FIRST-SHOT UNSUPERVISED ANOMALOUS SOUND DETECTION BASED ON SELF-SUPERVISED LEARNING

Technical Report

Jiawei Yin,Yu Gao,Wenbin Zhang

AI Innovation Center,Midea Group,Shanghai, China
{yinjw25,gaoyu11,zhangwb87}@midea.com

ABSTRACT

This technical report contains a description of Midea's submission to Task 2 "First-Shot Unsupervised Anomalous Sound Detection for Machine Condition Monitoring" of DCASE Challenge 2024. Compared with previous challenges, this task focuses on the first-shot problem, and some attribute information is unavailable, which brings many challenges. Our proposed system is based on a self-supervised learning approach by using a convolutional neural network to extract feature vectors from input sounds and an anomaly detection algorithm to detect abnormal sounds. The proposed method is evaluated using the DCASE 2024 Task 2 development dataset. The results show that the proposed method can effectively extract the sound features and significantly outperforms the baseline in detection performance.

Index Terms— anomalous sound detection, unsupervised learning, domain generalization, data augmentation

1. INTRODUCTION

DCASE 2024 Challenge Task 2 [1] aims at First-Shot Unsupervised Anomalous Sound Detection for Machine Condition Monitoring.The purpose of the task is to detect whether the sound emitted by the target machine type is abnormal. However, since the frequency of anomalies is very low, there is a lack of abnormal data, which makes it not a typical binary classification problem but an unsupervised problem. At the same time, the task focuses on the first-shot problem, which makes it impractical to use the test data in the development set to tune the hyperparameters of each machine type because the machine type may be novel. In addition, the source domain and target domain data are seriously unbalanced, which may cause the performance of the model to deteriorate on the target domain [2, 3]. The organizers proposed two baseline methods [4] based on autoencoders, using mean squared error (MSE) and Mahalanobis distance (MAHALA) as anomaly scores. Both methods performed well on the source domain, but poorly on the target domain. This is mainly attributed to the small amount of data in the target domain.

In the previous challenge, a common practice among the top-ranked teams was to train a classifier as an auxiliary task, training embeddings based on different machine types and working conditions, which effectively learned the distribution of normal samples. The final anomaly score is based on the similarity between the features extracted from the test data and the features extracted from normal sounds. This year, the task has put forward new requirements. The attribute information of the machine is not always available, which requires the system to work well with and without attribute information. This makes it more difficult to train embeddings using auxiliary classification tasks, and the model is more likely to overfit without attribute information. Therefore, we propose an improved baseline algorithm and use a variety of data augmentation methods to enhance the detection performance.

2. PROPOSED METHOD

2.1. Proposed baseline model

We used the model in [5] as our backbone network of our baseline model. For the input data, we first unified it into a fixed length by repeating and cropping the waveform, which facilitates the subsequent processing of the data. In order to capture dynamic and static frequency information, we used two different input feature representations, namely the magnitude spectrum and the magnitude spectrogram. Among them, we used the full magnitude spectrum to obtain a very high frequency resolution. For the magnitude spectrogram, we obtained it by short-time Fourier transform (STFT) and subtracted the time average to remove static frequency information, where the sampling window size was set to 1024, the hop size was 512, the maximum frequency was set to 8000Hz, and the minimum frequency was 200Hz. The dual-branch network consists of an improved ResNet architecture and multiple one-dimensional convolutions. For dual branches, we use Convolutional Block Attention Module (CBAM) [6] to adaptively adjust the channel and spatial weights of the feature map, highlight important features, suppress irrelevant features, and thus improve the classification performance. CBAM combines channel attention and spatial attention to effectively improve the expressiveness of the model. Compared with traditional convolutional modules, CBAM significantly enhances the representation ability of the model with limited increase in parameters, and improves the accuracy and robustness of the network. The outputs of the dual branches are concatenated to form an embedding of size 256, which we use to calculate cosine similarity as our anomaly score. The training is performed using the subcluster AdaCos loss [7] and the Adam optimizer. The loss function of the model consists of the cross entropy loss of the joint category of machine ID and attribute $\mathcal{L}_{joint}$ and the cross entropy loss of the individual machine ID $\mathcal{L}_{machine}$, defined as follows:

$$\mathcal{L} = \mathcal{L}_{joint} + \mathcal{L}_{machine} \tag{1}$$

2.2. Data augmentation

In the task, the data of different working states of each machine is unbalanced, especially the data of source domain and target domain

Table 1: Anomaly detection results [%] for different machine types

Machine type	Criteria	MSEAE	MHLAE	Submission 1	Submission 2	Submission 3	Submission 4
ToyCar	AUC(source)	66.90	63.01	55.92	53.08	56.72	48.88
	AUC(target)	33.70	37.35	57.72	48.48	47.76	46.76
	pAUC	48.77	51.04	48.68	50.53	49.53	48.00
ToyTrain	AUC(source)	76.63	61.99	68.56	65.72	68.20	68.04
	AUC(target)	46.92	39.99	53.76	67.40	65.44	66.20
	pAUC	47.95	48.21	50.26	58.16	60.58	53.74
bearing	AUC(source)	62.01	54.43	63.96	68.64	59.60	61.56
	AUC(target)	61.40	51.58	77.12	72.08	69.88	68.04
	pAUC	57.58	58.82	59.89	54.42	55.89	51.79
fan	AUC(source)	67.71	79.37	64.56	58.16	53.68	61.56
	AUC(target)	55.24	42.70	66.36	68.28	65.68	63.72
	pAUC	57.53	53.44	55.00	56.16	55.89	57.11
gearbox	AUC(source)	70.40	81.82	68.96	75.04	66.68	66.28
	AUC(target)	69.34	74.35	73.60	75.48	69.44	71.32
	pAUC	55.65	55.74	52.26	53.63	55.26	51.68
slider	AUC(source)	66.51	75.35	75.36	95.52	96.84	95.56
	AUC(target)	56.01	68.11	76.84	88.36	87.88	88.76
	pAUC	51.77	49.05	70.79	73.68	72.37	75.26
valve	AUC(source)	51.07	55.69	89.12	88.24	96.52	88.68
	AUC(target)	46.25	53.61	67.56	65.28	63.84	64.28
	pAUC	52.42	51.26	65.00	63.84	69.63	64.16
All(hmean)	AUC(source)	64.99	65.77	68.24	69.34	67.71	67.00
	AUC(target)	50.26	49.51	66.44	67.39	65.30	64.98
	pAUC	52.84	52.28	56.47	57.83	58.94	56.23

of different machines, which may be one of the reasons why the model performs well in the source domain but poorly in the target domain. To solve this problem, we use a variety of mixed data augmentation methods and apply them to our the baseline model as our submission of the task.

One of the most classic methods to solve data imbalance is the Synthetic Minority Over-sampling Technique (SMOTE) [8]. SMOTE balances the data set by generating synthetic samples. Compared with simple resampling, the synthetic samples generated by SMOTE are more diverse. For single sample data, SMOTE cannot enhance it. We will first use time warping [9] to increase the sample size of the data to ensure the enhancement effect. For categories without attribute information, we use the K-means clustering [10] method to cluster the data first to increase the number of joint categories and avoid model overfitting. During the training process, we use the Mixup [11] data enhancement technology to generate new training samples. Mixup helps improve the generalization ability of the model by mixing two groups of samples and their labels, and performs better when processing noisy data.

3. RESULTS AND SUBMISSIONS

A total of four different systems were submitted to the challenge. Table 1 shows the comparison of our submissions with two baseline methods.Submission 1 is the result of our proposed baseline model, submission 2 is the result after clustering the data without attribute information, submission 3 is the result of using Mixup technology, and submission 4 is the result of using time warping, Smote and Mixup technology.

4. REFERENCES

[1] T. Nishida, N. Harada, D. Niizumi, D. Albertini, R. Sannino, S. Pradolini, F. Augusti, K. Imoto, K. Dohi, H. Purohit, T. Endo, and Y. Kawaguchi, “Description and discussion on DCASE 2024 challenge task 2: First-shot unsupervised anomalous sound detection for machine condition monitoring,” *In arXiv e-prints: 2406.07250*, 2024.

[2] 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 *Proceedings of the Detection and Classification of Acoustic Scenes and Events Workshop (DCASE)*, Barcelona, Spain, November 2021, pp. 1–5.

[3] K. Dohi, T. Nishida, H. Purohit, R. Tanabe, T. Endo, M. Yamamoto, Y. Nikaido, and Y. Kawaguchi, “MIMII DG: Sound dataset for malfunctioning industrial machine investigation and inspection for domain generalization task,” in *Proceedings of the 7th Detection and Classification of Acoustic Scenes and Events 2022 Workshop (DCASE2022)*, Nancy, France, November 2022.

[4] N. Harada, D. Niizumi, Y. Ohishi, D. Takeuchi, and M. Yasuda, “First-shot anomaly sound detection for machine condition monitoring: A domain generalization baseline,” in *2023 31st European Signal Processing Conference (EUSIPCO)*, 2023, pp. 191–195.

[5] K. Wilkinghoff, “Design choices for learning embeddings from auxiliary tasks for domain generalization in anomalous sound detection,” in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.

[6] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, “Cbam: Convolutional block attention module,” in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 3–19.

[7] K. Wilkinghoff, “Sub-cluster adacos: Learning representations for anomalous sound detection,” in *2021 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2021, pp. 1–8.

[8] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, “Smote: synthetic minority over-sampling technique,” *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.

[9] B. K. Iwana and S. Uchida, “An empirical survey of data augmentation for time series classification with neural networks,” *Plos one*, vol. 16, no. 7, p. e0254841, 2021.

[10] K. P. Sinaga and M.-S. Yang, “Unsupervised k-means clustering algorithm,” *IEEE access*, vol. 8, pp. 80 716–80 727, 2020.

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