AI-Based PdM Platform in Deciding Failure of Automobile SCU Equipment

Sung Hyun Oh and Jeong Gon Kim *

Department of Electronic Engineering, Tech University of Korea, Si-Heung, Republic of Korea
Email: osh119@tukorea.ac.kr (S.H.O.); jgkim@tukorea.ac.kr (J.G.K.)
*Corresponding author

***Abstract*—Recently, factory automation has been implemented using sensor networks. In general, the equipment deployed in automated factories is expensive. Due to the huge maintenance expenses associated with manufacturing plant equipment, there is a growing need for technology that can predict the lifespan of equipment consumables. Real-time fault prediction technology is essential because downtime in a process can led to substantial financial losses for a factory. Predictive Maintenance (PdM), which predicts replacement cycles instead of relying on Preventive Maintenance (PM) following equipment failure, can enhance productivity. Hence, this paper developed a predictive maintenance technology based on Industrial Internet of Things (IIoT). The developed platform can predict and verify the state of equipment in real time. To predict faults, we generated virtual voltage and frequency data for the inspection equipment of the Shift-by-wire Control Unit (SCU). We then applied this data to three models: the Recurrent Neural Network (RNN), the Long Short-Term Memory (LSTM), and the Gated Recurrent Unit (GRU), and compared their performance. Among them, the GRU model achieved the highest prediction speed and accuracy, with an R^2-score of 0.992. We utilized this platform to develop a real-time AI prediction management system with the goal of improving productivity.**

***Keywords*—Predictive Maintenance (PdM), Artificial Intelligence (AI), Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), inspection equipment**

I. Introduction

Recently, the aging of core equipment within manufacturing companies worldwide has become a serious issue [1]. Among this equipment, 38.6% have been in operation for more than 10 years, while a total of 80% have been in operation for more than 5 years, indicating a significant proportion. Continuing production with aging equipment can result in a decline in productivity due to manufacturing equipment problems and even workplace accidents [2].

In addition, recent maintenance trends indicate that Predictive Maintenance (PdM) [3, 4], which utilizes Artificial Intelligence (AI) [5], big data, augmented reality, and other technologies, has become a crucial factor in reducing maintenance costs in the manufacturing industry [6]. Recent research has introduced new methods for predicting equipment maintenance, such as general Recurrent Neural Networks (RNN), which can uncover relationships within specific time series data using big data [7]. Furthermore, there has been an approach to predictive maintenance using an augmented reality smart glasses system [8]. These developments demonstrate the continuous advancements in predictive maintenance systems based on RNN.

Following the above developments, the PdM system is being applied to various fields, such as Smart IoT platform [9], nuclear power plant [10], Motor [11], Boilers [12], and Electrical Power Systems [13].

In Ref. [14], a study was conducted on an AI-based Internet of Things (IoT) prediction system aimed at improving network latency by developing AI-assisted distributed systems. This is achieved by deploying separate AI models on various edge nodes, allowing for data processing near the sensor and enhancing network latency. The authors validated their proposed system using the Tennessee Eastman dataset and demonstrated its superiority over existing techniques.

Liu *et al*. [15] proposed an IoT-based PdM method to optimize the manufacturing process using machine learning algorithms. The method analyzes correlations between datasets and detects outlier data patterns. It then utilizes a classification approach to identify issues according to the specific type of manufacturing process. The variables that contribute the most to manufacturing defects are identified and analyzed in order to optimize the manufacturing process.

Rahhal and Abualnadi [16] collected data from sensors connected to the central processing unit through IoT and applied two types of neural networks: vanilla-RNN and Long Short-Term Memory (LSTM)-RNN for prediction. Based on the prediction results, LSTM-RNN is recommended for important devices, while vanilla-RNN is suitable for devices that prioritize simplicity.

In this paper, we conduct research on a real-time monitoring program based on the Gated Recurrent Unit (GRU). Our study focuses on evaluating the prediction accuracy and processing time of regression neural networks. Initially, we set up the dataset and attempted to predict data using the simplest model for time series prediction, which is the RNN. However, in order to predict

Manuscript received August 3, 2023; revised August 29, 2023; accepted September 13, 2023; published February 5, 2024.

doi: 10.12720/jait.15.2.164-169

random numbers that follow a specific rule, we needed to input data into the dataset for a longer period of time. Unfortunately, the performance of the general RNN model deteriorates as the sequence length increases. Therefore, we introduced the LSTM model [17], which can handle longer sequence lengths without any problems, to enhance the accuracy of the predictions. Additionally, we introduced the GRU model, which has a similar simple structure to LSTM and aims to reduce training time. In addition to model selection and adjustment, we have developed a Graphical User Interface (GUI) platform for visualizing real-time data.

The main contributions of this paper are as follows. First, by comparing the performance of RNN, LSTM and GRU, which are famous prediction models, analyze each scheme and studied how to decide the optimal model for PdM in real environment. Next, the monitoring program shows the condition of the equipment in real time and also predict the expected lifetime by AI model to prevent the suspension of whole manufacturing due to the failure of some part process in IIoT based sensor network for commercial SCU productivity.

The structure of this paper is as follows. In Section II, the system configuration used in this study. Section III presents the proposed Predictive Maintenence (PdM) platform. In Section IV, the experiments and evaluations of the developed system are presented. In Section V, the final version of the developed platform is summarized, and in Section VI, the conclusion of this paper is presented.

II. System Configuration

In this section, we present system model comprehensively. The conceptual diagram of the proposed system model is shown in Fig. 1.

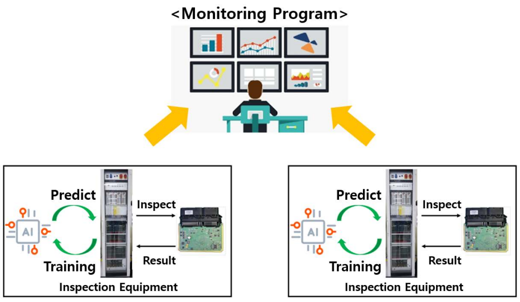


Fig. 1. Overall system configuration.

The proposed system model consists of three main components as depicted in Fig. 1. The first component is the inspection equipment hardware, which is responsible for conducting inspections on the Shift-by-wire Control Unit (SCU) product and storing the corresponding log data. The second component is the AI engine, which trains the model using the log data generated by the inspection equipment. It then performs diagnoses and predictions on the state of the equipment using real equipment data. Lastly, the third component is a monitoring program that runs on a Personal Computer (PC). This program receives diagnostic and predictive data from the AI engine and visualizes the equipment status. The monitoring program is designed to manage the status of multiple pieces of equipment, thereby enhancing the efficiency of equipment management in the production automation factory.

Fig. 2. Electric vehicle SCU controller.

The SCU and SCU test equipment will be described. Recently, most vehicles adopt automotive transmission, which can change park (P), reverse (R), neutral (N) and drive (D) by pressing the buttons or operating of rotary. The SCU is a device that enables automatic shifting. It automatically shifts gears based on the speed and load of the vehicle. Fig. 2 shows an automatic transmission SCU mounted on an electric vehicle. SCU inspection equipment is connected to the SCU through a functional test jig and performs inspections by measuring voltage and current, etc. However, if jig is used iteratively, it wears out gradually. As it last long, the contact resistance increases, which leads to a error in the signal transmitted to SCU from the equipment. It may results in mistake such that the operator decides it is defective while it is not actual fact due to malfunction of the jig pin and test equipment.

In this study, we aim to address malfunctions caused by aging equipment. Data generated from the test equipment hardware and function board is collected for this purpose. The collected data is transmitted to the server using a wireless network and is used for AI model learning. The AI model is described in detail in the following section.

III. Proposed PdM System

To implement the system proposed in this study, an AI model and monitoring program were developed based on Python. The proposed system consists of three phases: data training, testing, and visualization. In the data training phase, the model is trained using accumulated data. The dataset is provided in Comma Separated Value (CSV) format, and specific parameters are selected for training the model. The information of the trained model is saved, and early stopping is implemented to prevent overfitting and reduce training time.

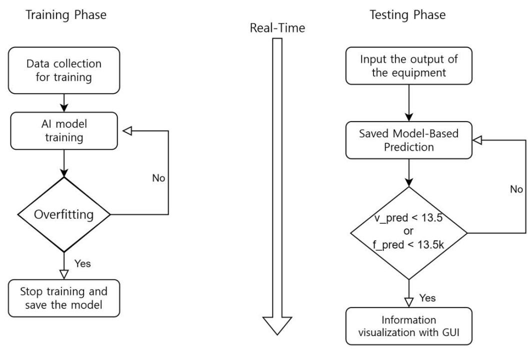


Fig. 3. Proposed system flow chart.

Moving to the testing phase, the saved model is loaded, and real-time input from the equipment's output values (test set) is utilized to predict the output values. The predicted values are then written to a CSV file and trans mitted to the GUI component for visualization. The system flow chart is summarized in Fig. 3.

A. Time Series Data Prediction Model

This subsection describes the structure of a time series data prediction model. First, the structures of RNN, LSTM, and GRU, which are models for predicting time series data, are depicted in Fig. 4.

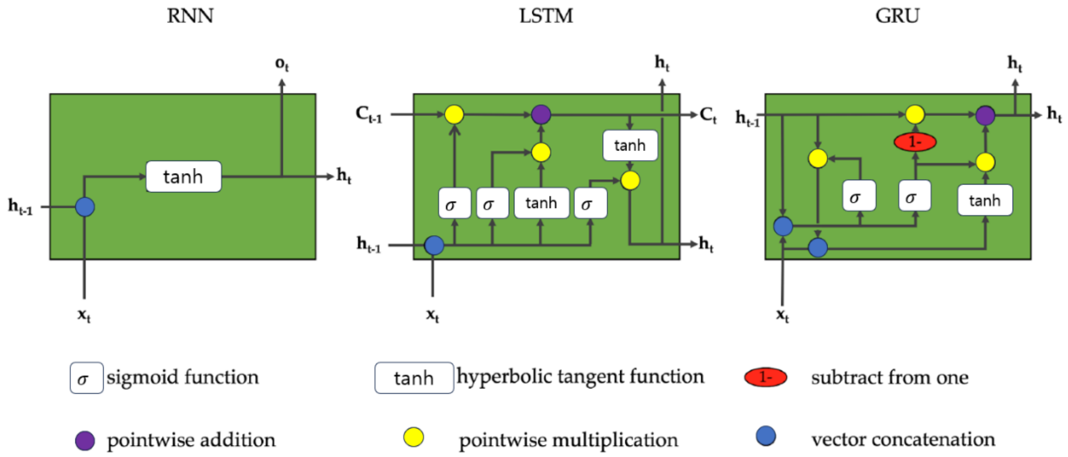


Fig. 4. Structure of time series data prediction model.

RNN is a model that processes input and output as sequences. A sequence refers to continuous data, making RNN a suitable neural network model for time-series data. Unlike a Deep Neural Network (DNN), RNN shares all parameters. RNN is advantageous in processing short sequences; however, its learning ability is diminished when there is a long distance between related pieces of information. As the data sequence becomes longer, the Back Propagation Through Time (BPTT) algorithm of RNN's backpropagation becomes challenging for learning long-term dependencies. During the weight updating process, a problem arises where the gradient diminishes as values less than 1 are continuously multiplied.

LSTM, unlike RNN, is capable of detecting long-term dependencies in data and trains efficiently. LSTM introduces a structure called the Cell State to facilitate the learning of long-term dependencies. It incorporates three gates, namely the Input, Forget, and Output gates, which operate during the data calculation step and store state values in memory cells. By adjusting the gate components that interact with the data, unnecessary operations and errors can be minimized, partially addressing the issue of long-term dependency.

On the other hand, the GRU is a modified version of the LSTM model that contains fewer parameters. While GRU may be slightly inferior to LSTM, it offers faster processing speed. GRU represents the latest algorithm that improves upon LSTM. However, since there is no significant performance difference between the two, it is crucial to select the appropriate model based on performance testing.

B. Performance Evaluation Criterion of Predictive Model (R^2-Score)

The R^2-score is a metric used to evaluate the performance of regression models compared to the Zero-R model, which predicts the mean value. Unlike other performance metrics such as Root Mean Square Error (RMSE) or Mean Absolute Error (MAE), which can be influenced by the scale of the data, the R^2-score is a relative performance metric that provides a more intuitive understanding of the model's performance. The R^2-score can be calculated using the following equation, denoted as Eq. (1).

$$R^2\text{-score} = \frac{SSE}{SST} = 1 - \frac{SSR}{SST} \tag{1}$$

The R^2-score is calculated by comparing the variance of the residuals to the total variance, which provides a measure of similarity between the actual values and the predicted values. A high R^2-score indicates a greater accuracy of the model, where the squared difference between the actual and predicted values is minimized.

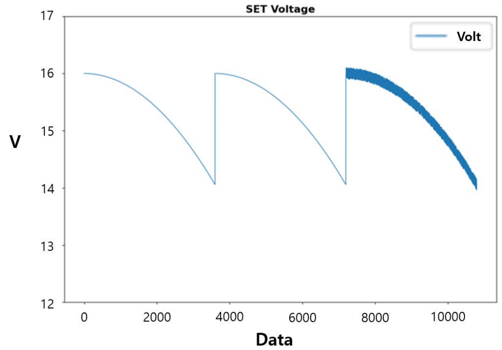
When using a model that estimates the mean value, the Sum of Squares Error (SSE) will tend to be close to 0. Consequently, the R^2-score, which evaluates the model's relative performance against the mean value model, will also be 0, indicating no improvement over the mean value model. Conversely, if an ideal regression model is employed, the difference between the estimated and actual values (SSR) will be 0, resulting in an R^2-score approaching 1.

In cases where the model's performance is exceptionally poor and the Sum of Squared Residuals (SSR) significantly exceeds the performance of the mean value model, a negative R^2-score may occur. This suggests that there may be significant issues with the dataset or the model.

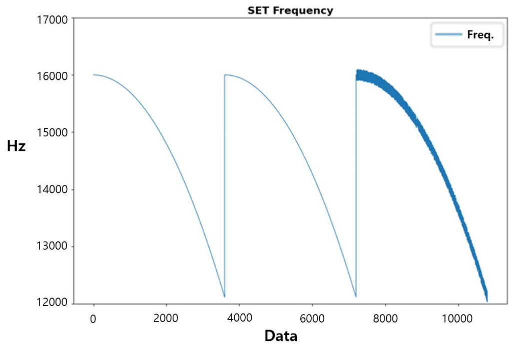
C. Data Set Construction

In this subsection, we provide an explanation of the dataset's composition. The global trend in manufacturing equipment aging indicates that approximately 77% of equipment has a lifespan of less than 10 years [6]. Based on this trend, we have established a 10-year replacement period for equipment, following the Preventive Maintenance (PM) approach. Additionally, we have defined the time frame for identifying abnormal equipment symptoms as 7 years. Consequently, we constructed the dataset to reflect the occurrence of abnormal equipment conditions after 7 years.

For the Shift-by-wire Control Unit (SCU) equipment, abnormal symptoms manifest as voltage drops or a decrease in the measured frequency. Hence, we used voltage drop and frequency decrease as the criteria for determining abnormal equipment conditions. The trend of the constructed dataset is illustrated in Fig. 5.



(a)



(b)

Fig. 5. Constructed data set. (a) Volt, (b) Frequency.

As depicted in the figure, the data were generated assuming daily recordings of the output values. A 10-year cycle was considered to align with the equipment's lifespan, resulting in 3,600 data points representing one cycle. For the training set, two cycles (7,200 data points) were utilized. For the test set, it was assumed that real-time data would be generated and inputted into the system on a daily basis. Fig. 5 illustrates a scenario where the equipment was replaced shortly before, and 10 years have passed since then.

IV. Experiment and Evaluation

There are several parameters involved in training an AI model, such as batch size, Learning rate (Lr), sequence length, and training epochs. As explained in Section III.A, there are different models for time series prediction, such as RNN, LSTM, and GRU. In order to select the appropriate type of time series prediction model and its parameters, we adjusted various parameters while taking into account the evaluation criterion of the R^2-score, as outlined in Section III.B. The goal was to assess the accuracy of the AI model's predictions by comparing the test set to the predicted data.

A. Batch Size, Learning Rate

Batch size is a parameter that determines the amount of data processed at once during training. A larger batch size can accelerate training speed but may lead to reduced accuracy and an increased risk of overfitting. On the other hand, a smaller batch size offers greater resilience to sudden changes. The learning rate is a value that determines how quickly the model adjusts its parameters based on the gradient. A higher learning rate can accelerate convergence but increases the risk of overshooting, while a lower learning rate increases training time.

In this paper, our aim was to identify the optimal combination of parameters by comparing commonly used values, such as a batch size of 64 and a learning rate of 0.01. Additionally, we set the sequence length to 7 and the number of epochs to 10. As shown in Fig. 5, the best match to the true values was achieved with a batch size of 128 and a learning rate of 0.01. The performance of each parameter combination shown in Fig. 6 can be further examined in Table I.

TABLE I. Performance of Each Combination

Model	Batch Size	Learning Rate	R^2-Score
RNN	64	0.01	0.9723
RNN	128	0.01	0.9810
RNN	128	0.1	0.9732
RNN	128	0.2	0.9771

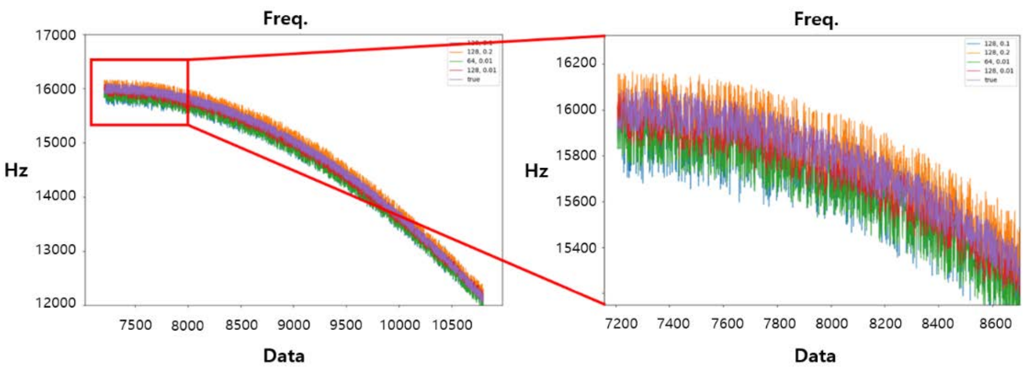


Fig. 6. Prediction graph according to batch size and learning rate.

B. *Train Epoch*

During the training of the model, the training loss typically decreases as the number of iterations increases. To prevent overfitting, early stopping was implemented when the loss started to increase. After conducting multiple tests, it was observed that early stopping occurred frequently around epoch 30, despite the initial setting being epoch 100, as depicted in Fig. 7. As a result, the results are presented in Table II.

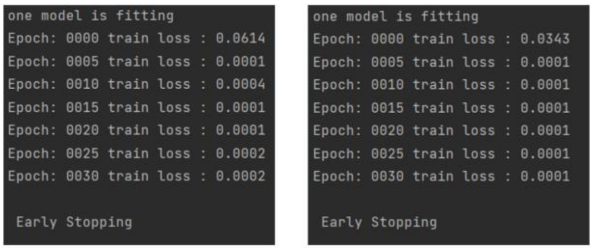


Fig. 7. Repeated test results.

TABLE II. R^2-SCORE AND LEARNING TIME ACCORDING TO MODEL TYPE AND SEQUENCE LENGTH

Model	Sequence Length	R^2-Score	Learning Time
RNN	7	0.9810	15 s
RNN	30	0.9650	45 s
LSTM	30	0.9930	62 s
GRU	30	0.9921	47 s

C. *Sequence length & Model*

Sequence length refers to the dimension of the input data set for RNN models. For instance, if the sequence length is set to 7, the input data set becomes a matrix with a dimensions format. Using a sequence length of 7 as a reference, we adjusted the values according to the parameters specified in Sections IV.A and IV.B. The test epoch value was adjusted to 30 to optimize the model's performance and prevent overfitting.

Upon examining the table, it is evident that increasing the sequence length of the RNN model in an attempt to achieve a higher R^2-score resulted in the problem of vanishing gradients. This issue ultimately led to a decrease in the R^2-score. To address the issue of vanishing gradients in the RNN model, we used an LSTM model, which exhibited enhanced performance in comparison to the RNN model under identical circumstances. Furthermore, the GRU model with a reduced tanh layer exhibited similar performance but with a 24% shorter training time compared to LSTM. Hence, after evaluating all the combinations, we selected the final prediction model, as shown in Table III.

TABLE III. FINAL SELECTED PARAMETER OF GRU MODEL

Model	Sequence Length	Epoch	Batch Size	Learning Rate
GRU	30	30	128	0.01

V. PREDICTION RESULT AND MONITORING GUI

This section explains the prediction results of the GRU model and the GUI of the monitoring program. First, the main screen of the GUI will be explained.

Fig. 8 shows the main screen of the developed GUI. On the main screen, you can access three types of information. Among them, the red rectangular area consists of two buttons. When the button is selected, the screen converts to display the status prediction graph of the selected equipment for easy reference. The blue square area provides information about the current equipment status. The status of the equipment is indicated through guidance messages and colors, with normal (green), caution (yellow), and warning (red).

Fig. 9 illustrates the predicted results obtained from the proposed model. Utilizing the selected GRU model and parameters established based on the rationale described in the main text, we simulated 3,600 (10 years) test data. We observed that the "pred" graph gradually descends with a slope similar to the true value. Furthermore, the R^2-score, which represents the similarity between the two graphs, is calculated to be 0.992, indicating a close resemblance to the ideal value of 1.

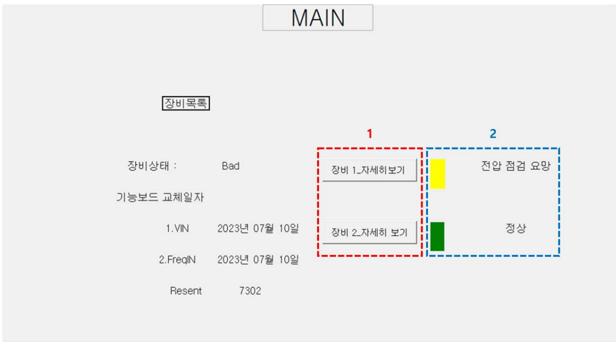


Fig. 8. Main screen of GUI.

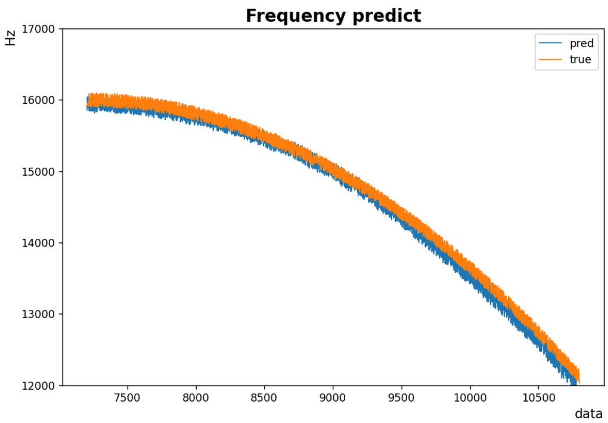


Fig. 9. Prediction result of GRU model.

Fig. 10 showcases the GUI of the monitoring program, which displays the prediction results in real-time. The program continuously updates the test set on the right and predicts the "pred" set one day in advance. As new data is added to the right, the previous data shifts towards the left of the graph.

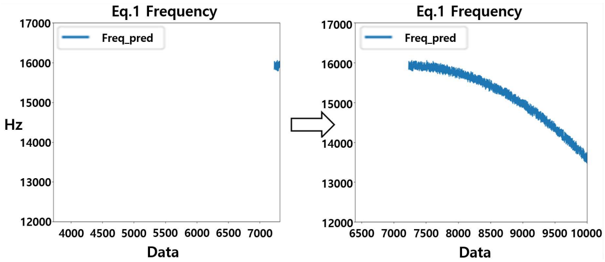


Fig. 10. Real-time visualizes the prediction results in GUI.

The x-axis (number of data) ranges from 3,600 to 7,200 and proceeds in real-time, displaying the predicted results on the right. After processing 7,230 data points, taking into account the train set and sequence length, the graph displays the predicted values based on the real-time output values of the device (test set). The transition from the left to the right graph represents approximately 3,000 data points, which can be interpreted as roughly 7 years. When the output value falls below the threshold, the system detects that the device has reached its end-of-life and triggers an alert or notification on the platform.

VI. Conclusion

This paper aims to reduce cost losses caused by equipment aging and traditional maintenance methods by implementing an AI-based equipment failure management platform using the emerging approach of predictive maintenance. In order to enhance the current factory maintenance data, which followed a 10-year cycle, the target for predictive maintenance was established at around 7 years. The selected GRU model, with parameters set at epoch = 30, sequence length = 30, batch size = 128, and learning rate = 0.01, demonstrated high prediction accuracy with a fast learning time of 47 s (R^2-score = 0.9921). Additionally, the platform presents the predicted values in real-time graph format, enabling the assessment of the current state of the equipment. Data processing techniques will be employed to optimize the prediction performance of the GRU model and improve the stability of manufacturing equipment. The plan involves implementing a real-time AI prediction platform and transforming it into a universal PdM platform by selecting and predicting various equipment features.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Sung Hyun Oh and Jeong Gon Kim conceived the idea of this study. Sung Hyun Oh made substantial contributions to the system implementation and data analysis. Jeong Gon Kim contributed significantly to the interpretation of the results and supervised the conduct of this study. Sung Hyun Oh drafted the original manuscript. Jeong Gon Kim critically revised the manuscript for intellectual content. All authors approved the final version of the manuscript.

References

[1] N. Davari, B. Veloso, G. D. A. Costa, P. M. Pereira, R. P. Ribeiro, and J. Gama, “A survey on data-driven predictive maintenance for the railway industry,” *Sensors*, vol. 21, 5739, Aug. 2021.

[2] K. S. Yu and Y. M. Moon, “Recent trends in smart factory equipment diagnosis and predictive maintenance technologies,” *Information and Communications Magazine*, vol. 37, pp. 36–48, Jun. 2020.

[3] X. Mehmeti, B. Mehmeti, and R. Sejdiu, “The equipment maintenance management in manufacturing enterprises,” *IFAC-PapersOnLine*, vol. 51, pp. 800–802, Nov. 2018.

[4] M. Jamal, S. T. Mohammad, and T. Paul, “Intelligent Predictive Maintenance (IPdM) in forestry: A review of challenges and opportunities,” *Forest*, vol. 12, 1495, Oct. 2021.

[5] H. Wang, W. Zhang, D. Yang, and Y. Xiang, “Deep-LEARNING-enabled predictive maintenance in industrial internet of things: Methods, applications, and challenges,” *IEEE Systems Journal*, vol. 17, pp. 2602–2615, June 2023.

[6] B. E. So and S. S. Shin, “The built of smart factory using sensors and virtual process design,” *J. KIECS*, vol. 12, pp. 1071–1080, Dec. 2017.

[7] S. Biswal and G. R. Sabareesh, “Design and development of a wind turbine test rig for condition monitoring studies,” in *Proc. Inter. Conf. Industrial Instrumentation and Control*, Pune, 2015, pp. 891–896.

[8] N. Slawomir, B. Mohammed-Rafik, and F. Handi, “On predictive maintenance in industry 4.0: Overview, models, and challenges,” *Applied Sciences*, vol. 12, 8081, Aug. 2022.

[9] J. Przepióra, P. Bałazy, P. Książek, P. Knap, P. Pasławski, and M. Pietrzak, “Smart IoT platform for big data analysis in predictive maintenance,” in *Proc. Inter. Carpathian Control Conf.*, Miskolc-Szilvásvárad, 2023, pp. 353–357.

[10] R. Han, P. Li, and Z. Shi, “Implementation strategy of predictive maintenance in nuclear power plant,” in *Proc. Prognostics and Health Management Conf.*, London, 2022, pp. 143–146.

[11] A. A. Manjare and B. G. Patil, “A review: Condition based techniques and predictive maintenance for motor,” in *Proc. Inter. Conf. Artificial Intelligence and Smart Systems*, Coimbatore, 2021, pp. 807–813.

[12] V. Prabhu and D. Chaudhary, “Machine learning enabled Condition monitoring models for predictive maintenance of boilers,” in *Proc. Inter. Conf. Recent Developments in Control, Automation & Power Engineering*, Noida, 2021, pp. 426–430.

[13] S. B. Righetto, M. A. I. Martins, E. G. Carvalho, L. T. Hattori, and S. Francisci, “Predictive maintenance 4.0 applied in electrical power systems,” in *Proc. IEEE Power & Energy Society Innovative Smart Grid Technologies Conf.*, Washington, 2021, pp. 1–5.

[14] Y. Liu, W. Yu, T. Dillon, W. Rahayu, and M. Li, “Empowering IoT predictive maintenance solutions with AI: A distributed system for manufacturing plant-wide monitoring,” *IEEE Trans. Industrial Informatics*, vol. 18, pp. 1345–1354, Feb. 2022.

[15] L. C. Him, Y. Y. Poh, and L. W. Pheng, “IoT-based predictive maintenance for smart manufacturing systems,” in *Proc. Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*, Lanzhou, 2019, pp. 1942–1944.

[16] J. S. Rahhal and D. Abualnadi, “IOT based predictive maintenance using LSTM RNN estimator,” in *Proc. Inter. Conf. Electrical, Communication, and Computer Engineering*, Istanbul, 2020, pp. 1–5.

[17] Y. J. Roh and S. H. Bae, “Forecasting of traffic accident occurrence pattern using LSTM,” *J. Korea Inst. Intell. Transp. Syst.*, vol. 3, pp. 59–73, Jun. 2021.