CRNNS FOR URBAN SOUND TAGGING WITH SPATIOTEMPORAL CONTEXT

Technical Report

*Augustin Arnault**

IMT Lille Douai
Lille, France
augustin.a@free.fr

Nicolas Riche

Multitel
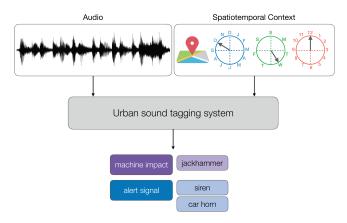Mons, Belgium
riche@multitel.be
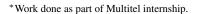
Figure 1: Task 5 of DCASE 2020 [1].

ABSTRACT

This paper describes CRNNs we used to participate in Task 5 of the DCASE 2020 challenge. This task focuses on hierarchical multi-label urban sound tagging with spatiotemporal context. The code is available on our GitHub repository at https://github.com/multitel-ai/urban-sound-tagging.

Index Terms— DCASE challenge, audio tagging, multilabel classification, metadata, CRNN, transformers

1. INTRODUCTION

This paper describes our submission to the urban sound tagging challenge, which is carried out as Task 5 (see Fig. 1) of the DCASE 2020 challenge. We built an urban sound tagging neural network which takes as input (log-mel) spectrograms and metadata (week, day, hour and location) and outputs mulitlabel prediction vector. There are two different levels of granularity as shown in Fig. 2. The first one returns whether each of 23 sources of noise (fine-grained tags) is audible in the recording or not. The second one predicts coarse-grained tags among a list of eight. The relationship between coarse-grained and fine-grained tags is hierarchical. It is, therefore, possible to derive coarse-grained labeling from fine-grained labeling, but not the other way around.
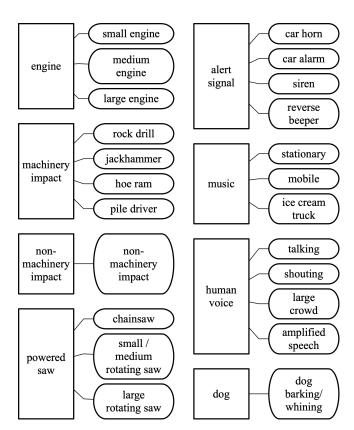

Figure 2: Hierarchical taxonomy of urban sound tags in the Task 5 of DCASE 2020.

Since 2013, the DCASE [1] challenges have provided numerous publicly available datasets and have gained an increasing research interest in audio pattern recognition. Though, there is still a need for a large scale dataset of generic real-world sound like ImageNet in image classification or Wikipedia data in natural language processing. To address this issue, in 2017, Google released AudioSet [2]. This dataset contains 2.1 millions of 10s audio sound grabbed from YouTube videos and annotated with presence / absence labeling of 527 types of sound events.

---

*Work done as part of Multitel internship.

The DCASE task 5 uses the SONYC-UST [2] [1] as main dataset. Similar to Audioset, it provides about 17 thousand 10s samples with coarse-grained and fine-grained tags (see Fig. 2) alongside a diversity of metadata such as spatio-temporal context. This dataset has been recorded by the SONYC acoustic sensor network and tagged by volunteers and SONYC team members.

2. RELATED WORK

Deep neural networks (DNN) have been extensively studied and applied over the last decade. Several deep learning neural networks have been proposed in machine listening research. First, as mentioned and summarized in [3], several CNN-based architectures have been applied to (log-mel) spectrogram of audio recordings followed by an activation function to predict the presence or absence of sounds.

Although CNNs act as a robust feature extractor, their receptive field have limited size. Therefore, they cannot capture long time dependency. To solve this problem, CRNNs were proposed to consider the long time information. The idea is to use a CNN combined with RNN or RNN-like layers such as GRU or LSTM. For example, in [4] a CNN followed by GRU layer is used. Moreover, this model includes a final pooling layer as it was designed for MIL [3] problems. Extensive comparison of pooling strategies has been made in [5].

Last but not least, as explained in [6], transformer has been proposed to take into account the long time dependency of time series. This approach is inspired of the "Attention Is All You Need" work and the success it acquired in the NLP field [7]. Replacing RNN layers by self attention layers has been proven to increase performances in general.

3. PREPROCESSING AND DATA AUGMENTATION

3.1. Spectrogram generation

Recordings are resampled to 44100 Hz and to generate (log-mel) spectrogram as the representation for the audio input data. Librosa [15] was used to compute these (log-mel) spectrograms. To compute and transforme the STFT to (log-mel) spectrogram, a Hanning window size of 2822 and hop length of 1103 samples were used with the number of bands being 64. The frequencies between 0Hz and 8000Hz are kept. Those numbers has been chosen to match those of TALNet original implementation [4].

3.2. Data augmentation

We used SpecAugment [20] which consists of warping the features, masking blocks of frequency channels, and masking blocks of time steps to supplement the training data. Moreover, several image data augmentation techniques [16] were employed such as ShiftScaleRotate, Grid distortion and Cutout. Mixup [17], a method that linearly mixes two random training examples with a scalar lambda sampled from a beta distribution, was used as well. We found out it was helping the model to obtain better scores.

---

[2]SONYC Urban Sound Tagging
[3]Multiple Instance Learning

3.3. Relabeling

A relabelling strategy has been used to relabel all the training set and a part of the validation set (the 4000 samples not labelled by the SONYCUST team). The 500 samples annotated by SONYCUST team remain untouched. The model used to relabel is System 2 (see below in Section 5) saved after reaching maximum macro AUPRC on coarse. Then, the 3 systems learn on the relabeled dataset.

4. FEATURE REPRESENTATION

4.1. Generic audio embeddings

One solution to generate generic audioset embeddings is to use released embeddings of audio clips extracted from a frozen neural network as feature extractor like OpenL3 [18] or VGGish [19]. However, as explained in [3], those methods did not work on improving systems obtaining better embedding features. Therefore, instead of using them, we decided to work on transfer learning.

One architecture have been extensively tested for this DCASE challenge: TALNet [4]. All parameters to calculate the embedding features are initialized from the pretrained audioset weights instead of being randomly initialized.

4.2. Specific audio embeddings

We trained a neural network from scratch to generate specific DCASE embeddings. A TALNet-like architecture has been improved. All parameters are randomly initialized.

Three improvements have been incorporated into the neural network. First, Group Normalization (GN) [10] is used instead of Batch Normalization (BN) to be independent of the batch dimension. Then, a second normalization technique called Weight Standardization [9] is used to accelerate training and smooth the loss and the gradients. Finally, the bi-GRU layer has been replaced by an encoder layer of a transformer [4] to decrease the number of parameters and increase performances. Each encoder consists of several encoder layers. For each layer, query, key and value transform matrices (see in [7]) were used on the outputs of the last convolution. After the computation of the feature correlation of different time steps, a softmax operation converts the correlation value to probability along the time steps.
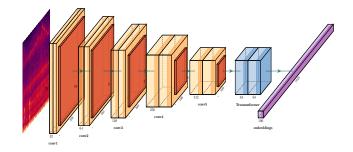

Figure 3: Overview of our TALNet-like architecture to compute specific audio embeddings.

---

[4]The decoder part which transforms an embedding back to output is not used.

4.3. Metadata embeddings

In [8], the author provides a model-agnostic vector representation for time, called Time2Vec, that can be easily imported into many existing and future architectures and improve their performances. The T2V representation was applied to all cyclic metadata because the representation is simple, invariant to re-scaling and captures both periodic and non-periodic patterns.
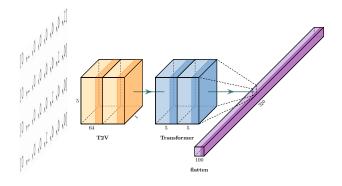

Figure 4: The architecture of metadata embeddings.

The encoder part of a transformer (see Section 4.2) is then applied to transform the T2V representation to a high level embedding.

5. MODELS

5.1. System1: Specific + Metadata

The first architecture takes as input the (log-mel) spectrogram and the metadata to generate two embeddings. First, a specific audio embedding is generated using the improved version of TALNet network. Then, the metadata embeddings are created by a T2V-Transformer network. Finally, the last layer is a fully-connected layer which converts these two embeddings (concatenated) into a multi-label classification problem. The neural network has been trained to label both coarse and fine tags jointly. System1 outputs a prediction vector of 31 sources of noise (8 coarse-grained tags + 23 fine-grained tags).

5.2. Systems 2 and 3: Generic + Specific + Metadata

As you can see in Fig. 5, the architecture of Systems 2 and 3 take as input the (log-mel) spectrogram and the metadata to generate three embeddings. First, a specific embedding is created by a TALNet-inspired CNN-Transformer. Then, a generic embedding is generated by a pre-trained TalNet network. Finally, the metadata embeddings are created by a T2V-Transformer network. The last layer is a fully-connected layer which converts these three embeddings (concatenated) into a multi-label classification problem.

The neural network has been trained to label both coarse and fine tags jointly. Two Systems have been submitted: System2 outputs a prediction vector of 31 sources of noise (8 coarse-grained tags + 23 fine-grained tags). System3 outputs 37 sources of noise. The fine other/unknown classes have been included during the training.
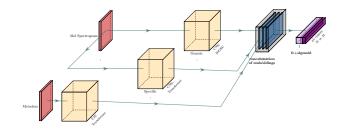

Figure 5: The structure of Systems 2 and 3 for urban sound tagging with spatiotemporal context.

6. TRAINING TECHNIQUES

Training was done on PyTorch Lightning [14], Ralamb (Radam [12] + LARS [13]) variant of the Adam algorithm with a learning rate of 10^{-3} was used as standard optimizer. On top of Ralamb, we use an algorithm called Lookahead which chooses a search direction, computes weight updates by looking ahead at the sequence of fast weights" generated by the Ralamb optimizer. In [11], the author show that Lookahead improves the learning stability and lowers the variance of its inner optimizer with negligible computation and memory cost.

The Systems 1 and 2 presented in Section 5 were trained with a jointly loss: Binary cross entropy (BCE) was used for the coarse-level and a Masked BCE loss (see in [1]) was used for the fine-level. System 3 outputs 37 predictions and a unique BCE loss was used for training.

7. RESULTS

We evaluate our models on the provided validation dataset. As the primary classification metric, the challenge uses the macro-averaged Area Under the Precision-Recall Curve (macro-averaged AUPRC) for ranking.

Our results can be found and compared to the baseline in Table 1 for coarse level and in Table 2 for fine level.

Table 1: Results for coarse level

	COARSE-GRAINED		
	Micro AUPRC	Micro F1	Macro AUPRC
Baseline	0.8352	0.7389	0.6323
System1	0.8622	0.7730	0.7601
System2	0.8906	0.7953	0.8011
System3	**0.8956**	**0.8039**	**0.8107**

Our methods were able to surpass the Micro and Macro AUPRC baseline scores in coarse-level and fine-level evaluation.

Table 2: Results for fine level

	FINE-GRAINED		
	Micro AUPRC	Micro F1	Macro AUPRC
Baseline	0.7329	0.6149	0.5278
System1	0.7983	0.6788	0.6349
System2	0.7932	0.6902	0.6817
System3	**0.8126**	**0.7116**	**0.7040**

8. CONCLUSIONS

This paper presents CRNNs for the Task 5 of the DCASE 2020. We investigated the performance of generic and/or specific audio embeddings with metadata embeddings.

In the future, we will continue to explore multilabel urban sound tagging, study the classwise performance and apply CRNNs in other tasks as sound event detection.

9. ACKNOWLEDGEMENT

10. REFERENCES

[1] Cartwright, M., Cramer, J., Mendez, A.E.M., Wang, Y., Wu, H., Lostanlen, V., Fuentes, M., Dove, G., Mydlarz, C., Salamon, J., Nov, O., Bello, J.P. SONYC-UST-V2: An Urban Sound Tagging Dataset with Spatiotemporal Context. In Proceedings of the Workshop on Detection and Classification of Acoustic Scenes and Events (DCASE), 2020.

[2] Jort F. Gemmeke and Daniel P. W. Ellis and Dylan Freedman and Aren Jansen and Wade Lawrence and R. Channing Moore and Manoj Plakal and Marvin Ritter, "Audio Set: An ontology and human-labeled dataset for audio events", Proc. IEEE ICASSP, New Orleans, LA, 2017.

[3] Kong, Qiuqiang, et al. "PANNs: Large-Scale Pretrained Audio Neural Networks for Audio Pattern Recognition." arXiv preprint arXiv:1912.10211, 2019.

[4] Yun Wang, "Polyphonic sound event detection with weak labeling", PhD thesis, Carnegie Mellon University, Oct. 2018.

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

[6] Kong, Qiuqiang, et al. "Sound Event Detection of Weakly Labelled Data with CNN-Transformer and Automatic Threshold Optimization." arXiv preprint arXiv:1912.04761, 2019.

[7] Vaswani, Shazeer, et al. "Attention Is All You Need" arXiv preprint arXiv:1706.03762, 2017.

[8] Kazemi, Seyed Mehran, et al. "Time2Vec: Learning a Vector Representation of Time." arXiv preprint arXiv:1907.05321, 2019.

[9] Qiao, Siyuan, et al. "Weight standardization." arXiv preprint arXiv:1903.10520, 2019.

[10] Wu, Yuxin, and Kaiming He. "Group normalization." Proceedings of the European Conference on Computer Vision (ECCV). 2018.

[11] Zhang, Michael, et al. "Lookahead Optimizer: k steps forward, 1 step back." Advances in Neural Information Processing Systems. 2019.

[12] Liu, Liyuan, et al. "On the variance of the adaptive learning rate and beyond." arXiv preprint arXiv:1908.03265, 2019

[13] You, Yang, Igor Gitman, and Boris Ginsburg. "Large batch training of convolutional networks." arXiv preprint arXiv:1708.03888, 2017.

[14] Falcon, W. A. "PyTorch Lightning." GitHub. Note: https://github. com/williamFalcon/pytorch-lightning, 2019.

[15] B. McFee, M. McVicar, S. Balke, V. Lostanlen, C. Thom,C. Raffel, D. Lee, K. Lee, O. Nieto, F. Zalkow, D. Ellis,E. Battenberg, R. Yamamoto, J. Moore, Z. Wei, R. Bittner,K. Choi, nullmightybofo, P. Friesch, F.-R. Stter, Thassilo,M. Vollrath, S. K. Golu, nehz, S. Waloschek, Seth,R. Naktinis, D. Repetto, C. F. Hawthorne, and C. Carr,librosa/librosa:0.6.3, Feb. 2019. [Online]. Available:https://doi.org/10.5281/zenodo.2564164

[16] E. K. V. I. I. A. Buslaev, A. Parinov and A. A. Kalinin, Albumentations: fast and flexible image augmentations, ArXiv e-prints, 2018.

[17] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, mixup: Beyond empirical risk minimization, arXiv preprint arXiv:1710.09412, 2017.

[18] Cramer, Jason, et al. "Look, listen, and learn more: Design choices for deep audio embeddings.", IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2019.

[19] Hershey, Shawn, et al. "CNN architectures for large-scale audio classification.", IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2017.

[20] Park, Daniel S., et al. "Specaugment: A simple data augmentation method for automatic speech recognition." arXiv preprint arXiv:1904.08779, 2019.