# Letter

## A Simple Framework to Generalized Zero-Shot Learning for Fault Diagnosis of Industrial Processes

Jiacheng Huang, Zuxin Li, and Zhe Zhou

Dear Editor,

This letter provides a simple framework to generalized zero-shot learning for fault diagnosis. For industrial process monitoring, supervised learning and zero-shot learning (ZSL) can only deal with seen and unseen faults, respectively. However, in the online monitoring stage of the actual industrial process, both seen and unseen faults may occur. This makes supervised learning and zero-shot learning impractical in industrial process monitoring. Generalized zero-shot learning (GZSL) can handle this problem, but its implementation process is too complicated. This letter introduces GZSL into industrial process fault diagnosis, and a simple end-to-end framework is provided to implement GZSL-based fault diagnosis. In this framework, GZSL-based fault diagnosis can be realized by using only a binary classification algorithm. Experimental results show that the proposed framework can accomplish this challenging task of GZSL for fault diagnosis.

With the further advancement of intelligent manufacturing, industrial processes are becoming more complex. At this time, the fault types will become more diversified, and this place higher demands on diagnostic methods. Traditional fault diagnosis methods based on supervised learning rely on a large number of labeled fault samples, which is difficult to meet in industrial processes, especially some zero-sample, but known faults (unseen faults), which cannot be handled by supervised learning. ZSL [1] method has been proven to classify unseen faults [2], and it only has seen fault (fault types with samples and participating in model training) samples in the training stage. However, the ZSL fault diagnosis method can only classify unseen faults during the online monitoring stage, it cannot classify the cases in which both seen and unseen faults may occur, which seriously limits its practical application. To clarify the basic concepts, in Fig. 1, we divide the faults into three categories: seen fault, unseen fault, and unknown fault. The methods we discuss in this letter are all trying to solve the classification problem of seen and unseen faults, and we believe it is difficult to deal with unknown faults at present.

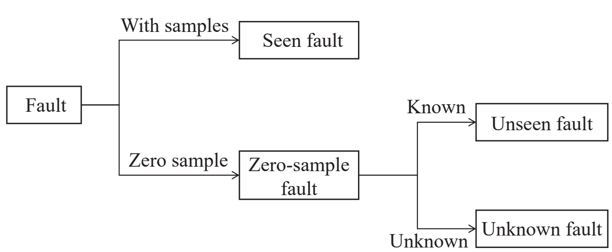


Fig. 1. The category of the fault.

Corresponding author: Zhe Zhou.

Citation: J. C. Huang, Z. X. Li, and Z. Zhou, “A simple framework to generalized zero-shot learning for fault diagnosis of industrial processes,” *IEEE/CAA J. Autom. Sinica*, vol. 10, no. 6, pp. 1504–1506, Jun. 2023.

J. C. Huang and Z. Zhou are with Huzhou Key Laboratory of Intelligent Sensing and Optimal Control for Industrial Systems, School of Engineering, Huzhou University, Huzhou 313000, China (e-mail: huangjiachenghjc@163.com; zzhou@zjhu.edu.cn).

Z. X. Li is with the School of Intelligent Manufacturing, Huzhou College, Huzhou 313000, China (e-mail: lzx@zjhu.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JAS.2023.123426

To solve the problems mentioned above, it is necessary to extend the ZSL method, which is GZSL [3], so that it can classify both seen and unseen faults during the online monitoring stage. By introducing semantic description information, GZSL can establish the connection between the disjoint seen faults and unseen faults, so that both seen faults and unseen faults can be handled during the online monitoring stage.

Semantic description information is the key to realizing GZSL, $\boldsymbol{V}_s = [v^s_{ij}]_{m\times r}$ and $\boldsymbol{V}_u = [v^u_{kj}]_{n\times r}$ are the semantic description information of seen faults and unseen faults, respectively. $v^s_{i,j}$ represents the $j$th attribute value of the $i$th seen fault and $v^u_{k,j}$ represents the $j$th attribute value of the $k$th unseen fault. $\mathbf{v}^i_s = [v^s_{i,1}, v^s_{i,2}, \ldots, v^s_{i,j-1}, v^s_{i,j}]$ is the semantic description information of $i$th seen fault. The attributes of seen and unseen faults are the same. Attribute values can come from sensors' measurements or artificial definition based on experience and are marked as one if an exception occurs or zero if not. Attributes can be semantically described, and semantic description information $\boldsymbol{V}_s$ and $\boldsymbol{V}_u$ can be obtained even if there are zero samples for unseen faults.

There is a lack of studies on ZSL/GZSL in the field of fault diagnosis, and most of them are concentrated in the field of rolling bearing fault diagnosis [4]–[8] and a few of them are for industrial process [9] and [10]. Feng and Zhao [2] first introduced ZSL into industrial process fault diagnosis. Our previous work transformed GZSL into ZSL and supervised learning through a domain discriminator [9]. However, our previous implementation of GZSL contains many learning steps and is too complex to implement. In addition, the existing ZSL and GZSL have the problem of the semantic bias because the artificial definition of semantic description information is too subjective, which can reduce the classification accuracy.

To this end, a simple end-to-end framework to GZSL for fault diagnosis of industrial processes is proposed. In this framework, GZSL can be realized only with a binary classification algorithm. To solve the semantic bias problem that the artificial definition of semantic description information subjectivity is too strong. This letter also develops an approach to correct the semantic description information of seen faults by predicting the error rate of attributes. Experiments on Tennessee-Eastman process show the effectiveness of the proposed method.

**Proposed framework:**

**GZSL problem formulation:** In this letter, the training set of GZSL is denoted by $D_{\text{train}} = \{(x^i_s, v^i_s, y^i_s)^{N_s}_{i=1} | x^i_s \in X_s, v^i_s \in V_s, y^i_s \in Y_s\}$, where $x^i_s \in \mathbb{R}^D$ represents the $D$-dimensional features; $v^i_s \in \mathbb{R}^K$ represents the semantic description information of seen faults; $Y_s = \{y^1_s, \ldots, y^{C_s}_s\}$ is the label set of the seen fault; $C_s$ is the number of seen faults.

The label set of unseen faults is represented by $Y_u = \{y^1_u, \ldots, y^{C_u}_u\}$, where $C_u$ is the number of unseen faults. The whole set of seen and unseen faults can be expressed as $Y = Y_s \cup Y_u$, where $Y_s \cap Y_u = \emptyset$. The aim of GZSL is to learn a model that can classify the online samples from both seen and unseen faults.

**Domain discriminator:** Although supervised learning and zero-shot learning can deal with seen and unseen faults, respectively, we cannot know in advance whether there are seen or unseen faults during online monitoring. If a domain discriminator can be designed to realize the identification of seen and unseen fault samples before classification, GZSL can be realized. However, it is difficult to train an effective domain discriminator because no unseen fault samples can be used for model training. In this letter, the domain discrimination is achieved by judging whether the predicted attribute vector of the online sample belongs to the semantic description information of seen fault. If the output attribute vector belongs to one of the semantic description information of seen faults, it is considered that the sample belongs to seen faults; otherwise, it belongs to unseen faults. The diagram of GZSL implementation based on domain discrimination is shown in Fig. 2.

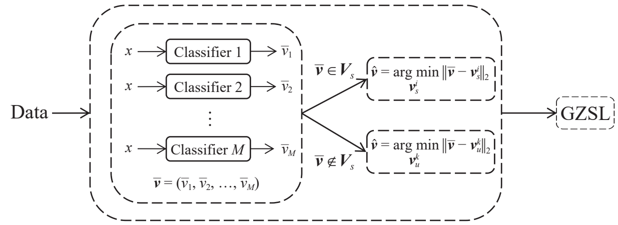


Fig. 2. The realization of GZSL based on domain discriminator.

**Semantic correction:** Binary semantic description information is used as auxiliary information, and the marking of its attributes is often affected by the subjectivity of the marker, which is called semantic bias.

To solve the semantic bias problem, a method was proposed to correct the semantic description information of the seen faults by predicting the error rate of attributes, so as to reduce the influence of human subjectivity on the results and improve the classification accuracy of the model.

$\mathbf{E} = [e_{ij}]_{m\times r}$ is the attribute prediction rate matrix obtained from the trained model with the input of the seen fault data that participating in the model training. $e_{i,j}$ represents the prediction error rate of the $j$th attribute of the $i$th seen fault. If $e_{i,j} > \lambda$, the value of $v_{i,j}$ will be modified according to (1)

$$\hat{v}_{i,j}^s = v_{i,j}^s \oplus I(e_{i,j} > \lambda) \tag{1}$$

where $I(\cdot)$ represents the characteristic function, $\oplus$ represents the exclusive OR; $\lambda$ is the threshold for correcting attributes. $\hat{V}_s$ is the modified semantic description matrix of seen fault.

**End-to-end framework overview:**

1) Prepare $j$ binary classifiers according to the number of attributes in the semantic description information (if an attribute is all 0 or all 1, delete the attribute directly).

2) In the training stage, $j$ binary classifiers are trained with seen fault samples and their corresponding semantic description information, and the label of each binary classifier is an attribute value in the corresponding semantic description information of the samples.

3) Input the training samples into the trained $j$ binary classifiers to obtain the attribute vector output by each sample. If the binary classifiers output continuous values, discretize the output results (if the value is greater than 0.5, it becomes 1; otherwise, it becomes 0).

4) Calculate the error rate for each attribute of each fault $e_{i,j}$. Correct the attributes of the seen fault according to (1).

5) Train the above $j$ binary classifiers again with the new semantic description $\hat{V}_s$ (if an attribute is all 0 or all 1 after correction, delete the attribute directly).

6) In the test stage, the predicted attribute vector of online samples are provided by those $j$ binary classifiers trained in Step 5.

7) If the predicted attribute vector (discretized if the output is continuous value) is the same as one of the semantic description of the seen fault, the nearest neighbor search is carried out between the original vector output of the binary classifier and the semantic description of the seen fault; otherwise, the nearest neighbor search is carried out between the original vector output of the binary classifier and the semantic description of the unseen fault.

**Main results:** The end-to-end (e2e) GZSL framework proposed in this letter uses Naive Bayes (NB), linear support vector machine (LSVM), RandomForest (RF), Adaboost (Ada), Gradient Boosted Decision Trees (GBDT), and fully connected Neural Network (NN) to conduct the experiments on Tennessee-Eastman process [11]. The fault semantic description information and the division of seen/unseen faults are the same as [2]. The proposed method is verified in groups A, B, C, D and E respectively according to the five different division of seen/unseen faults.

$A_s$ and $A_u$ are the average classification accuracy of the seen fault and unseen fault, respectively. $A$ represents the overall classification accuracy. The harmonic mean $H = 2\times(A_s\times A_u)/(A_s+A_u)$.

**Impact of semantic correction on results:** Fig. 3 shows the matrix of attribute prediction error rate on group A by using the NN classifier. The semantic description information of the seen fault can be corrected by (1).

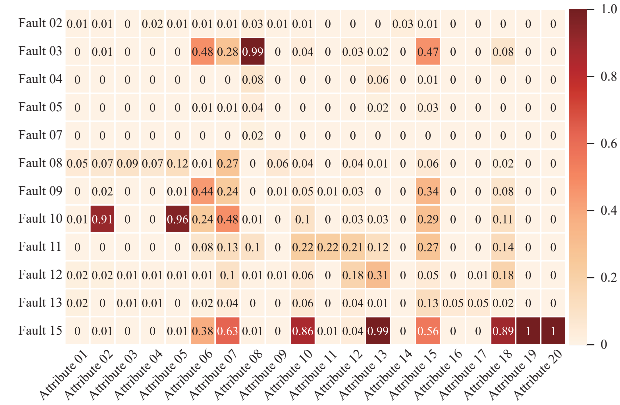


Fig. 3. Attribute prediction error rate.

Table 1 shows the overall attribute prediction error rate when the attribute classifier is NN, where $\lambda$ is 0.6, 0.7, 0.8, 0.9, and 1.0, respectively. When $\lambda$ is 1.0, the attribute will not be corrected. If $\lambda$ is less than 0.5 for attribute correction, its attribute prediction error rate will increase.

Table 1. Effects of Different Correction Thresholds on Attribute Error Rate

| $\lambda$ | A | B | C | D | E |
|---|---|---|---|---|---|
| 0.6 | 3.00 | 2.41 | 2.11 | 1.31 | 1.75 |
| 0.7 | 3.26 | 3.29 | 2.11 | 2.57 | 2.49 |
| 0.8 | 3.26 | 3.71 | 2.11 | 2.86 | 3.14 |
| 0.9 | 4.02 | 4.23 | 3.43 | 3.29 | 3.48 |
| 1.0 | 7.31 | 7.26 | 5.88 | 6.78 | 5.30 |

It can be seen from Table 1 that attribute correction can effectively reduce the error rate of attribute prediction. The attribute prediction error rate gradually increases with the increase of $\lambda$.

**Final classification results:** This section shows the performance of the proposed framework over a variety of binary algorithms and $\lambda = 0.9$. The effect of different $\lambda$ values on the results is discussed below. Tables 2 and 3 show the results of 5 groups.

In Table 4 shows the average of the final results of 5 groups, all the other methods except e2e-NB and e2e-LSVM can achieve good accuracy, but the harmonic mean of e2e-NN is much higher than other methods which is due to the multi-nearest neighbor problem,

Table 2. Classification Results

| Methods | A | | B | | C | | D | | E | |
|---|---|---|---|---|---|---|---|---|---|---|
| | $A_s$ | $A_u$ | $A_s$ | $A_u$ | $A_s$ | $A_u$ | $A_s$ | $A_u$ | $A_s$ | $A_u$ |
| e2e-NB | 22.53 | 34.51 | 31.32 | 5.35 | 39.93 | 14.37 | 21.20 | 11.32 | 32.17 | 5.62 |
| e2e-LSVM | 38.32 | 8.96 | 33.26 | 0 | 42.95 | 8.89 | 30.99 | 2.57 | 39.76 | 20.28 |
| e2e-RF | 48.45 | 8.47 | 52.73 | 19.44 | 60.05 | 21.60 | **57.20** | 51.53 | **73.40** | 25.56 |
| e2e-Ada | 43.72 | 20.07 | 49.38 | 22.99 | **63.25** | 25.83 | 51.84 | 31.32 | 66.06 | 29.10 |
| e2e-GBDT | 47.22 | 17.92 | **57.10** | 27.22 | 63.21 | 19.24 | 56.96 | 27.71 | 72.86 | 19.79 |
| e2e-NN | **49.22** | **74.72** | 51.55 | **34.86** | 61.93 | **35.62** | 49.97 | **56.18** | 58.39 | **36.94** |

Table 3. Results of $A$ and $H$

| Methods | A | | B | | C | | D | | E | |
|---|---|---|---|---|---|---|---|---|---|---|
| | $A$ | $H$ | $A$ | $H$ | $A$ | $H$ | $A$ | $H$ | $A$ | $H$ |
| e2e-NB | 24.93 | 27.26 | 26.13 | 9.14 | 34.82 | 21.13 | 19.22 | 14.76 | 26.86 | 9.57 |
| e2e-LSVM | 32.45 | 14.52 | 26.61 | 0 | 36.14 | 14.73 | 26.31 | 4.75 | 35.86 | 26.86 |
| e2e-RF | 40.45 | 14.42 | 46.07 | 28.41 | 52.36 | 31.77 | **56.07** | **54.22** | **63.83** | 37.92 |
| e2e-Ada | 38.99 | 27.51 | 44.10 | 31.37 | 55.77 | 36.68 | 47.74 | 39.05 | 58.67 | 40.40 |
| e2e-GBDT | 41.36 | 25.98 | **51.12** | 36.87 | 54.42 | 29.50 | 51.11 | 37.28 | 62.25 | 31.13 |
| e2e-NN | **54.32** | **59.35** | 48.21 | **41.59** | **56.67** | **45.23** | 51.21 | 52.89 | 54.10 | **45.25** |

Table 4. The Average of Final Results of Five Group

| | e2e-NB | e2e-LSVM | e2e-RF | e2e-Ada | e2e-GBDT | e2e-NN |
|---|---|---|---|---|---|---|
| $A$ | 26.39 | 31.27 | 51.76 | 49.05 | 52.05 | **52.90** |
| $H$ | 16.37 | 12.17 | 33.35 | 35.00 | 32.15 | **48.86** |

and we will be analyzed in the next section.

**Multi-nearest neighbor problem:** It can be seen from the previous section that the harmonic mean of NN is much higher than other methods. Due with the NN output continuous values, it is almost impossible to have multiple nearest neighbors that we can see result in Table 5.

Table 5. The Proportion of Multi-Nearest Neighbors

| | A | B | C | D | E |
|---|---|---|---|---|---|
| e2e-NB | 22.49 | 20.62 | 34.22 | 29.67 | 19.46 |
| e2e-LSVM | 13.56 | 10.71 | 40.62 | 8.00 | 10.67 |
| e2e-RF | 17.21 | 13.21 | 24.62 | 12.56 | 8.49 |
| e2e-Ada | 13.33 | 5.88 | 14.85 | 7.08 | 13.60 |
| e2e-GBDT | 7.10 | 3.32 | 18.19 | 7.04 | 8.75 |
| e2e-NN | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |

**The influence of different thresholds on the results:** In Table 6, when the binary classifier is NN. As $\lambda$ gradually increases, for most groups $A_s$ gradually decreases, and $A_u$ gradually increases. After attribute correction, the lower the attribute prediction error rate, the higher the $A_s$. Therefore, the intermediate value of 0.7 or 0.8 for $\lambda$ should be selected.

Table 6. The Influence of Different Thresholds on the Final Results

| $A_s$ | A | | B | | C | | D | | E | |
|---|---|---|---|---|---|---|---|---|---|---|
| | $A_s$ | $A_u$ | $A_s$ | $A_u$ | $A_s$ | $A_u$ | $A_s$ | $A_u$ | $A_s$ | $A_u$ |
| 0.6 | **51.37** | **76.25** | **56.18** | 23.06 | **65.09** | 32.15 | **56.18** | 36.60 | **67.45** | 11.46 |
| 0.7 | 50.47 | 74.86 | 53.85 | 39.03 | 65.09 | 32.15 | 54.64 | 47.15 | 61.68 | 24.79 |
| 0.8 | 50.47 | 74.86 | 53.49 | 34.58 | 65.09 | 32.15 | 50.35 | 52.22 | 59.22 | 35.90 |
| 0.9 | 49.22 | 74.72 | 51.55 | **34.86** | 61.93 | **35.62** | 49.97 | **56.18** | 58.39 | **36.94** |

**Shortcomings of the end-to-end GZSL framework:** The confusion matrix of the final classification results for group A of the e2e-NN method is shown in Fig. 4, which is the best result of all methods. Since no samples of unseen fault for model training and the conditions for identifying online samples as seen faults are very strict, the domain discriminator of the proposed method has a high probability of identifying the seen faults as unseen faults.

**Conclusion:** In this letter, a simple end-to-end framework to GZSL for fault diagnosis of industrial processes is proposed, which only needs a binary classification algorithm to achieve GZSL. It can be found that this framework performs well with most binary classification algorithms. Aiming at the problem that artificially defined semantic description information is too subjective, the proposed method can realize the correction of the semantic description information of seen faults. Finally, the multi-nearest neighbor problem is found, and experiments show that the neural network with continuous output value does not have the multi-nearest neighbor problem in the final fault classification, which makes the final performance better than other binary classification algorithms.

The quality of the samples generated by the variational autoencoder has a significant impact on the performance of the previous proposed method [9], and its training is not easy. Hence, designing a model that is easier to train and has better quality of the generated samples is expected to further improve the performance of the proposed method.

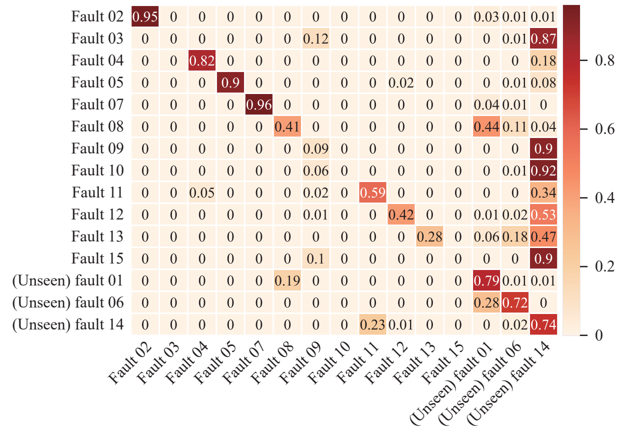


Fig. 4. The confusion matrix of the final classification result of the e2e-NN method on group A.

**Acknowledgments:** This work was supported by the National Natural Science Foundation of China (61703158), the Zhejiang Provincial Public Welfare Technology Application Research Project (LGG22F030023, LTGS23F030003), and the Huzhou Municipal Natural Science Foundation of China (2021YZ03).

## References

[1] C. H. Lampert, H. Nickisch, and S. Harmeling, "Learning to detect unseen object classes by between-class attribute transfer," in *Proc. IEEE Conf. CVPR*, Miami, USA, 2009, pp. 951–958.

[2] L. Feng and C. Zhao, "Fault description based attribute transfer for zero-sample industrial fault diagnosis," *IEEE Trans. Ind. Inf.*, vol. 17, no. 3, pp. 1852–1862, 2021.

[3] W.-L. Chao, S. Changpinyo, B. Gong, and F. Sha, "An empirical study and analysis of generalized zero-shot learning for object recognition in the wild," in *Proc. ECCV'16*, Amsterdam, The Netherlands, 2016, pp. 52–68.

[4] H. Lv, J. Chen, T. Pan, and Z. Zhou, "Hybrid attribute conditional adversarial denoising autoencoder for zero-shot classification of mechanical intelligent fault diagnosis," *Appl. Soft Comput.*, vol. 95, p. 106577, 2020.

[5] J. Xu, L. Zhou, W. Zhao, Y. Fan, X. Ding, and X. Yuan, "Zero-shot learning for compound fault diagnosis of bearings," *Expert Syst. Appl.*, vol. 190, p. 116197, 2022.

[6] Y. Gao, L. Gao, X. Li, and Y. Zheng, "A zero-shot learning method for fault diagnosis under unknown working loads," *J. Intell Manuf*, vol. 31, no. 12, pp. 899–909, 2019.

[7] A. He and X. Jin, "Deep variational autoencoder classifier for intelligent fault diagnosis adaptive to unseen fault categories," *IEEE Trans. Reliab.*, vol. 70, no. 4, pp. 1581–1595, 2021.

[8] S. Xing, Y. Lei, S. Wang, N. Lu, and N. Li, "A label description space embedded model for zero-shot intelligent diagnosis of mechanical compound faults," *Mech. Syst. Sig. Process.*, vol. 162, p. 108036, 2022.

[9] J. Huang, Z. Li, L. Ye, and Z. Zhou, "Fault classification of industrial processes based on generalized zero-shot learning," in *Proc. IEEE 10th DDCLS Conf.*, Suzhou, China, 2021, pp. 887–892.

[10] Y. Zhuo and Z. Ge, "Auxiliary information-guided industrial data augmentation for any-shot fault learning and diagnosis," *IEEE Trans. Ind. Inf.*, vol. 17, no. 11, pp. 7535–7545, 2021.

[11] J. Downs and E. Vogel, "A plant-wide industrial process control problem," *Comput. Chem. Eng.*, vol. 17, no. 3, pp. 245–255, 1993.