MULTI-TASK LEARNING PARADIGM FOR SOUND EVENT DETECTION

Technical Report

Krzysztof Rykaczewski

Samsung R&D Institute Poland, pl. Europejski 1, 00-844 Warsaw

ABSTRACT

In this technical report, we describe our system submitted to DCASE2020 task 4. This task evaluates systems for the detection of sound events in domestic environments using large-scale weakly labeled data. To perform this task, we propose resdual convolutional recurrent neural networks (CRNN) as our system and trained by datasets including strong and weak labels. We also use mean-teacher model based on confidence thresholding and smooth embedding method. In addition, we also apply specaugment for the labeled data shortage problem. Finally, we achieve better performance than DCASE2020 baseline system.

1. INTRODUCTION

This paper presents our approach to DCASE 2020 task 4. The task evaluates systems for the large-scale detection of sound events using weakly labeled data (without time boundaries). Sound Event Detection (SED) [1, 2] is a particularly challenging task because it consists of predicting not only possible event class but also their start and end times (the onset and offset of sound events). Moreover, recently it has attracted more and more attention because it can have a large impact on many fields, including smart cities (monitoring public security) and autonomous cars (abnormal sound detection) etc.

The dataset consists of weakly labeled audio clips, unlabeled in domain audio clips and unlabeled out domain audio clips taken from Youtube videos excerpt from domestic context. This dataset is a subset of Audioset [3] that consists of 10 classes of sound events: Speech, Dog, Cat, Alarm, Dishes, Frying, Blender, Running Water, Vacuum cleaner and Electric shaver. Each audio clip has a duration of 10 seconds and can be assigned to one or more labels.

Many methods has been applied in the sound event detection task, such as Gaussian Mixture Models (GMM), Hidden Markov Models (HMM), non-negative matrix factorization (NMF) and Deep Neural Network (DNN). Especially, convolutional neural networks that perform well in image recognition tasks also have good results in sound event detection problems.

We propose convolutional recurrent neural network (CRNN) architecture for sound event detection. Convolutional network can integrate information from different time resolutions and similarly to the multi-scale methods can capture both the fine-grained and coarse-grained features of sound events. Moreover, RNN structure followed the CNN can model the temporal dependencies including fine-grained dependencies and long-term dependencies [4].

This paper is organized as follows, Section 2 introduces related works. Section 3 shows selected methods. Section 4 shows experimental results. Section 5 concludes and forecasts our work.

2. RELATED WORK

Many methods like mixture Gaussian models (GMMs), non-negative matrix factorization (NMFs) and hidden Markov models (HMMs) have been used to model sound events. However, the emergence of deep learning has opened a new era in data mining and analysis, and in particular in the field of artificial intelligence. Deep neural networks have become the latest state-of-the-art in many fields of application including classification and detection tasks.

Recently, convolutional neural networks (CNNs) have achieved state-of-the-art performance in image classification [5]. A CNN consists of several convolutional layers followed by fully-connected layers. Each convolutional layer consists of filters to convolve with the output from the previous convolutional layer. In general, such filters can capture local patterns in feature maps, such as edges in lower layers and complex profiles in higher layers. In particular (after applying this to audio spectrograms) using such methods we can analyze the visual representation of sound.

3. SELECTED METHODS

At the beginning the data was subjected to preprocessing. Audio signal shorter than 10 seconds was padded with zero or minimal value in the signal. A simple algorithm was applied to cut silence. In general, we resample all audio files to 16 kHz and calculate features, i.e. log-mel spectograms with the following parameters [6]: STFT window size 512, hop length 882, number of mel-bins were 64, time duration 10 seconds. The window size of the two scales for short-time Fourier transform was 4096.

Because the training dataset is not balanced by classes, and also has a small enough size, two types of augmentation were applied to the data: time stretch and pitch shifting.

Since training dataset is small and not class-balanced we applied several augmentations to the data. We applied time stretch and pitch shift. We also used SpecAugment [7], a technique known for data augmentation in speech recognition.

The configuration of CRNN is summarized in Table 1.

We apply batch normalization (BN) [8] after each convolutional layer to stabilize training followed by a rectifier (ReLU) nonlinearity. Then we apply the global max pooling (GMP) operation on the feature maps of the last convolution layer to summarize the feature maps to the vector. GMP can maximize information about the time and frequency of sound events in the spectrogram, so this is invariant for time or frequency shifts. Finally, a fully connected layer is applied to the combined vector, followed by sigmoidal or softmax nonlinearity to derive the probabilities of audio classes.

In post-processing stage median filter was applied to the mask received from CRNN.

Layer
Input: Log mel-band energy (625×64)
Conv3D: 64 filters, 3×3, BN, ReLU
MaxPool2D: 1x4
Conv3D: 64 filters, 3×3, BN, ReLU
MaxPool2D: 1x4
Conv3D: 64 filters, 3×3, BN, ReLU
MaxPool2D: 1x4
Bidirectional LSTM: 128 units
TimeDistributed, Dense with 128 units, ReLU
GlobalMaxPooling
Dense: 1024 units, ReLU
Dense: 1024 units, ReLU
Dense: 10 units, ReLU

Table 1: Neural network configuration.

Overall F1-score: 36.01%	
Alarm bell ringing	46.9%
Blender	44.2%
Cat	45.1%
Dishes	22.3%
Dog	19.2%
Electric shaver toothbrush	40.8%
Frying	19.1%
Running water	26.6%
Speech	45.6%
Vacuum cleaner	50.3%

Table 2: F1-score results.

4. EXPERIMENT RESULTS

The results of audio tagging and weakly supervised sound event detection are given in Table 2.

We use Adam optimizer [9] with a learning rate of 0.001. A batch size of 64 is used to sufficiently use the GPU. We trained the model for 100 iterations.

In evaluation, a F1-score of 18.6% is achieved using our system.

5. CONCLUSIONS

Convolutional neural networks work well in image recognition tasks. However, considerable research is still needed to reliably detect sound events in realistic soundscapes in which many sounds are present.

Future work with unlabeled and unbalanced training data can improve system performance.

6. REFERENCES

[1] R. Serizel, N. Turpault, A. Shah, and J. Salamon, "Sound event detection in synthetic domestic environments," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 86–90.

[2] N. Turpault, R. Serizel, J. Salamon, and A. P. Shah, "Sound event detection in domestic environments with weakly labeled data and soundscape synthesis," 2019.

[3] J. F. Gemmeke, D. P. 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 *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 776–780.

[4] G. Parascandolo, H. Huttunen, and T. Virtanen, "Recurrent neural networks for polyphonic sound event detection in real life recordings," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 6440–6444.

[5] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[6] K. Choi, G. Fazekas, and M. Sandler, "Automatic tagging using deep convolutional neural networks," *arXiv preprint arXiv:1606.00298*, 2016.

[7] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "Specaugment: A simple data augmentation method for automatic speech recognition," *arXiv preprint arXiv:1904.08779*, 2019.

[8] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[9] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.