International Journal For Advanced Research In Science & Technology
A peer reviewed international journal www.ijarst.in
ISSN: 2457-0362

DETECTING ABNORMAL SOUNDS VIA SELF-SUPERVISED LEARNING

[1]Mrs. K. Jyothi, [2] Sheik Vanisha, [3] Nakka Vedhasudha, [4] Yedida Hema Priya,

[1]Assistant Professor, Dept. of CSE, Rajamahendri Institute of Engineering & Technology, Bhoopalapatnam, Near Pidimgoyyi, Rajahmundry, E.G. Dist. A.P. 533107.

[2,3,4] Students, Dept. of CSE, Rajamahendri Institute of Engineering & Technology, Bhoopalapatnam, Near Pidimgoyyi, Rajahmundry, E.G. Dist. A.P. 533107.

ABSTRACT

It is common practice to train state-of-the-art anomalous sound detection (ASD) systems by learning an embedding space using an auxiliary classification task. This allows the system to learn noise-tolerant embeddings that disregard off-target sound occurrences; however, it does need the usage of explicitly annotated meta information for class labels. But the embeddings become less useful and the ASD performance suffers as the classification challenge gets easier. Using self-supervised learning (SSL) is one way to fix this. A straightforward SSL method for ASD, feature exchange (FeatEx), is suggested in this paper. Also included are comparisons and combinations with current SSL techniques, as well as FeatEx itself. The primary outcome is a new benchmark performance for the DCASE2023 ASD dataset that far surpasses all previous reported findings on this dataset. Machine listening, self-supervised learning, domain generalization, and anomalous sound detection are some of the index terms.

1. INTRODUCTION

Unlike supervised learning, self-supervised learning (SSL) [1] can function without class labels that have been explicitly annotated. Rather, data is enhanced in many strategically selected ways, each of which defines a new artificially constructed class, and a model is trained to distinguish between these classes. Predicting the newly added artificial classes accurately requires the model to comprehend the data structure, which is the underlying assumption. SSL is an unsupervised learning method that has been used to learn representations of speech or general purpose audio from big datasets of unlabeled data [2, 3]. Anomaly sound detection (ASD) has also utilized SSL:
In [5], new classes are generated by combining pitch-shifting and time-stretching. As pseudo-anomalous classes, [6] creates target sounds using mixup [7] and then employs linear combinations of those sounds. One SSL method that combines first- and second-order statistics of time-frequency representations to generate new classes is Statistics Exchange (StatEx) [8]. To pre-train autoencoders, a variant of variance-invariance-covariance regularization, known as domain generalization mixup, is employed in [9]. It should be noted that SSL is used by certain ASD works to refer to supervised learning of embeddings with supplementary classification tasks [11, 12]. We shall use two distinct terminologies here because SSL does not necessitate any human annotations

while classification jobs do. As the model learns to pay close attention to the target sounds, class labels alone are proven to be highly effective for detecting anomalous sounds in noisy environments [13]. In an acoustic setting without class labels, the presence of numerous non-target sounds is far more noticeable than the little variations of the target sounds that must be recognized to identify unusual sounds. The problem is that the discriminating of classes yields fewer useful embeddings when the available classes are less comparable to one another. Using SSL is a good strategy in these situations as well, as it increases the amount of data gathered and should, in theory, boost ASD performance. The objective of this project is to explore various methods of implementing SSL for ASD. The following contributions are being made: We begin by taking a look at mixup and StatEx, two of the current SSL methods for ASD. The second proposal is a hybrid of the first two SSL methods plus a new one for ASD called feature exchange (FeatEx). The suggested method outperforms a baseline system that does not use SSL, according to experimental assessments performed on the DCASE2022 and DCASE2023 ASD datasets. Consequently, using the DCASE2023 ASD dataset1, a new state-of-the-art performance is achieved, surpassing all previously reported ASD findings by a wide margin.

2. STATE-OF-THE-ART BASELINE SYSTEM

The ASD system from our previous work [15] serves as a baseline system for this work. It is trained in a supervised manner utilizing an auxiliary classification job. Our objective is to enhance the system's performance by utilizing SSL techniques. In the DCASE2023 Challenge, this system came in at #4 [14], and the winner team [16] improved upon it by including an attention mechanism into the embedding model. Thus, using the ASD system as a baseline demonstrates that it is capable of being regarded as state-of-the-art. Figure 1 provides a high-level picture of the basic system. Using all accessible meta information, such as machine kinds, machine IDs, and attribute information, the system learns discriminative embeddings. For this purpose, a single embedding is obtained by merging the outputs of two convolutional sub-networks. To guarantee the best potential frequency resolution, one subnetwork uses the complete magnitude frequency spectrum as an input representation. In order to differentiate the two subnetworks' input representations, one employs magnitude spectrograms and subtracts the temporal mean to eliminate static frequency information. Using a batch size of 64 and reducing the angular margin loss sub-cluster AdaCos [17] with 16 sub-clusters, the neural network is trained for 10 epochs. The ASD performance is enhanced by not using bias terms in any network layer and by randomly initializing the cluster centers and not adjusting them during training. We don't employ any additional data augmentation techniques other than mixup. In the background, k-means is used to refine the embeddings that were produced from the source domain's normal training samples. This process requires a large number of training samples.

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal www.ijarst.in
ISSN: 2457-0362

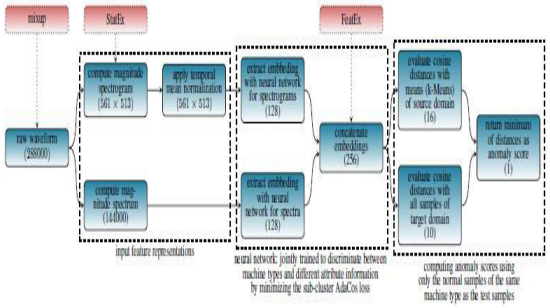

Fig. 1. Structure of the baseline system (blue boxes). SSL approaches are colored in red. Representation size in each step is given in brackets. This figure is adapted from [14] and originally adapted from [15].

We have availability. Anomaly scores are determined by finding the smallest cosine distance between these means and all samples of the target domain. This distance is calculated for domains that are different from the target domain and for which there are few training samples available. Refer to [15] for further information on this baseline system.

3.SELF-SUPERVISED LEARNING APPROACHES

Specifically, this part examines mixup [7] and StatEx [8], two SSL methods for ASD system training. In addition, a third method, FeatEx, is presented and explained in depth. Lastly, a unified SSL strategy is introduced, utilizing all three previously mentioned methods in tandem. 3.1. Collage Mixup[7] is a popular data augmentation method for ASD[14,16, 18–20] because it employs linear interpolations between two training samples and the related category labels. Mixup is applied by setting two randomly selected training samples x1, x2, together with their matching categorical class labels y1, y2 ∈ [0, 1]Nclasses, where Nclasses is the number of classes.

$$x_{\text{new}} = \lambda x_1 + (1-\lambda)x_2$$
$$y_{\text{new}} = \lambda y_1 + (1-\lambda)y_2$$

working with a random mixing coefficient λ that is within the range of 0 to 1. While mixup doesn't add any new classes to the data, it does extend the supervised training objective to include predicting the mixing coefficient along with the original classes. This makes it a type of SSL that requires class labels, even though mixup itself doesn't add any new classes. Using a method similar to the one suggested in [5], authors in [8, 14] and elsewhere created new pseudo-anomalous classes by assigning mixed-up samples the same treatment as non-mixed samples inside other classes. We found that this method actually hurt performance on some machine types, rather than helping ASD when compared to consistently applying mixup. Another option is to use mixup as a fully self-supervised method, in which case you just need to forecast the mixing coefficient and disregard the class labels. To train the system to focus on the machine sounds of interest while ignoring background noise and other nontarget events, it is very advantageous to use all available meta information for classification when dealing with noisy audio data [13]. In this study, we trained the baseline system with a 100% chance of using mixup and a 50% chance of using any other SSL technique. 3.2. Data sharing In StatEx [8], two training samples x1, x2 are compared along the frequency or time dimension using first- and second-order statistics of the time-frequency representations. The goal is to generate new classes of pseudo-anomalies artificially.

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal www.ijarst.in
ISSN: 2457-0362

From a mathematical perspective, this is equivalent to creating a new sample xnew ∈ RT×F by adjusting

$$x_{\text{new}} = \frac{x_1 - \mu_1}{\sigma_1}\sigma_2 + \mu_2$$

in which two randomly selected training samples' time-frequency representations are denoted by x1, x2 ∈ RT×F. In the time or frequency dimension, μ1, μ2 represent the first-order statistics of these samples, while σ1, σ2 stand for the second-order statistics in the same dimension. A new class is formed for every conceivable combination of classes, which adds N2 classes, a quadratic term, to the initial number of classes Nclasses ∈ N. A StatEx variation with the following changes was utilized in this work: When calculating statistics, we always utilize the full frequency band and all time steps for simplicity's sake, even though the original specification included subbands [8]. Third, using the data from the other sample x2 and the original sample x1, we train the model to predict their classes. The labels y1 and y2 from the category class are joined together to accomplish this:

$$y_{\text{new}} = (\mathbf{0}, 0.5 \cdot y_1, 0.5 \cdot y_2) \in [0,1]^{3N_{\text{classes}}}$$

inside the set of N classes, where 0 is a value between 0 and 1, inclusive. Therefore, the number of parameters does not explode, even though the number of classes is tripled. This is because the number of cluster centers rises proportionately with the number of classes. In addition, this makes it easy to combine with other data augmentation methods, like mixup, that assign many classes to each sample. The results of the ASD are improved by subtracting the temporal mean from the spectrograms, as demonstrated in [15]. This paper's version, then, solely makes use of temporal StatEx, which we applied to the frequency axis. In addition, the basic system makes use of two feature branches. Therefore, the spectrogram representations have only been subjected to temporal StatEx. Across this project, we utilized mixup and applied StatEx during training with a 50% probability. The new label of the training sample xnew = x1 is set to in the event that StatEx is not used.

$$y_{\text{new}} = (y_1, \mathbf{0}, \mathbf{0}) \in [0,1]^{3N_{\text{classes}}}.$$

Furthermore, for the newly added classes, we utilized trainable cluster centers. The results of the ablation trials discussed in paragraph 4.3 further support these specific decisions.

Table 1. Harmonic means of AUCs and pAUCs taken over all machine IDs obtained when using different SSL approaches. Highest AUCs and pAUCs in each row are highlighted in bold letters. Arithmetic mean and standard deviation over five independent trials are shown.

			baseline [15]		Stadfix [8] variant		Feadfix		regular and Stadfix [8] variant		regular and Feadfix		proposed approach	
dataset	split	domain	AUC	pAUC	AUC	pAUC	AUC	pAUC	AUC	pAUC	AUC	pAUC	AUC	pAUC
DCASE2022	dev	source	84.2 ± 0.8%	76.5 ± 0.9%	80.5 ± 1.8%	69.5 ± 1.9%	82.1 ± 0.9%	72.8 ± 0.8%	85.2 ± 0.9%	77.5 ± 1.2%	85.1 ± 0.9%	76.3 ± 1.8%	**86.0 ± 0.9%**	**77.6 ± 0.8%**
DCASE2022	dev	target	76.5 ± 0.9%	62.5 ± 0.9%	75.3 ± 1.7%	60.0 ± 0.9%	77.2 ± 1.0%	62.8 ± 0.8%	**78.0 ± 0.9%**	63.2 ± 1.8%	77.9 ± 0.9%	62.7 ± 0.8%	78.2 ± 0.7%	**64.4 ± 1.1%**
DCASE2022	dev	mixed	81.4 ± 0.7%	66.6 ± 0.9%	78.4 ± 1.5%	62.4 ± 1.2%	78.5 ± 0.6%	65.2 ± 0.6%	82.2 ± 0.6%	67.0 ± 1.0%	81.6 ± 0.7%	67.0 ± 0.9%	**82.5 ± 0.8%**	**68.2 ± 1.1%**
DCASE2022	eval	source	76.8 ± 0.8%	65.8 ± 0.2%	74.2 ± 0.6%	61.6 ± 1.4%	76.3 ± 0.9%	64.5 ± 1.2%	76.9 ± 0.4%	65.8 ± 0.9%	**78.1 ± 0.4%**	**67.0 ± 1.1%**	77.7 ± 0.8%	**67.0 ± 0.5%**
DCASE2022	eval	target	69.8 ± 0.5%	59.7 ± 1.1%	70.6 ± 0.5%	59.0 ± 0.7%	**72.3 ± 0.6%**	61.0 ± 0.7%	71.2 ± 0.2%	60.3 ± 0.7%	72.2 ± 0.4%	**61.3 ± 0.5%**	71.6 ± 1.0%	61.2 ± 0.9%
DCASE2022	eval	mixed	73.4 ± 0.5%	59.8 ± 0.8%	72.2 ± 0.2%	58.2 ± 0.7%	73.9 ± 0.5%	60.0 ± 0.9%	73.9 ± 0.3%	59.9 ± 0.6%	**74.0 ± 0.4%**	**61.5 ± 0.6%**	74.2 ± 0.2%	61.2 ± 0.3%
DCASE2023	dev	source	69.8 ± 1.8%	60.9 ± 0.9%	67.8 ± 1.5%	59.2 ± 0.8%	68.4 ± 1.0%	60.2 ± 0.5%	70.3 ± 1.8%	62.0 ± 1.6%	**72.9 ± 2.0%**	**65.0 ± 1.3%**	71.2 ± 1.0%	62.7 ± 1.3%
DCASE2023	dev	target	72.3 ± 1.8%	55.6 ± 0.9%	69.7 ± 1.7%	54.7 ± 1.1%	74.4 ± 0.7%	**57.6 ± 1.0%**	72.2 ± 1.4%	56.2 ± 1.2%	**75.7 ± 0.8%**	57.0 ± 1.6%	75.0 ± 1.5%	56.1 ± 1.4%
DCASE2023	dev	mixed	71.3 ± 0.6%	58.1 ± 0.8%	69.0 ± 1.2%	55.7 ± 1.0%	71.7 ± 0.4%	57.5 ± 0.7%	71.2 ± 0.7%	57.0 ± 1.4%	**74.4 ± 1.0%**	**58.0 ± 1.4%**	73.1 ± 0.9%	57.9 ± 0.6%
DCASE2023	eval	source	72.5 ± 0.8%	62.4 ± 1.2%	70.0 ± 1.2%	59.7 ± 0.9%	69.3 ± 2.0%	59.3 ± 1.2%	72.4 ± 2.4%	62.4 ± 1.3%	**75.9 ± 1.0%**	62.9 ± 1.2%	75.5 ± 0.8%	**64.5 ± 0.6%**
DCASE2023	eval	target	63.1 ± 2.6%	57.5 ± 0.8%	66.7 ± 1.8%	58.4 ± 0.7%	**69.1 ± 1.3%**	59.0 ± 1.3%	65.9 ± 1.9%	59.0 ± 1.4%	66.5 ± 1.9%	58.3 ± 1.0%	66.7 ± 2.2%	**59.3 ± 0.7%**
DCASE2023	eval	mixed	67.9 ± 1.0%	58.8 ± 0.8%	65.8 ± 0.6%	57.1 ± 0.8%	68.1 ± 1.2%	58.1 ± 0.9%	69.5 ± 1.8%	60.7 ± 0.9%	71.1 ± 1.1%	60.1 ± 1.2%	**72.6 ± 0.7%**	**61.6 ± 0.5%**

3.3. Feature exchange

To train look, listen, and learn (L3) embeddings [21–23], an audio and video subnetwork is used to predict if two one-second audio segments and video frames belong together. Using these pre-trained embeddings does not improve ASD performance compared to explicitly building an embedding model, as demonstrated empirically in [24]. Compared to supervised

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal www.ijarst.in
ISSN: 2457-0362

embeddings, self-supervised embeddings like L3-embeddings seem to perform better when evaluating numerous pre-trained embeddings. This calls for the creation of an equivalent SSL method for learning embeddings from audio data alone. We can utilize a similar SSL method, which we'll name feature exchange (FeatEx), as the baseline system also has two sub-networks that employ distinct input feature representations.

Given two random training samples x1 and x2, let e1 = (e11, e21) and e2 = (e12, e22) ∈ R2D with D = 128 represent the combined embeddings of the two sub-networks, and let y1, y2 stand for the associated categorical class labels. After that, specify a new embedding and what it will be called by changing

$$e_{\text{new}} = (e_1^1, e_2^2) \in \mathbb{R}^{2D}$$
$$y_{\text{new}} = (\mathbf{0}, 0.5 \cdot y_1, 0.5 \cdot y_2) \in [0,1]^{3N_{\text{classes}}}$$

The number of original classes is represented by Nclasses, where N is an integer between zero and one. As a result, there are three times as many classes as in the StatEx variant. Applying FeatEx also requires the network to learn if the sub-networks' embeddings belong together, which leads to the collection of more information.

All of this work has made use of a mixup, a 50% chance of using FeatEx during training, and trainable cluster centers for the newly added classes.

34. Losses that are both monitored and self-supervised combined

It was demonstrated in [15] that the ASD performance that follows from training without adjusting randomly started cluster centers is superior. When we tested the SSL methods with trainable cluster centers, we discovered that they worked much better. We utilized the normal supervised loss of the baseline system with non-trainable cluster centers as an equally weighted loss to further guarantee that only the SSL loss's cluster centers pertaining to the initial classes are non-trainable. It is also possible to view this as a type of disentangled learning [25] since the classes introduced by the SSL techniques are subdividing the original classes. Since no two SSL methods are identical, we also suggest combining the conventional loss with StatEx and FeatEx into a single loss function. So, for a sample x with a categorical label y, the total loss Ltotal(x, y) equals

$$\mathcal{L}_{\text{total}}(x, y) = \mathcal{L}(x, y) + \mathcal{L}(x_{\text{new}}, y_{\text{new}})$$

xnew and ynew are defined by applying all the SSL techniques in sequential order as mentioned in the preceding sections, and L is the categorical crossentropy. The outcome is a ninefold rise in the total number of courses. Hereafter, this method is referred to as the suggested strategy.

4. EXPERIMENTAL RESULTS

4.1. Datasets

The studies performed in this study make use of the DCASE2022 [26] and DCASE2023 ASD datasets [27]. Using recordings of different types of machine noises from ToyAdmos2 [28] and MIMII-DG [29], the two datasets are created for use in semi-supervised ASD for machine condition monitoring. When it comes to training, all that's accessible is regular

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal www.ijarst.in
ISSN: 2457-0362

sounds plus some extra meta data called attribute information, which includes things like the types of machines and their parameter values. In addition, the two datasets are structured for domain generalization, so they include information from two domains: one with 1000 training samples per machine ID in the source domain, and the other with 10 samples in the target domain, which differs in some way by adjusting the target machine's parameters or the background noise. Regardless of the domain a sample belongs to, the aim is to distinguish between normal and anomalous samples.

A development set and an evaluation set are created from the two datasets, with the former including a training subset of normally distributed data and the latter including both normally distributed and abnormally distributed samples. With three unique machine identifiers in the development set and three more in the evaluation set, the DCASE2022 ASD dataset contains recordings from seven distinct machine kinds. There are fourteen distinct kinds of machines included in the DCASE2023 ASD dataset. Each machine type in the evaluation and development sets has its own unique identifier, and these sets are mutually exclusive. Because of this, we may use SSL for ASD system training as well, as the classification job is considerably simpler on the DCASE2023 dataset than on the DCASE2022. This is because learning informative embeddings by the solution of an auxiliary classification task is significantly more challenging on the DCASE2023 dataset. 4.2. A Review of SSL Methods

The first experiment compares the baseline performance achieved without extra SSL losses against that of other SSL techniques. Table 1 and the notes that follow contain the findings.

Table 2. Harmonic means of AUCs and pAUCs taken over all machine types obtained on the DCASE2023 dataset by modifying design choices of the proposed approach. Arithmetic mean and standard deviation over five independent trials are shown.

		SSL loss without class labels		non-trainable class centers		no TMN and full StatEx	
split	domain	AUC	pAUC	AUC	pAUC	AUC	pAUC
dev	source	70.8 ± 1.5%	63.2 ± 1.1%	71.5 ± 0.9%	64.8 ± 1.9%	70.9 ± 0.7%	61.0 ± 1.5%
dev	target	74.7 ± 1.5%	58.1 ± 1.6%	74.0 ± 2.0%	56.7 ± 1.0%	72.1 ± 1.4%	55.2 ± 1.0%
dev	mixed	72.3 ± 1.2%	57.9 ± 1.3%	71.6 ± 1.1%	57.7 ± 0.7%	71.3 ± 0.7%	55.6 ± 0.9%
eval	source	73.5 ± 2.4%	63.8 ± 0.6%	74.2 ± 0.7%	63.9 ± 1.3%	73.8 ± 1.3%	62.4 ± 1.5%
eval	target	62.1 ± 1.5%	57.7 ± 0.9%	58.2 ± 3.3%	57.3 ± 0.9%	66.9 ± 2.4%	58.5 ± 1.9%
eval	mixed	68.6 ± 1.2%	59.1 ± 0.7%	65.0 ± 0.9%	57.7 ± 0.6%	70.9 ± 0.8%	59.9 ± 0.8%

is feasible: To start, the suggested FeatEx loss outperforms the StatEx loss on both datasets by a wide margin. Also, for most dataset splits, using only the FeatEx loss improves performance marginally over the baseline system, but using only the StatEx loss significantly worsens performance compared to the baseline system. On the other hand, taking a combination of the normal loss and one of the SSL losses yields better results than using either loss alone, particularly on the DCASE2023 dataset. As mentioned earlier, the most probable explanation is because the DCASE2022 dataset has many more machine types, making the classification process easier. As a result, the embeddings are less informative and less sensitive to anomalies. In order to train the system to learn nontrivial mappings for each class, which leads to more informative embeddings and the ability to detect small data outliers, SSL is necessary as a regularization. Finally, it's clear that merging all SSL methods into one loss somewhat increases performance for some dataset splits but decreases

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal www.ijarst.in
ISSN: 2457-0362

performance for others. The net benefit appears to outweigh the net cost, albeit by a little margin. Studying ablation (4.3) Three ablation tests on the DCASE2023 dataset have been carried out to demonstrate that the design choices of the suggested method optimize the ASD performance. To be more specific, we tested three different approaches to SSL loss calculation: 1) ignoring class labels, 2) utilizing non-trainable class centers, or 3) avoiding TNN and adding StatEx to the temporal axis to see if speeded up performance. Table 2 shows the results compared to the original ones in Table 1, and it's clear that changing the suggested method in any of the three methods lowers ASD performance, particularly on the assessment set. This strengthens faith in the suggested SSL method's architecture. 4.4. Evaluation in relation to other system publications The last test was comparing the suggested system to the 10 best entries in the DCASE2023 Challenge. For an equitable comparison, we utilized an ensemble that was generated by retraining the system five times and averaging all anomaly scores. Figure 2 displays the outcomes. We have achieved a new state-of-the-art performance with our suggested system, which significantly surpasses all previously reported systems. Take note that the system that came in fourth place [14] in the DCASE2023 Challenge is identical to the baseline system used in this study, and that the system that came in first place [16] is a tweaked version of this baseline system.

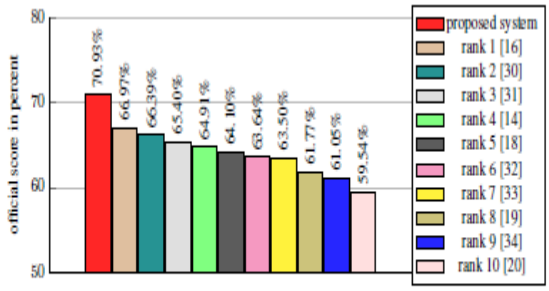

Fig. 2. Comparison between presented and ten top-performing systems of the DCASE Challenge 2023.

5. CONCLUSION

Application of SSL to ASD was explored in this study. So, we looked into mixup and StatEx and came up with a new SSL method for ASD called FeatEx. A single loss function was used to train an ASD system that was exposed to outliers, using all three techniques. Applying SSL to ASD yields excellent results, and testing on the DCASE2022 and DCASE2023 ASD datasets demonstrated that FeatEx works better than the current SSL methods. Thus, a new state-of-the-art performance was achieved on the DCASE2023 ASD dataset, far surpassing all previously published systems.

6. REFERENCES

[1] Shuo Liu, Adria Mallol-Ragolta, Emilia Parada-Cabaleiro, Kun Qian, Xin Jing, Alexander Kathan, Bin Hu, and Bj¨orn W. Schuller, “Audio self-supervised learning: A survey,” Patterns,
vol. 3, no. 12, pp. 100616, 2022.
[2] Abdelrahman Mohamed, Hung-yi Lee, Lasse Borgholt, Jakob D. Havtorn, Joakim Edin, Christian Igel, Katrin Kirchhoff,

Shang-Wen Li, Karen Livescu, Lars Maaloe, Tara N.
Sainath, and Shinji Watanabe, "Self-supervised speech representation learning: A review," IEEE J. Sel. Top. Signal Process., vol. 16, no. 6, pp. 1179–1210, 2022.
[3] Daisuke Niizumi, Daiki Takeuchi, Yasunori Ohishi, Noboru Harada, and Kunio Kashino, "BYOL for audio: Selfsupervised learning for general-purpose audio representation," in IJCNN. 2021, pp. 1–8, IEEE.
[4] Daisuke Niizumi, Daiki Takeuchi, Yasunori Ohishi, Noboru Harada, and Kunio Kashino, "BYOL for audio: Exploring pretrained general-purpose audio representations," IEEE/ACM
Trans. Audio Speech Lang. Process., vol. 31, pp. 137–151,
2023.
[5] Tadanobu Inoue, Phongtharin Vinayavekhin, Shu Morikuni, Shiqiang Wang, Tuan Hoang Trong, David Wood, Michiaki Tatsubori, and Ryuki Tachibana, "Detection of anomalous
sounds for machine condition monitoring using classification confidence," in DCASE, 2020, pp. 66–70.
[6] Jose A. Lopez, Hong Lu, Paulo Lopez-Meyer, Lama Nachman, Georg Stemmer, and Jonathan Huang, "A speaker recognition approach to anomaly detection," in DCASE, 2020, pp. 96–99.
[7] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz, "Mixup: Beyond empirical risk minimization," in ICLR, 2018.
[8] Han Chen, Yan Song, Zhu Zhuo, Yu Zhou, Yu-Hong Li, Hui Xue, and Ian McLoughlin, "An effective anomalous sound detection method based on representation learning with simulated anomalies," in ICASSP. IEEE, 2023.
[9] Ismail Nejjar, Jean Meunier-Pion, Ga¨etan Frusque, and Olga Fink, "DG-Mix: Domain generalization for anomalous sound detection based on self-supervised learning," in DCASE. 2022, Tampere University.
[10] Adrien Bardes, Jean Ponce, and Yann LeCun, "VICReg: Variance-invariance-covariance regularization for selfsupervised learning," in ICLR. 2022, OpenReview.net.
[11] Ritwik Giri, Srikanth V. Tenneti, Fangzhou Cheng, Karim Helwani, Umut Isik, and Arvindh Krishnaswamy, "Selfsupervised classification for detecting anomalous sounds," in DCASE, 2020, pp. 46–50.
[12] Kota Dohi, Takashi Endo, Harsh Purohit, Ryo Tanabe, and Yohei Kawaguchi, "Flow-based self-supervised density estimation for anomalous sound detection," in ICASSP. 2021, pp. 336–340, IEEE.
[13] Kevin Wilkinghoff and Frank Kurth, "Why do angular margin losses work well for semi-supervised anomalous sound detection?," IEEE/ACM Trans. Audio Speech Lang. Process, vol.
32, pp. 608–622, 2024.
[14] Kevin Wilkinghoff, "Fraunhofer FKIE submission for task 2: First-shot unsupervised anomalous sound detection for machine condition monitoring," Tech. Rep., DCASE2023 Challenge, 2023.
[15] Kevin Wilkinghoff, "Design choices for learning embeddings from auxiliary tasks for domain generalization in anomalous sound detection," in ICASSP. 2023, IEEE.
[16] Wang Junjie, Wang Jiajun, Chen Shengbing, Sun Yong, and Liu Mengyuan,

"Anomalous sound detection based on selfsupervised learning," Tech. Rep., DCASE2023 Challenge,
2023.

[17] Kevin Wilkinghoff, "Sub-cluster AdaCos: Learning representations for anomalous sound detection," in IJCNN. 2021, IEEE.

[18] Jia Yafei, Bai Jisheng, and Huang Siwei, "Unsupervised abnormal sound detection based on machine condition mixup," Tech. Rep., DCASE2023 Challenge, 2023.

[19] Lei Wang, Fan Chu, Yuxuan Zhou, Shuxian Wang, Zulong Yan, Shifan Xu, Qing Wu, Mingqi Cai, Jia Pan, Qing Wang, Jun Du, Tian Gao, Xin Fang, and Liang Zou, "First-shot unsupervised anomalous sound detection using attribute classification and conditional autoencoder," Tech. Rep., DCASE2023 Challenge, 2023.

[20] Wang JiaJun, "Self-supervised representation learning for firstshot unsupervised anomalous sound detection," Tech. Rep., DCASE2023 Challenge, June 2023.

[21] Relja Arandjelovic and Andrew Zisserman, "Look, listen and learn," in ICCV. 2017, pp. 609–617, IEEE Computer Society.

[22] Relja Arandjelovic and Andrew Zisserman, "Objects that sound," in ECCV. 2018, vol. 11205 of Lecture Notes in Computer Science, pp. 451–466, Springer.

[23] Aurora Cramer, Ho-Hsiang Wu, Justin Salamon, and Juan Pablo Bello, "Look, listen, and learn more: Design choices for deep audio embeddings," in ICASSP. 2019, pp. 3852–3856, IEEE.

[24] KevinWilkinghoff and Fabian Fritz, "On using pre-trained embeddings for detecting anomalous sounds with limited training data," in EUSIPCO. 2023, pp. 186–190, IEEE.

[25] Agrawal, K. K. ., P. . Sharma, G. . Kaur, S. . Keswani, R. . Rambabu, S. K. . Behra, K. . Tolani, and N. S. . Bhati. "Deep Learning-Enabled Image Segmentation for Precise Retinopathy Diagnosis". International Journal of Intelligent Systems and Applications in Engineering, vol. 12, no. 12s, Jan. 2024, pp. 567-74, https://ijisae.org/index.php/IJISAE/article/view/4541.

[26] Samota, H. ., Sharma, S. ., Khan, H. ., Malathy, M. ., Singh, G. ., Surjeet, S. and Rambabu, R. . (2024) "A Novel Approach to Predicting Personality Behaviour from Social Media Data Using Deep Learning", International Journal of Intelligent Systems and Applications in Engineering, 12(15s), pp. 539–547. Available at: https://ijisae.org/index.php/IJISAE/article/view/4788

[27] Kota Dohi, Keisuke Imoto, Noboru Harada, Daisuke Niizumi, Yuma Koizumi, Tomoya Nishida, Harsh Purohit, Ryo Tanabe, Takashi Endo, and Yohei Kawaguchi, "Description and discussion on DCASE 2023 challenge task 2: First-shot unsupervised anomalous sound detection for machine condition monitoring," in DCASE. 2023, pp. 31–35, Tampere University.

[28] Noboru Harada, Daisuke Niizumi, Daiki Takeuchi, Yasunori Ohishi, Masahiro Yasuda, and Shoichiro Saito, "Toy-ADMOS2: Another dataset of miniature-machine operating sounds for anomalous

sound detection under domain shift conditions," in DCASE, 2021, pp. 1–5.

[29] Kota Dohi, Tomoya Nishida, Harsh Purohit, Ryo Tanabe, Takashi Endo, Masaaki Yamamoto, Yuki Nikaido, and Yohei Kawaguchi, "MIMII DG: sound dataset for malfunctioning industrial machine investigation and inspection for domain generalization task," in DCASE. 2022, pp. 26–30, Tampere University.

[30] Zhiqiang Lv, Bing Han, Zhengyang Chen, Yanmin Qian, Jiawei Ding, and Jia Liu, "Unsupervised anomalous detection based on unsupervised pretrained models," Tech. Rep.,
DCASE2023 Challenge, 2023.

[31] Anbai Jiang, Qijun Hou, Jia Liu, Pingyi Fan, Jitao Ma, Cheng Lu, Yuanzhi Zhai, Yufeng Deng, andWei-Qiang Zhang, "Thuee system for first-shot unsupervised anomalous sound
detection for machine condition monitoring," Tech. Rep., DCASE2023 Challenge, 2023.

[32] Yifan Zhou and Yanhua Long, "Attribute classifier with imbalance compensation for anomalous sound detection," Tech. Rep., DCASE2023 Challenge, 2023.

[33] Jiantong Tian, Hejing Zhang, Qiaoxi Zhu, Feiyang Xiao, Haohe Liu, Xinhao Mei, Youde Liu, Wenwu Wang, and Jian Guan, "First-shot anomalous sound detection with gmm clustering
and finetuned attribute classification using audio pretrained model," Tech. Rep., DCASE2023 Challenge, 2023.

[34] Noboru Harada, Daisuke Niizumi, Daiki Takeuchi, Yasunori Ohishi, and Masahiro Yasuda, "First-shot anomaly detection for machine condition monitoring: A domain generalization
baseline," in EUSIPCO. 2023, pp. 191–195, IEEE.