Anomalous Sound Detection Using a Binary Classification Model and Class Centroids

Ibuki Kuroyanagi*, Tomoki Hayashi*†, Kazuya Takeda*, Tomoki Toda*,
*Nagoya University, Nagoya, Japan
†Human Dataware Lab. Co., Ltd., Nagoya, Japan
E-mail: *{kuroyanagi.ibuki, hayashi.tomoki}@g.sp.m.is.nagoya-u.ac.jp,
†hayashi@hdwlab.co.jp, *takeda@i.nagoya-u.ac.jp, *tomoki@icts.nagoya-u.ac.jp

***Abstract*—An anomalous sound detection system to detect unknown anomalous sounds usually needs to be built using only normal sound data. Moreover, it is desirable to improve the system by effectively using a small amount of anomalous sound data, which will be accumulated through the system's operation. As one of the methods to meet these requirements, we focus on a binary classification model that is developed by using not only normal data but also outlier data in the other domains as pseudo-anomalous sound data, which can be easily updated by using anomalous data. In this paper, we implement a new loss function based on metric learning to learn the distance relationship from each class centroid in feature space for the binary classification model. The proposed multi-task learning of the binary classification and the metric learning makes it possible to build the feature space where the within-class variance is minimized and the between-class variance is maximized while keeping normal and anomalous classes linearly separable. We also investigate the effectiveness of additionally using anomalous sound data for further improving the binary classification model. Our results showed that multi-task learning using binary classification and metric learning to consider the distance from each class centroid in the feature space is effective, and performance can be significantly improved by using even a small amount of anomalous data during training.**

***Index Terms*—anomalous sound detection, binary classification, class centriods, semi-supervised learning, metric learning, multi-task learning**

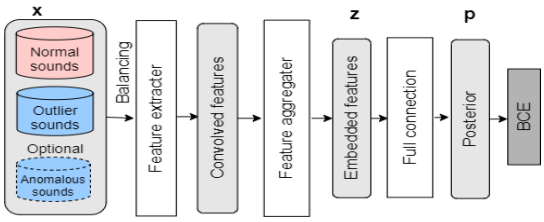
I. Introduction

Anomalous sound detection (ASD) is the task of identifying whether a sound emitted from a particular object is normal or anomalous. Here, an anomalous sound is caused by an atypical event, such as an accident or the malfunction or breakdown of a machine. The detection of anomalous sounds can also be used to improve the efficiency of maintenance work on manufacturing equipment and infrastructure and to monitor equipment installed in difficult locations for people to enter. The use of this technology is expected to become widespread during the coming fourth industrial revolution, e.g., factory automation utilizing artificial intelligence [1], [2].

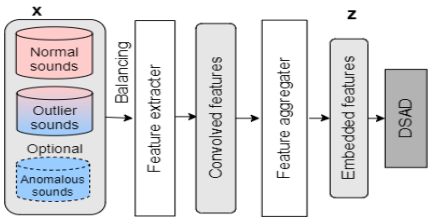
It would be difficult to collect data representing every possible anomalous sound because these sounds rarely occur during the normal operation of factory equipment, and the possible types of anomalous sounds are very diverse. Therefore, when constructing an ASD model, it is often the case that only normal data is used, or that only a small amount of anomalous data is used in addition to the normal data. One detection method that is used when only normal data is being utilized is outlier detection, which models normal data and detects data that does not correspond to the model, categorizing it as anomalous. Typical methods include generative modeling approaches which utilize probabilistically modeling of the distribution of normal data using Gaussian mixture models [3], and one-class support vector machines [4], [5] using acoustic features such as Mel frequency cepstrum coefficients. As a result of advances in deep learning technology, methods based on neural networks are also gaining attention [6]. These methods train autoencoders (AE) or autoregressive models with recursive neural networks to reconstruct normal data, and calculate the reconstruction error for use as an anomaly score [7]–[10]. Although these methods can achieve a high level of performance, they use only normal data during training, so it is difficult to make effective use of anomalous data.

In contrast, binary classification approaches utilize outlier sound data in addition to normal sound data such as typical target machine operating noise [11]–[13]. These methods assume that anomalous data is distributed outside the normal data domain, and outlier data is distributed further outside of normal data. Based on this assumption, a binary classifier is trained using the normal data as positive examples, and the outlier data as pseudo-negative examples, so that distance from the decision boundary can be used as an anomaly score for the data. Therefore, unlike methods that model normal data, a small amount of anomalous data, usually obtained during the ASD system's routine operation, is directly used for training. It is expected that these types of binary classification-based ASD methods will continuously improve due to the long-term operation of the system as more and more anomalous data is collected.

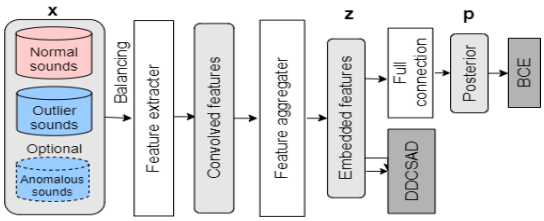
In this paper, we propose a new method for detecting anomalous sounds based on a binary classification model using outlier data. A new loss function is introduced to the binary classification model to learn the relative distances between each sound class's centroids in the feature space. By using both multi-task learning of the classification task and metric learning, we should be able to map the feature space that minimizes within-class variance and maximizes between-class variance, while allowing linear separation between classes. We also investigate the relationship between the amount of anomalous data used for training and classification performance to clarify



(a) Classification-based method



(b) Distance-based method



(c) Proposed method

Fig. 1: Architectures of existing and proposed ASD methods.

how the amount of anomalous data used impacts the detection of anomalous sounds in binary classification models. We conducted our experimental evaluation using the same dataset used in the DCASE 2020 Task 2 anomaly detection task [6]. Our results showed that 1) multi-task learning using binary classification and metric learning to consider the distance from each class centroid in the feature space is effective, and 2) performance can be significantly improved by using even a small amount of anomalous data during training.

II. Related work

This section provides a brief overview of previous research on ASD using binary classification. We also describe a metric learning approach for improving anomaly detection, originally proposed for image processing.

A. Anomalous Sound Detection Based on Binary Classification

Several methods for detecting anomalous sounds based on binary classification using outlier data have been proposed [11]–[13]. An overview of this approach is shown in Fig. 1 (a). These methods assume that even task-irrelevant outlier data can be substituted as anomalous data if carefully selected. These methods allow a model to be trained to solve a classification problem that discriminates between anomalous and normal sounds, even when anomalous data is not available. In [12], the important aspects of outlier data selection has been identified as the matching of recording conditions, similarity to the target sound, and content diversity.

Consider a set $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N\}$ that has N samples of outlier data, and a set $\tilde{\mathbf{X}} = \{\tilde{\mathbf{x}}_1, \tilde{\mathbf{x}}_2, ..., \tilde{\mathbf{x}}_M\}$ that has M samples of normal and anomalous data. Normal and anomalous data sets are assigned labels $\tilde{y}_j \in \{+1, -1\}$ $(j = 1, 2, ..., M)$ for each data sample. Here, $\tilde{y} = +1$ indicates that the data is normal, and $\tilde{y} = -1$ indicates that the data is anomalous. The outlier data set is labeled $y \in \{+1, -1\}$, which is common to all of the data, based on a prior assumption as to whether it is closer to the normal or the anomalous data. When performing ASD using a method based on binary classification, the network is trained to minimize the following binary cross-entropy (BCE) loss function:

$$\mathcal{L}_{\mathrm{BCE}} = -\frac{1}{N+M}\Bigg\{\sum_{i=1}^{N}\log\left(1-p_i\right) + \sum_{j=1}^{M}\{u(\tilde{y}_j)\log\left(\tilde{p}_j\right) + (1-u(\tilde{y}_j))\log(1-\tilde{p}_j)\}\Bigg\}, \tag{1}$$

where p and $\tilde{p}$ are the posterior probabilities output by network ϕ_p, such as $p = \phi_p(\mathbf{x})$, which minimizes (1) when $\mathbf{x}$ or $\tilde{\mathbf{x}}$ is used as input and $u(y)$ is a binary function that takes 1 for $y > 0$ and 0 for $y \leq 0$. In this paper, the outlier data is always treated as pseudo-anomalous data by setting y to -1. This assumption allows learning to be performed even when no anomalous data actually exists. During inference, the posterior probability p output by the network is used to calculate and use the anomaly score s as $s = 1 - p$.

B. Anomaly Detection Based on Metric Learning

Another method of anomaly detection, based on metric learning using outlier data, is deep semi-supervised anomaly detection (DSAD) [14]. Fig. 1 (b) shows an overview of this approach. During classification, data that falls closer to the centroid is deemed normal, while data falling farther from the centroid are considered anomalous. The centroid in the feature space is obtained using the pre-trained model. The DSAD loss function is expressed as follows, using a set of N outlier data $\mathbf{X}$ and a set of M normal and anomalous data $\tilde{\mathbf{X}}$:

$$\mathcal{L}_{\mathrm{DSAD}} = \frac{1}{N+M}\left\{\sum_{i=1}^{N} \| \mathbf{z}_i - \mathbf{c} \|^{2y} + \eta \sum_{j=1}^{M} \| \tilde{\mathbf{z}}_j - \mathbf{c} \|^{2\tilde{y}_j}\right\}, \tag{2}$$

where $\mathbf{z}$ and $\tilde{\mathbf{z}}$ are the embedding vector output by encoder network ϕ_z, such as $\mathbf{z} = \phi_z(\mathbf{x})$, which minimizes (2) when $\mathbf{x}$ or $\tilde{\mathbf{x}}$ are used as input. $\mathbf{c} \in \mathbb{R}^D$ is the centroid in the feature space of the normal data $(\tilde{y} = +1 \cup y = +1)$, and $\eta > 0$ is the hyperparameter that weights the data. For the data deemed normal, a loss is imposed on the distance between centroid $\mathbf{c}$ and the mapping point, and learning is performed to minimize the within-class variance of the data. Note that DSAD can be used even when there is no anomalous data. In this paper, the outlier data is always treated as pseudo-anomalous data by setting y to -1.

The training procedure is as follows. First, the encoder network is trained as an AE using only data considered normal to obtain the parameters' initial values. Then, using the trained encoder, the average vector is calculated for the embedding vectors of the normal data, which is used as the centroid of

the normal data. The data deemed anomalous is also used, and learning is performed by minimizing (2). Note that centroid $\mathbf{c}$ is not updated and does not change from its initial value. During inference, the distance between embedding vector $\mathbf{z}$ and centroid $\mathbf{c}$ is used to calculate the anomaly score.

III. Proposed Method

An overview of the proposed semi-supervised ASD system is shown in Fig. 1 (c). We propose a new loss function based on metric learning, which we call a Deep Double-Centroids Semi-supervised Anomaly Detection (DDCSAD) loss function. The proposed loss function, which is based on metric learning, is used for the embedding vector. The DDCSAD loss function is an extension of the DSAD loss function, that considers the centroid of normal data and the centroid of outlier data. The DDCSAD loss function is calculated by first extending (2) as follows:

$$\begin{aligned}\mathcal{L}_{\mathrm{DDCSAD}} = &\frac{1}{N+M}\sum_{i=1}^{N}\left\{\| \mathbf{z}_i - \mathbf{c}_p \|^{2y} + \| \mathbf{z}_i - \mathbf{c}_n \|^{-2y}\right\} \\ &+ \frac{\eta}{N+M}\sum_{j=1}^{M}\left\{\| \tilde{\mathbf{z}}_j - \mathbf{c}_p \|^{2\tilde{y}_j} + \| \tilde{\mathbf{z}}_j - \mathbf{c}_n \|^{-2\tilde{y}_j}\right\},\end{aligned} \tag{3}$$

where, $\mathbf{c}_p \in \mathbb{R}^D$ and $\mathbf{c}_n \in \mathbb{R}^D$ represent the centroid of the normal and outlier data, respectively. In this paper, the outlier data is always treated as pseudo-anomalous data by setting y to -1. The following equation expresses the final loss function:

$$\mathcal{L} = \mathcal{L}_{\mathrm{BCE}} + \lambda\mathcal{L}_{\mathrm{DDCSAD}}, \tag{4}$$

where $\lambda > 0$ is a hyperparameter that controls the balance between the loss functions. It is expected that multi-task learning using both the cross-entropy of the posterior probability and the DDCSAD loss function will improve the use of data and increase accuracy when learning the decision boundaries, resulting in more accurate ASD.

During training, outlier data is used as pseudo-anomaly data. If anomalous data is available, it is also used together with the outlier data. Unlike DSAD, the proposed method does not perform pre-training to initialize the weight parameters but instead uses randomly initialized parameters. The initial values of the two centroids $\mathbf{c}_p$ and $\mathbf{c}_n$ are also calculated using randomly initialized parameters. And then, they are updated at each epoch by recalculating the centroids using the entire training data set.

During inference, posterior probability p (which is the output of the full connection layer), and distance $d = \| \mathbf{z} - \mathbf{c}_p \|^2$ between embedding vector $\mathbf{z}$ and centroid $\mathbf{c}_p$ of the normal class, are used to obtain the anomaly score. First, we compute distance d across the entire set of evaluation data, and then calculate the standardized distance d' (within the range of a maximum value one and a minimum value zero) across the entire data. Finally, anomaly score s is calculated using the following equation:

$$s = \alpha \times (1-p) + (1-\alpha) \times d', \tag{5}$$

where, α is a hyperparameter that determines the proportion of anomaly scores using posterior probability p.

TABLE I: α for BCE+DSAD and BCE+DDCSAD.

Method	fan	pump	slider	ToyCar	ToyConv.	valve
BCE+DSAD	0.1	0.2	0.0	0.0	0.0	0.0
BCE+DDCSAD	0.1	1.0	1.0	0.1	0.0	1.0

IV. Experimental evaluation

A. Experimental conditions

To evaluate the performance of the proposed method, we conducted an experiment using the DCASE 2020 Task 2 [6] data, which consists of two datasets, ToyADMOS [15] and MIMII [16]. From the ToyADMOS dataset, we used audio data for two types of machines, *ToyCar* and *ToyConveyor*, while MIMII provided audio data for four types of machines, *fan, pump, slider,* and *valve*, for a total of six machine types. Each set of audio data for each type of machine consists of seven or eight different machines of that type, and ID information is provided to indicate exactly which machine the data belongs to. For each machine (i.e., each ID), about 1,000 samples of normal sound are provided as training data, about 200 to 400 samples of normal and anomalous sounds from some of the machines are provided as validation data, and about 400 samples of normal and anomalous sounds from machines different from those included in the validation data are provided as evaluation data. Each sample is about 10 seconds in duration, and includes the target machine's operational and environmental sounds, with a sampling rate of 16 kHz on one recording channel.

We compared the results when using each of the following five loss functions:

- **BCE**: A function which divides the feature space linearly. For the loss function, we used (1).
- **DSAD**: The centroid of the normal data is defined to minimize within-class variance. For the loss function, we used (2). However, we did not use the pre-training model to initialize the AE's weight parameters, and we randomly initialized both the weight parameters and the parameters of the centroid of the normal data $\mathbf{c}$ because we found that the random initialization tended to outperform the AE-based initialization.
- **DDCSAD**: The centroids of both the normal and outlier data are defined to minimize within-class variance and maximize between-class variance. For the loss function, we used (3).
- **BCE+DSAD**: We defined the centroid of the normal data to minimize within-class variance while creating a function that linearly divides the feature space. The loss function is derived by replacing $\mathcal{L}_{\mathrm{DDCSAD}}$ in (4) with $\mathcal{L}_{\mathrm{DSAD}}$ in (2). In this case, as in the case of DSAD, we did not use pre-training to initialize the AE's weight parameters, and we randomly initialized both the weight parameters and centroid $\mathbf{c}$ of the normal data parameters.
- **BCE+DDCSAD**: A function which divides the feature space linearly, while also defining the centroid for normal and outlier data to minimize within-class variance and maximize between-class variance. For the loss function, we used (4).

In order to accurately compare differences in performance related to the use of these various loss functions, we used the same pre-processing and network structure for all of the methods being compared, and only varied the loss function and presence of the full connection layer. As a pre-processing step, we calculated each machine's amplitude values, normalized them to have a mean of 0 and variance of 1, and then extracted 128-dimensional logarithmic Mel filter-banks, which were used as input features for the network using 1,024-sample windows and a hop-size of 512 samples. The network structure consisted of a feature extractor with a convolutional layer that takes a series of acoustic feature as input, an aggregator that aggregates the acoustic feature and transforms them into fixed-length embedding vectors, and a full connection layer that performs binary classification using the embedding vectors. As our feature extractor, we used the ResNet38 framework [17] proposed for pretrained audio neural networks (PANNs) [18]. Global average pooling, which averages in frequency and time information, was used as the aggregator. We performed learning for each particular machine ID. The normal data of the target ID of the target machine type was used as the normal data, and the normal data of other IDs of the target machine type, and the normal data of all the IDs of the other machines in the same data set, were used as outlier data [12]. The outlier data was used as pseudo-anomalous data in all of the methods compared. We used different learning rates for each layer, 0.0001 for the convolutional layer and 0.001 for the full connection layer. We used Adam [19] as the optimization method and multiplied the learning rate by 0.5 after every 1,000 iterations. We set the total number of iterations to 4,000. We set the value of λ to 1.0 and the value of η to 2.0. During inference, we divided the number of frames in each sample's acoustic feature series into ten sections, with overlap allowed so that the frame length was equal to 256. We calculated anomaly scores for each segment of each series, and these scores were then averaged and used as the final anomaly score. The values of α for inference were decided using the validation data, and are shown in Table I.

We used the area under the receiver operating characteristic (ROC) curve (AUC) as an evaluation metric, which is calculated as follows:

$$\mathrm{AUC} = \frac{1}{N_- N_+} \sum_{i=1}^{N_-} \sum_{i=j}^{N_+} \mathcal{H}\left(\mathcal{A}_\theta\left(\mathbf{x}_j^+\right) - \mathcal{A}_\theta\left(\mathbf{x}_i^-\right)\right), \tag{6}$$

where $\mathcal{H}(a)$ represents a binary function that returns 1 when $a > 0$ and 0 when $a \leq 0$, and where $\mathcal{A}_\theta(\mathbf{x})$ represents a function that returns an anomaly score when $\mathbf{x}$ is input. $\{\mathbf{x}_i^-\}_{i=1}^{N^-}$ and $\{\mathbf{x}_j^+\}_{j=1}^{N^+}$ represent normal and anomalous data, respectively, and are sorted to rank each sample's anomaly scores in descending order. N_- and N_+ represent the number of normal and anomalous data samples, respectively.

B. Performance evaluation when anomalous data is not used

First, we investigated the performance of ASD when no anomalous data was used for training. Our experimental results are shown in Table II. The averaged results (Machine Average) in Table II show that the DDCSAD loss function outperformed the DSAD loss function. This result suggests that it is important to consider not only the centroid of normal data but also the centroid of outlier data in order to increase between-class variance, which is achieved by making the centroids updatable. Furthermore, the improvement in performance when using BCE+DDCSAD over that of using either DDCSAD or BCE alone confirms the effectiveness of multi-task learning.

C. Relationship between amount of anomalous training data and performance

We added a small amount of anomalous data to the training data to investigate how this affected performance. For each machine ID, 64 samples were randomly selected from the validation and evaluation data's anomalous data, and moved to the training data. We then increased the number of anomalous samples used for training to $[1, 2, 4, ..., 64]$ for each method and observed the performance change. The pre-processing, learning and inference procedures were identical to those described in IV-B, but we made the following two changes in the event that new anomalous data became available during the operation of the ASD system:

1) The ratio of normal data, outlier data and anomalous data was set to 32:31:1 so that there was always one anomalous data sample representing the outlier class in each mini-batch.
2) We used the model trained without anomalous data as the initial value, and then halved the total number of iterations to 2,000.

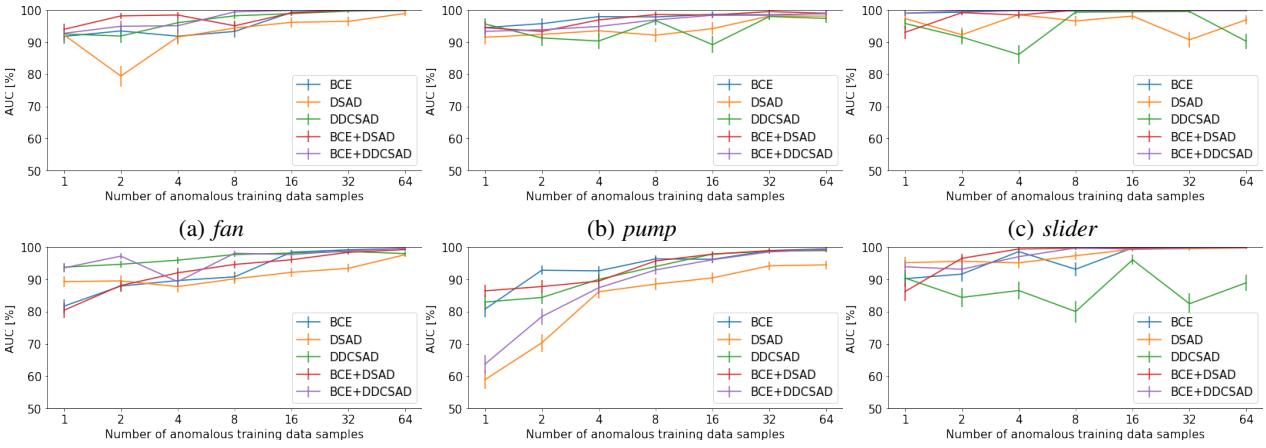
Experimental results when using the evaluation data are shown in Fig. 2. We can see from these results that each method's performance improves as more anomalous training data is added. In other words, we can improve the performance of all of these methods by simply adding anomalous data to the training data without changing the systems' structure. Furthermore, by comparing methods with and without BCE, we can also confirm that BCE-based methods receive a more dramatic boost in performance by utilizing anomalous data during training. This suggests that the proposed multi-task learning makes more effective use of small amounts of anomalous training data.

V. Conclusion

In this paper, we have proposed a multi-task learning method for detecting anomalous sounds which uses a binary classification model based on outlier data, and a loss function based on metric learning. We also proposed DDCSAD, a loss function that considers the class centroids of both normal and anomalous data in the feature space, and demonstrated its effectiveness through experimental evaluation. The relationship between the amount of anomalous data used during training and detection performance was also investigated. Our experimental results showed that the more anomalous data added to the outlier class during training, the better detection performance becomes, especially when using a binary classification model. In future work, we will develop a metric for

TABLE II: Detection results AUC [%] for all machines and loss functions (95% confidence interval [20]).

Machine Type	BCE	DSAD	DDCSAD	BCE+DSAD	BCE+DDCSAD
fan	92.69 ± 2.06	82.33 ± 3.00	91.05 ± 2.25	92.96 ± 2.02	$\mathbf{95.14 \pm 1.70}$
pump	$\mathbf{94.39 \pm 1.88}$	88.88 ± 2.60	91.71 ± 2.27	89.91 ± 2.49	92.14 ± 2.21
slider	90.30 ± 2.43	93.46 ± 2.01	89.97 ± 2.46	93.38 ± 2.03	$\mathbf{97.60 \pm 1.24}$
ToyCar	87.82 ± 1.77	84.10 ± 2.00	91.18 ± 1.51	83.72 ± 2.03	$\mathbf{93.85 \pm 1.26}$
ToyConveyor	75.21 ± 2.54	68.51 ± 2.74	68.37 ± 2.74	64.13 ± 2.82	$\mathbf{82.05 \pm 2.25}$
valve	89.92 ± 2.45	90.80 ± 2.35	88.62 ± 2.59	$\mathbf{98.61 \pm 0.94}$	96.08 ± 1.57
Machine Average	88.39	84.68	86.82	87.12	**92.81**



(a) *fan* (b) *pump* (c) *slider*

(d) *ToyCar* (e) *ToyConveyor* (f) *valve*

Fig. 2: Change in AUC [%] for each machine and loss function when adding anomalous data (95% confidence interval [20]).

selecting outliers to use for training from large data sets to further improve performance.

Acknowledgment

This paper was partly supported by a project, JPNP20006, commissioned by NEDO.

References

[1] B. Bayram, T. B. Duman, and G. Ince, "Real time detection of acoustic anomalies in industrial processes using sequential autoencoders," *Expert Systems*, vol. 38, no. 1, p. e12564, 2021.

[2] D. Huang, C. Lin, C. Chen, and J. Sze, "The internet technology for defect detection system with deep learning method in smart factory," in *2018 4th International Conference on Information Management (ICIM)*, 2018, pp. 98–102.

[3] D. W. Scott, "Outlier detection and clustering by partial mixture modeling," in *COMPSTAT 2004 — Proceedings in Computational Statistics*, Springer. Physica-Verlag HD, 2004, pp. 453–464.

[4] D. M. Tax and R. P. Duin, "Support vector data description," *Machine learning*, vol. 54, no. 1, pp. 45–66, 2004.

[5] Y. Chung, S. Oh, J. Lee, D. Park, H.-H. Chang, and S. Kim, "Automatic detection and recognition of pig wasting diseases using sound data in audio surveillance systems," *Sensors*, vol. 13, no. 10, pp. 12 929–12 942, 2013.

[6] Y. Koizumi, Y. Kawaguchi, K. Imoto, T. Nakamura, Y. Nikaido *et al.*, "Description and discussion on DCASE2020 challenge task2: Unsupervised anomalous sound detection for machine condition monitoring," in *arXiv e-prints: 2006.05822*, June 2020, pp. 1–4. [Online]. Available: https://arxiv.org/abs/2006.05822

[7] T. Hayashi, T. Yoshimura, and Y. Adachi, "Conformer-based id-aware autoencoder for unsupervised anomalous sound detection," DCASE2020 Challenge, Tech. Rep., July 2020.

[8] P. Malhotra, L. Vig, G. Shroff, and P. Agarwal, "Long short term memory networks for anomaly detection in time series," in *Proceedings*, vol. 89. Presses universitaires de Louvain, 2015, pp. 89–94.

[9] T. Hayashi, T. Komatsu, R. Kondo, T. Toda, and K. Takeda, "Anomalous sound event detection based on wavenet," in *2018 26th European Signal Processing Conference (EUSIPCO)*. IEEE, 2018, pp. 2494–2498.

[10] R. Giri, S. V. Tenneti, K. Helwani, F. Cheng, U. Isik, and A. Krishnaswamy, "Unsupervised anomalous sound detection using self-supervised classification and group masked autoencoder for density estimation," DCASE2020 Challenge, Tech. Rep., July 2020.

[11] L. Ruff, R. A. Vandermeulen, B. Franks, K.-R. Muller, and M. Kloft, "Rethinking assumptions in deep anomaly detection," *arXiv*, vol. abs/2006.00339, 2020.

[12] P. Primus, V. Haunschmid, P. Praher, and G. Widmer, "Anomalous sound detection as a simple binary classification problem with careful selection of proxy outlier examples," *arXiv*, vol. abs/2011.02949, 2020.

[13] P. Primus, "Reframing unsupervised machine condition monitoring as a supervised classification task with outlier-exposed classifiers," DCASE2020 Challenge, Tech. Rep., July 2020.

[14] L. Ruff, R. A. Vandermeulen, N. Görnitz, A. Binder, E. Müller, K.-R. Müller, and M. Kloft, "Deep semi-supervised anomaly detection," in *International Conference on Learning Representations*, 2020. [Online]. Available: https://openreview.net/forum?id=HkgH0TEYwH

[15] Y. Koizumi, S. Saito, H. Uematsu, N. Harada, and K. Imoto, "ToyADMOS: A dataset of miniature-machine operating sounds for anomalous sound detection," in *Proceedings of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. IEEE, November 2019, pp. 308–312.

[16] H. Purohit, R. Tanabe, T. Ichige, T. Endo, Y. Nikaido, K. Suefusa, and Y. Kawaguchi, "MIMII Dataset: Sound dataset for malfunctioning industrial machine investigation and inspection," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2019 Workshop (DCASE2019)*, November 2019, pp. 209–213.

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

[18] Q. Kong, Y. Cao, T. Iqbal, Y. Wang, W. Wang, and M. D. Plumbley, "PANNs: Large-scale pretrained audio neural networks for audio pattern recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2880–2894, 2020.

[19] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2015.

[20] J. A. Hanley and B. J. McNeil, "The meaning and use of the area under a receiver operating characteristic (ROC) curve," *Radiology*, vol. 143, no. 1, pp. 29–36, 1982.