IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 3 July 2023, accepted 16 August 2023, date of publication 25 August 2023, date of current version 5 September 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3308698

RESEARCH ARTICLE

Enhanced Anomaly Detection in Manufacturing Processes Through Hybrid Deep Learning Techniques

KYUNG SUNG LEE, SEONG BEOM KIM, AND HEE-WOONG KIM, (Member, IEEE)
Graduate School of Information, Yonsei University, Seoul 03722, South Korea

Corresponding author: Hee-Woong Kim (kimhw@yonsei.ac.kr)

ABSTRACT Smart factory systems have been introduced to prevent the decline in overall equipment effectiveness caused by the presence of defects within factory manufacturing equipment. In this context, it is crucial to predict process downtime using manufacturing equipment data and take preemptive actions. However, anomaly detection models for preemptive actions have limitations in labeling anomaly alarms. Moreover, from real-time data collection to model deployment, it needs to consider the model-based service implementation for stakeholders. Our research develops a hybrid deep learning-based anomaly detection model that does not require data labeling. The advantage of this model stems from its ability to identify anomalous patterns via the reconstruction of the sequential progression inherent in the data. According to our experimental findings, the proposed hybrid model demonstrated superior efficacy compared to alternative anomaly detection algorithms. By preemptively predicting downtime within the manufacturing process, it contributes to enhanced production efficiency. Furthermore, we develop an anomaly detection system based on a real-time service framework from the data collection step to improve the activation of smart factories. This research contributes to the literature on the monitoring and management of anomaly detection in smart factories. It also has practical implications for the manufacturing industry by recommending efficiency measures for smart factories to reduce downtime in manufacturing processes and improve product quality.

INDEX TERMS Anomaly detection, deep learning, predictive maintenance, smart factory, manufacturing process.

I. INTRODUCTION

In recent years, there has been a surge in the adoption of smart factories as manufacturers strive to increase their competitiveness in response to changing market demands In smart factories, manufacturing equipment data, collected from connected Internet of Things devices or sensors, is monitored and controlled in real-time using sophisticated software systems [1], [2]. As a result, manufacturers can expect increased productivity, decreased downtime and maintenance costs, and improved product quality [3], [4], [5]. Despite data-driven efforts, however, smart factories have not effectively addressed the issue of delays caused by downtime during manufacturing processes [1], [6]. Downtime reduces the output and is associated with machine repair and maintenance costs, potentially resulting in significant financial losses. Early detection and adjustment of anomalies can increase equipment stability by up to 40% and extend the equipment life by up to 25%.[1] It is thus critical to prevent downtime in smart factories by detecting anomalies in the manufacturing process beforehand.

Various efforts have been made to prevent process downtime in smart factories using equipment data [5], [6], [7]. These efforts aim to build a supervisory control and data acquisition (SCADA) based smart factory system that can

---

The associate editor coordinating the review of this manuscript and approving it for publication was Parikshit Sahatiya.

[1] https://www.ibm.com/blogs/internet-of-things/iot-predictive-maintenance-optimize/

preemptively handle downtime using various time-series data, such as equipment status, temperature, and humidity, originating from process sensors [2], [8]. As a result, factors such as product types and weather temperatures that significantly influence production efficiency degradation caused by manufacturing downtime have been identified [1]. Despite the need for substantial investment in infrastructure through technology implementation within smart factories, preemptive measures against downtime through technical capabilities are still required [8]. In other words, the reality is insufficient to tackle tasks such as the development of capabilities for new technologies among workers [9]. This can be explained by the fact that, as more complexity is added to AI technology operations, the sustainability of technology operations decreases [10], [11]. From an organizational perspective, solutions, including technology, should be tailored for self-operation by relevant organizations with minimal effort [12].

It is crucial to increase production efficiency and reduce production costs by detecting potential errors in advance using manufacturing process time-series data and securing additional production volumes [13]. Previous studies have detected anomalies using statistics-based estimation and machine learning (ML)-based algorithms. Most of these studies, however, have not considered the time-series characteristics of the data [14]. Long short-term memory (LSTM) has recently emerged as a powerful technique for learning long-term dependencies and effectively representing relationships between current and previous events, suggesting that stacked LSTM networks are suitable for anomaly detection in time series [13], [15], [16]. LSTM-based methods perform better in multivariate time-series data than ML-based methods [13], [17]. Furthermore, although earlier research was inclined towards supervised learning due to data labeling issues, it has now expanded to embrace unsupervised learning-based multivariate anomaly detection methods [15], [18]. There is still a limit to having high noise for abnormal labels [19]. It also needs to consider model deployment from real-time data collection to model placement. In smart factories, detecting anomalies early can reduce unexpected failures through immediate repairs or system maintenance. Abnormal conditions must be identified in advance through data-driven approaches during normal manufacturing processes.

This research aims to identify abnormalities in the production process by utilizing an unsupervised learning LSTM-autoencoder model. The model considers the manufacturing equipment anomaly alarm data's noise, which is capable of functioning effectively in actual factory settings. To evaluate the appropriateness of the proposed model, we perform model training using 12-month equipment data collected from actual manufacturing and conduct experiments on three real product codes. Through this, we verify an effective method for detecting early signs of anomalies in multivariate time-series data. From the model deployment perspective, our study further presents a systematic approach that can proactively respond in the field by predicting downtime situations in the packaging process using BentoML, which has high practical applicability. We anticipate practical implications from this research, such as strengthening the business value chain and improving work processes, by monitoring anomaly detection within smart factories.

II. RELATED WORKS

It is crucial to detect anomalies in the manufacturing equipment data collected in smart factories. Through this, better decision-making can improve production efficiency in manufacturing processes. We investigate deep learning (DL)-based anomaly detection models suitable for multivariate time-series data, drawing insights from the relevant literature for practical applications in factories.

The auto-regressive integrated moving average (ARIMA) statistical model has been used for anomaly detection in prediction problems involving linear time-series data. However, it proves unsuitable for nonlinear multivariate time-series data [20]. The recurrent neural network (RNN) model fundamentally learns time-series characteristics using nonlinear activation functions in the data. A comparison of ARIMA and LSTM for time-series predictions revealed an average error reduction rate of approximately 80% [21]. This demonstrates that DL models surpass statistical models.

Chalapathy and Chawla [22] comprehensively investigated anomaly detection methodologies suitable for multivariate time-series data through hybrid DL approaches. Hybrid DL approaches primarily use autoencoder algorithms as variable extractors within hidden variable vectors and then employ traditional anomaly detection algorithms such as one-class support vector machine (OCSVM) and support vector data description (SVDD) [16]. These anomaly detection model structures integrate separate networks like convolutional neural networks (CNNs) and LSTM with OCSVM or SVDD, which are not typically applied to multivariate time-series data. To handle multivariate time-series data for anomaly detection, structures formed by DL network combinations such as CNN–fuzzy neural network (FNN), LSTM–FNN, CNN–bidirectional LSTM, and restricted Boltzmann machine–LSTM–FNN have been introduced [23], [24], [25]. More complex structures should thus be used in DL models to achieve high performance in anomaly detection with multivariate time-series data [18], [19], [26]

Nguyen et al. [13] proposed an LSTM–autoencoder model for anomaly detection using multivariate time-series data. They, however, did not consider the practical application in actual factory environments [16], [27]. Because of this issue, SCADA systems were integrated with manufacturing systems to make appropriate decisions in real-time [8], [28]. The need for efficient utilization of real-time integrated systems was emphasized for monitoring manufacturing processes and detecting real-time anomalies [16]. Antonini et al. [29] proposed an end-to-end adaptive and configurable

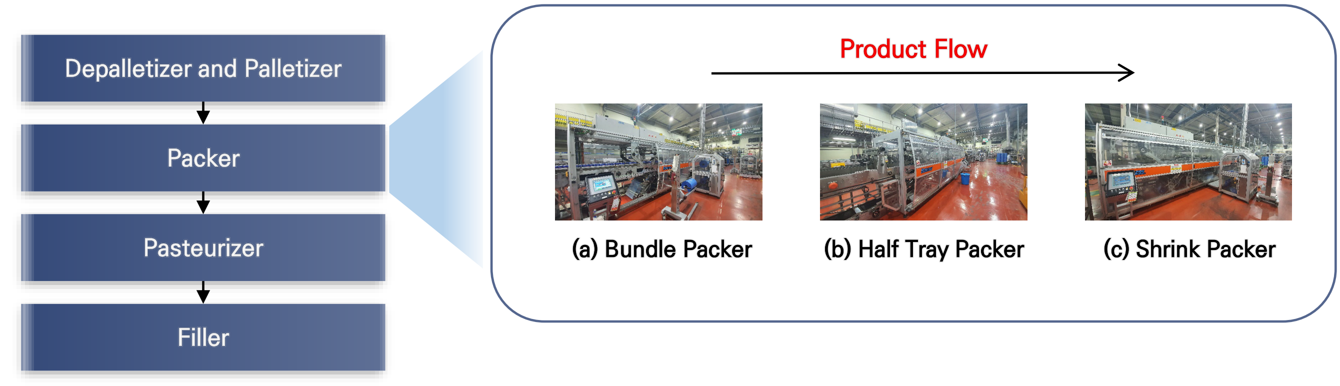


FIGURE 1. **Manufacturing factory - product flow process.**

anomaly detection system using the Tiny-MLOps methodology. They designed a TinyML pipeline considering industrial environments from data sampling, feature extraction, model training, and inference to anomaly event notifications. The proposed Tiny-MLOps approach demonstrates the suitability of real-time anomaly detection models in smart systems [10]. Previous research has not yet considered another important issues such as the optimization process for training the best model considering the factory environment and methods to apply the model within the manufacturing process [2], [30].

TABLE 1. **Related studies on anomaly detection.**

Study	Method	Hybrid Approach	Application
Chalapathy et al. (2019)	OCSVM, SVDD	-	-
Wang et al. (2022)	Encoder-double decoder	Yes	-
Nguyen et al. (2021)	LSTM–autoencoder	Yes	-
Rousopoulou et al. (2022)	AI Platform	-	Yes
Antonini et al. (2023)	Tiny-ML	-	Yes
Our Study	LSTM–autoencoder	Yes	Yes

As presented in Table 1, previous studies have focused on developing DL-based anomaly detection models using multivariate time series data from manufacturing processes [27], [31]. The objective of these studies was to improve the predictive performance of the models. These studies have employed complex DL architectures rather than simple network structures to achieve enhanced outcomes [2], [13]. Moreover, these studies have sought to advance smart systems by proposing ML frameworks that consider real-world factory environments [11], [32]. Consequently, there is a need to introduce practical anomaly detection models suitable for manufacturing processes. Our study aims to address the limitations identified in the literature. Building upon prior research, we adopt a hybrid DL model to analyze multivariate time series data and propose an approach to enhance the deployment of the model by considering the specific context of manufacturing processes.

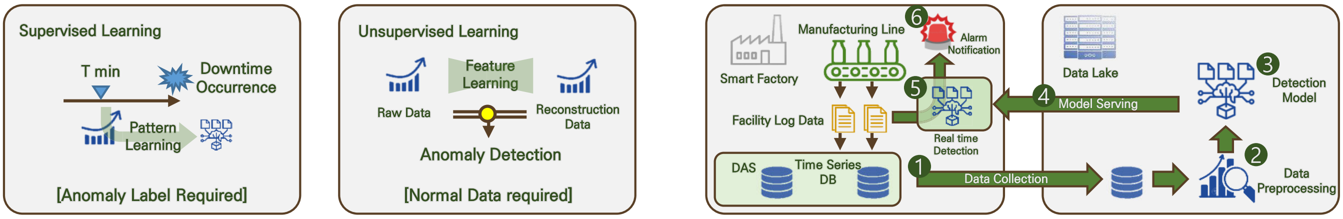
III. METHODOLOGY

A. PRODUCT FLOW PROCESS

Downtime in the product-packaging line occurs at the Packer step (see Fig. 1). Products move from the Bundle Packer step (Fig. 1(a)) to the Half Tray Packer step (Fig. 1(b)). During this product flow, environmental factors such as mechanical malfunction may result in conducting or packaging defects. Because of this downtime, the product cannot move to the Shrink Packer step (Fig. 1(c)), causing time delays in the manufacturing process. The equipment problem is acknowledged to have resulted from multiple environmental factors present during the manufacturing process. Conducting data-driven analysis is thus to find viable solutions. This study builds an anomaly detection model for the product-packaging line to increase the efficiency of the factory manufacturing equipment. Furthermore, the proposed anomaly detection model should be operated for the management and maintenance of a smart system. A real-time framework procedure is imperative to identify anomalies and emit warning alerts within a smart system.

B. APPROACHES FOR ANOMALY DETECTION

Our research considers two approaches for detecting anomalies in the manufacturing process based on equipment data (Fig. 2 (a)). First, we consider a supervised learning approach using anomaly alarm data. This approach identifies downtime points using data on previously occurred anomaly alarms [33]. However, the results of the anomaly detection model are significantly affected by the anomaly alarm selection results [19], [34]. In contrast, there is an unsupervised learning-based anomaly detection model approach that uses patterns in manufacturing equipment data. This approach automatically learns the widespread characteristics of the data and detects



(a) Supervised and Unsupervised Learning

(b) Applied Anomaly Detection Procedure

FIGURE 2. Two approaches for anomaly detection.

unusual data as anomalies. There is, however, a drawback in that it is difficult to validate the performance of the anomaly detection model in this approach [18], [19]. Considering the challenges, this study employs a suitable detection model approach within the manufacturing context. This is achieved by conducting interviews with practitioners and combining their domain expertise with AI technology to develop an efficient system architecture [9], [11], [35]. Furthermore, this study proposes real-time model utilization methods, such as early warnings in the manufacturing process, by building an anomaly detection model through a step-by-step procedure (Fig. 2 (b)).

Our study is based on the design science research methodology [36], which combines information systems and computer science. According to the design science methodology, we define the limitations and problem phenomena of existing situations and propose solutions. Following the guidelines of design science research [37], we develop an anomaly detection model, verify the prediction results, and evaluate its performance for validity. We further consider real-time prediction model application methods, taking into account the process environment [9], [12].

Stemming from the design science paradigm, the sequence for real-time deployment of the anomaly detection model is articulated as follows: Step (1) involves collecting manufacturing equipment data from the factory. We collect data from the data lake, including bundle packer (BDP) equipment data, anomaly alarms, and production information data. Step (2) involves the analysis of manufacturing equipment data and data preprocessing for the anomaly detection model. We identify downtime factors through exploratory data analysis and select variables to perform data cleaning and merging. Step (3) involves the development process of the anomaly detection model. We develop a prediction model for anomaly detection based on previous manufacturing equipment data and evaluate it by comparing it with downtime points. Step (4) proposes application methods for the real-time anomaly detection prediction model within a factory. This study proposes specific methods for applying the model created in the data lake to the factory system. Step (5) is the phase where anomalies in the process are predicted in real-time. It refers to the process of detecting anomalies by collecting data generated in the factory in real-time. Finally, Step (6) performs prediction using the applied anomaly detection model in the system and provides early alarms for the results.

C. DATA COLLECTION

Our study uses manufacturing equipment data stored in a smart factory operated by Company XYZ. Company XYZ is the largest beverage company in South Korea and has a strong presence in overseas markets such as China, Southeast Asia, and the United States. The company extensively utilizes process data as part of its adoption of smart factory systems to enhance business competitiveness and production efficiency. When the factory operates continuously to produce products, a time-series log record is automatically stored on the factory server. Facility inspections are conducted regularly, typically at least once a month. Additionally, there is an annual one-month period dedicated to comprehensive facility overhaul, involving complete disassembly, inspection, and repair. Prior to initiating standard production cycles, preliminary test runs spanning approximately 0.5 to 1 hour are executed. These measures ensure the strict maintenance management of the facility and help control any potential impact of machine conditions.

From November 2021 to November 2022, we collected process equipment data at 3-second intervals, such as time-series sensor log and anomaly alarms. The data used for our study was collected from the production environment, specifically from the Bundle Packer (BDP) and Half-Tray Packer (HTP) systems. In our study, we collected data on diverse variables indicative of the production environment, complemented by readings from other sensors deployed within the facility. For example, temperature modules provide information about conditions within the system while environmental variables capture the surrounding indoor temperature and humidity. Such variables elucidate the environmental dynamics influencing the production mechanism. Lastly, the aging variable represents the elapsed time since the manufacturing units started and can provide insights into the operational duration of the system. These can be integrated with product information to identify the history of each product. Each product contains a number of product codes (PO_CDs) as product information. For example, a product (e.g., a can product of Coke) has its unique product code over

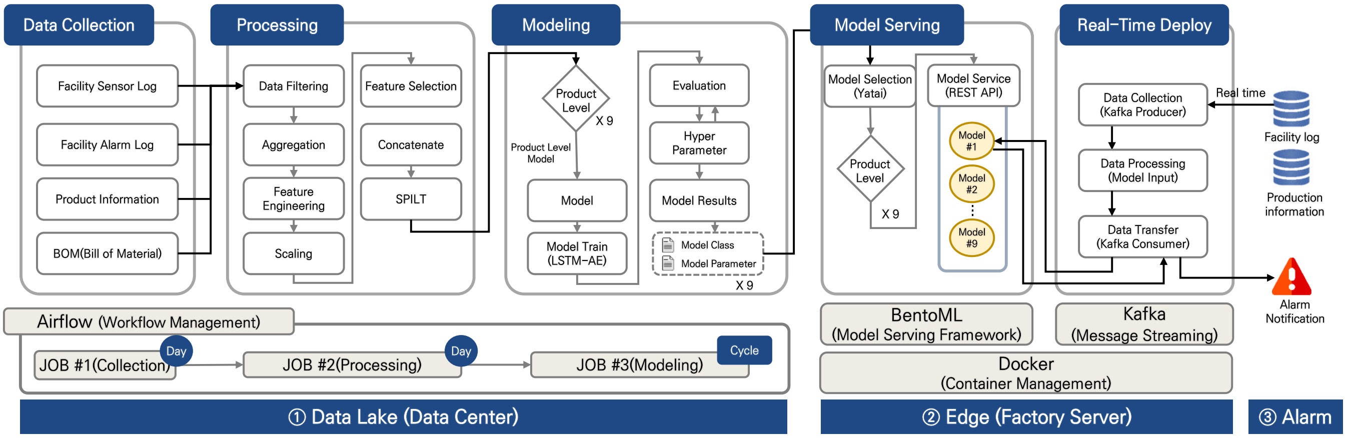


FIGURE 3. Framework overview. To apply the developed anomaly detection to the actual smart factory, the following steps were taken: (1) a workflow for data preprocessing and modeling was implemented, (2) the trained model was implemented for real-time service. And real-time data collection was performed to obtain input data for the trained model, including preprocessing, and (3) an alarm notification indicating the anomaly status was presented through real-time inference.

TABLE 2. Data variables summary.

Category	Variables	Description	Type
Product Information	PO_CD	Production order number	object
BDP (Bundle Packer)	System time	Timestamp of the log occurrence with a 3-second interval	Date time
	RIDS	The current state of the half-tray packer, including RUN, IDLE, DOWN, and STOP	object
	CPM	Number of products packed per minute	Integer
	Setting temperature	Setting the value of module temperature for the three tunnels	Float
	Temperature module	The actual temperature reading from the modules in the three tunnels	Float
	Air volume	Setting the value of fan air volume from the six modules	Float
	Environmental variables	Indoor temperature and humidity recorded around the packer	Float
	G tags	Equipment alarm status from nine sections of the bundle packer	object
	Aging	Count of time (in units) since the manufacturing units started	Integer
HTP (Half-Tray Packer)	HTP Alarm	Alarm status from the half-tray packer	object
	RIDS	The current state of the half-tray packer, including RUN, IDLE, DOWN, and STOP	object

time (e.g., *-00 at 09:00am,*-60 at 09:01am). We developed an anomaly detection model for the collected manufacturing equipment data using the features listed in Table 2.

D. MODEL SERVICE FRAMEWORK

Fig. 3 describes the real-time anomaly detection framework in our study. The collected manufacturing equipment data undergo preprocessing, followed by the anomaly detection modeling process. We then establish an in-plant model service for real-time anomaly detection applications and develop a model that detects anomalies in real-time by receiving input data from the process.

To interpret the data collection and processing in the target factory context, we conducted domain-based interviews with five practitioners (two plant operation managers, one production operation manager, and two data organization members). The interview revealed that the collected anomaly alarm data includes noisy data and is not accurate regarding downtime occurrence. Based on the feedback, we refined the anomaly alarm data and defined an abnormal pattern interval for downtime. Specifically, we did not adopt a supervised learning-based model approach because of the problem of noisy data labeling. Instead, we developed an unsupervised learning-based model that learns only from normal data and detects abnormal patterns. The unsupervised learning-based model needs to learn only normal data, which requires data refinement. Because the collected alarm data allow us to understand the normal time-series flow of equipment data, BDP RIDS and CPM variables are used for normal data refinement.

In addition, to minimize noisy and duplicate values in normal data, we preprocessed the manufacturing equipment data at 3-second intervals into 1-minute intervals, which will be used only for model learning. The 1-minute training data is transformed using a time-series smoothing method to address unnecessary variability. In the actual factory environment, the test data uses 3-second intervals. To develop a real-time anomaly detection model, we need to consider model learning and the number of models. However, learning individual models using product codes creates difficulties in generating models for all products. In addition, there are distribution differences in time-series data (e.g., temperature, airflow) by product type. For this reason, we develop a model by selecting 9 product types.

This study thus develops an unsupervised learning-based anomaly detection model that considers only normal patterns due to the issue of anomaly alarm data labeling. We refine the existing manufacturing equipment data into normal data and preprocess them to reflect time-series properties. Furthermore, we consider real-time anomaly detection models and construct models and pipelines considering data attributes by product type. To optimize the performance of the anomaly detection model, we use three datasets for development. We construct a training set for model learning, use a validation set for parameter estimation, and a test set to verify whether the model classifies abnormal and normal intervals from manufacturing equipment data.

IV. RESULTS AND EVALUATION

A. MODEL DEVELOPMENT

Our study conducted domain interviews with industry experts to understand the data characteristics and system environment of smart factories. First, the data collected from the manufacturing process was noisy in terms of abnormality labels. Furthermore, for real-time anomaly detection, we need a model that receives and processes sequentially flowing data as input. Therefore, we considered an unsupervised learning-based model without abnormality labels. We developed an unsupervised learning-based model for the application of a real-time anomaly detection algorithm (Fig. 4). This concept involves detecting anomaly sequence data in a real environment when a model pre-trained with only normal data receives input.

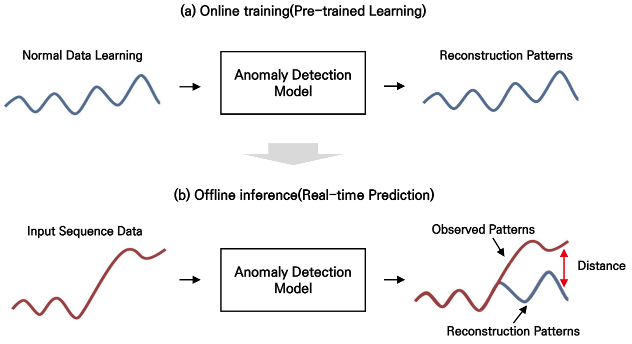


FIGURE 4. Conceptual anomaly detection model development.

Our manufacturing process involves multivariate time series data. The utilization of LSTM-autoencoder for anomaly detection in such data has demonstrated numerous advantages in various research studies [27], [30], [31]. An autoencoder is an unsupervised neural network model that uses unlabeled data. It compresses input data to reduce their dimensions and then expands and restores the data to reconstruct them [38]. An encoder converts high-dimensional input to a low-dimensional representation and a decoder expands the dimensions to restore the data. The model learns by updating parameters to make the input data in the encoder and the restored data in the decoder identical. The ability of an autoencoder to restore input data can be used to detect abnormal anomalies [2]. That is, it trains on normal data only and detects abnormal patterns in parts that cannot be reconstructed [13]. The input data (i.e., process sensor data) require sequential information flow characteristics [39]. Therefore, we develop an LSTM–autoencoder that combines an autoencoder capable of detecting anomalies with an LSTM capable of reflecting sequential time-series information. The LSTM-autoencoder comprises a network of LSTM units serving as both the encoder and decoder. The LSTM's capability to capture temporal dependencies in multivariate data over extended sequences makes it well-suited for tasks such as time series prediction and anomaly detection [13]. Specifically, it can be leveraged to detect anomalies in multivariate time series solely based on normal instances [15]. This holds significant practical implications for our manufacturing process, as access to abnormal data is not always feasible.

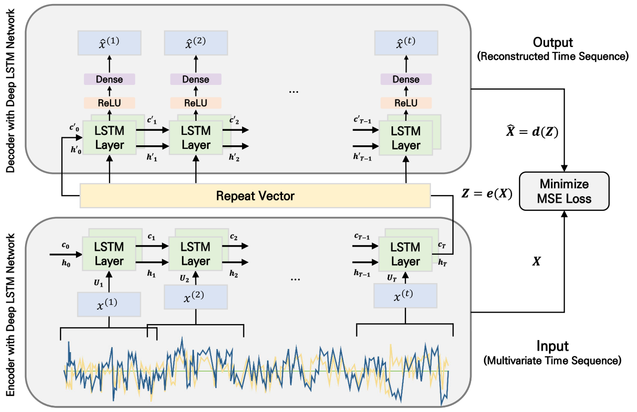


FIGURE 5. LSTM-autoencoder network architecture illustration.

The LSTM–autoencoder model is a neural network model that uses an autoencoder (encoder–decoder) structure and an LSTM architecture for time sequence data (Fig. 5). In general, the architecture comprises an input layer, an output layer, an encoder-decoder neural network, and a repeat vector. The input sequence data are moved to overlap the window size from multivariate variables. This compresses the input sequence data into a repeat vector through the LSTM module in the encoder. In the decoder, the feature vector converted from the encoder is restored in the reverse order of the input sequence data and then output. In this process, the difference between the input sequence data and the reconstructed output sequence data is calculated. Specifically, when presented with an input $X \in R_m$, the encoder compresses x to derive an encoded representation $Z = e(x) \in R_n$. Subsequently, the decoder reconstructs this representation to produce the output $\hat{X} = d(z) \in R_m$. During training, the autoencoder is optimized by minimizing the reconstruction error (MSE). The newly defined unsupervised learning approach of LSTM–autoencoder is applied to the manufacturing equipment data to learn normal time-series data. It can detect abnormal intervals based on the difference between actual data flow and predicted patterns (Fig. 4). This approach is suitable for environments that receive real-time data from manufacturing processes and detect anomalies [39], [40].

In our study, we develop an anomaly detection model using unsupervised learning. To evaluate the performance of the unsupervised learning model, we consider reconstruction error and anomaly score [41].

$$e^{(i)} = \left\| x^{(i)} - \widehat{x}^{(i)} \right\|, \tag{1}$$

$$a^{(i)} = \left(e^{(i)} - \mu\right)^{T} \Sigma^{-1} \left(e^{(i)} - \mu\right), \tag{2}$$

The reconstruction error is a method for calculating the error for restoration during the anomaly inference process, utilizing the absolute error, as shown in Equation (1). The anomaly score, as shown in Equation (2), represents the abnormality measure of point i using the mean and covariance. If the anomaly score exceeds a specified $Threshold(\tau)$, point (i) is defined as an abnormal interval when $a^{(i)} > \tau$. Through this, it is possible to determine the abnormality of each point and interval during the inference phase and by using specific criteria, it can be used as an alarm for anomaly indicators such as warning lights or anomaly monitoring [18].

B. MODEL EVALUATION

This study evaluates the performance of the unsupervised learning-based LSTM–autoencoder using a test dataset considering real-time environments. The anomaly detection model was trained on a 1-minute basis using a lot of Product Information (PO_CD) collected over a year. The evaluation units for the test set, which is considered a real-time factory environment, are set at 3-second intervals to evaluate the performance of the model.

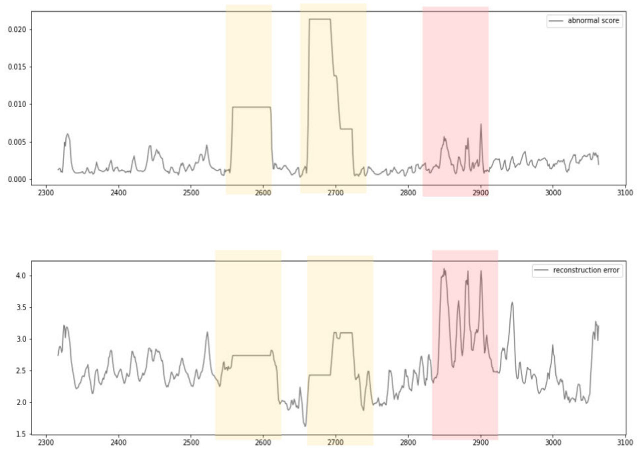
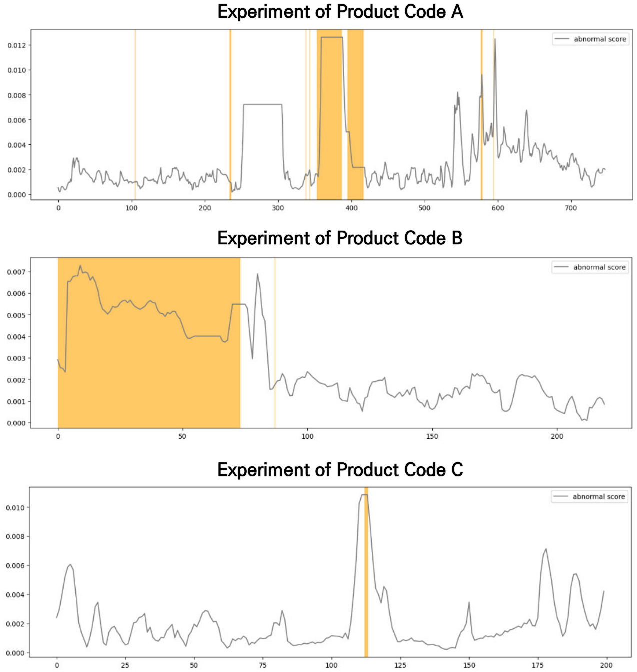


FIGURE 6. Examples of anomaly score by calculating reconstruction error (x-axis: timestamp, y-axis: reconstruction error or anomaly score).

The anomaly detection model calculates the reconstruction loss, computes the anomaly score, and derives the anomaly classification results. When the volatility of the time series trend of the reconstruction loss changes, the anomaly score increases due to the covariance calculation (Fig. 6). The performance of the anomaly detection model is evaluated by checking the HTP Alarm and HTP RIDS at points with high anomaly scores (Fig. 7). The yellow areas indicate anomaly patterns according to the actual HTP variables. The experiment tests three product codes (PO_CDs) from one product in an actual process environment. In this setting, the product operates for 1–6 hours, with a test set recorded at 3-second intervals. We also benchmark our proposed model's performance against a baseline model to validate its feasibility. The base model employs a supervised learning-based LSTM.


Experiment of Product Code A

Experiment of Product Code B

Experiment of Product Code C

FIGURE 7. Experimental results. We selected three random PO_CDs from products with high production. The experimental times were divided into different time bands (x-axis: timestamp, y-axis: anomaly score).

C. MODEL HYPER-PARAMETER

Among the collected data, 90% of the HTP variable consists of alarms associated with signs of abnormalities in the bundle packaging machine (packaging defects) and serves as an indirect signal for manufacturing downtime. The HTP-labeling alarm can confirm the downtime status in the actual manufacturing process and identify the abnormal point in the manufacturing process equipment data [18]. As a result, our study adjusts the threshold, which is the criterion for the anomaly score, and the parameters of the unsupervised learning-based LSTM–autoencoder using the HTP Alarm and HTP RIDS variable (Fig. 8). Anomaly alarms for the manufacturing process could not be used because of labeling noise issues [37]. However, with status labels like the IDEL state in HTP RIDS, we can pinpoint the abnormal sections in the actual process [18]. The anomaly detection model used in our study explores the threshold value for the anomaly score using HTP RIDS. The optimal threshold is determined using a grid search method based on the minimum and maximum anomaly scores derived from the validation set. The anomaly pattern of the LSTM–autoencoder model is optimized by comparing it with the HTP RIDS, defined as an abnormal duration section.

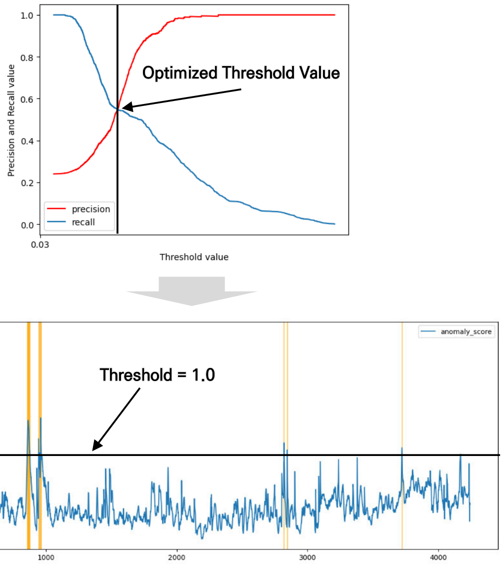


FIGURE 8. Examples of optimizing thresholds for anomaly alarms (x-axis: timestamp, y-axis: anomaly score).

After constructing the LSTM–autoencoder model, we initiate a parameter optimization process to fine-tune each anomaly detection model's performance (Table 3). This study determines the parameters by searching for the best-performing parameters using the HTP RIDS variable, which can identify abnormal sections. The parameters of thresholds for anomaly alarms and the LSTM-autoencoder model are set using the validation set. The anomaly pattern detection performance of the prediction model is evaluated on the test set that considers a real-time factory environment.

TABLE 3. Hyper-parameters for the best parameters of each model.

Hyperparameters Search Space	Product A	Product B	Product C
Threshold	0.013	0.002	0.010
Learning Rate	10^{-3}	10^{-3}	10^{-2}
Dropout	0.1	0.1	0.3
Batch Size	64	32	32
Latent Size	3	5	3
Window Size	5	3	3
LSTM Layer	5	4	3
Activation function		ReLU	
Optimizer		Adam	
Early Stopping		True	
Loss function		MSE Loss	

We explored various hyper-parameter configurations during the model development process. The learning rate was considered in the range of 10^{-5} to 10^{-1}, and batch sizes were varied from 32 to 128. Among different activation functions including Softmax, Sigmoid, tanh, and ReLU, we adopted ReLU. For the optimizer, we selected Adam, preferring it over SGD, Momentum, and others. To investigate the impact of network structure, we experimented with 2, 3, 4, and 5 LSTM layers, along with window sizes of 3, 5, and 10. The autoencoder's repeat vector was examined with latent sizes of 3, 5, 7, and 9. To mitigate the risk of overfitting in deep neural networks, we implemented early stopping and incorporated dropout layers into the model architecture. To optimize the performance, all case models in the study underwent hyperparameter tuning. Due to space limitations, only the performance statistics of the models with the best-performing hyperparameters are reported.

D. COMPARISONS OF MODEL RESULTS

The experimental results demonstrate that the anomaly score sharply increases in abnormal sections while staying low in normal sections (see Fig. 7). This pattern allows us to establish a threshold based on the anomaly score. This threshold can be used to set the alarm status for periods of anomalous patterns, especially in real-time anomaly alarm services. For instance, in the case of Product A, an alarm can be triggered for an outlier interval when the threshold is greater than or equal to 0.013. We conducted a comprehensive comparison of the manufacturing process status by utilizing the HTP variables in the abnormal sections identified through the model (high-score points). The yellow area denotes the occurrence of an HTP Alarm and signifies the downtime point in the manufacturing process. It enables effective identification of the HTP status in sections with relatively high anomaly scores. Conversely, sections with low anomaly scores exhibit a generally low frequency of HTP alarms and HTP RIDS. Quantification of the model's performance was accomplished by specifying thresholds for anomaly alarms using HTP variables (Table. 4). Despite being an unsupervised learning method, the proposed model demonstrates superior performance compared to the baseline model.

TABLE 4. Comparison of baseline model performance results.

Evaluation Metric	Baseline Model			LSTM-AE
	OCSVM	SVDD	LSTM	
Accuracy	0.407	0.816	0.854	0.963
Precision	0.143	0.750	0.581	0.821
Recall	0.510	0.820	0.262	0.808
F1-Score	0.193	0.780	0.363	0.772
AUC	0.480	0.620	0.613	0.928

Note: The baseline model encompasses a combination of several techniques. For each baseline model, the hyper-parameters were optimized. It follows the same alarm labeling scheme as the proposed model. The evaluation metrics of the baseline model were computed by averaging the results obtained from three distinct products (Product A, B, C). These metrics provide a quantitative assessment of the model's performance.

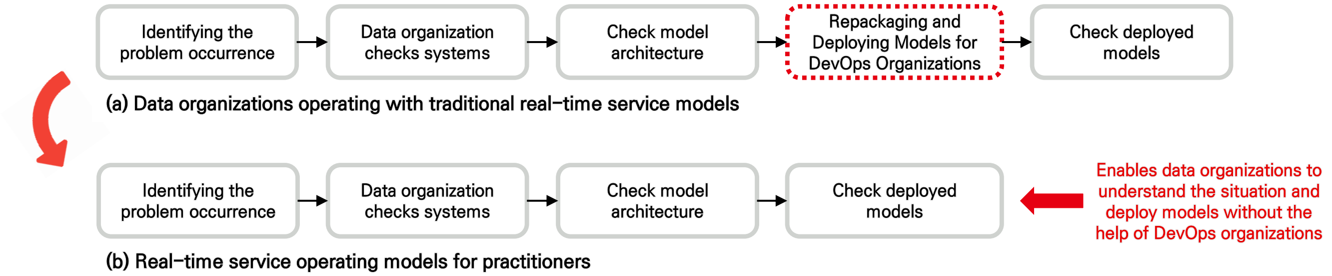


FIGURE 9. **Considerations for configuring a real-time service model based on the practitioner's organizational environment.**

TABLE 5. **Model performance for each product based on a threshold.**

Evaluation Metric	Product A	Product B	Product C
Accuracy	0.949	0.944	0.995
Precision	0.933	0.863	0.667
Recall	0.438	0.986	1.000
F1-Score	0.595	0.920	0.800
AUC	0.809	0.977	0.998

We conducted performance comparisons with other baseline models to justify our model selection [2], [13], [22]. One-Class Support Vector Machine (OCSVM) is a variant of Support Vector Machines that addresses binary classification problems by distinguishing normal and anomalous data. Support Vector Data Description (SVDD) is an unsupervised learning algorithm that leverages only normal data from a single class to establish a boundary and detect outliers. Long Short-Term Memory (LSTM), as a recurrent neural network, is well-suited for capturing long-term dependencies and modeling sequences of extended length. The comparison with the baseline model shows that the LSTM-AE model is superior. This supports that the unsupervised learning-based reconstruction approach is suitable and superior in our manufacturing process context [13]. We further confirmed the performance of our proposed model, focusing specifically on three product codes as shown in Table 5. Our findings demonstrate that the proposed method exhibits superior capabilities in forecasting multivariate time series data. It is important to note, however, that this comparison is based on actual anomaly labels and may not necessarily reflect a fully precise outcome. In other words, it is worth noting that false alarms with high anomaly scores may occasionally arise in normal sections. To address this concern, future real-time services can adjust false alarm rates by modifying the number of parameters of learning for the specific product [37]. Additionally, we propose adopting a multi-label (normal, warning, abnormal) approach for anomaly alarms instead of a binary (normal and abnormal) classification [32].

E. SERVICE MODEL DEVELOPMENT

To use the anomaly detection model in a real-world manufacturing environment, it is necessary to consider optimizing the model for optimal performance, such as hyperparameter estimation and determining the threshold for abnormal sections. The convenience of redeploying the anomaly detection model, learned from new data, should also be considered [11], [29]. Traditionally, real-time service models are operated by data operations organizations, which are well suited to re-modify models or operate real-time models (see Fig. 9 (a)). However, interviews with practitioners reveal that manufacturing processes in smart factories lack data development organizations and that it is difficult to redeploy models in practice. The convenience of model optimization should be considered without the help of a data development organization in many cases [10], [32]. We consider a design that allows for repackaging and deployment of models without involving the development and operations (DevOps) organization (see Fig. 9 (b)).

Our study uses the BentoML library to build a real-time service model, considering the convenience of real-time deployment of the anomaly detection model. BentoML is an online prediction-serving library that returns results as soon as the model requests them and supports various ML frameworks (e.g., Scikit-Learn, PyTorch, and Tensorflow2) and DL libraries (e.g., Transformers and PyTorch Lightning). BentoML is used because convenience in smart factory management and inference speed is important.

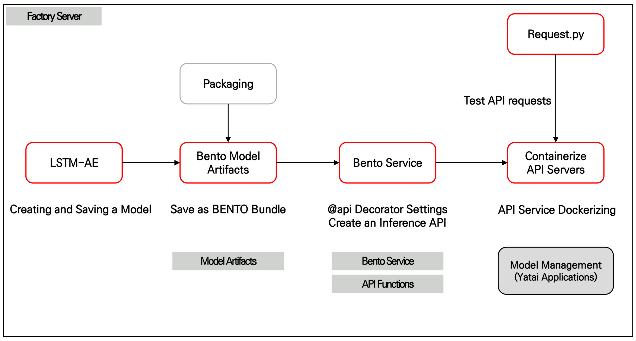
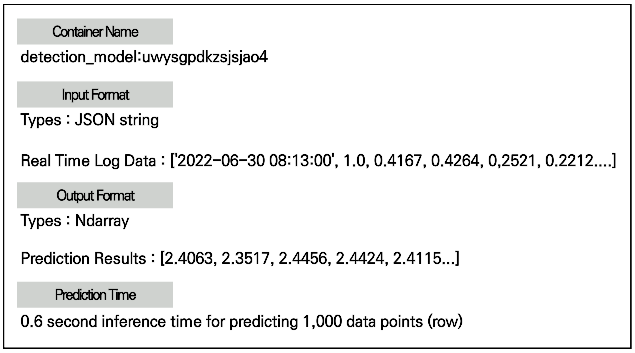


FIGURE 10. **Configuring the service model in the factory.**

Fig. 10 illustrates the procedure for using BentoML. First, we develop the LSTM–autoencoder model and store it on the server. The model artifact step involves saving it in a format suitable for the BentoML library. BentoML has the advantage of being able to package artifacts optimized according to the model type (e.g., Transformers and Scikit-Learn), and the artifacts can include model weight files, tokenizers, and others. Bento Service is the most basic component for configuring the inference service of the model using BentoML. It is a single unit containing a summary of everything required for the prediction execution of the model. An application programming interface (API) object can be declared in the decorator format of Bento Service (@api). It defines the inference logic for the input data, and settings such as input/output format and batching are possible through Adapters. Finally, containerization is performed through the dockerization of an API server to generate prediction results according to data requests. BentoML has the advantage of easy handling during the dockerization process and supporting integration with the Model Management library (Yatai), which serves as storage for trained models.



Container Name

detection_model:uwysgpdkzsjsjao4

Input Format

Types : JSON string

Real Time Log Data : ['2022-06-30 08:13:00', 1.0, 0.4167, 0.4264, 0,2521, 0.2212....]

Output Format

Types : Ndarray

Prediction Results : [2.4063, 2.3517, 2.4456, 2.4424, 2.4115...]

Prediction Time

0.6 second inference time for predicting 1,000 data points (row)

FIGURE 11. Real-time service model API test results.

As depicted in Fig. 11, a real-time service model is built using BentoML, and it provides the response result for the requested data. When sending data in the JSON string format, the trained anomaly detection model returns a predicted value. This inferred return value can be used to check anomaly alarms by specifying anomaly sections through the threshold among the previously trained parameters. The prediction time can be confirmed immediately using real-time data. Thus, our study presents specific utilization strategies for the BentoML library, specialized in real-time service deployment, to develop a fast anomaly detection infrastructure for smart factories.

V. DISCUSSION AND IMPLICATIONS

A. DISCUSSION

Smart factories are actively investing in AI fields such as anomaly detection to improve their business competitiveness and production efficiency [9], [11]. Previous studies have not considered issues arising from the development of anomaly detection models, such as data labeling. There also has been a lack of research reflecting the real-world environment within smart factory systems [2], [13], [22]. Going beyond previous research, our study developed an anomaly detection model that can apply real-time services in the manufacturing process following the procedures of the design science method, considering the data labeling issue.

Previous studies on anomaly detection have attempted to predict anomalies in the manufacturing process using various AI technologies. However, they have faced obvious limitations regarding data noise and real-time operation models [34], [42]. A supervised learning-based LSTM model with alarm notification was suggested for detecting anomalies in manufacturing process data. This approach could provide predictive maintenance alerts before anomalies occur. However, when manufacturing equipment data contains significant labeling noise, supervised learning-based models may struggle to accurately detect anomalous patterns [18], [34]. In contrast, our study developed an unsupervised learning-based LSTM-autoencoder model that does not require target labeling. This approach has the advantage of reliably detecting anomalous patterns in real-time time-series process data. It overcame the limitations arising from insufficient abnormal data [42]. Our study also proposed an optimal hyperparameter exploration method using various manufacturing process data and equipment alarm data. To verify the feasibility of the proposed method, we evaluated it using three product codes. The performance evaluation results showed that abnormal patterns occurred more frequently in sections with high anomaly scores. Our study highlights the need to identify and address early downtime situations such as "molding defects" and "product falling" in the manufacturing process by detecting anomaly patterns in real-time through unsupervised learning.

To enhance the practical applicability of our work, we suggest providing simple physical explanations for the predicted failures. In line with this, we conducted interviews with domain experts to gain deeper insights into the potential causes of anomalies in manufacturing processes. These interviews revealed the main empirical factors that lead to anomalies. environmental factors such as temperature and humidity cause more anomalous patterns, which in turn lead to "molding defects". We propose labeling anomalies based on these critical empirical factors. We are exploring the incorporation of a multi-class classification scheme that differentiates between 'normal', 'warning', and 'abnormal' states within the manufacturing process. By employing this expanded taxonomy, we can shed light on the potential causes of anomalous conditions and enhance the interpretability of our anomaly detection system [5], [32]. This labeling approach will offer a more meaningful understanding of the underlying causes behind the predicted failures.

Furthermore, we developed a model considering real-time operation services and presented practical application methods that can be used within the manufacturing process. Literature related to anomaly detection has focused on developing algorithms to predict abnormal patterns using previous data [2], [8], [28]. However, these studies do not consider the data operation environment for developing models with real-time operation services in mind. In contrast, our study developed an anomaly detection model that considers real-time operation services. This model is designed for convenient deployment within smart factories using BentoML. This approach was taken with the convenience of direct operation in the manufacturing process in mind, rather than having data development organizations develop and operate the models themselves [12], [35]. BentoML was also suitable for manufacturing equipment data at 3-second intervals, requiring fast inference speeds. Through the anomaly detection model considering real-time operation services, initiative-taking responses to abnormalities are expected. As a result, a more efficient smart factory system can reduce downtime during the manufacturing process [43].

B. LIMITATIONS AND FUTURE RESEARCH

Despite our efforts regarding the application of LSTM-Autoencoder for anomaly detection in a smart factory setting, our study has several limitations and future research directions. First, our LSTM–autoencoder model is a DL algorithm composed of many parameters and layers. However, there still exists the black box issue: it is difficult to understand the prediction results of anomalies using DL technologies [5]. Explanations are required for the results predicted as anomalies [5], [32]. We propose future research directions that involve incorporating explanatory artificial intelligence (XAI) methodologies, such as surrogate models, local interpretable model-judgment explanation (LIME), and Shapley Additive exPlanations (SHAP). These approaches aim to provide insights into the underlying causes of anomalous patterns and offer explanations for why anomalies occur. By considering these methodologies, we aim to enhance the practical applicability of our work by providing simple physical explanations for predicted failures and addressing their root causes. These additions will bring significant value to smart factory operators and practitioners seeking to understand and mitigate anomalies in their operations.

Second, our research advances an unsupervised learning-based anomaly detection model, obviating the need for ground normal labels. Conventional anomaly alarm often exhibit limitations in predictive efficiency due to significant noise, necessitating an accurate assessment of the developed model concerning anomaly alarms [19], [37]. In this context, reconstruction loss and anomaly scores serve as quantitative evaluation metrics; however, it is inherently limited in its ability to definitively signal anomaly alarms. If future research accomplishes precise data labeling for abnormal alarms within manufacturing equipment and identifies the optimal threshold, it could substantially further the field of anomaly detection research [18].

Finally, our study developed an anomaly detection model, optimizing the threshold parameter for anomaly scores utilizing data from abnormal sections. Nevertheless, we did not consider additional equipment alarm data, including optimal learning cycles for products or specific product characteristics (e.g., capacity, type, packaging material). By optimizing the product cycle for future training, we anticipate improvements in the performance of anomaly detection.

C. IMPLICATIONS FOR RESEARCH AND PRACTICE

The academic implications of our study are as follows. First, our study differentiates itself from previous studies by developing an anomaly detection model using the design science research methodology [9], [18], [37]. The significance of our research lies in following the design science procedure for problem-solving within smart factories, such as providing a process and verifying prediction performance. The anomaly detection model developed through this process can improve the limitations of existing anomaly detection models in smart factories by defining and proposing solutions to real-world problems [9].

Second, this study has a contribution by developing and applying a DL model based on unsupervised learning for anomaly detection in smart factories [6]. Previous anomaly detection literature has developed supervised learning-based anomaly detection models based on anomaly alarm data. However, these studies lacked concrete plans for real-time services [33]. Our study has differentiated significance compared with previous research by developing an unsupervised learning-based anomaly detection model to address data labeling issues and proposing a real-time prediction model application plan within smart factories.

Finally, this study proposed a framework for real-time anomaly detection integrated within smart factories. To mitigate the downtime during the manufacturing process, this study proposed a classification and prediction model that helps detect signs of anomalies using previous data from SCADA systems. This integrated framework has demonstrated applicability for detecting anomalies in real-time operation services. This approach can effectively perform periodic learning of training data. As a result, by regularly updating the parameters of the anomaly detection model within the manufacturing system, the efficiency of real-time operation model can be enhanced [8], [28]. By proposing an anomaly detection framework from such an integrated perspective, it is possible to explore implicit information about the performance degradation state of subsystems in complex systems. It is also possible to identify abnormal events during operation, which can contribute to a real-time maintenance planning program.

This study also has several practical implications. First, by detecting and addressing issues that may arise in the manufacturing process using anomaly detection models, it is possible to improve production efficiency [43]. That is, by proactively detecting and addressing abnormal situations occurring in the packaging line of the manufacturing process, we can reduce operating costs [9].

Second, anomaly detection can reduce downtime and the impact of equipment errors on production. That is, anomaly detection can help identify equipment failures early, enabling preventive maintenance and reducing maintenance costs [1].

Finally, understanding anomalies in advance using real-time systems is critical for cost savings and improving equipment availability [40]. Practitioners can verify the process status based on monitoring signals. They can analyze signals generated in smart factories to determine the source of anomalies and whether repairs are required [9]. For the target system under consideration for in-operation state monitoring, it is expected that the system can distinguish between normal and abnormal states.

In conclusion, the equipment data collected in smart factories can be used for real-time anomaly detection. The proposed anomaly detection model proposed in this study for smart factories can offer substantial advantages in terms of enhanced production efficiency, minimized downtime, improved product quality, and cost-effectiveness.

REFERENCES

[1] H. M. Hashemian, "State-of-the-art predictive maintenance techniques," *IEEE Trans. Instrum. Meas.*, vol. 60, no. 1, pp. 226–236, Jan. 2011.

[2] Y. Wang, M. Perry, D. Whitlock, and J. W. Sutherland, "Detecting anomalies in time series data from a manufacturing system using recurrent neural networks," *J. Manuf. Syst.*, vol. 62, pp. 823–834, Jan. 2022.

[3] J. Davis, T. Edgar, J. Porter, J. Bernaden, and M. Sarli, "Smart manufacturing, manufacturing intelligence and demand-dynamic performance," *Comput. Chem. Eng.*, vol. 47, pp. 145–156, Dec. 2012.

[4] A. Kusiak, "Smart manufacturing must embrace big data," *Nature*, vol. 544, no. 7648, pp. 23–25, Apr. 2017.

[5] H. Choi, D. Kim, J. Kim, J. Kim, and P. Kang, "Explainable anomaly detection framework for predictive maintenance in manufacturing systems," *Appl. Soft Comput.*, vol. 125, Aug. 2022, Art. no. 109147.

[6] G. Pang, C. Shen, L. Cao, and A. V. D. Hengel, "Deep learning for anomaly detection: A review," *ACM Comput. Surv.*, vol. 54, no. 2, pp. 1–38, Mar. 2021.

[7] B. Steenwinckel, D. De Paepe, S. Vanden Hautte, P. Heyvaert, M. Bentefrit, P. Moens, A. Dimou, B. Van Den Bossche, F. De Turck, S. Van Hoecke, and F. Ongenae, "FLAGS: A methodology for adaptive anomaly detection and root cause analysis on sensor data streams by fusing expert knowledge with machine learning," *Future Gener. Comput. Syst.*, vol. 116, pp. 30–48, Mar. 2021.

[8] T. Skripcak and P. Tanuska, "Utilisation of on-line machine learning for SCADA system alarms forecasting," in *Proc. Sci. Inf. Conf.*, Oct. 2013, pp. 477–484.

[9] K. Choi, J. Yi, C. Park, and S. Yoon, "Deep learning for anomaly detection in time-series data: Review, analysis, and guidelines," *IEEE Access*, vol. 9, pp. 120043–120065, 2021.

[10] M. Razavian, G. Procaccianti, and D. A. Tamburri, "Four-dimensional sustainable e-services," in *Proc. EnviroInfo*, 2014, pp. 221–228.

[11] V. Rousopoulou, T. Vafeiadis, A. Nizamis, I. Iakovidis, L. Samaras, A. Kirtsoglou, K. Georgiadis, D. Ioannidis, and D. Tzovaras, "Cognitive analytics platform with AI solutions for anomaly detection," *Comput. Ind.*, vol. 134, Jan. 2022, Art. no. 103555.

[12] D. A. Tamburri, "Sustainable MLOps: Trends and challenges," in *Proc. 22nd Int. Symp. Symbolic Numeric Algorithms Sci. Comput. (SYNASC)*, Sep. 2020, pp. 17–23.

[13] H. D. Nguyen, K. P. Tran, S. Thomassey, and M. Hamad, "Forecasting and anomaly detection approaches using LSTM and LSTM autoencoder techniques with the applications in supply chain management," *Int. J. Inf. Manage.*, vol. 57, Apr. 2021, Art. no. 102282.

[14] L. Bontemps, V. L. Cao, J. McDermott, and N.-A. Le-Khac, "Collective anomaly detection based on long short-term memory recurrent neural networks," in *Future Data and Security Engineering*. Can Tho, Vietnam: Springer, Nov. 2016, pp. 141–152.

[15] P. Malhotra, L. Vig, G. Shroff, and P. Agarwal, "Long short term memory networks for anomaly detection in time series," in *Proc. ESANN*, 2015, p. 89.

[16] M. Hu, Z. Ji, K. Yan, Y. Guo, X. Feng, J. Gong, X. Zhao, and L. Dong, "Detecting anomalies in time series data via a meta-feature based approach," *IEEE Access*, vol. 6, pp. 27760–27776, 2018.

[17] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, "Estimating the support of a high-dimensional distribution," *Neural Comput.*, vol. 13, no. 7, pp. 1443–1471, Jul. 2001.

[18] A. Lundström, M. O'nils, F. Z. Qureshi, and A. Jantsch, "Improving deep learning based anomaly detection on multivariate time series through separated anomaly scoring," *IEEE Access*, vol. 10, pp. 108194–108204, 2022.

[19] S. Bulusu, B. Kailkhura, B. Li, P. K. Varshney, and D. Song, "Anomalous example detection in deep learning: A survey," *IEEE Access*, vol. 8, pp. 132330–132347, 2020.

[20] G. P. Zhang, "Time series forecasting using a hybrid ARIMA and neural network model," *Neurocomputing*, vol. 50, pp. 159–175, Jan. 2003.

[21] S. Siami-Namini, N. Tavakoli, and A. Siami Namin, "A comparison of ARIMA and LSTM in forecasting time series," in *Proc. 17th IEEE Int. Conf. Mach. Learn. Appl. (ICMLA)*, Dec. 2018, pp. 1394–1401.

[22] R. Chalapathy and S. Chawla, "Deep learning for anomaly detection: A survey," 2019, *arXiv:1901.03407*.

[23] K. Deng, X. Zhang, Y. Cheng, Z. Zheng, F. Jiang, W. Liu, and J. Peng, "A remaining useful life prediction method with long-short term feature processing for aircraft engines," *Appl. Soft Comput.*, vol. 93, Aug. 2020, Art. no. 106344.

[24] A. L. Ellefsen, E. Bjørlykhaug, V. Æsøy, S. Ushakov, and H. Zhang, "Remaining useful life predictions for turbofan engine degradation using semi-supervised deep architecture," *Rel. Eng. Syst. Saf.*, vol. 183, pp. 240–251, Mar. 2019.

[25] T. Xia, Y. Song, Y. Zheng, E. Pan, and L. Xi, "An ensemble framework based on convolutional bi-directional LSTM with multiple time windows for remaining useful life estimation," *Comput. Ind.*, vol. 115, Feb. 2020, Art. no. 103182.

[26] B. Lim and S. Zohren, "Time-series forecasting with deep learning: A survey," *Phil. Trans. Roy. Soc. A*, vol. 379, no. 2194, 2021, Art. no. 20200209.

[27] O. I. Provotar, Y. M. Linder, and M. M. Veres, "Unsupervised anomaly detection in time series using LSTM-based autoencoders," in *Proc. IEEE Int. Conf. Adv. Trends Inf. Theory (ATIT)*, Dec. 2019, pp. 513–517.

[28] I. Garitano, R. Uribeetxeberria, and U. Zurutuza, "A review of SCADA anomaly detection systems," in *Soft Computing Models in Industrial and Environmental Applications, 6th International Conference SOCO 2011*. Berlin, Germany: Springer, 2011, pp. 357–366. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-642-19644-7_38

[29] M. Antonini, M. Pincheira, M. Vecchio, and F. Antonelli, "An adaptable and unsupervised TinyML anomaly detection system for extreme industrial environments," *Sensors*, vol. 23, no. 4, p. 2344, Feb. 2023.

[30] E. Principi, D. Rossetti, S. Squartini, and F. Piazza, "Unsupervised electric motor fault detection by using deep autoencoders," *IEEE/CAA J. Autom. Sinica*, vol. 6, no. 2, pp. 441–451, Mar. 2019.

[31] J. Pereira and M. Silveira, "Unsupervised anomaly detection in energy time series data using variational recurrent autoencoders with attention," in *Proc. 17th IEEE Int. Conf. Mach. Learn. Appl. (ICMLA)*, Dec. 2018, pp. 1275–1282.

[32] C. Shang and F. You, "Data analytics and machine learning for smart process manufacturing: Recent advances and perspectives in the big data era," *Engineering*, vol. 5, no. 6, pp. 1010–1016, Dec. 2019.

[33] X. Xie, B. Wang, T. Wan, and W. Tang, "Multivariate abnormal detection for industrial control systems using 1D CNN and GRU," *IEEE Access*, vol. 8, pp. 88348–88359, 2020.

[34] T. Lee, K. B. Lee, and C. O. Kim, "Performance of machine learning algorithms for class-imbalanced process fault detection problems," *IEEE Trans. Semicond. Manuf.*, vol. 29, no. 4, pp. 436–445, Nov. 2016.

[35] L. Zhou, S. Pan, J. Wang, and A. V. Vasilakos, "Machine learning on big data: Opportunities and challenges," *Neurocomputing*, vol. 237, pp. 350–361, May 2017.

[36] A. R. Hevner, S. T. March, J. Park, and S. Ram, "Design science in information systems research," *MIS Quart.*, vol. 28, no. 1, p. 8, Mar. 2004.

[37] A. Garg, W. Zhang, J. Samaran, R. Savitha, and C.-S. Foo, "An evaluation of anomaly detection and diagnosis in multivariate time series," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 6, pp. 2508–2517, Jun. 2022.

[38] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," 2013, *arXiv:1312.6114*.

[39] B. Choi and J. Jeong, "ViV-Ano: Anomaly detection and localization combining vision transformer and variational autoencoder in the manufacturing process," *Electronics*, vol. 11, no. 15, p. 2306, Jul. 2022.

[40] B. Chen, J. Wan, L. Shu, P. Li, M. Mukherjee, and B. Yin, "Smart factory of industry 4.0: Key technologies, application case, and challenges," *IEEE Access*, vol. 6, pp. 6505–6519, 2018.

[41] K. Qian, Y. Zhang, S. Chang, X. Yang, and M. Hasegawa-Johnson, "Autovc: Zero-shot voice style transfer with only autoencoder loss," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 5210–5219.
[42] F. B. Saghezchi, G. Mantas, M. A. Violas, A. M. de Oliveira Duarte, and J. Rodriguez, "Machine learning for DDoS attack detection in industry 4.0 CPPSs," *Electronics*, vol. 11, no. 4, p. 602, Feb. 2022.
[43] J. Friederich, D. P. Francis, S. Lazarova-Molnar, and N. Mohamed, "A framework for data-driven digital twins of smart manufacturing systems," *Comput. Ind.*, vol. 136, Apr. 2022, Art. no. 103586.

KYUNG SUNG LEE is currently pursuing the Ph.D. degree with the Graduate School of Information, Yonsei University, Seoul, South Korea. His research interests focus on business value chain optimization. Specifically, he is interested in topics related to big data analytics for manufacturing.

SEONG BEOM KIM is currently pursuing the Ph.D. degree with the Graduate School of Information, Yonsei University, Seoul, South Korea. His research interests focus on applied AI (deep learning applications), recommender systems, and data science. Specifically, he is interested in topics related to social media and customer behavior in e-commerce.

HEE-WOONG KIM (Member, IEEE) is a Professor with the Graduate School of Information, Yonsei University, South Korea. Before joining Yonsei University, he was a Faculty Member with the Department of Information Systems and Analytics, National University of Singapore (NUS). He has served on the Editorial Board of the *MIS Quarterly*, *Journal of the Association for Information Systems*, and IEEE Transactions on Engineering.

• • •