Performance Improvement of Abnormal Condition Diagnosis Model Using Data Augmentation Methods in Nuclear Power Plants

Ji Woo Hong, Sang Won Oh, and Man Gyun Na*
Dept. of Nuclear Engineering, Chosun Univ., 10, Chosundae 1-gil, Dong-gu, Gwangju, 61452
**Corresponding author: magyna@chosun.ac.kr*

1. Introduction

Actual data from nuclear power plants (NPPs) is limited to the collection. Accordingly, in most studies [1, 2], data are collected through simulators. The simulator data are bound to be different from the real data, and the amount of data is limited.

When a model performs well on the training data but poorly on the test data, it is called overfitting [3]. Insufficient data causes overfitting. A sufficient amount of data is essential for successful model training.

A proposed method to solve this problem is data augmentation. Data augmentation means artificially generated data while preserving existing data [4]. Data augmentation not only expands the data set but also increases the diversity of the data set. Data augmentation helps to make better the overall performance of the model [5]. Data augmentation on limited data can make better the scale and quality of the training data set and build better deep learning models. It also enhances the performance of the model on untrained data to prevent overfitting [6].

Therefore, this study proposes a data augmentation method for applying NPP data and performs classification tasks using LSTM. For the diagnosis of 12 scenarios, five data augmentation methods are adjusted to the time-series data set and evaluated.

The goal of this paper is to help select which technique is most effective when augmenting NPP data consisting of time-series data.

2. Methods

2.1 Data Augmentation

When training a deep learning model, it is essential to have an adequate amount of training data set. However, in reality, it is not easy in terms of time and cost. Using data augmentation techniques, a model can be successfully trained with a small amount of data.

Data augmentation is a technique for making new data based on the original data set. Although image data augmentation methods are already known, cases of time-series augmentation are relatively few. Image data augmentation methods include 1) geometric transformation, 2) flipping, 3) color space, 4) cropping, 4) rotation, 5) translation, 6) noise injection, and 7) color space transformations [6]. Most time-series data augmentation methods are inspired by image data augmentation methods. Similar to image augmentation methods, time-series data augmentation is performed through random transformations such as adding random noise to the training data, slicing or scaling, and warping. However, since time-series data have properties different from images, not all image augmentation methods may be applied to time-series data. Fig. 1 is a visualization of each augmentation method.

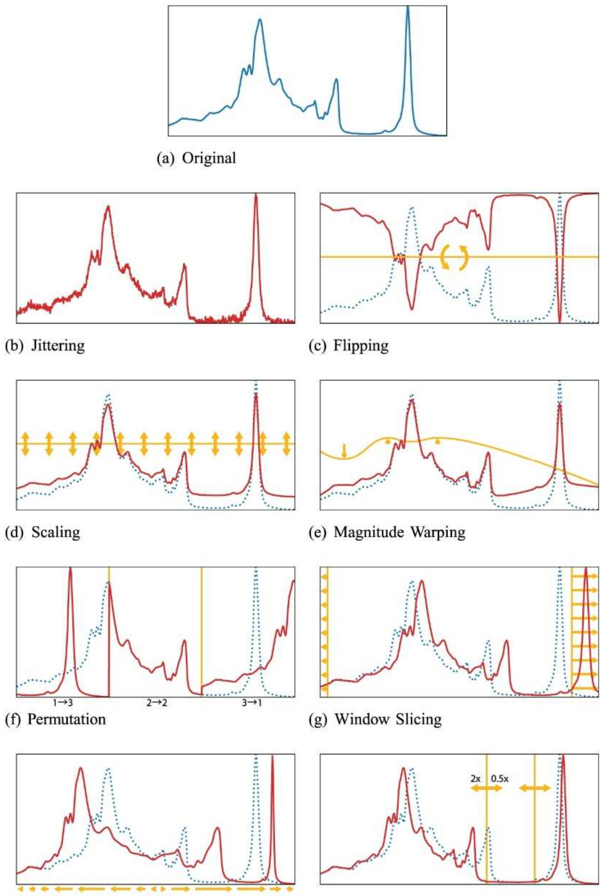

Fig. 1. Visualizing and explaining data augmentation [8].

2.1.1 Jittering

Adding noise to data is the best-known data augmentation method. Jittering is a method of adding a small amount of noise or outliers into the original data. The standard deviation of the noise following a Gaussian distribution serves as a hyperparameter. ε is the Gaussian noise added to each time step t and $\theta \sim N(0,\sigma^2)$. The standard deviation of the noise is a hyperparameter. This data augmentation method provides robust characteristics in spite of noise addition and enhances performance [7]. Jittering is calculated by Eq. (1).

$$x' = x_1 + \varepsilon_1, \cdots, x_t + \varepsilon_t, \cdots, x_T + \varepsilon_T, \quad (1)$$

2.1.2 Rotation

Rotation in multivariate time-series data means an arbitrary rotation matrix according to an angle [8]. The degree of rotation acts as a hyperparameter. R is the rotation matrix for angle $\theta \sim \mathrm{N}(0, \sigma^2)$. The stability of rotation is determined by the degree of rotation. It should be noted that the original data is not preserved if the degree of rotation is increased excessively [7]. The rotation equation is expressed as Eq. (2).

$$x' = Rx_1, \cdots, Rx_t, \cdots, Rx_T, \quad (2)$$

2.1.3 Scaling

Unlike scaling in image data, in time-series data, the size of each element is adjusted with a random scalar value, rather than expanding the data. It is created by multiplying all elements of data by a scaler, that is, an arbitrary scalar value α, which follows a Gaussian distribution $\alpha \sim \mathrm{N}(1, \sigma^2)$ and acts as a hyperparameter [7]. The scaling equation is expressed as Eq. (3).

$$x' = \alpha x_1, \cdots, \alpha x_t, \cdots, \alpha x_T, \quad (3)$$

2.1.4 Magnitude Warping

The size of each data is changed through a convolution operation between the data window and a smooth curve that changes around 1. The size of each time-series data is multiplied by the curve generated by the number of knots set to an arbitrary size. $\beta_1, \cdots, \beta_t, \cdots, \beta_T$ is a sequence produced by interpolating a cubic spline $S(u)$. Each knot u is taken from a distribution $\mathrm{N}(1, \sigma^2)$ [7]. The number of knots acts as a hyperparameter [9]. The equation of magnitude warping is expressed as Eq. (4).

$$x' = \beta_1 x_1, \cdots, \beta_t x_t, \cdots, \beta_T x_T, \quad (4)$$

2.1.5 Window Warping

The window warping is implemented by warping the original data of each activity by speeding it up or down [10].

Fig. 2 is a visual representation of the data applied with various augmentation methods compared with the original data.

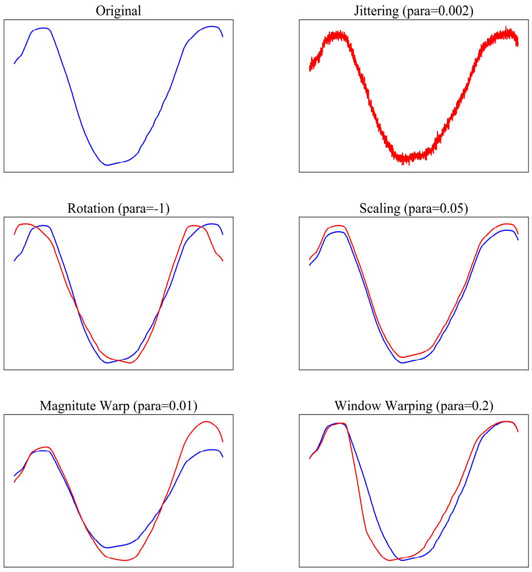

Fig. 2. Results of data augmentation methods

2.2 Long Short-Term Memory (LSTM)

LSTM is a recurrent neural network model suggested to solve the long-term dependency problem of RNNs. LSTM is a method in which forget, input, and output gates are added to the memory cell of the hidden layer. This deletes unnecessary memories and decides what to remember. At each point in time, the information is deleted or retained, selectively passing data [11].

The forget gate determines the information to be deleted from the cell state through the sigmoid layer. The forget gate is calculated as Eq. (5).

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (5)$$

The input gate determines the information to be stored in the cell state among the new input information. The input gate is calculated as Eq. (6).

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (6)$$

The current cell state updates the information to be forgotten and the information to be stored. The previous cell state is passed to the next cell state. The cell state is calculated as Eq. (7).

$$C_t = f_t * C_{t-1} + i_t * C'_t \quad (7)$$

Finally, the output information is determined and sent to the output gate. The output gate is calculated as Eq. (8).

$$o_t = \sigma\left(W_o\left[h_{t-1}, x_t\right] + b_o\right) \quad (8)$$

3. Data collection

The compact nuclear simulator (CNS) is a simulator designed based on the Westinghouse pressurized light water reactor NPP. 38,041 training data were collected through the CNS. Five different parameters were applied to each augmentation method. The amount of the augmented data increased 5 times compared to the original data. A total of 228,246 training data were generated by synthesizing the original data and the augmented data.

The model was trained by generating a new training set by synthesizing the original data with the generated data using the data augmentation methods.

All trainings were conducted in the same environment using the model of the same structure. We designed the LSTM model to run for 500 epochs using Adam optimization. By applying the early stopping, training can be stopped when the model performance is most optimal.

4. Result

By performing training and testing through the LSTM model, it is possible to look into the result of the data augmentation method on the classification model accuracy.

Balanced accuracy was used as an indicator to evaluate performance. The balanced accuracy is used to prevent exaggerated performance estimation when evaluating the performance of a classification model including unbalanced data. The balanced accuracy is calculated as Eq. (9).

$$\begin{aligned} balanced\_accuracy &= \frac{1}{2}(Sensitivity + Specificity) \\ Sensitivity &= \frac{TP}{TP+FN} \\ Specificity &= \frac{TN}{TN+FP} \end{aligned} \quad (9)$$

Sensitivity (i.e., True Positive(TP) rate) is the probability that a positive case is correctly forecast as positive, and Specificity (i.e., True Negative(TN) rate) is the probability that a negative case is correctly forecast as negative [12].

As a result of the experiment, it can be confirmed that the validation data set and the test data set show higher performance than the model trained with the original data set. It shows an accuracy improvement of at least 2% or more. Table I shows the accuracy of the model trained with the data to which each augmentation technique is applied.

Table I: Classification model performance results using various data augmentation methods

	Accuracy(%)	Balanced Accuracy(%)
Original	92.61	93.86
Jittering	98.24	95.56
Rotation	97.81	96.98
Scaling	99.27	97.16
Magnitude Warping	**99.36**	**98.85**
Window Warping	95.58	94.58

5. Conclusions

This study proposes an effective data augmentation technique appropriate for time-series NPP data using five data augmentation methods: jittering, rotation, scaling, magnitude warping, and window warping. Training and testing were conducted using LSTM, and the performance improvement of the classification model was tested by calculating the balanced accuracy. As a result, it was shown that data augmentation helps to enhance the generalization ability of the model and the overall performance of the model. In addition to the data augmentation method proposed in this study, there are also data generation models such as MODALS [13], Deep Autoregressive Networks (DARN), VAE [14], and GAN [15]. In future studies, the comparative evaluation of the performance of the data generation model and the data augmentation technique is also expected to be significant.

Acknowledgment

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. RS-2022-00144239).

REFERENCES

[1] J. Hou, K. Ni, and A. Hawari, An Artificial Neural Network Based Anomaly Detection Algorithm for Nuclear Power Plants, Proceedings of American Nuclear Society, June 9-13, 2019, Minneapolis, Minnesota.

[2] M. D. Wang, T. H. Lin, K. C. Jhan, and S. C. Wu, Abnormal event detection, identification and isolation in nuclear power plants using LSTM networks, Progress in Nuclear Energy, Vol.140, pp.103928, 2021.

[3] Ying. X, An Overview of Overfitting and its Solutions, Journal of physics: Conference series. Vol.1168. No. 2. IOP Publishing, 2019.

[4] L. Taylor, G. Nitschke, Improving Deep Learning with Generic Data Augmentation, Progress in 2018 IEEE Symposium Series on Computational Intelligence (SSCI), pp.1542-1549, 2018.

[5] M Banko, E. Brill, Scaling to Very Very Large Corpora for Natural Language Disambiguation, Proceedings of the 39th annual meeting of the Association for Computational Linguistics, July 6-11, 2001, Toulouse France.

[6] C. Shorten, T. M. Khoshgoftaar, A survey on Image Data Augmentation for Deep Learning, Progress in Journal of Big Data, Vol.6, pp.1-48, 2019.
[7] T. T. Um, F. M. J. Pfister, D. Pichler, S. Endo, M. Lang, S. Hirche, U. Fietzek, D. Kulic, Data augmentation of wearable sensor data for parkinson's disease monitoring using convolutional neural networks, Proceedings of International Conference on Multimodal Interaction, November 13-17, 2017, Glasgow UK.
[8] B. K. Iwana, S. Uchida, An Empirical Survey of Data Augmentation for Time Series Classification with Neural Networks, Progress in PLOS ONE, Vol.16, pp.1-32, 2021.
[9] B. K. Iwana, S. Uchida, Time Series Data Augmentation for Neural Networks by Time Warping with a Discriminative Teacher, Proceedings of 2020 25th International Conference on Pattern Recognition (ICPR), January 10-15, 2021, Milan Italy.
[10] A. Le. Guennec, S. Malinowski, and R. Tavenard. Data Augmentation for Time Series Classification using Convolutional Neural Networks, Proceedings of ECML/PKDD workshop, September 19-23, 2016, Riva del Garda, Italy.
[11] S. Hochreiter, J. Schmidhuber, Long Short-Term Memory, Progress in Neural Computation, Vol.9, pp.1735-1780, 1997.
[12] D. R. Velez, B. C. White, A. A. Motsinger, W. S. Bush, M. D. Ritchie, S. M. Williams, J. H. Moore, A balanced accuracy function for epistasis modeling in imbalanced datasets using multifactor dimensionality reduction, Progress in Genetic Epidemiology, Vol.31, pp.306-315, 2007.
[13] T. H. Cheung, D. Y. Yeung, MODALS: Modality-agnostic Automated Data Augmentation in the Latent Space, Proceedings of International Conference on Learning Representations, April 30, 2020, Addis Ababa Ethiopia.
[14] R. Child, Very Deep VAEs Generalize Autoregressive Models and Can Outperform Them on Images, Proceedings of International Conference on Learning Representations, April 30, 2020, Addis Ababa Ethiopia.
[15] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative Adversarial Nets, Advances in neural information processing systems 27, 2014.