THE SMALL RICE SUBMISSION TO THE DCASE2021 TASK 2 CHALLENGE: SEMI-SUPERVISED ANOMALY DETECTION USING CONTRASTIVE LEARNING

Technical Report

Xinyu Cai[1,2], *Heinrich Dinkel*[1], *Zhiyong Yan*[1], *Yongqing Wang*[1], *Junbo Zhang*[1], *Zhiyong Wu*[2], *Yujun Wang*[1]

[1]Xiaomi Corporation, Beijing, China
[2]Shenzhen International Graduate School, Tsinghua University, Shenzhen, China
{caixinyu,dinkelheinrich,yanzhiyong, zhangjunbo1,wangyongqing3,wangyujun}@xiaomi.com
zywu@sz.tsinghua.edu.cn

ABSTRACT

This paper describes our submission to the DCASE 2021 Task 2 challenge. The objective is identifying whether the sound emitted from a machine is normal or anomalous without having access to large amounts of anomalous samples. Our anomaly score calculator system is a combination of two models: i) AutoEncoder-based unsupervised training and ii) EfficientNet-based supervised model. To alleviate the problem of domain shift, we train the models with contrastive loss and hard example mining manner, which leads to a substantial improvement with regards to the main omega evaluation metric. Further we investigate the use of median-filtering, timemasking, time shifting and mixup augmentation for this task, which further boosts performance. Our best single model submission achieves an official omega score of 71.72, 70.05, 72.14, 67.26, 66.17, 71.97, 68.47 for Fan, Gearbox, Slider, Toy Train, Toy Car, Pump, Valve on the development dataset, respectively.

Index Terms— Unsupervised anomaly sound detection, AutoEncoder, Convolutional neural networks, Few shot learning.

1. INTRODUCTION

Anomaly sound detection has a wide range of applications, such as Machine Condition Monitoring (MCM). The aim of acoustic MCM is to monitor whether a machine is working properly through the sound signal collected by a microphone. This technology can help realize unattended factories, and thus reducing labor costs.

The DCASE 2021 Task 2 [1] has two main challenges: i) only normal sound clips are provided as training data and ii) the training data and the test data are in different domains. These restrictions reflect the problems encountered when applying anomaly sound detection system to real factory scenes, where it is changeable and difficult to collect exhaustive anomalous sounds.

The main idea of unsupervised anomaly sound detection is to learn the properties of the normal sounds, and then classify samples as anomalous or normal by the deviation of the sample from the normal sound properties. Statistic based methods such as Hidden Markov Model [2] and Gaussian Mixture Model [3] attempt to model the probability distribution of normal sound, and determine whether the sound is abnormal by posterior probability. The Non-negative Matrix Factorization method [4] and Autoencoder method [5] are both trained to compress and reconstruct normal sounds, such that those models will predict large reconstruction errors when encountering abnormal sounds.

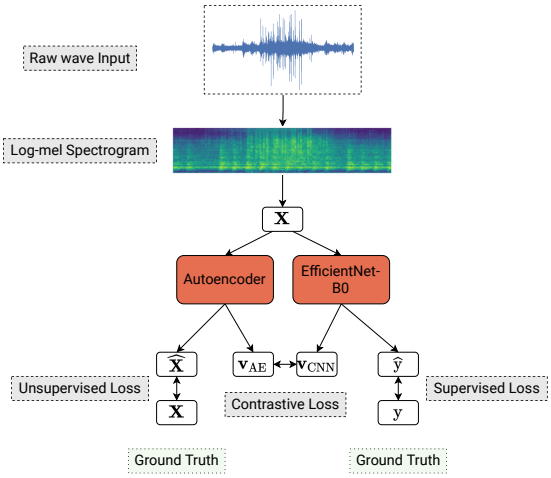

Figure 1: The overall architecture used in this work. A spectrogram feature is first extracted from an input waveform. Then the feature is fed into two separate models: An Autoencoder (AE) and a EfficientNet-B0. The model is jointly optimized to reconstruct the input spectrogram, a section label and minimize the contrastive loss between hidden representations.

Another modern type of approach fitting to the DCASE2020 and DCASE2021 datasets is supervised anomaly detection. Since the recent DCASE 2020 competition, the training data is composed of normal sounds from different operating conditions with different section IDs. The main idea for supervised anomaly detection is to use the section ID as a label and perform classification. Since we have access to the section ID during testing, a classifier could perform anomaly sound detection by identifying misclassified samples as anomaly sounds. In the previous competition, supervised classifiers have seen to perform well [6, 7, 8].

Inspired by the recent success of contrastive learning approaches for self-supervised audio pretraining [9, 10, 11], we aim to enhance our model's capability to detect unseen events by linking multiple views together. In order to solve the challenges within the DCASE2021 Task2 dataset, our proposed system is a novel combination of two mainstream anomaly detection models trained with an additional contrastive loss function.

The paper is structured as follows: In Section 2 we introduce

our approach. Further, in Section 3 details regarding the dataset and experimental setup are provided. Results can be seen in Section 4 and the conclusion is given in Section 5.

2. PROPOSED APPROACH

Our approach is the fusion of two individual approaches: unsupervised autoencoder-based training combined with a supervised convolutional neural network (CNN). The architecture can be seen in Figure 1.

2.1. Autoencoder-based unsupervised classification

Our autoencoder (AE) baseline model is trained to reconstruct training samples. The motivation is that a well trained AE will produce a low error if a new data sample has been seen during the training phase (normal sample) and a large error when it encounters unseen anomalous sounds. Formally, let x be an input sample and AE be the autoencoder, our training objective follows:

$$\begin{aligned} \mathrm{AE}(x) &\mapsto \hat{x}, \\ \mathcal{L}_{\text{unsup}}(\cdot) = \mathcal{L}_{\mathrm{AE}}(x) &= \mathcal{L}_{\mathrm{MSE}}(\hat{x} - x), \end{aligned} \tag{1}$$

where the training loss is chosen to be the mean square error (MSE).

2.2. EfficientNet-based supervised classification

Our supervised approach uses the provided section ID as classification target. The model outputs the softmax value that is the predicted probability for each section. Formally, for a sample x and corresponding one-hot target y, we compute the standard cross entropy (CE) loss, as seen in Equation (2).

$$\begin{aligned} \mathrm{CNN}(x) &\mapsto \hat{y}, \\ \mathcal{L}_{\text{sup}}(\cdot) = \mathcal{L}_{\mathrm{CE}}(\hat{y}, y) &= -\frac{1}{N}\sum_{i}^{N} y_i \log \hat{y}_i, \end{aligned} \tag{2}$$

where CNN represents the CNN-based classifier and N the number of samples. Then the anomaly score $A(x)$ is calculated as:

$$A(x) = \log \frac{1 - \hat{y}_i}{\hat{y}_i}, \tag{3}$$

where $\hat{y}_i$ is the softmax output for the correct section. Note that if the sample x is divided into consecutive segments $(x_1, x_2, ..., x_P)$, the anomaly score will be $\frac{1}{P}\sum_i^P A(x_i)$.

2.3. Proposed contrastive semi-supervised learning

We train these models with an additional contrastive loss [12]. The contrastive loss $\mathcal{L}_{\text{contrastive}}$ is added between the hidden representations of both models $(\mathbf{v}_{\mathrm{AE}}, \mathbf{v}_{\mathrm{CNN}})$ as:

$$\begin{aligned} \mathbf{p} &= \mathbf{v}_{\mathrm{AE}}, \\ \mathbf{u} &= \mathbf{v}_{\mathrm{CNN}}, \\ \mathcal{L}_{\text{contrastive}}(\cdot) &= -\sum_i \log \frac{\exp(\langle \mathbf{u}_i, \mathbf{p}_i \rangle / \rho)}{\sum_{j \neq i} \exp(\langle \mathbf{u}_i, \mathbf{p}_j \rangle) / \rho)}, \end{aligned} \tag{4}$$

where $\langle , \rangle$ represents an inner product, $\rho \in \mathbb{R}$ is a scalar hyperparameter and $\mathbf{v}_{i,j} \in \mathbb{R}^{256}$ are hidden vector representations obtained by both model via projection. Concretely speaking, we transform the output vector of Autoencoder's bottleneck layer and EfficientNet's feature layer into same dimension by linear transformation, then map representations to the space where contrastive loss is applied via a shared MLP projection layer with one hidden layer. So to say, our approach aims to obtain two different representations of a single sample, which is reminiscent of SimCLR [9], unsupervised data augmentation (UDA) [13] and other semi and self-supervised approaches.

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{unsup}} + \mathcal{L}_{\text{sup}} + \mathcal{L}_{\text{contrastive}} \tag{5}$$

The final loss for optimization can be seen in Equation (5).

2.4. Data Augmentation

Additionally to the above techniques, we explore the use of data augmentation techniques. Regrading conventional techniques, we explore the use of Mixup [14] along with time masking and shifting for model training. Further, our intuition is that the input audio data contains large amounts of short-time noise, thus an input feature might contain a surplus of unreliable information, which can affect the performance of our supervised training method. We propose a median filtering approach applied on the input spectrogram feature along frequency axis aiming to reduce distracting noise.

3. EXPERIMENTAL SETUP

Log Mel-spectrogram (LMS) features are chosen as the default front-end feature for the task. Overall, seven models are trained in our approach, one for every machine type.

For the supervised CNN training, each 128-filter LMS is extracted from a 64 ms window with a stride of 32 ms, resulting in an approximately 128×311 dimensional input tensor. If segments are shorter than 10 seconds (or 311 samples), we zero-pad the input to the longest sample within a batch. We also explore using the model pretrained on Audioset [15].

Regarding the AE training, we follow the baseline approach by combining a 2 left-right frame window of a single feature into a single input vector of size 640 $(128 * 5)$. All experiments are run for 300 epochs, with learning rate halving every 30 epochs. The batchsize is set to 32 for training and we set the hyperparameter $\rho = 0.07$. Our proposed median filtering approach uses a window size of 30 frames for each filter bank respectively.

PyTorch [16] was used as the default neural network toolkit.

3.1. Evaluation metric

The evaluation metric used in the challenge are the area under curve (AUC) and partial-AUC (pAUC) scores respectively. The scores are defined as:

$$\begin{aligned} \mathrm{AUC}_{m,n,d} &= \frac{1}{N_- N_+} \sum_{i=1}^{N_-} \sum_{j=1}^{N_+} \mathcal{H}(\mathcal{A}_\theta(x_j^+) - \mathcal{A}_\theta(x_i^-)), \\ \mathrm{pAUC}_{m,n,d} &= \frac{1}{\lfloor pN_- \rfloor N_+} \sum_{i=1}^{\lfloor pN_- \rfloor} \sum_{j=1}^{N_+} \mathcal{H}(\mathcal{A}_\theta(x_j^+) - \mathcal{A}_\theta(x_i^-)), \end{aligned} \tag{6}$$

where m represents the index of a machine type, n represents the index of a section and $d = \{\text{source}, \text{target}\}$ represents a domain. The final official score Ω is computed as the harmonic mean (h) of the AUC and pAUC scores:

Model	Fan	Gearbox	Slider	Toy Train	Toy Car	Pump	Valve	Avg.
Autoencoder Baseline	57.90	58.64	60.87	57.50	56.99	57.93	51.94	57.39
MobileNet Baseline	62.43	63.04	58.05	54.49	56.53	59.68	54.57	58.39
EfficientNet-B0	66.11	62.21	67.46	53.39	55.47	63.25	67.06	62.14
+ Pretrain	68.03	65.60	61.78	58.91	61.68	67.46	**68.47**	64.56
+ Pretrain, Mixup	67.14	64.17	68.19	59.44	65.61	68.64	67.02	65.74
+ Pretrain, Median Filter, Mixup	66.06	67.27	65.42	65.10	57.30	62.24	57.73	63.02
+ Median Filter	68.12	67.32	67.47	57.32	60.52	65.68	56.28	63.24
+ Median Filter, Mixup	**71.72**	**70.05**	**72.14**	**67.26**	**66.17**	**71.97**	56.82	68.01
S1 (Best Single Model)	71.72	70.05	72.14	67.26	66.17	71.97	68.47	**69.68**
S2 (Mean Anomaly Score Ensemble)	71.86	72.20	70.63	67.50	64.40	71.66	56.25	67.78
S3 (Max Anomaly Score Ensemble)	67.14	76.85	71.38	68.90	67.83	74.53	58.71	69.33
S4 (Mean softmax Ensemble)	73.94	72.42	73.89	67.32	66.71	73.57	57.07	69.27

Table 1: Main results proposed in our work for the DCASE 2021 Task2 challenge on the held-out development dataset in regards to the main evaluation metric Ω (see Equation (7)). Note that a single model is trained for each machinetype. The average performance across machine types is also provided. The submissions are S1 and S2.

Layer	Output size	Trainable
Input	2560	✗
LMS	128×5	✗
Reshape	640	✗
Enc-Block1	128	✓
Enc-Block2	128	✓
Enc-Block3	128	✓
Enc-Block4	128	✓
Bottleneck	32	✓
Dec-Block1	128	✓
Dec-Block2	128	✓
Dec-Block3	128	✓
Dec-Block4	128	✓
Output	640	✓

Table 2: Autoencoder architecture used in this work. Different to the baseline, we use a smaller bottleneck.

$$\Omega = h\left\{\mathrm{AUC}_{m,n,d},\ \mathrm{pAUC}_{m,n,d}\right\} \tag{7}$$

3.2. Dataset

The data used for this task consists of running sounds of seven machine types being "ToyCar", "Fan", "ToyTrain","Valve", "Gearbox", "Silder" and "Pump", including two recent machine audio datasets, ToyADMOS [17] and MIMII [18].

Notably all provided data samples by the challenge authors have a length of 10 seconds and each section as well as machine type have a near uniformly distributed duration. The overall data length is 70 hours of which the large majority belongs to source domain. Because of the imbalance of dataset, we applied a weighted sampler to ensure 20% samples come from target domain during trainning.

The two models used in this work are described. First, our Autoencoder is similar to the one provided by the challenge baseline, where we changed the bottleneck block size to 32 as seen in Table 2. Second, the EfficientNet-B0 architecture is directly taken from [19], where our approach differs from the standard architecture:

- We use global average and max pooling (GAMP) as our aggregation method compared to the standard global average pooling (GAP).
- The number of input channels is set to 1.

During training both the AE and EfficientNet-B0 models are jointly optimized given the total loss Equation (5). During evaluation, we remove the AE branch and only obtain predictions from the EfficientNet-B0 model used to score our results.

4. RESULTS

The results are displayed in Table 1. As it can be seen, our EfficientNet-B0 baseline approach, which uses the proposed contrastive loss training paradigm, is effective in improving the average result. Furthermore, using our proposed median filtering and mixup data augmentation techniques leads to large gains against the baseline Mobilenet and Autoencoder approaches on most machine types except Valve. Meanwhile, we can get a large gain on Valve when using pretrained model. However, experimental result shows that finetuning pretrained model with the median filter is not a good choice, due to the change of the distribution of input spectrogram.

Based on existing experimental results, our submission system 1 (S1) is an ensemble of single model that perform best on each machine type respectively. Submissions system 2 to 4 (S2 to S4) adopt different model fusion methods on 5 best models for each machine type. S2 outputs average anomaly score of 5 models, while S3 outputs maximum anomaly score. S4 calculates anomaly score using the mean of model's softmax output. Due to the poor performance on Valve dataset, we use the model for Valve from S1 to replace Valve models in other submissions.

5. CONCLUSION

This paper proposes our submission to the DCASE2020 Task2 challenge. Our work includes a novel contrastive loss training scheme for semi-supervised training. Our proposed single model approach improves against the baseline (in terms of Ω) by an average of 10 points absolute on the development dataset, while not being significantly larger in size than the baseline.

6. REFERENCES

[1] Y. Kawaguchi, K. Imoto, Y. Koizumi, N. Harada, D. Niizumi, K. Dohi, R. Tanabe, H. Purohit, and T. Endo, "Description and discussion on dcase 2021 challenge task 2: Unsupervised anomalous sound detection for machine condition monitoring under domain shifted conditions," *arXiv preprint arXiv:2106.04492*, 2021.

[2] E. Dorj and E. Altangerel, "Anomaly detection approach using hidden markov model," in *Ifost*, vol. 2. IEEE, 2013, pp. 141–144.

[3] S. Ntalampiras, I. Potamitis, and N. Fakotakis, "Probabilistic novelty detection for acoustic surveillance under real-world conditions," *IEEE Transactions on Multimedia*, vol. 13, no. 4, pp. 713–719, 2011.

[4] A. Sasou and N. Odontsengel, "Acoustic novelty detection based on ahlac and nmf," in *2012 International Symposium on Intelligent Signal Processing and Communications Systems*, 2012, pp. 872–875.

[5] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *nature*, vol. 323, no. 6088, pp. 533–536, 1986.

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

[7] 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.

[8] P. Daniluk, M. Gozdziewski, S. Kapka, and M. Kosmider, "Ensemble of auto-encoder based systems for anomaly detection," DCASE2020 Challenge, Tech. Rep., July 2020.

[9] T. Chen, S. Kornblith, M. Norouzi, and G. E. Hinton, "A simple framework for contrastive learning of visual representations," *CoRR*, vol. abs/2002.05709, 2020. [Online]. Available: https://arxiv.org/abs/2002.05709

[10] L. Wang, K. Kawakami, and A. van den Oord, "Contrastive Predictive Coding of Audio with an Adversary," in *Proc. Interspeech 2020*, 2020, pp. 826–830. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2020-1891

[11] L. Wang, P. Luc, A. Recasens, J. Alayrac, and A. van den Oord, "Multimodal self-supervised learning of general audio representations," *CoRR*, vol. abs/2104.12807, 2021. [Online]. Available: https://arxiv.org/abs/2104.12807

[12] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan, "Supervised Contrastive Learning," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2020, pp. 18 661–18 673. [Online]. Available: http://arxiv.org/abs/2004.11362

[13] Q. Xie, Z. Dai, E. Hovy, M.-T. Luong, and Q. V. Le, "Unsupervised Data Augmentation for Consistency Training," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2019, pp. 6256–6268. [Online]. Available: http://arxiv.org/abs/1904.12848

[14] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," in *International Conference on Learning Representations*, 2018. [Online]. Available: https://openreview.net/forum?id=r1Ddp1-Rb

[15] 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," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 776–780.

[16] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "PyTorch: An Imperative Style, High-Performance Deep Learning Library," in *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d\textquotesingle Alché-Buc, E. Fox, and R. Garnett, Eds. Curran Associates, Inc., 2019, pp. 8026–8037.

[17] 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," *arXiv preprint arXiv:2106.02369*, 2021.

[18] R. Tanabe, H. Purohit, K. Dohi, T. Endo, Y. Nikaido, T. Nakamura, and Y. Kawaguchi, "Mimii due: Sound dataset for malfunctioning industrial machine investigation and inspection with domain shifts due to changes in operational and environmental conditions," *arXiv preprint arXiv:2105.02702*, 2021.

[19] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proceedings of the 36th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, K. Chaudhuri and R. Salakhutdinov, Eds., vol. 97. PMLR, 09–15 Jun 2019, pp. 6105–6114. [Online]. Available: http://proceedings.mlr.press/v97/tan19a.html