ENSEMBLE OF ARCFACE BASED SYSTEMS FOR UNSUPERVISED ANOMALOUS SOUND DETECTION UNDER DOMAIN SHIFT CONDITIONS

Technical Report

Qiping Zhou

PFU SHANGHAI Co., LTD
46 Building 4~5 Floors, 555 GuiPing Road
XuHui District, Shanghai 200233, CHINA
qpzhou.pfu@cn.fujitsu.com

ABSTRACT

In this report, we outline our ensemble of models solution for the DCASE 2021 challenge's Task 2 (Unsupervised Anomalous Sound Detection for Machine Condition Monitoring under Domain Shifted Conditions) [1]. The basic approach follows our DCASE2020 Task 2 system [2]. In 2021 we diversify our CNN backbone architecture and input size. The final submissions are the ensemble of 6 models for each machine type. Models are trained on source domain data and fine-tuned on target domain data to improve the performance on the domain shifted data.

Index Terms**—** DCASE2021, anomalous sounds detection, metric learning, ArcFace

1. INTRODUCTION

DCASE2021 Task2 has two main challenges:

1. To detect unknown anomalous sounds under the condition that only normal sound clips have been provided as training data, as in DCASE2020 Task2.
2. To perform under the conditions that the acoustic characteristics of the training data and the test data are different.

For challenge 1, we train a section classification network for each machine type, which tries to identify each section under a certain machine type. In test phase, the last classification layer of the network is removed. Each input spectrogram is mapped into a 128-dim embedding vector, which is used for measuring cosine similarity in angular space.

For challenge 2, the models trained on source domain data are further fine-tuned on the target domain data. We use serval fine-tuning strategies to improve performance.

1.1. DCASE 2021 Task2 Dataset

The data used for this task comprises parts of MIMII DUE [3] and the ToyADMOS2 Dataset [4] consisting of the normal/anomalous operating sounds of seven types of toy/real machines. Each recording is single-channel, 10-second audio sampled at 16,000 Hz that includes both the sounds of a machine and its associated equipment as well as environmental sounds. All the training data (normal) in development dataset and additional training dataset is used for training our models. The performance is evaluated by using the test data in development dataset.

Table 1: Architecture of MobileFaceNet based network

Input	Operator	t	c	n	s
1024×32×1	conv3x3	-	64	1	2
512×16×64	depthwise conv3x3	-	64	1	1
512×16×64	bottleneck	2	64	5	2
256×8×64	bottleneck	4	128	1	2
128×4×128	bottleneck	2	128	6	2
64×2×128	bottleneck	4	128	1	2
32×1×128	bottleneck	2	128	2	1
32×1×128	conv1x1	-	512	1	1
32×1×512	linear GDConv32x1	-	512	1	1
1×1×512	linear conv1x1	-	128	1	1

1.2. Audio preprocessing

Follow [5], we load the audio clips with their raw sampling rate (16,000 Hz), and the spectrogram is adopted through a Short-Time Fourier Transform (STFT). We use librosa package [6] to apply STFT, the length of the window (nFFT) is 2046, the hop length is 512, so the height of the spectrogram is 1024 (1 + nFFT/2). Then spectrogram is split into 32 columns piece and each piece is normalized by subtracting the mean and dividing by the standard deviation. We use these 1024 × 32 shape data to train our network, and the 512 × 32 reshaped version is also trained, we ensemble all the models' predictions scores finally.

2. SOLUTIONS

2.1. Training Loss

ArcFace loss [7] is employed to train our machine section classification network. By incorporating additive angular margin penalty, ArcFace loss help the network to learn discriminative feature representation that has high intra-class similarity and low inter-class similarity, leading to find an accurate decision boundary between trained(normal) and un-trained(i.e., anomalies).

ArcFace loss has two hyper-parameter: the additive angular margin penalty parameter m and the re-scale factor s. We use our DCASE2020 Task's setting m=0.05 and s=30, respectively.

We also tried Sub-center ArcFace loss [8], but the score was not good enough in this challenge, we left it for future work.

Table 2: Evaluation results: Harmonic mean of the AUC [%]/partial AUC [%] on Development Dataset

Algorithm	Toy Car	Toy Train	Fan	gearbox	pump	slider	valve
Baseline1 (Autoencoder)	62.49/52.36	61.71/53.81	63.24/53.38	65.97/52.76	61.92/54.41	66.74/55.94	53.41/50.54
Baseline2 (MobileNetV2)	56.04/56.37	57.46/51.61	61.56/63.02	66.70/59.16	61.89/57.37	59.26/56.00	56.51/52.64
Submission 1	76.12/62.65	**72.00/59.92**	**76.61/71.45**	80.58/55.32	71.66/62.74	**81.67/68.16**	**70.11/56.23**
Submission 2	77.34/62.81	71.42/60.43	76.26/70.91	**80.68/55.59**	**72.73/62.89**	81.48/68.50	69.82/56.41
Submission 3	**78.03/65.37**	71.30/60.49	75.10/70.58	80.26/55.22	71.73/62.71	81.65/68.82	69.99/56.83
Submission 4	77.69/64.20	70.90/58.98	72.97/69.03	79.71/54.00	71.37/62.67	80.88/68.45	69.38/57.20

2.2. Models

Three backbone architectures are incorporated: MobileFaceNet [9] based network, ResNet-50 [10] and EfficientNet-B0 [11]. Table 1 shows the architecture of our MobileFaceNet based network. For each backbone architecture, we trained 2 input shape version: 1024 × 32 and 512 × 32. So finally we got six models for each machine type.

2.3. Data Augmentation

For data augmentation, we employed Mixup [12] strategy, which can help avoid data memorization and improve model generalization. Follow [13], Mixup is performed on the raw audio data to generate the additional virtual audio for training. The mixup weights are sampled from the uniform distribution.

2.4. Training Details

All models are trained from scratch without using any pre-trained model or external data resources. We use SGD to optimize models. The learning rate is set to 0.05. The training of each model contains two phase:

- **Phase 1**: Train models on source domain data.

Each section's 10-second training audios are randomly picked from the source domain audios. A certain section's audio is mixed with the randomly selected other section's audio to generate the new 10-second augmented audio. Then spectrograms are extracted from the new 10-second audio and the 1024 × 32 or 512 × 32 pieces are randomly cropped to feed to the network. We train each model for 250 epochs.

- **Phase 2**: Fine-tune models trained in phase 1.

In one training step, we randomly select one section's audios from target domain and the other section's audios from source domain. The same Mixup strategy as in the phase 1 is employed. We train each model for 5000 steps.

In phase 2, we tried serval fine-tuning strategies, such as:

- Freeze layers until the last residual block
- Freeze all layers in CNN backbone
- Freeze ArcFace layer's weights

Table 3: Summarization of hyper-parameters

Parameters for signal processing	
Sampling rate	16,000 Hz
FFT length	2046 pts
FFT hop length	512 pts
ArcFace loss parameters	
Margin Parameter (m)	0.05
Re-scale factor (s)	30
Learning strategy	
learning rate	0.05
Mixup rate(uniform distribution)	[0, 0.5)
Other parameters	
Batch size	48
K (for embedding's similarity calculation)	10

2.5. Submissions

In Table 2, we present harmonic mean of the AUC and partial AUC for 2 baseline systems and our 4 submissions. The 4 submissions are implemented as follows:

- Submission 1: For each machine type, we use the best single model's prediction results.
- Submission 2: Top-3 best prediction results are averaged.
- Submission 3: Top-5 best prediction results are averaged.
- Submission 4: All prediction results are averaged.

3. CONCLUSIONS

This technique report briefly presents our ensemble of ArcFace based systems for the task 2 of DCASE2021 challenge. On the basis of our 2020 Task 2 approach, we diversify CNN backbone and fine-tuned on target domain data to improve the performance on the domain shifted data. Our method significantly outperforms the baseline systems.

4. REFERENCES

[1] Y. Kawaguchi, K. Imoto, Y. Koizumi, N. Harada, D. Niizumi, K. Dohi, R. Tanabe, H. Purohit, and T. Endo, “Description and discussion on DCASE 2021 challenge task 2: Unsupervised anomalous sound detection for machine condition monitoring under domain shifted conditions,” in arXiv preprint arXiv:2106.04492, 2021.

[2] Q. Zhou, “ArcFace based Sound MobileNets for DCASE 2020 Task 2,” Tech. report in DCASE2020 Challenge Task 2, 2020.

[3] R. Tanabe, H. Purohit, K. Dohi, T. Endo, Y. Nikaido, T. Nakamura, and Y. Kawaguchi, “MIMII DUE: sound dataset for malfunctioning industrial machine investigation and inspection with domain shifts due to changes in operational and environmental conditions,” in arXiv e-prints: 2006.05822, 1–4, 2021.

[4] N. Harada, D. Niizumi, D. Takeuchi, Y. Ohishi, M. Yasuda, and S. Saito, “ToyADMOS2: another dataset of miniature-machine operating sounds for anomalous sound detection under domain shift conditions,” in arXiv preprint arXiv:2106.02369, 2021.

[5] O. Dong and I. Yun, “Residual Error Based Anomaly Detection Using Auto-Encoder in SMD Machine Sound,” Sensors, 2018, 18(5), pp. 1308–.

[6] B. McFee, C. Raffel, D. Liang, D. P.W. Ellis, M. McVicar, E. Battenberg, and O. Nieto, “librosa: Audio and Music Signal Analysis in Python,” in Proceedings of the 14th Python in Science Conference, Kathryn Huff and James Bergstra, Eds., 2015, pp. 18 – 24.

[7] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, “ArcFace: Additive Angular Margin Loss for Deep Face Recognition,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4690–4699, 2019.

[8] J. Deng, J. Guo, T. Liu, M. Gong, and S. Zafeiriou, “Sub-center ArcFace: Boosting Face Recognition by Large-Scale Noisy Web Faces,” in Proceedings of European Conference on Computer Vision. Springer, 2020, pp. 741–757.

[9] S. Chen, Y. Liu, X. Gao, and Z. Han, “MobileFaceNets: Efficient CNNs for Accurate Real-Time Face Verification on Mobile Devices,” in Chinese Conference on Biometric Recognition. Springer, 2018, pp. 428–438.

[10] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[11] M. Tan and Q. V. Le, “EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks,” in Proceedings of International Conference on Machine Learning, 2019.

[12] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, “Mixup: beyond empirical risk minimization,” in Proceedings of International Conference on Learning Representations, 2018, pp. 1–8.

[13] Y. Zhu, T. Ko, and B. Mak, “Mixup Learning Strategies for Text-Independent Speaker Verification,” in Proceedings of Interspeech, 2019, pp. 4345–4349.