UNSUPERVISED ADVERSARIAL DOMAIN ADAPTIVE ABNORMAL SOUND DETECTION FOR MACHINE CONDITION MONITORING UNDER DOMAIN SHIFT CONDITIONS

Technical Report

Renjie Li[1], *Xiaohua Gu*[1], *Fei Lu*[1], *Hongfei Song*[1], *Jutao Pan*[1],

[1] Chongqing University of Science and Technology, China.
{2019204002}@stu.cqust.edu.cn

ABSTRACT

In the industrial field, it is very important to detect unknown anomalies based on normal production data. Facing the actual production situation, it is also of great significance to study the abnormal detection of the machine under the condition of constantly changing operating conditions. In dcase2021 task 2, we propose to use an unsupervised abnormal sound detection method based on adversarial domain adaptation. This method proposes a framework of adding domain discriminator and one-class classifier on the basis of auto-encoder extreme learning machine, and achieves good results on the development dataset provided by the contest.

Index Terms— unsupervised, abnormal sound detection, adversarial domain adaptation, domain shift, extreme learning machine

1. INTRODUCTION

For the health monitoring of complex industrial system, the traditional method based on fault classification limits the types of faults that may exist in the machine. Because the machine is in normal state most of the time, the faults are rare (the design performance of complex industrial system is reliable, and the failure rate is low), but the number of possible fault types is large, Therefore, it is impractical to collect a marked data set with enough of each possible failure instance. In addition, in the actual production process, the industrial system needs to constantly adjust the machine operating conditions according to the environmental changes, production demand and other conditions. These adjustments are often carried out in a short time. In this case, we can not collect the data set representing all the operating conditions in a short time.

Domain adaptation can be used to transfer information between sections or between different operating conditions in the same section when there are no or not enough similar sections[1]. In fact, in the data set provided by the task, the data in the target domain is seriously insufficient, and the data between the source domain and the target domain cannot be matched one by one. Therefore, domain migration under domain shift conditions will expand the operating conditions on the target domain and expand the representativeness of the dataset. Since the task is unsupervised, we must consider the distribution of data in the source domain and target domain in the input space. For task 2, based on the research of unsupervised fault detection in hierarchical machine learning[2][3][4] combined with deep adversarial learning, we use an unsupervised adversarial domain adaptive anomaly detection method proposed by Michau et al[5][6]. The task of anomaly detection is to detect abnormal mechanical sound, and the task of domain adaptation is to align different source domains to target domains in the same section.

2. HIERARCHICAL EXTREME LEARNING MACHINE FOR UNSUPERVISED ANOMALY DETECTION

Extreme Learning Machine(ELM) is a kind of feed-forward neural network. Its feature is to randomly extract the weights and deviations between the input layer and the hidden layer, and only learn the weights between the hidden layer and the output layer[7].

Hierarchical Extreme Learning Machine (HELM) is constructed by stacking networks. This stacking method includes extracting the hidden layer for each elm and using it as the input of the next elm. Since back propagation can not be used, information can only flow forward. This method requires sequential training for each elm. The underlying network is usually trained in an unsupervised way, such as automatic encoder, while only the last network is trained for anomaly detection. In fact, in helm, the auto-encoder is a feature learner. In addition, in the case that only health data can be used for training, it is impossible to quantify the features in the training process, and it is impossible to transfer the recognition ability about the wrong features. Therefore, it is impossible to realize the loss back propagation of one-class classification to the feature learner. For the above situation, the architecture based on elm has good applicability.

2.1. ELM model

Elm is essentially a single layer feed-forward neural network. The connection weights of the input layer and the hidden layer, and the threshold of the hidden layer can be set randomly and need not be adjusted after setting. Instead of iterative adjustment, the connection weight between hidden layer and output layer, is determined by solving the equations at one time. The most important feature of elm is that for the traditional neural networks, especially the single hidden layer feed-forward neural networks (SLFNs), it is faster than the traditional learning algorithm on the premise of ensuring the learning accuracy.

The single layer feed-forward network is defined as follows:

$$Y = F(M, X, N) \cdot \gamma \tag{1}$$

X is the input, Y is the output, F is the activation function, M and N correspond to the weight and deviation between the input layer and the hidden layer respectively, γ Is the weight that connects the hidden layer to the output layer. The training of elm model includes three steps. First, the values of M and N are given randomly, and

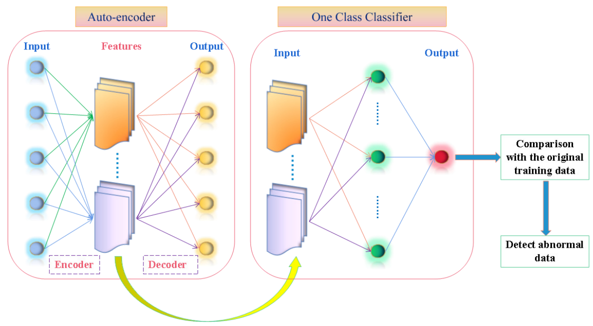

Figure 1: **HELM architecture**. The HELM consists of an arbitrary number of stacked unsupervised auto-encoder ELM and a one-class classifier ELM.

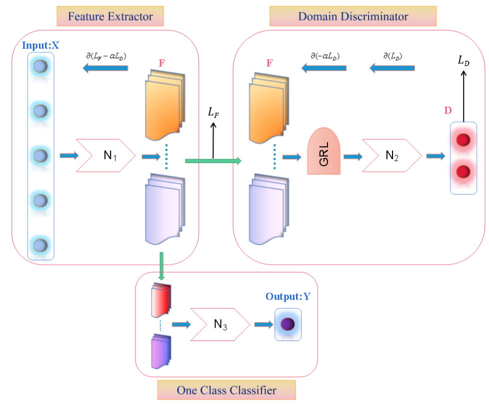

Figure 2: **Unsupervised Adversarial Domain Adaptive Abnormal Sound Detection(UADA-ASD)**. The feature extractor N_1 is trained to minimise the multidimensional scaling loss L_F and to maximise the domain discriminator loss L_D. The domain discriminator N_2 is a traditional feed-forward dense classifier. Once trained, features are fed to the one-class classifier N_3 for anomaly detection.

then the Eq.(1) Y is solved to make it as close as possible to a desired target T. Finally, given an architecture, M, N and a target T, Training elm model includes first drawing M, N randomly, the value of γ can be calculated to satisfy the optimal ELM.

2.2. HELM model

The helm architecture is shown in Fig. 1. A single compressed elm auto-encoder is used as a feature learner, and then the feature is used as the input of one-class classifier for anomaly detection.

3. UNSUPERVISED ADVERSARIAL DOMAIN ADAPTIVE ABNORMAL SOUND DETECTION

3.1. Network structure

In this paper, we plan to use the ELM architecture. The structure consists of three parts, as shown in Fig. 2. It consists of a feature coder N_3, a confrontational feature domain discriminator N_3 connected to a network N_3 with a gradient inversion layer, and one-class classifier N_3 performing an anomaly detection task. The emphasis of this method is the feature alignment strategy and its influence on the performance of anomaly detector.

Since the anomaly detection in task is unsupervised, its purpose is to detect the abnormal instances that are not available in the source domain and target domain during training[8]. Therefore, our goal is to collect domain independent features as much as possible, so as to monitor the health of the target data, including the operation mode not obtained during the training[9].

In order to align features without labels, the method used tries to ensure that the features are meaningful relative to the information originally contained in the data. Therefore, in our method, we want to minimize the multi-dimensional scale loss affected by the dimension reduction tools in the auto-encoder. Here, the feature extractor N1 can be trained in a confrontational manner, first, to minimize the proposed multi-dimensional scale loss, and second, to maximize the domain discriminator loss. After training, the extracted features will be used for one-class classification in N3.The threshold of one-class classification is calculated in the same way as the baseline system. The anomaly detection threshold is determined as the 90-th percentile of the gamma distribution.

3.2. Multidimensional scale loss

The feature space comes from the feature encoder N1, which is actually a coding part of the auto-encoder, transforming the input nonlinearity into a new feature space with different dimensions. In order to ensure that the feature space contains data distribution information similar to the input space, this paper redefines the loss calculation method which can better maintain the sample relationship between the source domain dataset and the target domain dataset.

The input data is expressed as X, and the feature of neural network learning is expressed as F. we propose to define multidimensional scaling loss as:

$$L_F = \sum_{D \in \left\{ \begin{array}{c} Src \\ Tgt \end{array} \right\}} \frac{1}{|D|} \sum_{(i,j)\in D} \| \| \|X_i - X_j\|_2 - \widehat{\eta_D} \| F_i - F_j \|_2 \|_2 \tag{2}$$

Where

$$\forall D \in \left\{ \begin{array}{c} Src \\ Tgt \end{array} \right\}, \quad \widehat{\eta_D} = \mathrm{Argmin}_{\tilde{\eta}_D} L_F\left(\tilde{\eta}_D\right) \tag{3}$$

For every pair of samples i and j, the L_F compares their Euclidean distance in the input space X with their distance in the feature space F , given a global scaling factor $\tilde{\eta}_D$. The loss is minimised when, given the scaling factors, the distances in the feature space between every pair of samples are the same as in the input space. The scaling factors $\tilde{\eta}_D$ are computed independently for each dataset to allow for independent scaling and to mitigate different distribution shifts such as translations, noise, rotations, and scales. Since these shifts are not known a priori, the variables $\tilde{\eta}_D$ must either be learned or chosen arbitrarily. In Eq. (2), we propose to learn their value as the value that minimises the loss, since a closed-form solution exists.

From an optimisation perspective, the minimisation of the loss (1) is equivalent to that where $\eta_{Src} = 1$ and $\eta_{Tgt} = \widehat{\eta_{Src}}/\widehat{\eta_{Tgt}}$, such that we can define the loss as:

$$L_F = \sum_{D \in \left\{ \begin{array}{c} Src \\ Tgt \end{array} \right\}} \frac{1}{|D|} \sum_{(i,j) \in D} | \, \|X_i - X_j\|_2 - \eta_D \, \|F_i - F_j\|_2 \, \|_2 \tag{4}$$

Where

$$\eta_{Src} = 1, \quad \eta_{Tgt} = \underset{\tilde{\eta}_{Tgt}}{\operatorname{Arg\,min}} L_F \left(\tilde{\eta}_{Tgt} \right) \tag{5}$$

4. EXPERIMENTAL RESULTS

In this part, we do experiments on the development dataset[10][11] in task 2. The training set is divided into source domain training set and target domain training set. The data processing method is consistent with that of task 2 auto-encoder baseline system[12].

4.1. Parameter setting

In our method, the feature extractor N_1 is designed as a two-layer feed-forward dense network with 640 neurons. The domain discriminator is a dense network with two five neuron layers. The gradient inversion layer super parameter weights the back-propagation domain discrimination loss gradient to the feature extractor parameter, which is set to 0.1.

The corresponding AUC and pAUC are calculated respectively, and then the values are compared with the baseline system based on the auto-encoder.

4.2. Experiments Summary

It can be known from table 1 and table 2 that compared with the AE method used in the baseline system, the AUC and pAUC obtained by our method are improved in different machine types. On the value dataset, the performance is improved obviously, the average AUC is increased by 15.31%, and the average pAUC is increased by 5.32%. Meanwhile, we should also aware that our method is not effective on the ToyTrain and Slider datasets, and the generalization ability of the model still needs to be improved.

5. CONCLUSION

For task 2, we propose the UADA-ASD method. According to the AUC results of the development dataset, our method is basically better than the baseline system, except for the ToyTrain and Slider datasets. In the future work, we hope to further improve the generalization ability of the model, enhance the feature alignment strategy in adversarial domain adaptation, and use the idea of generating adversarial network to enhance the sample and feature of the target domain data.

Table 1: AUC of UADA-ASD and AE method

			UADA-ASD	**Baseline(AE)**
	section	domain	AUC	AUC
ToyCar	0	source	65.98%	67.63%
	1	source	**75.32%**	**61.97%**
	2	source	42.89%	74.36%
	0	target	65.77%	54.50%
	1	target	**72.10%**	**64.12%**
	2	target	62.71%	56.57%
		arithmetic mean	64.13%	63.19%
		harmonic mean	62.00%	62.49%
ToyTrain	0	source	**88.19%**	**72.67%**
	1	source	74.47%	72.65%
	2	source	71.54%	69.91%
	0	target	38.17%	56.07%
	1	target	**61.25%**	**51.13%**
	2	target	52.91%	55.57%
		arithmetic mean	64.42%	63.00%
		harmonic mean	59.90%	61.71%
dgearbox	0	source	**84.95%**	**56.03%**
	1	source	54.81%	72.77%
	2	source	79.70%	58.96%
	0	target	70.18%	74.29%
	1	target	47.93%	72.12%
	2	target	**71.97%**	**66.41%**
		arithmetic mean	68.25%	66.76%
		harmonic mean	65.52%	65.97%
fan	0	source	72.15%	66.69%
	1	source	**82.37%**	**67.43%**
	2	source	70.42%	64.21%
	0	target	53.27%	69.70%
	1	target	**77.04%**	**49.99%**
	2	target	61.85%	66.19%
		arithmetic mean	69.52%	64.03%
		harmonic mean	68.73%	63.24%
pump	0	source	69.72%	67.48%
	1	source	**89.36%**	**82.38%**
	2	source	61.37%	63.93%
	0	target	61.96%	58.01%
	1	target	**65.58%**	**47.35%**
	2	target	50.04%	62.78%
		arithmetic mean	66.34%	63.66%
		harmonic mean	64.38%	61.92%
slider	0	source	69.73%	74.09%
	1	source	**83.70%**	**82.16%**
	2	source	81.17%	78.34%
	0	target	**57.50%**	**67.22%**
	1	target	38.08%	66.94%
	2	target	50.64%	46.20%
		arithmetic mean	63.47%	69.16%
		harmonic mean	58.82%	66.74%
value	0	source	70.92%	50.34%
	1	source	56.86%	53.52%
	2	source	**72.71%**	**59.91%**
	0	target	**78.57%**	**47.12%**
	1	target	75.07%	56.39%
	2	target	60.18%	55.16%
		arithmetic mean	69.05%	53.74%
		harmonic mean	68.10%	53.41%

Table 2: pAUC of UADA-ASD and AE method

			UADA-ASD	Baseline(AE)
	section	domain	pAUC	pAUC
ToyCar	0	source	65.11%	51.87%
	1	source	**66.84%**	**51.82%**
	2	source	47.58%	55.56%
	0	target	54.16%	50.52%
	1	target	**65.16%**	**52.14%**
	2	target	53.63%	52.61%
		arithmetic mean	57.75%	52.42%
		harmonic mean	57.81%	52.36%
ToyTrain	0	source	**66.16%**	**69.38%**
	1	source	56.32%	62.52%
	2	source	47.84%	47.48%
	0	target	49.79%	50.62%
	1	target	49.84%	48.60%
	2	target	**51.79%**	**50.79%**
		arithmetic mean	53.62%	54.90%
		harmonic mean	52.99%	53.81%
dgearbox	0	source	**70.10%**	**51.59%**
	1	source	54.14%	52.30%
	2	source	69.70%	51.82%
	0	target	63.74%	55.67%
	1	target	53.43%	51.78%
	2	target	**64.08%**	**53.66%**
		arithmetic mean	62.53%	52.80%
		harmonic mean	61.80%	52.76%
fan	0	source	51.26%	57.08%
	1	source	**79.79%**	**50.72%**
	2	source	76.58%	53.12%
	0	target	56.05%	55.13%
	1	target	64.37%	48.49%
	2	target	**73.00%**	**56.93%**
		arithmetic mean	66.84%	53.58%
		harmonic mean	65.09%	53.38%
pump	0	source	61.05%	61.83%
	1	source	**64.16%**	**58.29%**
	2	source	53.79%	55.44%
	0	target	51.79%	51.53%
	1	target	**54.47%**	**49.65%**
	2	target	50.74%	51.67%
		arithmetic mean	56.00%	54.74%
		harmonic mean	55.59%	54.41%
slider	0	source	53.84%	52.45%
	1	source	59.11%	60.29%
	2	source	**74.20%**	**65.16%**
	0	target	54.53%	57.32%
	1	target	48.86%	53.08%
	2	target	**54.41%**	**50.10%**
		arithmetic mean	57.49%	56.40%
		harmonic mean	56.52%	55.94%
value	0	source	51.00%	50.82%
	1	source	52.16%	49.33%
	2	source	**65.74%**	**51.96%**
	0	target	57.95%	48.68%
	1	target	**59.05%**	**53.88%**
	2	target	49.68%	48.97%
		arithmetic mean	55.93%	50.61%
		harmonic mean	55.40%	50.54%

6. REFERENCES

[1] I. Redko, E. Morvant, A. Habrard, M. Sebban, and Y. Bennani, *Advances in domain adaptation theory*. Elsevier, 2019.

[2] G. Michau, T. Palmé, and O. Fink, "Deep feature learning network for fault detection and isolation," in *Annual Conference of the Prognostics and Health Management Society 2017*, 2017.

[3] G. Michau, Y. Hu, T. Palmé, and O. Fink, "Feature learning for fault detection in high-dimensional condition monitoring signals," *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*, vol. 234, no. 1, pp. 104–115, 2020.

[4] G. Michau, O. Fink, and T. Palmé, "Fleet phm for critical systems: Bi-level deep learning approach for fault detection," in *Proceedings of the European Conference of the PHM Society 2018*, 2018.

[5] G. Michau and O. Fink, "Unsupervised transfer learning for anomaly detection: Application to complementary operating condition transfer," *Knowledge-Based Systems*, vol. 216, p. 106816, 2021.

[6] G. Michau and O. Fink, "Unsupervised fault detection in varying operating conditions," in *2019 IEEE International Conference on Prognostics and Health Management (ICPHM)*. IEEE, 2019, pp. 1–10.

[7] G.-B. Huang, L. Chen, C. K. Siew, *et al.*, "Universal approximation using incremental constructive feedforward networks with random hidden nodes," *IEEE Trans. Neural Networks*, vol. 17, no. 4, pp. 879–892, 2006.

[8] R. Zhang, H. Tao, L. Wu, and Y. Guan, "Transfer learning with neural networks for bearing fault diagnosis in changing working conditions," *IEEE Access*, vol. 5, pp. 14 347–14 357, 2017.

[9] X. Li, W. Zhang, and Q. Ding, "Cross-domain fault diagnosis of rolling element bearings using deep generative neural networks," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 7, pp. 5525–5534, 2018.

[10] 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," *In arXiv e-prints: 2006.05822, 1–4*, 2021.

[11] 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.

[12] 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," *In arXiv e-prints: 2106.04492, 1–5*, 2021.