EAI Endorsed Transactions
on Scalable Information Systems
Research Article EAI.EU

Human-Centric Smart Manufacturing under Industry 5.0:IIoT-Enabled Machine Learning for Real-Time Fault Identification and Adaptive Workforce Scheduling

Yaocong Xie [1], Ning Wang [2,3,*]

[1] South China Agricultural University , Guangzhou, China
[2] INTI International University , NiLai , Malaysia
[3] Guangzhou Huashang College, Guangzhou, China

Abstract

This study addresses Industry 5.0's demand for human-machine collaboration and manufacturing resilience by developing machine learning models for rapid fault identification and adaptive personnel scheduling. When a production-line fault occurs, the proposed approach uses IIoT data to identify the fault category quickly, support targeted intervention, and facilitate the restoration of normal production. XGBoost achieved near-perfect fault identification accuracy in model evaluation and accuracy and recall of 1.0 in validation, significantly outperforming the Random Forest baseline. Analysis revealed that experienced operators increased product qualification rates by 15% but reduced output by 10% due to physical factors. Decision tree regression minimized scheduling errors with an MSE of 0.49 and an R2 of 0.83. These results demonstrate that integrating IIoT-driven rapid fault identification with intelligent staffing provides a practical basis for shortening the response cycle, improving recovery capability after disruptions, and strengthening the resilience of human-centric manufacturing systems.

Keywords: rapid fault identification; XGBoost; personnel scheduling; manufacturing resilience; Industrial Internet of Things

Received on 04 June 2026, accepted on 29 June 2026, published on 20 July 2026

doi: 10.4108/ eetsis.13326

1. Introduction

1.1. Research Background

The evolution towards Industry 4.0/5.0 has heightened the importance of real-time fault detection in manufacturing [1]. Traditional statistical methods, though capable of identifying basic anomalies, often struggle with complex industrial data [2-3]. This limitation underscores the need for adaptive, data-driven approaches that integrate fault prediction with human-centric scheduling, a gap central to achieving responsive smart factories [4-5]. In this context, fault identification should not be treated only as an accuracy-oriented classification task. Its practical value lies in helping managers recognize the fault type promptly after a disruption, organize targeted inspection and repair, and restore normal production with less operational interruption. This rapid recognition-response-recovery capability is an important component of manufacturing resilience.

1.2. Literature Review

Machine learning is increasingly applied in manufacturing for quality control, fault diagnosis, and predictive maintenance. Supervised methods such as SVM

* Corresponding author. Email: 3219783281@qq.com

and Random Forests are widely used for anomaly classification [6], while deep learning models like CNNs and autoencoders excel in complex defect detection [7-8]. Image-based deep learning further enhances prognostics accuracy [9].
XGBoost has become prominent for fault prediction due to its efficiency and interpretability, with applications in pipeline safety and system diagnostics [10-11]. Its feature importance analysis supports operational transparency [12]. Optimization algorithms like WOA improve model performance through hyperparameter tuning [13-14].
IIoT and big data ecosystems have advanced predictive maintenance [15-16], and model management platforms aid industrial ML deployment [17]. However, challenges remain, including model adaptability to dynamic processes [18], the handling of heterogeneous data, and the need for interpretability in human-machine collaboration [19]. A key unmet need is the integration of real-time fault identification with adaptive staffing and recovery-oriented decision support. Such an integration is essential not only for optimizing equipment and human resources but also for improving the ability of production lines to respond to disturbances and recover normal operations [20-21].

1.3. Manufacturing Resilience Significance

Manufacturing resilience refers here to a production system's capacity to respond to faults, reduce operational disruption, and recover its normal operating state. The proposed framework is designed to support this capacity in two ways. First, rapid classification of the fault category provides a basis for directing maintenance resources toward the affected device rather than relying on broad manual inspection. Second, personnel scheduling can allocate suitable operators during recovery and subsequent operation. By connecting fault recognition, targeted response, and staffing adjustment, the framework supports a more continuous recovery process and improves the production line's ability to withstand operational disturbances.

2. Model Establishment

2.1. Data feature analysis

Data preprocessing and visualization were performed using Python libraries including Pandas, Matplotlib, and Seaborn. The production line fault data were cleaned and analyzed to identify nine primary fault types and their frequencies. As shown in Table 1, the material push device 1001 exhibited the highest fault count with 37,008 instances, followed by faults in the screw capping and filling devices.

These failures primarily reflect issues in positioning systems, sensor detection, and operator execution, indicating the need for improved maintenance, system optimization, and operator training to enhance production stability. From a resilience perspective, high-frequency fault points should receive priority in inspection planning because rapid localization can help reduce disruption and support the recovery of normal production.

Table 1. Failure frequency table

Fault name	frequency
Material push device failure 1001	37008
Material detection device failure 2001	23254
Fill device detection fault 4001	28062
Fill device positioning fault 4002	26371
Facking fault 4003	27020
Cover the device location fault 5001	25892
Cdevice cover fault 5002	34125
Screwing cover device positioning fault 6001	25962
Wwing cover device twist cover fault 6002	28998

2.2. Division of the test set and the training set

The dataset comprises annual, daily, and basic data components, which were processed through the Jupyter Python interface. Using the train_test_split function from sklearn.model_ selection, the data were partitioned into training and test sets with a 70%:30% split ratio.

Model performance was assessed using three key evaluation metrics: accuracy, recall, and the ROC curve.

An imbalanced distribution between positive and negative samples was observed in the dependent variable, as illustrated in Figure 1.

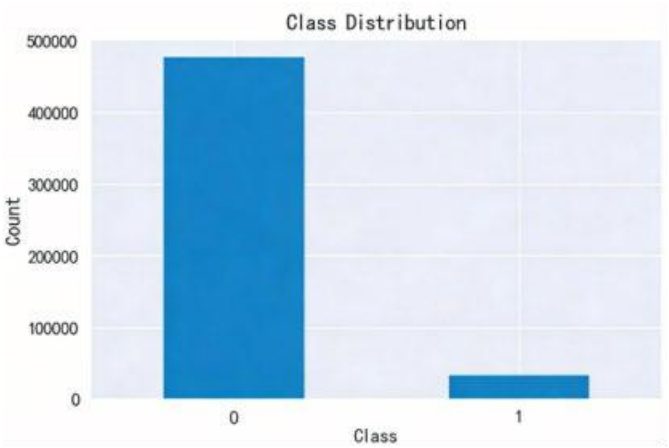


Figure 1. Class distribution

2.3. Model construction

Six machine learning models were implemented to classify production line faults, including Logistic Regression, Random Forest, XGBoost, AdaBoost, Decision Tree, and GBDT. The dependent variables comprised specific fault categories such as material push device failure 1001 and material detection device failure 2001. Independent variables encompassed operational states and numerical indicators related to material handling, filling, capping, and quality inspection, such as timestamp and material push cylinder status.

All models achieved an AUC value of 1.00 on the ROC curve, indicating excellent classification performance on the given dataset, as shown in Figure 2.

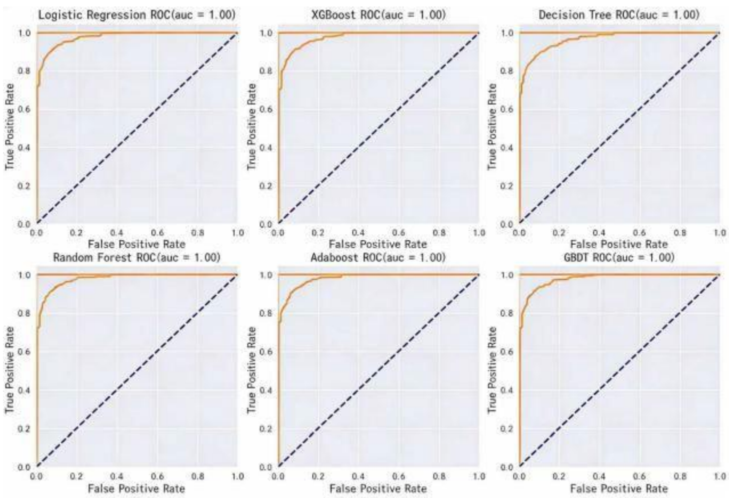


Figure 2. ROC plot

Features were extracted from the provided data, covering device type, fault occurrence time, fault duration, and other relevant attributes.

Given that all six models performed strongly in preliminary evaluations, subsequent analysis focuses on comparing two representative algorithms: XGBoost and Random Forest.

3. Testing, Application, and Analysis

3.1. Random Forest Model Evaluation

As shown in Table 2, the Random Forest model achieved an accuracy of 93%, indicating its ability to correctly classify most samples. However, its recall was extremely low, approaching zero, leading to a correspondingly low F-score. This suggests that the model performed poorly in detecting positive cases, likely due to class imbalance within the dataset, scarcity of positive samples, or inherent limitations in identifying minority-class instances.

Table 2. Summary table of random forest assessment

	Inspection value
ACC	0.93
REC	0.00
F-score	0.07

3.2. XGBoost Evaluation results

Based on Table 3, the model achieves an accuracy and F-score both above 0.99, demonstrating its ability to correctly classify nearly all samples and maintain a strong balance between precision and recall. Furthermore, with recall reaching 1.0, the model successfully identifies all true positive samples without omission. These results collectively indicate excellent performance in the classification task, reflecting high accuracy and comprehensive detection capability.

Table 3. Summary table of the XGBoost assessments

	Inspection value
ACC	0.9999739845858672
REC	1.0
F-score	0.9998186927749071

Table 4 presents the model′s performance metrics for the binary classification task. As shown, precision, recall, and F1-score all achieve 1.00 for both class 0 and class 1, indicating perfect classification without any misidentification or omission. The overall accuracy also reaches 1.00, reflecting correct predictions across all samples. Furthermore, both macro-average and weighted-average metrics attain optimal values, further confirming the model's outstanding classification capability.

Table 4. XGBoost Classification detailed assessment table

	precision	recall	f1-score	support
0	1.00	1.00	1.00	142726
1	1.00	1.00	1.00	11029
accuracy			1.00	153755
Macro avg	1.00	1.00	1.00	153755
Weighted avg	1.00	1.00	1.00	153755

3.3. Application and Validation of the Fault Alarm Model

The XGBoost algorithm in Problem 1 is applied to model the data in Attachment 2. From Table 5, the accuracy rate (ACC) reaches a perfect 1.0, which means that the model can accurately identify the true categories of all the samples. Meanwhile, the recall rate (REC) is also 1.0, indicating that the model did not miss any samples belonging to the target category. The F-score, as the harmonic mean of precision and recall, also reached 1.0, further validating the perfect performance of the model on the classification task.

Table 5. Results of the XGBoost evaluation

	Inspection value
ACC	1.0
REC	1.0
F-score	1.0

This data clearly demonstrates the excellent performance of the model in the binary classification task. For categories 0 and 1, the model precision, recall and F1-score were perfect 1.00. This means that the model neither misjudges nor misses when identifying samples in each category, showing extremely high accuracy and completeness.

According to Table 6, the overall accuracy also reached 1.00, further verifying the model's excellent classification ability across the entire dataset. Both the macro-average and weighted-average indicators demonstrate consistently high performance.

From an operational perspective, the value of the model is not limited to classification accuracy. Once an alarm is triggered, the predicted fault code can be mapped to the corresponding device and failure type, allowing managers to narrow the inspection scope and initiate targeted maintenance more quickly. The resulting response process can be summarized as fault occurrence - rapid identification - targeted intervention - production recovery. In this study, the model provides the analytical foundation for the rapid-identification stage; the actual reduction in downtime should be quantified further in future on-site deployment. Nevertheless, embedding this stage into the production response workflow is important for manufacturing resilience because it supports the faster restoration of normal production after disruptions.

Table 6. XGBoost Classification detailed evaluation Table

	precision	recall	f1-score	support
0	1.00	1.00	1.00	11131
1	1.00	1.00	1.00	11074
accuracy			1.00	22205
Macro avg	1.00	1.00	1.00	22205
Weighted avg	1.00	1.00	1.00	22205

3.4. Temporal and Duration Analysis of Faults

In order to analyze the total number of monthly failures and the longest and shortest duration, the monthly failure record data should be collected first, including the number of failure occurrences and the duration of each failure. The data are then organized to calculate the total number of failures in each month. Subsequently, the records with the longest and shortest durations are selected. These analysis results are organized in a table or chart to present the monthly fault situation visually, including fault frequency and duration. Figure 3 shows the total number of failures and the longest and shortest durations per month for fault number 1001 in M201. The total number of monthly failures of each unit and the longest and minimum durations are presented in the appendix. These temporal and duration indicators can also be used to identify fault points that require priority attention in resilience-oriented maintenance planning.

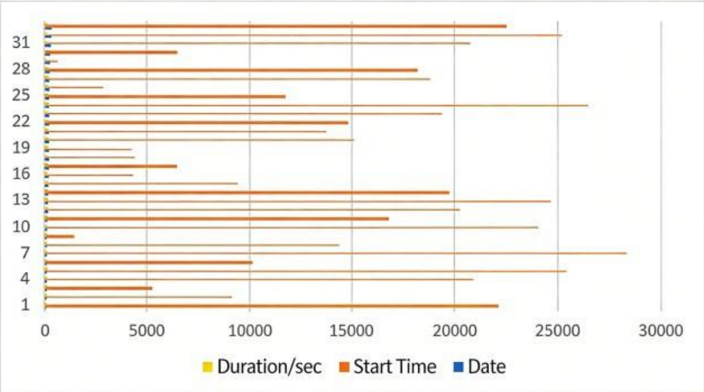


Figure 3. Bar graph of the fault characteristics

3.5. Analysis of Work Tenure Influence Based on Kruskal-Wallis Test

Data from ten production lines were integrated, encompassing metrics such as output, qualification rate, operator information, and fault records. The combined dataset was cleaned to remove errors and missing values, ensuring reliability for subsequent analysis. Data quality and sample sufficiency are crucial for result accuracy and statistical validity, laying a robust foundation for evaluating production performance.

According to Table 7，the relationship between operator tenure and production outcomes including total output, total defects, average efficiency, and qualification rate was analyzed using the Kruskal–Wallis test, a nonparametric method suitable for non-normally distributed data. Normality tests indicated significant departures from normality across all dependent variables, with p values below 0.05, justifying the use of nonparametric analysis.

Table 7. Test of normality

variable name	P values
Total production quantity	0.00
percent of pass	0.00
Average production efficiency	0.00
Total unqualified quantity	0.00

Given the non-normal distribution of the data, nonparametric analysis methods were employed. The Kruskal-Wallis test was selected as it is suitable for comparing median differences across three or more independent groups, aiding in understanding variations between different tenure levels and their impact on production indicators.

As shown in Table 8, the p-values for total output, qualification rate, average production efficiency, and total defect quantity are all 0.00, below the 0.05 significance threshold. This indicates statistically significant differences across tenure groups in these production metrics. These findings are essential for interpreting how operator experience influences production performance and for informing personnel management strategies.

Table 8. Results of the Kruskal-Wallis test

Analysis items	P values
Total production quantity	0.00
percent of pass	0.00
Average production efficiency	0.00
Total unqualified quantity	0.00

Beyond its impact on the four key production variables, the relationship between operator tenure and other factors was further examined through correlation analysis. A heatmap was generated using Python's visualization libraries to illustrate these associations. In multivariate settings, partial correlation coefficients were employed to better capture the direct relationships between variables, reducing confounding effects from other factors.

Figure 4 and Figure 5 show that tenure is positively correlated with product failure rate, as well as with faults at filling device 4002 and screw cover device 6001. Conversely, tenure exhibits negative correlations with total output and several other fault points.

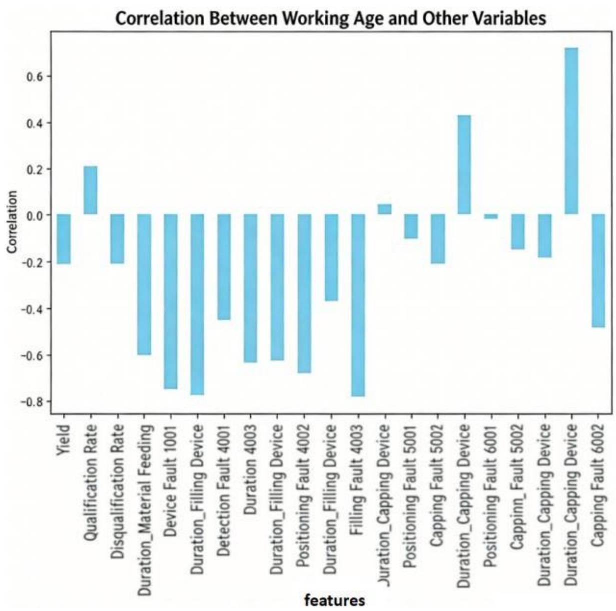


Figure 4. Correlation of length of service and other variables

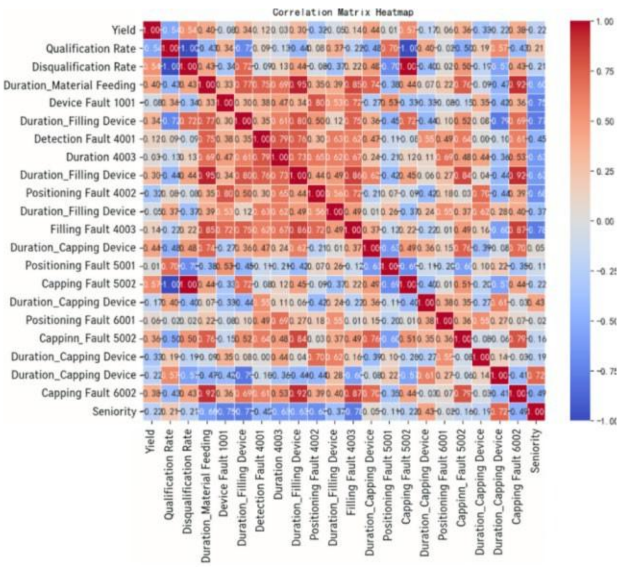


Figure 5. Heat map of length of service and other variables

3.6. Personnel Scheduling Optimization Based on Decision Tree Regression

Based on the significant tenure-based differences in production metrics and their distinct correlations with key fault points, personnel scheduling was optimized by maximizing the product of pass rate and output quantity. Multiple regression models, including linear regression, decision tree, random forest, support vector machine, and neural network, were trained using data from ten production lines and evaluated via MSE and R2. This staffing component complements rapid fault identification by supporting recovery arrangements after equipment disruptions and stable operation after production resumes.

As can be seen from Table 9 and Figure 6, the decision tree regression model had the best performance, with the lowest MSE value, only 0.49 and the highest R2, reaching 0.83. So the model effect was excellent, so the decision tree regression model was selected as the scheduling analysis model of this problem.

Table 9. Fitting effects of each model

	MSE	R^2
linear regression	2.61	0.07
decision tree	0.49	0.83
random forest	0.72	0.74
support vector machine	2.72	0.03
neural network	48641.26	-17309.06

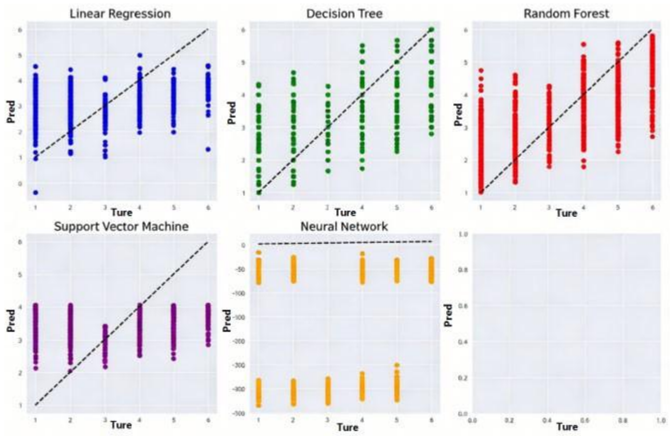


Figure 6. Evaluation results of each model

Because using machine learning model can only get the final result, to observe the internal impact of the data, this paper use the SHAP model to observe the decision tree regression model, the size of the SHAP value is used to represent the influence of each variable on the model output, the positive value of the variable has a positive effect on the model output, the negative value has a negative effect. Results are shown in the Figure 7 and Figure 8.

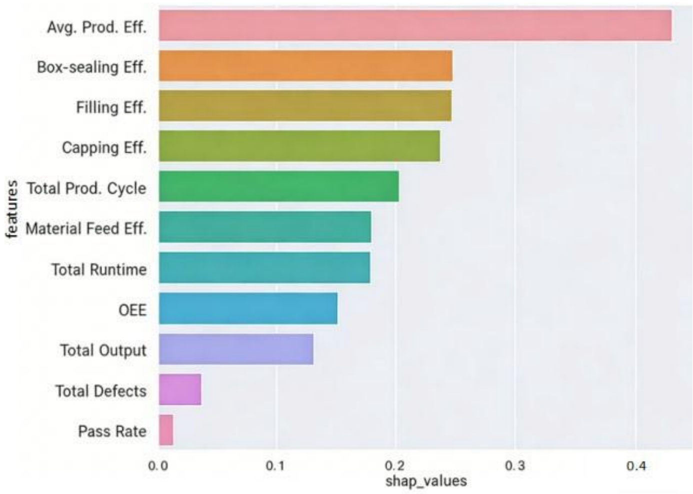


Figure 7. A plot for SHAP values

Figure 8. The Decision tree model affects the effect

4. Conclusion and Outlook

This study confirms that XGBoost can accurately identify production-line fault categories on the available dataset, while decision tree regression effectively supports staffing optimization. More importantly, the proposed IIoT-based ML framework links fault occurrence with rapid identification, targeted intervention, and recovery-oriented staffing. Its practical significance is not limited to improving classification accuracy or production efficiency: it also provides a decision-support basis for restoring normal production more quickly after faults and strengthening manufacturing resilience. The framework therefore aligns with Industry 5.0 principles by combining operational continuity with human-centric resource allocation.

Future work should validate the framework in real-time production environments and quantify resilience-oriented indicators, including fault recognition time, maintenance response time, downtime, recovery time, and post-recovery production stability. Integrating real-time IIoT data with digital twins and reinforcement learning may further improve adaptive scheduling and recovery decisions.

Acknowledgements

This research was supported by the Young Mentor Program of Guangzhou Huashang College: Research on the Development and Application of Data Mining Technology in Various Industries (No. 2024HSDS14).

References

[1] Paolanti M, Romeo L, Felicetti A, Mancini A, Frontoni E, Loncarski J. Machine learning approach for predictive maintenance in Industry 4.0. In: IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications; 2018. p. 1–6.

[2] Crespino AM, Corallo A, Lazoi M, Barbagallo D, Appice A, Malerba D. Anomaly detection in aerospace product manufacturing: Initial remarks. In: IEEE 2nd International Forum on Research and Technologies for Society and Industry Leveraging a Better Tomorrow; 2016. p. 1–4.

[3] Proto S, di Corso E, Apiletti D, Cagliero L, Cerquitelli T, Malnati G, mazzucchi D. REDTag: A predictive maintenance framework for parcel delivery services. IEEE Access. 2020;8:14953–14964.

[4] Stojanović D, Joković J, Tomašević I, et al. Algorithmic approach for the confluence of lean methodology and industry 4.0 technologies: Challenges, benefits, and practical applications. Journal of Industrial Intelligence. 2023;1(2):125–135.

[5] Gunasegaram DR, Barnard AS, Matthews MJ, Jared BH, Andreaco AM, Bartsch K, Murphy AB. Machine learning-assisted in-situ adaptive strategies for the control of defects and anomalies in metal additive manufacturing. Additive Manufacturing. 2024;82:104105.

[6] Kang Z, Catal C, Tekinerdogan B. Machine learning applications in production lines: A systematic literature review. Computers & Industrial Engineering. 2020;149:106773.

[7] Nakazawa T, Kulkarni DV. Anomaly detection and segmentation for wafer defect patterns using deep convolutional encoder-decoder neural network architectures in semiconductor manufacturing. IEEE Transactions on Semiconductor Manufacturing. 2019;32(2):250–256.

[8] Scime L, Siddel D, Baird S, Paquit V. Layer-wise anomaly detection and classification for powder bed additive manufacturing processes: A machine-agnostic algorithm for real-time pixel-wise semantic segmentation. Additive Manufacturing. 2020;36:101453.

[9] Aydemir G, Paynabar K. Image-based prognostics using deep learning approach. IEEE Transactions on Industrial Informatics. 2020;16(9):5956–5964.

[10] Liu W, Chen Z, Hu Y. XGBoost algorithm-based prediction of safety assessment for pipelines. Journal of Natural Gas Science and Engineering. 2020;73:103377.

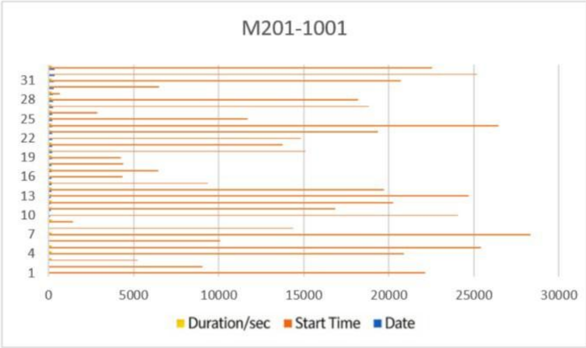
[11] Zhang S, Zhu X, Anduv B, et al. Fault detection and diagnosis for the screw chillers using multi-region XGBoost model. Science and Technology for the Built Environment. 2021;27(5):608–623.

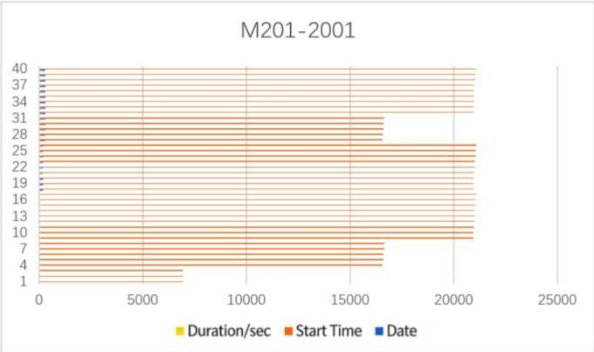
[12] Zhang C, Wang D, Song C, et al. Interpretable learning algorithm based on XGBoost for fault prediction in optical network. In: Optical Fiber Communications Conference and Exhibition; 2020. p. 1–3.

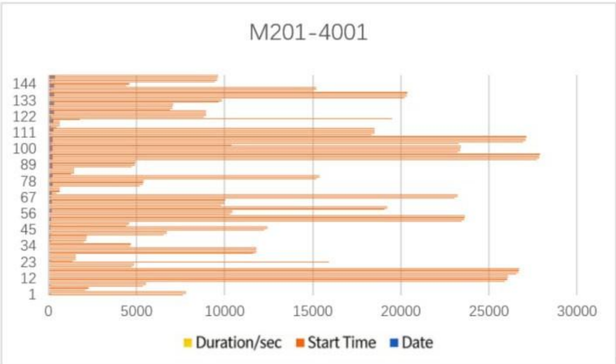
[13] Lu L, Zhang Y, Si Z, Dou Z. Research on anomaly detection of parts in workshop production line

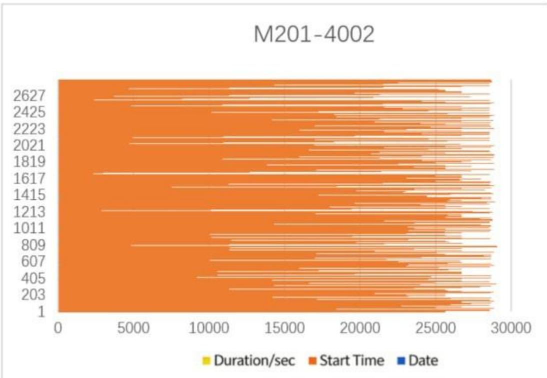
based on BO-XGBoostLSS. In: Advanced Intelligent Computing Technology and Applications; 2024. p. 282–291.
[14] Henriques J, Caldeira F, Cruz T, Simões P. Combining k-means and XGBoost models for anomaly detection using log datasets. Electronics. 2020;9(7):1164.
[15] Yu W, Dillon T, Mostafa F, Rahayu W, Liu Y. A global manufacturing big data ecosystem for fault detection in predictive maintenance. IEEE Transactions on Industrial Informatics. 2020;16(1):183–192.
[16] Sharma A, Aslekar A. IOT based predictive maintenance in Industry 4.0. In: International Interdisciplinary Humanitarian Conference for Sustainability; 2022. p. 143–145.
[17] Weber C, Reimann P. MMP - A platform to manage machine learning models in Industry 4.0 environments. In: IEEE 24th International Enterprise Distributed Object Computing Workshop; 2020. p. 91–94.
[18] Mrazovac B, Ilian V, Hulea M. Beyond Industry 4.0: Leveraging AI-powered anomalous sound detection for smart maintenance. In: Zooming Innovation in Consumer Technologies Conference; 2021. p. 43–47.
[19] Bozcan I, Korndorfer C, Madsen MW, Kayacan E. Score-based anomaly detection for smart manufacturing systems. IEEE/ASME Transactions on Mechatronics. 2022;27(6):5233–5242.
[20] Jeon K, Yoo D, Park J, Lee KD, Lee JJ, Kim CW. Reliability-based robust design optimization for maximizing the output torque of brushless direct current motors considering manufacturing uncertainty. Machines. 2022;10(9):797.
[21] Pang JL. Adaptive Fault Prediction and Maintenance in Production Lines Using Deep Learning. International Journal of Simulation Modelling. 2023;22(4):734–745.

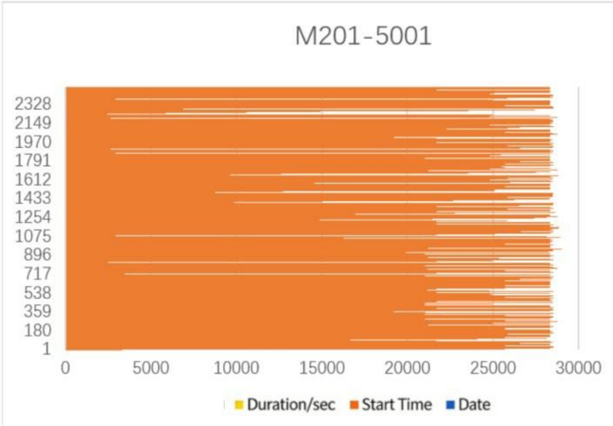
Appendix

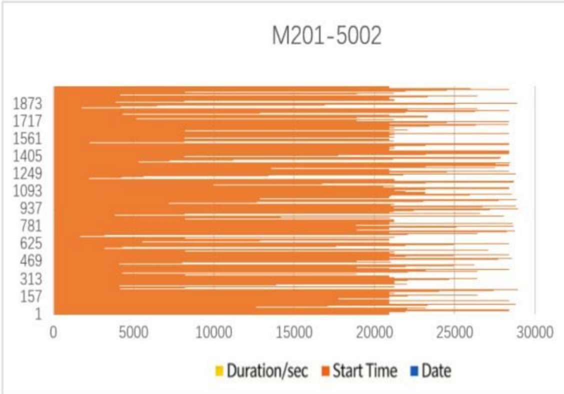

M201-1001
31 28 25 22 19 16 13 10 7 4 1
0 5000 10000 15000 20000 25000 30000
Duration/sec Start Time Date

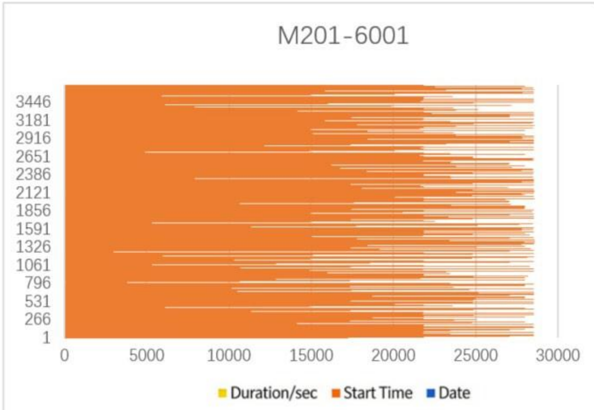

M201-2001
40 37 34 31 28 25 22 19 16 13 10 7 4 1
0 5000 10000 15000 20000 25000
Duration/sec Start Time Date

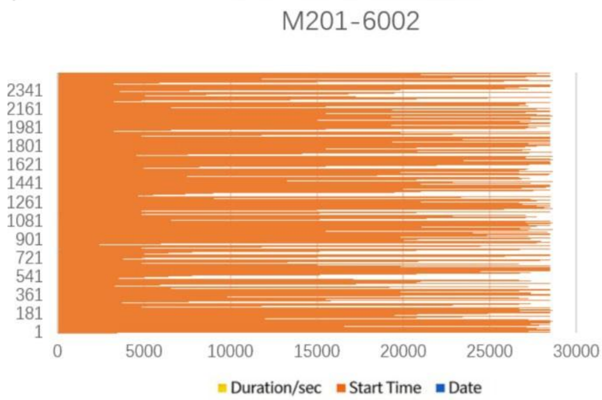

M201-4001
144 133 122 111 100 89 78 67 56 45 34 23 12 1
0 5000 10000 15000 20000 25000 30000
Duration/sec Start Time Date


M201-4002
2627 2425 2223 2021 1819 1617 1415 1213 1011 809 607 405 203 1
0 5000 10000 15000 20000 25000 30000
Duration/sec Start Time Date


M201-5001
2328 2149 1970 1791 1612 1433 1254 1075 896 717 538 359 180 1
0 5000 10000 15000 20000 25000 30000
Duration/sec Start Time Date


M201-5002
1873 1717 1561 1405 1249 1093 937 781 625 469 313 157 1
0 5000 10000 15000 20000 25000 30000
Duration/sec Start Time Date


M201-6001
3446 3181 2916 2651 2386 2121 1856 1591 1326 1061 796 531 266 1
0 5000 10000 15000 20000 25000 30000
Duration/sec Start Time Date


M201-6002
2341 2161 1981 1801 1621 1441 1261 1081 901 721 541 361 181 1
0 5000 10000 15000 20000 25000 30000
Duration/sec Start Time Date