Transactions of the Korean Nuclear Society Spring Meeting
Jeju, Korea, May 9-10, 2024

A Study on an Adaptable Deep Learning Algorithm for Dynamic Environments using CycleGAN

Doyeob Yeo[a*], Hyunseok Lee[b]
[a]*Nuclear System Integrity Sensing & Diagnosis Division, Korea Atomic Energy Research Institute (KAERI), Daejeon, Korea*
[b]*Daegu-Gyeongbuk Medical Innovation Foundation, Daegu, Korea*
[*] *Corresponding author: yeody@kaeri.re.kr*

***Keywords : leak detection, adaptable, dynamic environments, CycleGAN**

1. Introduction

Recently, as the demand for integrating deep learning technology into the anomaly detection of plant pipelines has risen, there is a growing emphasis on collecting high-quality data from operational plants for training deep learning models. However, in actual plant environments, anomalies rarely occur, making it challenging to collect data on abnormal conditions. Additionally, creating artificial abnormal conditions in real plant environments is difficult due to the high probability of significant damage occurring when anomalies do happen. For these reasons, experimental data is collected in a testbed environment that simulates real plant conditions for the development of deep learning models. Furthermore, research on applying deep learning models developed in the testbed to real-world scenarios has become increasingly important.

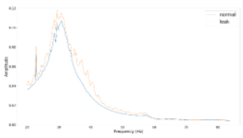
Recently, there have been numerous efforts to develop deep learning algorithms that can be utilized in environments where obtaining real-world data is challenging, by simulating real environment data using simulators. However, when using self-developed sensors instead of commercially available sensors to collect time-series data, simulating data using simulators becomes very challenging.

Therefore, there is a need to develop an adaptable deep learning algorithm for dynamic environments based entirely on collected data. This paper introduces research on an adaptable deep learning algorithm for dynamic environments using CycleGAN. The algorithm utilizes CycleGAN to learn a transformation from normal state data to leak state data and generate leak state data from normal state data in a new environment.

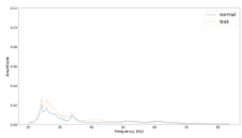
2. Necessity of Adaptable Deep Learning Algorithm for Dynamic Environments

The Korea Atomic Energy Research Institute (KAERI) is currently developing a wireless acoustic sensor module designed to detect micro-leaks in plant pipelines [1]. Once the sensor collects data, it undergoes several signal preprocessing steps and is subsequently analyzed using a deep learning model to identify the presence of leaks in the pipeline. In this process, data are collected in the frequency domain by the sensor module.

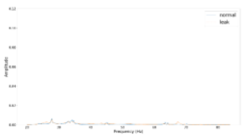
When training the deep learning model with learning data collected in a testbed located in the laboratory, validation in different environments exposes its incapacity to detect leaks. This is attributed to significant variations in the collected data based on the collection environment (time, temperature, ambient conditions, etc.), as illustrated in Figure 1. In Figure 1, there are differences even among normal state data (blue solid line) depending on the environment, as well as variations among leak state data (orange dashed line). Furthermore, there are differences between normal state data and leak state data, attributed to variations in factors such as leak pressure and the distance between the leakage area and the sensor.



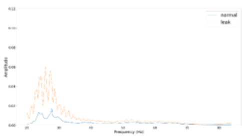
(a) Environment A



(b) Environment B



(c) Environment C



(d) Environment D

Fig. 1. Data collected in various environments (A~D) using a wireless acoustic sensor module. The average data collected over 10 minutes are illustrated. The orange dashed line represents data collected during a leak state, while the blue solid line represents data collected during a normal state.

Table I represents the classification accuracy of deep learning model training and inference using data in the four environments shown in Figure 1. Supervised learning techniques, specifically classification methods, were employed for training deep learning models.

As shown in Table I, classification performance decreases when the training environment differs from the inference environment, except in the case where a model trained with data from environment B is inferred with data from environment D. In these cases, the classification models, instead of randomly determining normal or leak during inference, consistently classified all data as either entirely normal or entirely leak. The data from environment C showed lower inference performance compared to other environments, as the number of normal state data samples and leak state data samples had an approximate ratio of 2:1.

Table I. Classification accuracy (%) in various collecting environments (A~D). The classification performance was measured by varying the environments for both training data collection and inference data collection.

Training / Inference	A	B	C	D
A	99.92	50.25	49.32	49.38
B	46.06	99.70	54.43	46.10
C	32.35	32.35	99.28	31.80
D	46.94	99.93	51.70	99.91

3. Adaptable Deep Learning Algorithm for Dynamic Environments using CycleGAN

CycleGAN [2] is a model derived from Generative Adversarial Network (GAN) [3], designed to learn the translation of an image from a source domain to a target domain in the absence of paired examples. While GAN requires paired data for training, CycleGAN can be trained even when paired data is not available.

Generally, collecting normal state data is easier than collecting leaked state data. Therefore, the objective is to use GAN to learn transformation maps from normal state data to leak state data. This allows the generation of leak state data from normal state data in a new environment, enabling training using classification methods. As the normal state data and leak state data are not in a 1:1 correspondence, CycleGAN, capable of learning from unpaired data, is employed to train the transformation.

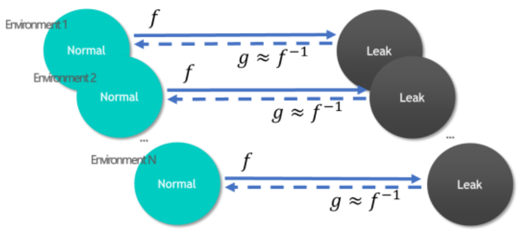


Fig. 2. Proposed CycleGAN training procedure.

The algorithm proposed in this paper is as follows:

1. As illustrated in Figure 2, normal and leak state data collected across various environments are paired according to their respective environmental conditions. It is not necessary for the number of samples of normal and leak state data from the same environment to be equal.
2. Train generators and discriminators of CycleGAN using the dataset to obtain the transformation from the normal state to leak state, denoted as f. It is essential to train pairs of normal state data and leaked state data collected from the same environment.
3. Generate leak state data by applying the transformation f to the collected normal state data in a new environment. The number of generated leaked state data samples equals the number of normal state data samples in the new environment.
4. Train a classification model for the new environment using the normal state data and generated leak state data in step 3. Utilize this model as the leak detection model in the new environment.

4. Experimental Results

The algorithm proposed in Chapter 3 was implemented using Python 3.8 and PyTorch 1.12.1. All generators and discriminators within the CycleGAN architecture adopt a 1D-CNN structure, with the generators being modified versions of the U-Net structure. The training process utilized data collected from two environments, A and C of Figure 1. The trained transformation was validated using data collected from environment B. The number of data samples used for training in each environment is detailed below in Table II.

Table II. Environments and the quantity of data samples utilized in CycleGAN training.

	Environment A	Environment B
Normal	1001	778
Leak	1001	363

a. Validation of Trained Transformation using Training Environment Data (Environment A)

In Figure 3, the first row represents the leak state data from the environment A, and the second row shows the leak state data generated from the normal state data in environment A and the trained transformation. In the figure, the x-axis represents Frequency, and the y-axis represents amplitude values. As shown in Figure 3, the collected leak state data and the generated leak state data are similar.

Figure 4 illustrates the clustering results of collected leak data and generated leak data using t-distributed Stochastic Neighbor Embedding (t-SNE) [4]. If the generated data is well-created and indistinguishable from real data, the two sets of data should not form separate clusters but rather be interspersed. In Figure 4, the two sets of data are intermingled, confirming that the generated data is well-produced.

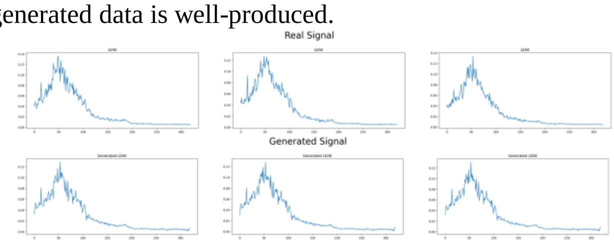


Fig. 3. The first row represents the leak state data from the environment A, and the second row shows the leak state data generated from the normal state data in environment A and the trained transformation

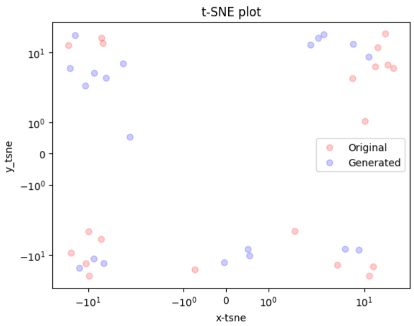


Fig. 4. Clustering results of collected leak data and generated leak data using t-SNE.

b. Validation of Trained Transformation using Non-Training Environment Data (Environment B)

In Figure 5, the first row represents the leak state data from the environment B, and the second row shows the leak state data generated from the normal state data in environment B. In Figure 5, the collected leak state data and the generated leak state data are different.

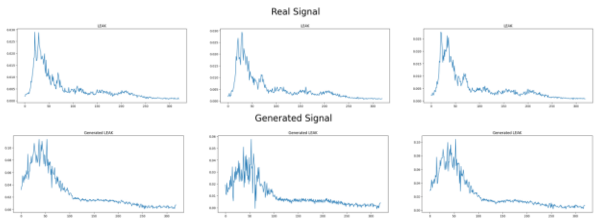


Fig. 5. The first row represents the leak state data from the environment B, and the second row shows the leak state data generated from the normal state data in environment B.

Through the experiments, it was confirmed that the leak data from the environment used for training was well-generated. However, leak data from an environment not used for training was observed not to be as well-generated.

5. Conclusion

In this paper, we proposed an adaptable deep learning algorithm for dynamic environments using CycleGAN. The goal was to learn transformation from normal state data to leak state data using data from various environments, and subsequently use this transformation to generate leak state data from normal state data in a new environment. Finally, we aimed to learn a classification model for the new environment using the generated leak state data and normal state data. From experiments, it was verified that leak state data was well-generated in the environment used for training; however, it was not as well-generated in an environment not used for training. In the experiments, CycleGAN was trained using data collected from two environments only. However, it is necessary to train the transformation using data collected from a more extensive range of environments. Furthermore, it is suggested that the training of the transformation could be further improved by incorporating factors that distinguish the data collection environment, enabling the transformation to learn information about the environment.

ACKNOWLEDGEMENT

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (Ministry of Science and ICT) (No. RS-2022-00144000).

REFERENCES

[1] Yeo, D., Lee, G., & Lee, J. C. (2020). Pipe Leak Detection System using Wireless Acoustic Sensor Module and Deep Auto-Encoder. *Journal of the Korea Society of Computer and Information*, 25(2), 59-66.
[2] Zhu, J. Y., Park, T., Isola, P., & Efros, A. A. (2017). Unpaired image-to-image translation using cycle-consistent adversarial networks. *In Proceedings of the IEEE international conference on computer vision* (pp. 2223-2232).
[3] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2014). Generative adversarial nets. *Advances in neural information processing systems*, 27.
[4] Van der Maaten, L., & Hinton, G. (2008). Visualizing data using t-SNE. *Journal of machine learning research*, 9(11).