ANOMALOUS SOUND DETECTION BY SELF-SUPERVISED LEARNING OF VAE WITH DOMAIN SHIFT TECHNIQUES

Technical Report

Jianghan Hai[1], *Dengjian Zhou*[1], *Ruihang Liu*[2], *Yue Ivan Wu*[3]*,

[1] School of Software Engineering, Sichuan University,
[2] School of Computer Science, Sichuan University,
[3] IEEE Senior Member, School of Electronics and Information Engineering,
Sichuan University, Chengdu, Sichuan, 610065, China
ivan.wuyue@scu.edu.cn

ABSTRACT

In this paper, we introduce an advanced abnormal sound detection (ASD) system that integrates two innovative approaches to improve detection performance. Firstly, we utilize a self-supervised learning method known as Feature Exchange (FeatEx) to map raw data to meta data and obtain robust feature embeddings for enhanced discrimination of non-target sound events. Secondly, we expand a serial approach by employing an outlier-exposed feature extractor and an anomaly detector based on ResNet18 inlier modeling. The model undertake end-to-end joint optimization utilizing a variational autoencoder (VAE) feature extractor based on the intermediate inner-layer model vectors. Additionally, domain generalization techniques such as domain-invariant latent space modeling in normalized and hybrid streams are introduced to address the domain drift problem by resolving ”transfer entanglement”. To further enhance performance, our data is augmented using Mixup and Smote methods respectively. Furthermore, our system employs an ensemble learning strategy that combines anomaly scores calculated directly through normalization process with earlier models trained using optimized feature embeddings. Ultimately, our system achieves state-of-the-art performance on the DCASE 2024 ASD dataset for two machine types through weighted anomaly scores on the development set. Through the integration of self-supervised learning and domain generalization techniques, our ASD system not only reduces reliance on manual annotation but also enhances adaptability and robustness across different sound environments. This research outcome offers a novel perspective and solution for abnormal sound detection field.

Index Terms— anomaly sound detection, domain shift, DCASE Challenge

1. INTRODUCTION

Upon the occurrence of a risk or hazard, there is typically a shared attribute in the form of warning acoustic events. However, within industrial production, these acoustic events, such as abnormal noise, exhibit complexity in their origins and possess strong characteristics of suddenness and concealment. This results in diverse types and significant variations in the sound source characteristics of these events within complex spatial environments.

The DCASE 2024 Challenge 2[1] builds upon the follow-up tasks of Task 2 from DCASE 2020 to DCASE 2023. In comparison to previous iterations, the new task incorporates additional attribute information aimed at enhancing detection performance. However, it is important to note that such attribute information may not always be readily available. Consequently, the system must demonstrate robust performance under conditions where attribute information is both present and absent.

Given the nature of this year's data[2], when encountering a completely novel type of machine, the available data may not be sufficient for hyperparameter adjustment. To address this challenge, we have implemented data augmentation techniques for preprocessing and analysis. The summary of DCASE2023 challenge also recognizes the reasonableness and effectiveness of these processing methods. Additionally, in situations where only partial data is accessible for each machine type, we believe that the outlier exposure method offers unique capabilities. Therefore, we have retained some content for processing and utilized a classification model-based approach to compensate for missing information. Finally, in cases where additional attribute information is unavailable for certain machine types, after experimental verification we find setting these vectors to zero more appropriate as knowledge accuracy and bias cannot be arbitrarily transferred or supplemented.

The article begins with a comprehensive introduction to the DCASE task in the opening paragraph, followed by a detailed description of the enhancement processing steps employed, and an outline of the entire model's steps in Chapter 3. The comparison results of the model's performance with the baseline system[3] are presented in a table in Chapter 4. Finally, Chapter 5 provides a summary of the entire article.

2. THE PREPROCESSING STEP FOR THE WHOLE DETECTION CASCADE SYSTEM

2.1. Mixup

Mixup was evaluated on various standard image classification benchmarks, showing significant improvements in accuracy and robustness compared to traditional augmentation techniques. The technique also demonstrated enhanced performance in adversarial robustness and calibration of neural networks.

Synthetic data can be utilized to accurately model the distribution of normal data and enhance the robustness of the detection

*This work is supported by grant #62271333 from the National Natural Science Foundation of China.

model. By training on convex combinations of examples, Mixup forces the model to predict intermediate representations, leading to smoother and more generalizable decision boundaries. It reduces the risk of overfitting and improves model performance on unseen data.

2.2. Smote

Imbalanced datasets, where some classes are underrepresented, pose significant challenges for machine learning classifiers. Traditional methods to address this issue often involve either oversampling the minority class (by duplicating minority class examples) or under-sampling the majority class (by removing majority class examples). Both methods have drawbacks, such as overfitting and loss of valuable information.

Because the number of samples in the datasets DCASE2024 is imbalanced across domains and attributes, compensating for these class imbalances can improve the detection performance. SMOTE (Synthetic Minority Over-sampling Technique) is a popular method used for handling imbalanced datasets, especially in classification tasks. It generates synthetic samples for the minority class to balance the class distribution.

SMOTE offers a novel approach to over-sampling by generating synthetic examples for the minority class rather than simply duplicating existing ones. This is achieved by interpolating between existing minority class examples. The synthetic samples are generated by selecting two or more similar instances and introducing new samples along the line segments connecting these instances. This process leads to a more generalized decision boundary and mitigates overfitting.

3. PROPOSED METHOD

Our final model is composed of the integration of two major models, and the specific structure of these two major models are as follows:

3.1. Classification-Based Model

We use Wilkinghoff[4] as one of our backbone networks, which uses spectra obtained from Fourier transform and Mel spectrograms as features. These features are input into a dual-branch network for encoding to extract embeddings. Subsequently, the two extracted embeddings are concatenated to obtain a joint embedding. The model encodes metadata information such as machine type, ID, and attributes to obtain one-hot vectors as classification labels, and it is trained using the SCadaCos [5] loss function. Afterwards, the distance between the sample embedding vector and the centroids of the clusters formed by k-means clustering in the source domain, as well as the cosine distance between the embedding vector and all samples in the target domain, are computed separately. The minimum of these distances is taken as the anomaly score.

In addition to using FFT spectrum and Mel spectrum as input features, to further increase the available features, we have added a third branch to the classification network, using the phase spectrum of short-term Fourier transform as the third type of feature input for embedding extraction. The structure of the third branch network used is the same as the branch network structure for processing the Mel spectrogram. In addition, we use Adamax as the optimizer and replace the relu activation function in the backbone with prelu [6] for all of network's layers.

3.2. OEVAE

Building on the success of the outlier exposure (OE) [7] and inlier model (IM), we have extended it by jointly optimizing the feature extractor and the Variational Autoencoder (VAE) [8]. Figure 1 provides an overview of the model. In the first step, we use the following L_{OEVAE} joint optimization network.

$$L_{OEVAE} = L_{machine} + \lambda_{vae}(L_{vae1} + L_{vae2}) \tag{1}$$

$$\begin{aligned} L_{machine} = -\frac{1}{M}\sum_{i=1}^{M}\{t_i log(\phi(g_{machine}(f_{FE}(\mathbf{X_i})))) \\ + (t_i - 1)log(\phi(g_{machine}(f_{FE}(\mathbf{X_i}))))\} \end{aligned} \tag{2}$$

$$\begin{aligned} L_{vae1} = \frac{1}{M}\sum_{i=1}^{M}(KL(N(\mu^i_{\mathbf{z}_d}, (\delta^i_{\mathbf{z}_d})^2), N(kv, 1)) \\ + KL(N(\mu^i_{\mathbf{z}_c}, (\delta^i_{\mathbf{z}_c})^2), N(0, 1))) \end{aligned} \tag{3}$$

$$L_{vae2} = \frac{1}{M}\sum_{i=1}^{M}||f_{FE}(\mathbf{X_i}) - \hat{f}_{FE}(\mathbf{X_i})||^2 \tag{4}$$

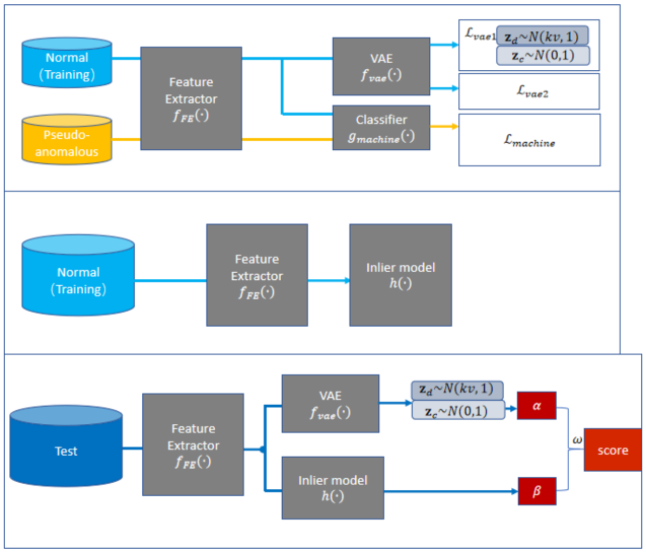
Normal (Training)
Pseudo-anomalous
Feature Extractor $f_{FE}(\cdot)$
VAE $f_{vae}(\cdot)$
Classifier $g_{machine}(\cdot)$
$\mathcal{L}_{vae1}$
$\mathbf{z}_d \sim N(kv, 1)$
$\mathbf{z}_c \sim N(0,1)$
$\mathcal{L}_{vae2}$
$\mathcal{L}_{machine}$
Normal (Training)
Feature Extractor $f_{FE}(\cdot)$
Inlier model $h(\cdot)$
Test
Feature Extractor $f_{FE}(\cdot)$
VAE $f_{vae}(\cdot)$
Inlier model $h(\cdot)$
$\mathbf{z}_d \sim N(kv, 1)$
$\mathbf{z}_c \sim N(0,1)$
α
β
ω
score

Figure 1: The structure of the OEVAE model.

The classification loss $L_{machine}$ involves both the target machine type and other machine types, where $\mathbf{X}_i$ represents the extracted Mel-spectrogram, $f_{FE}(\mathbf{X_i})$ is the output of the feature extractor, M is the number of samples, t_i is 1 indicates the target machine type, and t_i is 0 indicates other machine types (pseudo-anomaly data), and ϕ is the non-linear activation function. L_{vae1} and L_{vae2} represent the VAE loss and the KL respective the Kullback-Leibler divergence, which measures the difference between two distributions. To mitigate domain shift in the VAE for ASD, we use domain-invariant latent space modeling [9]. In this framework, some latent variables $\mathbf{z}_d$ are constrained to follow $N(kv, 1)$, while other latent variables $\mathbf{z}_c$ are constrained to follow $N(0, 1)$, where k is a hyperparameter and v represent physical parameters causing domain shift (e.g., operational speed).The segmentation of $\mathbf{z}_d$ is to make $\mathbf{z}_c$ invariant to physical parameters. The domain-shift-invariant anomaly score is calculated as

$\alpha = -logp_{\mathbf{z}_c}(\mathbf{z}_c)$, where $p_{\mathbf{z}_c}$ is constrained to follow a standard normal distribution $N(0,1)$. We introduce this domain generalization technique by splitting the VAE's latent variable channels into $\mathbf{z}_c$ and $\mathbf{z}_d$. $\hat{f}_{FE}(\mathbf{X_i})$ is the embedding reconstructed by the Variational Auto Encoder, and L_{vae2} is the reconstruction loss of the Auto Encoder. We aim to use L_{OEVAE} for end-to-end training so that the feature extractor can extract embeddings more suitable for the Variational Auto Encoder, and differentiate pseudo-anomaly data from normal data, thereby forming a better embedding space.

In the second step, we utilize the trained feature extractor to extract high-quality embeddings of normal data and train a k-nearest neighbors algorithm (KNN) as the anomaly detector $h(\cdot)$.The anomaly score β_i is calculated as follows:

$$\beta_i = h(f_{FE}(\mathbf{X_i})) \tag{5}$$

During the testing phase, for each sample in the test set, we first extract high-quality embeddings using $f_{FE}(\cdot)$. Subsequently, we compute anomaly scores α based on the negative log-likelihood derived from the latent variables output by the VAE. Simultaneously, we calculate anomaly scores β using $h(\cdot)$ according to Eq. 5. Finally, we integrate these anomaly scores using a weighting factor ω.

4. EXPERIMENTAL RESULTS

4.1. System

We developed an OEVAE model. For the target machine type with attributes, we randomly select an attribute parameter to train the model, for the target machine type without attributes, we calculate the average power of each audio sample as the attribute to train the model.We integrate the OEVAE model with the Classification-Based model and use the official Auto Encoder-based baseline system, including the selective Mahalanobis method as the baseline.

4.2. Experimental setups

We conducted an experimental evaluation using the DCASE 2024 Task 2 Challenge development sets (ToyADMOS2 [2], MIMII DG [10]). The development sets included seven machine types: bearing, fan, gearbox, valve, slider, ToyCar, and ToyTrain. The training data had 1,000 samples of normal data for each machine type, of which 990 samples are in the source domain and ten samples are in the target domain. The test data had 50 samples of normal and anomalous data for each machine type and each domain. Each recording was a 10 or 12-second single channel segment sampled at 16 kHz. During the training process, we use both the Smote and Mixup augmented data along with the original training set. For pseudo-anomaly data, we use the training data from non-target machine types and the additional training dataset from the DCASE 2024 Task 2 challenge.

The amplitude of the audio input sequence was standardized to have a mean of 0.0 and a variance of 1.0.Due to the variable lengths of audio samples, each audio signal is divided into 8 segments, each lasting 2 seconds. Mel spectrograms are extracted using a window size of 64ms and a hop length of 16ms, covering 224 mel bins spanning frequencies from 50 Hz to 7800 Hz, which serve as input for $f_{FE}(\cdot)$. ResNet-18 [11] is employed as $f_{FE}(\cdot)$, $g_{machine}(\cdot)$ is a fully connected network used for binary classification. The entire network is trained for 20 epochs with a fixed learning rate of 0.001 using the Adam optimizer [12], and a batch size of 128.During the random selection of pseudo-anomaly data, we ensure an equal number of normal and pseudo-anomaly data samples, maintaining a 1:1 ratio (denoted as t). In Eq. 1 is set to 10^{-7}. The VAE network is trained exclusively on normal samples that do not contain pseudo-anomalies, modeling domain-invariant latent spaces where v represents the attribute value of the target machine type. We utilize advantageous parameters v for data generated using Mixup and Smote. The hyperparameter k is set to 5 divided by the minimum distance parameter. We use 8 out of 64 latent variable channels as $\mathbf{z}_d$.

In the second step, we also utilize the data generated by Smote and Mixup along with the original training set for training. An anomaly score is computed for each two-second segment of audio, and these scores are averaged over the eight segments to derive the anomaly score for each audio sample.For the inner model, we employ KNN with the hyperparameter being the number of neighbors, chosen from 1, 2 and 4. KNN calculates the anomaly score by averaging the distances to the nearest k points based on Euclidean distance.

During the testing process, we similarly divide segments of the test set into eight equal two-second segments. Embeddings are extracted using the feature extractor $f_{FE}(\cdot)$, which are then input into VAE to obtain anomaly scores. Simultaneously, these embeddings are input into the inner model $h(\cdot)$ to obtain anomaly scores β, The weights are set to ω, and the final anomaly score is computed according to Eq. 6.

$$score = \omega * \alpha + (1-\omega) * \beta \tag{6}$$

4.3. Results

Table 1 shows the AUC and pAUC performance of the source domain and the target domain. Compared with the baselines, our model achieved the best results in two machine types on the development set. It can be observed that the OEVAES system exhibits better performance primarily in the target domain, with relatively minor performance improvements in the source domain. This indicates that domain generalization techniques significantly enhance detection performance in the target domain.

5. CONCLUSION

In this work, applying Classification-Based and OEVAE models to ASD was investigated. To this end, mixup and Smote were reviewed. We employed several techniques to address domain adaptation and event sparsity issues. On the development set, we observed two machine types outperforming the baseline. Moreover, our results indicate that jointly optimizing multiple features in parallel can construct a better feature space for the ASD real extractor. Future work includes a detailed analysis of our joint optimization framework and its enhancements

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,” *In arXiv e-prints: 2406.07250*, 2024.

Table 1: Results of our models

Machine Types	Metrics	Baseline	Baseline(MHLAE)	OEVAES
bearing	AUC(source)	65.92	65.16	63.80
	AUC(target)	55.75	55.28	69.34
	pAUC	50.42	51.33	60.36
fan	AUC(source)	80.19	87.10	70.02
	AUC(target)	36.18	45.98	33.00
	pAUC	59.04	59.33	48.95
gearbox	AUC(source)	60.31	71.88	61.12
	AUC(target)	60.78	70.78	63.92
	pAUC	53.22	54.34	50.47
slider	AUC(source)	70.31	84.02	65.00
	AUC(target)	48.77	73.29	61.36
	pAUC	56.37	54.72	48.89
valve	AUC(source)	55.35	56.31	50.04
	AUC(target)	50.69	51.40	52.54
	pAUC	51.18	51.08	50.94
ToyCar	AUC(source)	70.10	74.53	47.66
	AUC(target)	46.89	43.42	51.62
	pAUC	52.47	49.18	51.21
ToyTrain	AUC(source)	57.93	55.98	67.32
	AUC(target)	57.02	43.42	42.24
	pAUC	48.57	48.13	48.78

[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] 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.

[4] K. Wilkinghoff, “Self-supervised learning for anomalous sound detection,” in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 276–280.

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

[6] Y. Hu, Y. Liu, S. Lv, M. Xing, S. Zhang, Y. Fu, J. Wu, B. Zhang, and L. Xie, “Dccrn: Deep complex convolution recurrent network for phase-aware speech enhancement,” *arXiv preprint arXiv:2008.00264*, 2020.

[7] D. Hendrycks, M. Mazeika, and T. Dietterich, “Deep anomaly detection with outlier exposure,” *arXiv preprint arXiv:1812.04606*, 2018.

[8] D. P. Kingma and M. Welling, “Auto-encoding variational bayes,” *arXiv preprint arXiv:1312.6114*, 2013.

[9] K. Dohi, T. Endo, and Y. Kawaguchi, “Disentangling physical parameters for anomalous sound detection under domain shifts,” in *2022 30th European Signal Processing Conference (EUSIPCO)*. IEEE, 2022, pp. 279–283.

[10] 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,” *arXiv preprint arXiv:2205.13879*, 2022.

[11] 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.

[12] I. Loshchilov and F. Hutter, “Decoupled weight decay regularization,” *arXiv preprint arXiv:1711.05101*, 2017.