DATA AUGMENTATION BASED SYSTEM FOR URBAN SOUND TAGGING

Technical Report

Jisheng Bai[1], *Chen Chen*[1], *Mou Wang*[2],
Jianfeng Chen[1], Xiaolei Zhang[2]

Northwestern Polytechnical University
Xi'an, 710072, China
{baijs, cc_chen524, wangmou21}@mail.nwpu.edu.cn
{cjf, xiaolei.zhang}@nwpu.edu.cn

Qingli Yan[2]

Xi'an University of Post and Telecommunications
Xi'an, 710121, China
yql@xupt.edu.cn

ABSTRACT

In this report, we introduce our system for Task5 of Dcase 2020 challenges (Urban Sound Tagging with Spatiotemporal Context). We presents a fusion system for Task5 based on different features and data augmentation. The task focuses on predicting whether each of 23 sources of noise pollution is present or absent in a 10-second scene with original recordings and addtional spatiotemporal context [1]. There are two levels of taxonomies to train a model. To explore features in detecting various sources of urban sound, we extracted four different features from original recordings. We applied a 9-layer convolutional neural network(CNN) as our primary classifier. To prevent the imbalance between classes, we applied data augmentation method. The experiment results show that our system performed better than baseline on validation data.

Index Terms— Urban Sound Tagging, features, data augmentation, CNN

1. INTRODUCTION

The city of New York, like many others, has a "noise code".The noise code presents a plan of legal enforcement and thus mitigation of harmful and disruptive types of sounds. Although harmful levels of noise predominantly affect low-income and unemployed New Yorkers, these residents are the least likely to take the initiative of filing a complaint to the city officials. For reasons of com-fort , public health and improving fairness, accountability, and transparency in public policies against noise pollution, to control and learn the distribution of noise is essential for government.

Meanwhile some of the most successful techniques in the challenge could inspire the development of an embedded solution for low-cost and scalable monitoring, analysis, and mitigation of urban noise. Due to the strong ability of feature extraction, CNNs have been widely used in computer vision. In sound tagging and classification, CNNs achieve great results as well, such as bird sound detection [2], acoustic scene classification [3] and domestic activities [4].

In this paper, we explore features and data augmentation methods in detecting various sources of urban sound. Firstly we try a set of hyper parameters to obtain a better result on validate set. Secondly We use randomly add and mix-up augmentation to make the train set more balance. Finally, we explore log-mel, log-linear and HPSS (harmonic percussive source separation) features for tagging different source of urban sounds [5]. We make a fusion system to get a better performance by taking advantages of different features and data augmentation methods.

The paper is organized as follows: In section 2, we analyse the imbalance of train set classes. In section 3, we introduce our data augmentation method details. In section 4, we show the details of feature extraction and the architecture of the network. In section 5, we show the experiment results of features and data augmentation. The method of making a fusion system is introduced in the end.

2. TRAIN SET ANALYSIS

The development dataset contains all of the recordings and annotations from DCASE 2019 Task 5 (both development and evaluation sets), plus almost 15000 additional recordings with crowdsourced annotations. All of the recordings are grouped into a train split (13538 recordings) and validate split (4308 recordings). The train and validate splits are disjoint with respect to the sensor from which each recording came.

The two-level urban sound taxonomy consists of 8 coarse-level and 23 fine-level sound categories, e.g., the coarse alert signals category contains four fine-level categories: reverse beeper, car alarm, car horn, siren.

For train set, we count presence of each class class for every recording from two aspects. Firstly, for each recording, if there is only one class presented in a 10-second recording, it is counted. Secondly, we count the presence of each class of all recordings. The two aspects are showed in Figure 2 and 3.

It can be concluded that the train set is imbalanced. There are less recordings for class 2,3,4,6 and 8 (machinery-impact, non-machinery-impact, powered-saw, music and dog), especially for class 3,4,6 and 8.

3. DATA AUGMENTATION

To handle the imbalanced dataset, improve the generalization ability of model and prevent overfitting, we present two methods of data augmentation.

As it is widely applied in bird sound classification, we randomly add only-one-class-present recordings of class 2,3,4,6 and 8 into a new recording, the labels are generated at the same time. During the processing, loud sounds like machinery-impact and powered-saw could mask the music or non-machinery-impact sounds, so amplitude factors are applied for each augmentation class.

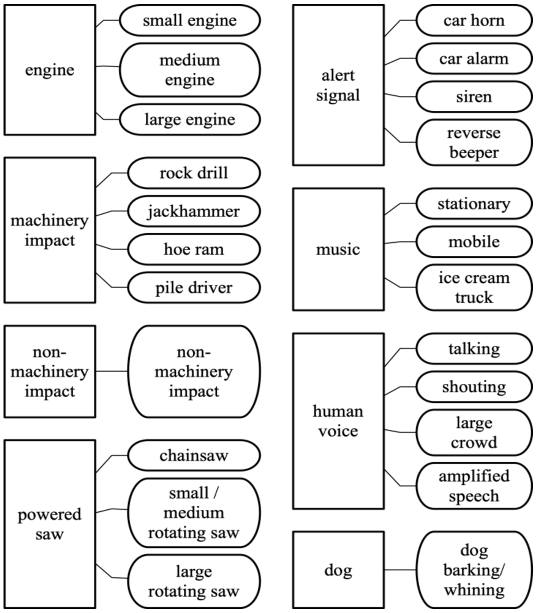

Figure 1: Taxonomy of urban sound tags in the DCASE Urban Sound Tagging task.

	CNN9	
features	Log-mel	Log-linear
Conv1	(3*3@64,BN,Relu)*2	
Pool1	2*2 average pooling	
Conv2	(3*3@128,BN,Relu)*2	
Pool2	2*2 average pooling	
Conv3	(3*3@256,BN,Relu)*2	
Pool3	2*2 average pooling	
Conv4	(3*3@256,BN,Relu)*2	
Pool4	1*1 average pooling	
Dense	TimeDistributed	
Dense	TimeDistributed	
AutoPool	AutoPool1D	

Table 1: Model architecture

On the other hand, we applied mix-up data augmentation during training[6]. It can be expressed as:

$$\tilde{x} = \lambda x_i + (1 - \lambda) x_j \tag{1}$$

$$\tilde{y} = \lambda y_i + (1 - \lambda) y_j \tag{2}$$

where x_i, x_j are input features, y_i, y_j are target labels, $\lambda \in [0, 1]$ is a random number drawn from the Beta (c, a) distribution. We can get more training samples without extra computing resource, new samples are linear interpolated of real samples.

4. FEATURE AND MODEL ARCHITECTURE

4.1. Features

Recordings are resampled to 22050 Hz and to generate short time Fourier transform (STFT) spectrograms with a Hanning window size of 1024 and hop length of 512 samples. Mel filters with bands of 64 are used to transformed STFT spectrogram to mel spectrogram, and frequencies lower than 50 Hz and beyond 10000 Hz are removed. Meanwhile, linear filters are also applied to generate linear spectrograms.

The harmonic percussive source separation (HPSS) [7] can split a signal $w(t)$ into harmonic part $h(t)$ and percussive part $p(t)$ and there are several approaches to separate. We can simplify the separation procedure as follows [8]

$$w(t) \xrightarrow{\mathrm{HPSS}(l)} h(t), p(t) \tag{3}$$

The harmonic spectrogram is generated from STFT spectrogram with librosa toolkit. To decrease the input size, mel filters with 64 bands are applied to generate mel-harmonic spectrogram. All the spectrograms are calculated by the log algorithm to get log spectrograms.

As the spatial context, via latitude and longitude values, and temporal context, via hour of the day, day of the week, and week of the year, are processed by the same method in baseline, finally giving a 85 values for each clip. And we concatenate them with feature vectors on frame level after the convolutional blocks. Then the concatenated feature is fed into the dense layer.

4.2. Model architecture

The CNN architecture is a VGG-like model. Batch normalization [9] is applied to speed up and prevent overfitting during train steps. And leaky_Relu or gated function are used as a non-linear activation after batch normalization. Average pooling with size of 2*2 to reduce the feature map. Then the frequency axis is aver-aged out and frame axis is maxed out after the last convolutional layer.

For training, Tensorflow and Keras are implemented. Sigmoid cross entropy is utilized as loss function and AdamOptimizer as optimizer with a learning rate of 0.001. Training is done with batch size of 64 and we early stop the training if the macro-auprc does not improve in last 3 steps.

5. EXPERIMENT RESULTS AND DISCUSSION

We experiment log-mel, log-linear and log-mel-h on CNN9 and best scores of each coarse class are showed in Table 2.

Log-mel spectrogram preforms better than log-linear spectrogram in non-machinery, powered-saw and human-voice, especially dog. As for machinery, log-linear gets 0.1 improvement than log-mel. Log-mel-h achieves the best score of 0.69. It may be explained that log-linear spectrogram shows high resolution in higher frequency areas, this could take advantages of some machinery sounds consist of high frequency components. Harmonic components can explore music sounds from other classes.

Then we applied two data augmentation methods based on log-mel and CNN9. We finally get 5,000 more training recordings though random-adding method. And for mixup, we apply mixup not only on one-hot labels, but also on spatial and temporal context. The results are showed in Table 3, the evaluation metrics on validate split are macro-auprc, micro-auprc and F-score.

Compared to the no aug method, the randomly add decreases about 0.4 to 0.6 on three scores, mix up improves the macro-auprc score from 0.68 to 0.72. The reason of decrease of randomly add is confused, may partly due to the inappropriate amplitude factors or the labels.

A CRNN architecture is also experimented. We just feed the feature maps which are extracted from the CNN9 into a simple

Coarse class	Log-mel	Log-linear	Log-mel-h
1_Engine	0.859	0.860	0.854
2_Machinery	0.621	0.728	0.685
3_Non-machinery	0.558	0.539	0.505
4_Powered-saw	0.713	0.705	0.699
5_Alert	0.940	0.926	0.941
6_Music	0.639	0.524	0.692
7_Human-voice	0.978	0.968	0.968
8_Dog	0.552	0.183	0.073

Table 2: Best macro-auprc of coarse classes of two features

Data augmentation	Mi-auprc	Ma-auprc	Mi-F1
No aug	0.86	0.68	0.78
Randomly add	0.82	0.62	0.74
Mix up	0.86	0.72	0.77

Table 3: Data augmentation results

GRU. And mixup augmentation is applied on the CRNN as well. The CRNN can achieve 0.67/0.70 macro-auprc score W/O mixup.

As the fusion system, we extract log-mel, log-linear and log-mel-h as input features, and apply mixup during the training steps. Then we get three prediction output files, the machinery predictions of log-linear, the music predictions of log-mel-h and the predictions of the rest classes of log-mel are merged into one final prediction.

6. CONCLUSION

In this paper, we present a data augmentation based system for Urban Sound Tagging. In our approach, three different features are generated as inputs of the networks based on a VGG-like CNN architectures for urban sound tagging. Then, we apply randomly add and mixup data augmentation methods during the training steps. Finally, we fused different feature and mixup based results as the system output. For further work, the data augmentation of randomly adding sounds will be studied.

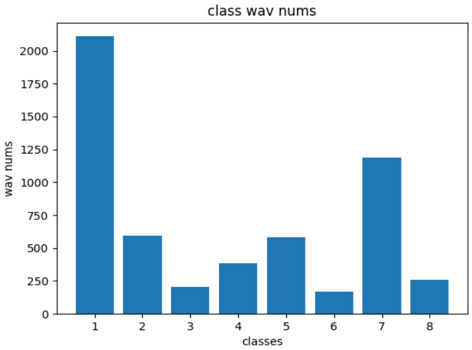

Figure 2: Only one class presented numbers of all recordings.

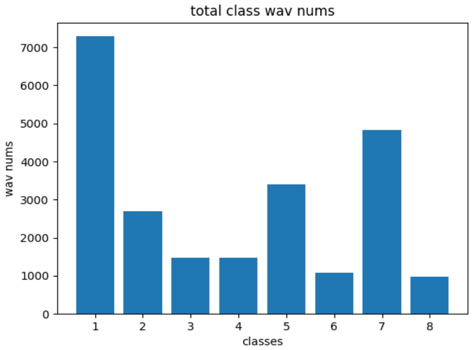

Figure 3: Each class presented numbers of all recordings.

7. REFERENCES

[1] M. Cartwright, A. E. M. Mendez, J. Cramer, V. Lostanlen, G. Dove, H.-H. Wu, J. Salamon, O. Nov, and J. Bello, “SONYC urban sound tagging (SONYC-UST): A multilabel dataset from an urban acoustic sensor network,” in *Proceedings of the Workshop on Detection and Classification of Acoustic Scenes and Events (DCASE)*, October 2019, pp. 35–39. [Online]. Available: http://dcase.community/documents/workshop2019/proceedings/DCASE2019Workshop_Cartwright_4.pdf

[2] M. Lasseck, “Acoustic bird detection with deep convolutional neural networks,” in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018)*, 2018, pp. 143–147.

[3] Y. Xu, Q. Kong, W. Wang, and M. D. Plumbley, “Surrey-CVSSP system for DCASE2017 challenge task4,” DCASE2017 Challenge, Tech. Rep., September 2017.

[4] L. Vuegen, P. Karsmakers, B. Vanrumste, *et al.*, “Weakly-supervised classification of domestic acoustic events for indoor monitoring applications,” in *In proceedings of IEEE Conference on Biomedical and Health Informatics 2018*. IEEE, 2018.

[5] J. Bai, C. Chen, and J. Chen, “A multi-feature fusion based method for urban sound tagging,” in *2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2019, pp. 1313–1317.

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

[7] D. Fitzgerald, “Harmonic/percussive separation using median filtering,” 2010.

[8] H. Tachibana, N. Ono, and S. Sagayama, “Singing voice enhancement in monaural music signals based on two-stage harmonic/percussive sound separation on multiple resolution spectrograms,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 1, pp. 228–237, 2013.

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