ANOMALOUS SOUND DETECTION BASED ON PRE-TRAINED MODELS WITH DIFFERENT AUDIO REPRESENTATIONS

Technical Report

Hyun Jun Kim, Hyeon Gyu Bae*, Min Jun Kim*, Yun Seo Lee*, Jaeheon Lee*, Changwon Lim*,*

* Department of Applied Statistics, Chung Ang University, Seoul, Korea
{hyunjun0615, amysst11, goodwill1669, lee340466, jaeheon, clim}@cau.ac.kr

ABSTRACT

DCASE 2024 challenge Task2 is about first-shot unsupervised anomalous sound detection. To solve this problem, we employed self-supervised learning with various methods to enable the model to achieve general and robust performance on diverse machine sounds with limited information. The methods include combining embeddings from pre-trained models based on different audio representations, attentive statistics pooling, and a memory bank. By applying these methods, we successfully achieved a higher score on development dataset compared to the baseline.

Index Terms— Self-supervised learning, first-shot, unsupervised anomalous sound detection

1. INTRODUCTION

The task of anomalous sound detection (ASD) involves determining whether the sounds produced by a target machine are normal or anomaly. Since its inception, DCASE Challenge Task 2 has focused on addressing the task of ASD. Each year, the task has been adapted to better reflect real-world challenges and constraints. The focus of this year's task [1] remains on the first-shot problem with key modifications from the task of last year to reflect real-world constraints more, which are as follows:

- Tuning hyperparameters using test data is often impractical due to the potential for encountering completely new machine types or insufficient test data. To address this, the development and evaluation datasets contain completely different machine types.
- Practical limitations may result in only a few machines per type, unlike previous tasks where multiple sections from different machines were available. Hence, only one section per machine type is provided in this year's task.
- In real-world scenarios, information on machine conditions or noise types may not always be available. To simulate this, additional attribute information is hidden for some machine types.

[1]Transfer learning using pre-trained models has achieved significant success in various fields, such as BERT [2] and GPT [3]. Inspired by this, we decided to leverage pre-trained models in our approach. Additionally, we hypothesized that employing pre-trained models with different audio representation methods simultaneously would enable robust learning for various machine sounds. Therefore, we devised a self-supervised learning framework that combines embeddings from models trained with waveforms and spectrograms as inputs to classify machine sounds. Furthermore, to achieve generalized performance on machines without attribute information, which is a key modification in this year's task, we divided our framework into two sub-systems. One subsystem is designed to classify 16 types of machines, while the other sub-system classifies combined classes of machine types and attribute information. We obtained the final embedding for each audio sample by taking a weighted mean of the embeddings produced by these two sub-systems. Finally, in the target domain, only a very limited number of training samples were allowed. Therefore, instead of using them for training, we introduced the concept of a memory bank. At the final stage of testing, we incorporated an additional process of comparing cosine distances with all samples from the target domain. As a result, we were able to observe a better performance on development dataset compared to the baseline [4].

2. PRE-TRAINED MODELS

In this section, we briefly introduce the pre-trained models we used for different audio representations.

2.1. Wav2Vec2.0

Wav2vec2.0 [5], developed by Facebook AI, demonstrates a significant advancement in speech recognition by leveraging self-supervised learning to utilize vast amounts of unlabeled audio data, thereby reducing dependence on labeled datasets. Utilizing a transformer-based architecture, the model captures long-range dependencies in audio sequences, and its quantization of latent speech representations into discrete units enhances speech modeling. Achieving state-of-the-art performance on benchmark datasets such as LibriSpeech [6], Wav2vec2.0 demonstrates superior accuracy and efficient fine-tuning with minimal labeled data.

2.2. AST

Audio Spectrogram Transformer (AST) [7], developed by MIT, introduces a novel approach to audio processing by applying a Transformer architecture to audio spectrograms, traditionally used in natural language processing, such as BERT. By converting

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Ministry of Science and ICT (RS-2024-00360176).

audio signals into 2-dimensional time-frequency representations, AST leverages the powerful self-attention mechanism of Transformers to capture complex temporal and spectral dependencies in audio data. This method allows for effective modeling of audio signals without relying on convolutional neural networks (CNNs). AST achieves state-of-the-art performance on various audio classification benchmarks, such as AudioSet [8], by demonstrating superior accuracy and robustness.

3. APPROACHES

3.1. Speed perturbation

Speed perturbation [9] is a data augmentation technique widely used in speech processing to improve model robustness and performance. It involves altering the speed of audio recordings without changing their pitch, creating variations of the original audio. This method generates additional training data by speeding up or slowing down the audio, effectively simulating different speaking rates. Speed perturbation helps models generalize better by exposing them to a wider range of acoustic conditions, thereby enhancing their ability to handle diverse and real-world scenarios.

3.2. Attentive statistics pooling

Attentive Statistics Pooling [10] is an advanced pooling technique used in neural network-based audio processing, particularly in speaker recognition systems. It employs an attention mechanism to assign weights to different frame-level features, emphasizing the most relevant segments of the audio for the task. This approach computes weighted mean and standard deviation, capturing both the central tendency and variability of the important features. Incorporating standard deviation is crucial as it provides information about the dispersion and variability of the features, enabling the model to differentiate between speakers more effectively. By dynamically focusing on the most informative parts of the audio signal, Attentive Statistics Pooling enhances the discriminative power and robustness of the feature representation, leading to improved performance in noisy and variable conditions.

3.3. Combine embeddings

To combine the representations from the two pre-trained models mentioned in Section 2, we utilized a Convolutional Neural Network (CNN). By feeding the outputs of these models into a CNN, we can effectively capture and integrate the intricate patterns and features from both sets of representations.

The Wav2vec2.0 model incorporates an attentive statistics pooling layer that compresses the output into a tensor of shape (batch size, 2, hidden size). Concurrently, for the AST model, the CLS token is utilized. The embeddings from both models are concatenated along the second dimension, resulting in a combined embedding. This concatenated embedding is then passed through a CNN to further compress it into a single unified embedding. This approach leverages the strengths of both Wav2vec2.0 and AST models, facilitating efficient and robust feature extraction for subsequent tasks.

3.4. Classification with pre-trained models

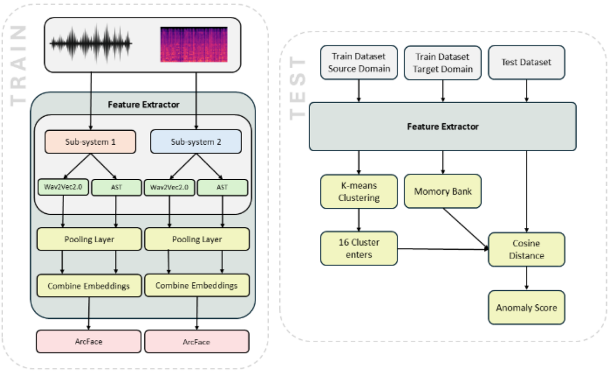

Figure 1: Overview of the self-supervised training network (left) and the testing phase based on K-means clustering and memory bank (right). During training, waveform and spectrogram of an audio sample are trained with two different pre-trained model, which are then combined through a convolutional layer. During testing, all training samples in source domain are used to determine cluster centers. Additionally, all target domain samples are stored in the memory bank, and for each test sample, the smallest cosine distance between the test sample and either the cluster centers or the memory bank entries is selected as the final anomaly score.

We employed ArcFace [11] to learn embeddings that effectively represent the input data. ArcFace is used for classification tasks, enabling the learning of embeddings with clear inter-class separability and intra-class compactness. The function is defined as follows:

$$L = -\frac{1}{N}\sum_{i=1}^{N} \frac{\log e^{s\left(\cos\left(\theta_{y_i}+m\right)\right)}}{e^{s\left(\cos\left(\theta_{y_i}+m\right)\right)} + \sum_{j=1,j\neq y_i}^{N} e^{s\cos\left(\theta_j\right)}}, \tag{1}$$

where N denotes the total number of samples, s is the scaling factor, m is the angular margin, θ_{y_i} represents the angle between the sample i and its corresponding class y_i, and θ_j represents the angle between the sample i and any other class j. This function aims to maximize the softmax value for the correct class, thereby minimizing the angle between the sample and the class center. Conversely, it drives the softmax values for the incorrect classes towards zero, thereby maximizing the angle between the sample and the centers of the other classes.

Once the model becomes saturated through the series of processes mentioned above, embeddings are extracted from the network for the training samples. These embeddings are then partitioned into 16 clusters using K-means clustering, with each cluster's center representing the embedding of a respective machine. Subsequently, for a test sample, the minimum cosine distance to any of the cluster centers is used as the anomaly score. The overall train and test process is shown in figure 1.

3.5. Memory bank

The concept of a memory bank is utilized to enhance the model's performance in scenarios with limited training samples in the target domain. A memory bank serves as a repository for storing high-quality feature representations of the target domain samples. During the final stage of testing, the model compares the test

System Index	Weight	All Hmean	Machines						
			ToyCar	ToyTrain	Bearing	Fan	Gearbox	Slider	Valve
Sub-system 1	N/A	53.79	50.82	50.74	59.55	51.60	51.25	54.63	57.63
Sub-system 2	N/A	55.25	47.32	50.98	54.22	55.64	58.85	68.70	55.77
1	0.62	56.68	48.34	53.99	56.79	57.40	56.86	65.60	58.44
2	0.64	56.60	48.46	53.99	56.20	57.37	56.50	65.54	58.27
3	0.65	56.64	48.43	54.02	55.89	57.49	56.61	65.77	58.13
4	0.67	56.66	48.52	54.15	55.50	57.53	56.68	66.06	58.13

Table 1: Results of sub-systems as well as the final submitted systems. Sub-system 1 refers to a classification system for 16 machine types, while sub-system 2 classifies combined classes of machines and their attributes. If the weight is denoted as w, then the system calculates the embedding using the formula $(1 - w) \times$ sub-system 1 + $w \times$ sub-system 2. This weighted mean allows the model to balance the contributions of both sub-systems to produce a final embedding that incorporates both machine types and their associated attributes.

samples with the stored representations in the memory bank using cosine distance. This approach allows for a more comprehensive and discriminative comparison by leveraging the entire set of target domain samples, thus mitigating the challenges posed by the scarcity of training data. The memory bank technique improves the model's ability to generalize and recognize patterns in the target domain by providing a richer context and additional reference points, ultimately leading to better performance and accuracy in real-world applications.

4. RESULTS

In Table 1, the results of each sub-system of our framework, as well as the ensemble results, can be observed. The experimental results indicate that performing separate classifications for the machine and for the combined class of machine and attribute, and then ensembling these results, improved the performance of anomalous sound detection when attribute information was limited.

5. CONCLUSIONS

In this task, we attempted various approaches to solve the ASD problem, reflecting numerous practical constraints. By combining information learned from different audio representations, we aimed to train a model that generalized well to various machine sounds. For target domains with a small sample size, we maximized the use of our network through a memory bank to achieve robust results. Additionally, by employing two sub-systems, we sought to achieve generalized performance even in situations where attribute information was limited.

6. 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." *arXiv preprint arXiv:2406.07250*, 2024.

[2] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding." *arXiv preprint arXiv:1810.04805*, 2018.

[3] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al., “Language models are few-shot learners,” *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020

[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." *2023 31st European Signal Processing Conference (EUSIPCO)*. IEEE, 2023.

[5] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations." *Advances in neural information processing systems* 33, 2020.

[6] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an asr corpus based on public domain audio books." *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2015.

[7] G., Yuan, Y. Chung, and J. Glass. "Ast: Audio spectrogram transformer." *arXiv preprint arXiv:2104.01778*, 2021.

[8] J. F. Gemmeke, D. P. W. 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." *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2017.

[9] Tom Ko, Vijayaditya Peddinti, Daniel Povey, and Sanjeev Khudanpur, "Audio augmentation for speech recognition." *Interspeech*. Vol. 2015, 2015.

[10] K. Okabe, T. Koshinaka, and K. shinoda, “Attentive statistics pooling for deep speaker embedding.” *arXiv preprint arXiv:1803.10963*, 2018.

[11] J, Deng, J, Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition." *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019.

[12] N. Harada, D. Niizumi, D. Takeuchi, Y. Ohishi, M. Yasuda, S. Saito, "ToyADMOS2: Another Dataset of Miniature-Machine Operating Sounds for Anomalous Sound Detection Under Domain Shift Conditions." *Proceedings of the Detection and Classification of Acoustic Scenes and Events Workshop (DCASE)*, 2021.

[13] 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.” *Proceedings of the 7th Detection and Classification of Acoustic Scenes and Events 2022 Workshop (DCASE 2022)*, 2022.