labeling”, in which we consider that all the frames are positive for a given class. MIL consists instead of considering that at least one instance is positive, i.e. the highest score should be equal to the weak label, as written in the following relation for the i^{th} instance: $\max_j \hat{y}_{ikj} = y_{ik}$, where $\hat{y}_{ikj}$ and y_{ik} are the prediction for frame j and class k and the reference tag for class k, respectively.

There are drawbacks to this approach. For instance, the training of the model will focus only on the highest scored frame and ignore the other ones. To remedy to this issue, Morfi and Stowell [14] proposed a loss function that takes into account frames with the lowest prediction scores, which should tend towards zero, and also a naïve assumption that in general a specific event will be present in half of the frames. They applied this idea successfully on a binary classification problem, namely the presence/absence of singing birds in audio recordings. For the need of the present challenge, we generalized their loss function to $K > 2$ classes, as written in equation 3, where binCE stands for the binary cross-entropy loss. In our experiments, a first network is used to identify which classes should be considered for localization by a second network trained to minimize the MIL loss function.

$$\begin{gathered}\text{loss} = \sum_{k=1}^{K} \mathrm{binCE}(y_{ik}, \max_j \hat{y}_{ikj}) + \\ \mathrm{binCE}(y_{ik}/2, \operatorname*{mean}_j \hat{y}_{ikj}) + \mathrm{binCE}(0, \min_j \hat{y}_{ikj})\end{gathered} \tag{3}$$

4. EXPERIMENTAL SETUP

4.1. Audio material

The DCASE 2018 Task 4 is related to discovering audio events from a set of 10 sound categories occurring in domestic environments, namely Speech, Dog, Cat, Alarm/Bell ringing, Dishes, Frying, Blender, Running water, Vacuum cleaner, and Electric shaver/toothbrush. All the files are 10-second clips extracted from Youtube user videos and are part of the Audioset corpus [2]. The recordings most often contain several overlapping event categories.

The challenge corpus is divided into three subsets: the training, test and evaluation subsets. Three different splits of training data were provided: a labeled, an unlabeled in domain and an unlabeled out of domain training sets. In our work, we only used the labeled subset to train our models, and, in the case of GRU-WGRU only, we also tried to extend it by using pseudo-labels made on the unlabeled in domain subsets. The labeled training subset is comprised of 1578 clips (2244 class occurrences) for which weak annotations have been verified and cross-checked.

The test subset is comprised of 288 files for which we were provided with strong labels. We will thus report performance results on this subset. The results obtained on the evaluation subset, comprised of 880 files, are not available.

4.2. Audio features

For both approaches, only the labeled (*weak* labels) and the *unlabeled in-domain* subsets were used. We used the first one to train a classifier (for both WGRU & MIL), and also to add weak annotations to the *unlabeled in domain* subset. Finally, both subsets were used to retrain the model and perform localization. The use of the unlabeled in-domain subset proved useful for WGRU only, but not for MIL. More experiments are needed to draw conclusions for this semi-supervised setting.

As input to the networks, 64 log-Mel filter-bank (F-BANK) coefficients were extracted every 23 ms on 100 ms duration frames, with 20 Hz and 11025 Hz as minimum and maximum frequency values to compute the Mel bands, respectively. Hence, for each 10-second file, a 431×64 matrix is extracted. This matrix is used as a single input image fed to the networks.

Different normalization and features scaling methods were tested as pre-processing stage such as *global mean removal*, *mean and variance standardization*, but no gains were observed compared to using raw F-BANK.

4.3. Neural networks

The networks used in our work are very similar to the baseline one in terms of number and types of layers: three blocks each comprised of a convolution (64 $3, 3$ kernels) layer - batch-normalization - Rectifier Linear Units (ReLu) as the non-linear activation function - sub-sampling by max-pooling ($2, 4$ for the first layer, $1, 4$ for the following blocks) - 2-d Spatial Dropout (dropping factor=20%). Then follows a bi-directional GRU layer with the tanh activation function with 64 cells, a time-distributed dense layer with 64 neuron units, a global-average-pooling layer to obtain 10 scores with a sigmoid function on each of the ten output neurons.

For the MIL approach, the first network used to classify the sound events at file level did not comprise the GRU layers but instead a dense one with 1024 neurons. The input of this layer was the concatenation of a 2-d average- and a 2-d max- global pooling. This network was found to perform better than the recurrent one, for classification at least. It yielded 85.84% and 82.86% f1 scores on our training and validation subset (proportion: 80/20 % of the weakly labeled training set).

Regarding the second network used for localization in the MIL approach, its architecture is the same as the CRNN one used for WGRU, except that the last dense layer is a time-distributed dense layer with ten units. The output of this network for a single recording is of dimension 431×10, 431 being the number of time frames, and 10 the number of classes. The score curves are then individually rescaled to the $[0, 1]$ interval. The final event segments are obtained by first smoothing the score curves with a moving-average filter of size 19 frames, second by binarizing the curves with a threshold of 0.07. Neighbor segments are merged when separated by less than 200 ms, the tolerance margin used for evaluation.

In all cases, we used the Adam optimizer and a simple learning rate decay policy: dividing it by two after 30 epochs and 60 epochs. All the networks were trained on 100 epochs except the MIL localization network trained on 10 epochs only.

4.4. Threshold optimization

We used an ad-hoc threshold optimization algorithm to set the different thresholds on the classes. Our method is used for the classification (WGRU and MIL) and detection (WGRU only) tasks. It consists of a genetic algorithm inspired by simulated annealing [18]. With optimizing the threshold modifications, it allows sharp reduction of the number of combinations and leads quickly to a near optimal solution.

5. RESULTS

Approach	Baseline	WGRU	MIL
F-score (%)	14.06	16.77	24.58
Alarm_bell_ringing	3.9	17.6	28.3
Blender	15.4	11.6	10.1
Cat	0.0	0.0	48.9
Dishes	0.0	0.0	0.0
Dog	0.0	4.8	18.6
Electric_shaver_toothbrush	32.4	33.3	28.6
Frying	31.0	29.5	26.7
Running_water	11.4	7.1	10.3
Speech	0.0	19.4	22.3
Vacuum_cleaner	46.5	40.0	52.2

Table 1: Global and class-wise F-measures (F-scores) on the test subset.

Table 1 shows the performance results on the test subset in terms of F-measure (F-score) for the baseline and our approaches. Both WGRU and MIL outperform the 14.06% baseline F-score with 16.34% and 24.58% scores, respectively. MIL behaves better than GRU-WGRU for all the classes.

	Weight(s)	F-score (%)	ER
Baseline	_	14.06	1.54
WGRU	1.	6.68	2.55
	0.50	4.69	2.92
	0.30	8.24	3.18
	0.20	11.35	3.37
Combined WGRUs	1. and 0.20	16.77	1.60

Table 2: Performance comparison and impact of the weight used with WGRU.

We only reported our best results in this table, using classification thresholds optimized on our validation subset (20% of the training data). Classification performance decreases by about 1% in F-score if using the default 0.5 threshold for all the classes.

5.1. WGRU

5.1.1. Temporal dependency weakening improves localization

Figure 2 gives an example in which the standard CRNN (GRU) does not allow to localize the sound event *Dog* that was correctly detected for that particular recording. The curves are the output of the last time-distributed layer of the network, the Green one being the curve using GRU and the orange one using WGRU. The vertical rectangles denote the ground truth and represent the segments where the dog should be detected. For GRU, *Dog* is detected throughout the whole audio clip. For WGRU, with $\omega = 0.25$, the *dog* segments are properly localized based on the curve peaks.

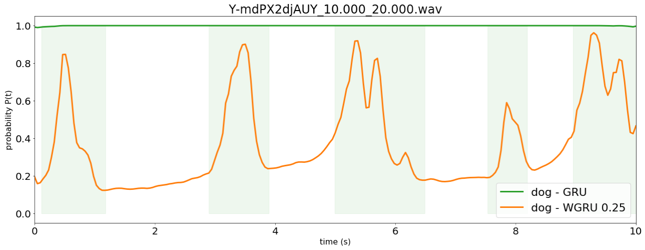

Figure 2: GRU (green) and WGRU (orange) score curves for the correct class *Dog*. The vertical rectangles denote the groundtruth and represent the segments where *Dog* should be detected. *Green* Localization results of the CRNN with GRU. The class *dog* is detected but in the entire clip. *Orange:* The prediction of the CRNN with WGRU with a temporal weight of 0.25. The dog barking segments are detected and localized properly.

This failure of the standard CRNN may be due to the fact that it is trained to detect a sound event in 10-s duration recordings, regardless of where in the file. Thus, the memory brought by the recurrent cell states keeps the information that *Dog* is present as soon as it is detected in the file, either at the beginning or at the end of the recording since bi-directional GRU layers are used.

5.1.2. *Combination of WGRUs*

Table 2 allows to compare the baseline performance of 14.06 in F-score and 1.54 in Error Rate (ER) to WGRU with different values of the weighting scalar ω. As one can see, WGRUs alone are worse than the baseline. The best weighting factor value was found to be estimated to about 0.20, with a 11.35% F-score. Smaller weight values revealed less efficient, showing that keeping some information from the previous hidden cell state is important. The best results, also reported in Table 1, were obtained by combining two WGRUs with 1. and 0.20 weights.

The combination of WGRU predictions is subjective and made after observation using the test dataset. The classes have been divided into two categories: stationary sounds (*Blender*, *Electric_shaver_toothbrush*, *Running_water*, *Vacuum_cleaner*) and short sounds (*Alarm_bell_ringing*, *Cat*, *Dog*, *Speech*). The predictions of the WGRU weighted at 1 are kept for the stationary sounds and the predictions of the WGRU weighted at 0.20 for the short sounds.

5.2. MIL

Figure 3 shows a successful example of the MIL model for a test file that contains speech and dog barking in segments given below the spectrogram. The first classification CNN correctly identified these two classes at file level, and the second MIL-CNN provides the two peaky curves, blue for *Speech*, red for *Dog*.

As shown in Table 1, MIL outperformed the baseline by a large margin for about haf of the classes and especially for *Cat* with a 48.9% F-score. For the other classes, its performance is lower but close. It is remarkable that all approaches failed for *Dishes*.

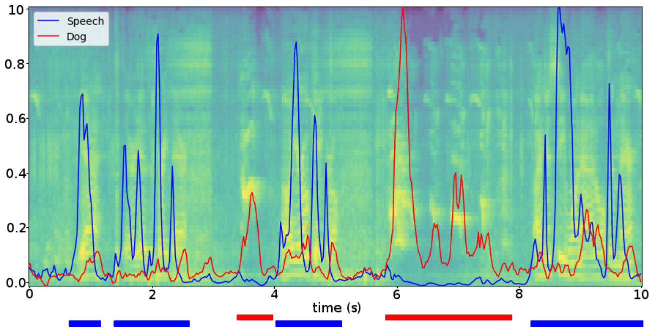

Figure 3: Score curves by MIL for the two correctly detected classes 'Speech' (Blue) and 'Dog' (Red). Below the spectrogram is represented the groundtruth.

By observing the localization predictions, it appears that the MIL model confuses *Dishes* and *Frying*. In the training subset, about 46% of the Dishes samples also contain *Frying*. There are even more files with *Dishes* and *Speech*, namely 52%, and 32% with the three classes together. *Dishes* is not confounded that much with *Speech* probably because there are many more Speech files than Dishes files: 550 versus 184 files.

6. CONCLUSION

In this paper, we reported experiments in the framework of the task four of the DCASE 2018 challenge. We had a two-fold objective of first, detecting sound events globally in audio recordings, second, localizing as precisely as possible where the detected event categories occur in time. We trained our models on weakly annotated dataset. The weak annotations provide tags from audio event but does not provide their temporal boundaries.

We explored two new approaches: 1) a "weighted-GRU" one (WGRU), in which we train a Convolutional Recurrent Neural Network for classification and then exploit its frame-based predictions at the output of the time-distributed dense layer to perform localization. We propose to lower the influence of the hidden states to avoid predicting a same score throughout a recording ; 2) An approach inspired by Multi-Instance Learning (MIL), in which we train a CNN to give predictions at frame-level, using a custom loss function based on the weak label and statistics of the frame-based predictions. Both approaches outperform the baseline of 14.06% in F-measure by a large margin, respectively 16.34% and 24.58% for combined WGRUs and MIL, on a test set comprised of 288 files.

Limitations of our best approach, MIL, were described. In particular, when two classes of sound events occur very often together, such as Dishes and Frying, MIL fails to learn how to distinguish between them. Our next step will be to modify the MIL loss function to penalize the fact that the prediction outputs for two different classes are too similar. Another improvement would be to achieve the same performance but using a single neural network rather than two.

References

[1] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, p. 436, 2015.

[2] 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 *Proc. IEEE ICASSP 2017*, New Orleans, LA, 2017.

[3] R. Serizel, N. Turpault, H. Eghbal-Zadeh, and A. Parag Shah, "Large-Scale Weakly Labeled Semi-Supervised Sound Event Detection in Domestic Environments," July 2018, submitted to DCASE2018 Workshop. [Online]. Available: https://hal.inria.fr/hal-01850270

[4] T. Virtanen, M. D. Plumbley, and D. Ellis, *Computational analysis of sound scenes and events*. Springer, 2018.

[5] D. Stowell, M. Wood, Y. Stylianou, and H. Glotin, "Bird detection in audio: a survey and a challenge," *arXiv preprint arXiv:1608.03417*, 2016.

[6] M. Crocco, M. Cristani, A. Trucco, and V. Murino, "Audio surveillance: a systematic review," *ACM Computing Surveys (CSUR)*, vol. 48, no. 4, p. 52, 2016.

[7] P. Guyot, J. Pinquier, and R. André-Obrecht, "Water sound recognition based on physical models," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 793–797.

[8] E. Cakir, T. Heittola, H. Huttunen, and T. Virtanen, "Polyphonic sound event detection using multi label deep neural networks," in *Neural Networks (IJCNN), 2015 International Joint Conference on*. IEEE, 2015, pp. 1–7.

[9] H. Zhang, I. McLoughlin, and Y. Song, "Robust sound event recognition using convolutional neural networks," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 559–563.

[10] G. Parascandolo, H. Huttunen, and T. Virtanen, "Recurrent neural networks for polyphonic sound event detection in real life recordings," *arXiv preprint arXiv:1604.00861*, 2016.

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

[12] A. Kumar and B. Raj, "Audio event detection using weakly labeled data," in *Proceedings of the 2016 ACM on Multimedia Conference*. ACM, 2016, pp. 1038–1047.

[13] ——, "Deep cnn framework for audio event recognition using weakly labeled web data," *arXiv preprint arXiv:1707.02530*, 2017.

[14] V. Morfi and D. Stowell, "Data-efficient weakly supervised learning for low-resource audio event detection using deep learning," *arXiv preprint arXiv:1807.06972*, 2018.

[15] R. Dey and F. M. Salem, "Gate-variants of gated recurrent unit (GRU) neural networks," *CoRR*, vol. abs/1701.05923, 2017. [Online]. Available: http://arxiv.org/abs/1701.05923

[16] K. Cho, B. van Merriënboer, Ç. Gülçehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder–decoder for statistical machine translation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 1724–1734. [Online]. Available: http://www.aclweb.org/anthology/D14-1179

[17] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez, "Solving the multiple instance problem with axis-parallel rectangles," *Artificial intelligence*, vol. 89, no. 1-2, pp. 31–71, 1997.

[18] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by Simulated Annealing," *Science*, vol. 220, pp. 671–680, May 1983.