Proc. of the European Safety and Reliability Conference (ESREL 2026)
Edited by José C. Matos, Paulo B. Lourenço, Michael Beer, Daniel Oliveira,
Edoardo Patelli, Hélder S. Sousa, Rui A. Silva and Monica Santamaria
 Published by Research Publishing, Singapore.
doi: 10.3850/ESREL2026061419_esrel26-p26345-cd

Remaining Useful Life Prediction of Imaging Devices via Multi-Iteration Semi-Supervised Modeling and Real-World Evaluation

Pankaz Das, Bulent Alpay, Xi Wang
GE Healthcare Technologies, The United States of America.
E-mail: (pankaz.das, bulent.alpay, xi.wang1)@gehealthcare.com

Obtaining a reliable and substantial amount of ground truth from medical imaging devices is challenging due to several factors: simultaneous degradation of multiple components, reversible and irreversible failure modes, operators' sensitivity to degradations, external stressors, varying operating environments, and identical symptoms arising from different root causes. This paper proposes an iterative semi-supervised approach for training a machine learning (ML) model to predict the remaining useful life (RUL) of imaging devices. The method employs an iterative approach to model degradation modes in complex systems. In the initial iteration, an ML model is trained on all available ground truth data for a specific system, subsystem, or component (SSC) failure, with the objective of maintaining the false positive rate (FPR) below a predefined threshold. This step learns signatures of the most dominant degradation mode (a subset of the full dataset). In the subsequent iteration, the predicted ground truth labels from the first model, corresponding to the dominant degradation mode, are used to train a new model. Additional iterations are performed if the FPR for this mode exceeds the threshold. The FPR threshold is an optimization parameter, varying across modes and determined by evaluating model performance on validation data. Additionally, we introduce a novel evaluation method that simulates real-world deployment. Specifically, we evaluate the model by its initial predictions for each failure, focusing on whether it can detect failure signatures within a set time window after those predictions. This approach captures realistic failure scenarios, such as, operators' sensitivity to degradations, reversible failures, and environmental variations. Results on a component failure in a computed tomography (CT) machine demonstrate that the proposed multi-iteration semi-supervised modeling and evaluation approach significantly improves performance metrics (achieving over 80% recall and 90% precision) compared to traditional ML modeling and evaluation methods.

Keywords: Predictive Maintenance, Semi-Supervised Learning, Medical Imaging Devices, Remaining Useful Life, Real-World Evaluation.

1. Introduction

Predictive maintenance (PdM) has become an essential component of modern industrial operations in the Industry 4.0, fueled by rapid advances in sensing technologies, data acquisition pipelines, and data-driven artificial intelligence and machine learning (ML) approaches Zonta et al. (2020); Zhu et al. (2019). At its core, PdM seeks to anticipate system degradation before it disrupts operations, allowing organizations to replace reactive or rigid preventive schedules with data-driven interventions Zhang et al. (2019). Specifically, prognostics and Remaining Useful Life (RUL) models aim to extract reliable degradation patterns from noisy, imbalanced, or sparsely labeled data. These approaches are designed to remain robust as data quality and operating conditions evolve in real-world environments Tsallis et al. (2025); Zhou et al. (2024); Meddaoui et al. (2024).

In healthcare, PdM is vital for maintaining diagnostic quality, throughput, and patient safety. With richer data streams, PdM can detect early degradation Manchadi et al. (2023). Recent studies show that ML-based PdM improves early fault detection and reduces downtime in imaging equipment Shah Ershad et al. (2024). However, imaging devices, such as CT, MRI, and X-ray systems, rely on many interdependent hardware and software components, making them vulnerable to performance loss when any element degrades. While PdM is critical for early detection, creating reliable ground truth is challenging due to following limitations.

- Limitations of service-action–based ground

truth: Service-action–based ground truth is often unreliable. Misdiagnoses, incorrect or preventative part replacements, temporary fixes, and overlapping symptoms all lead to labels that do not reliably represent the true failure mode. Moreover, a single service request (SR) may involve multiple simultaneous actions, making it unclear which intervention actually resolved the issue. This multi-modality results in ambiguous labels for service-based ground truth.

- Variability in perceived root causes and customer sensitivity to degradation: Similar symptoms can arise from different failure modes, leading service personnel to diagnose issues inconsistently. This variability makes service-action labels unreliable. Customer reporting behavior also varies widely, e.g., some report early symptoms, others only near failure, further degrading the consistency of service-based ground truth.
- Reversible versus irreversible failures: Because some failures are irreversible while others temporarily self-recover, service events are difficult to interpret for labeling ground-truth.
- Environmental stress and operational variability: Intermittent abnormalities caused by operational stress or environmental fluctuations introduce noise, making ground-truths less reliable.

In this paper, we introduced a multi-iterative semi-supervised modeling approach and a novel real-world evaluation framework. Used together, these methods effectively address the challenges outlined above. Specifically, This paper makes three contributions: (i) a multi-iteration semi-supervised self-training strategy to refine noisy service-derived labels; (ii) a deployment-oriented, event-level evaluation framework based on first actionable prediction; and (iii) real-world validation on CT collimator failures demonstrating improved recall/precision and actionable lead times. The remainder of the paper is structured as follows. Section II provides an overview of related work and situates our approach within the existing literature. Sections III and IV detail the proposed semi-supervised methodology and the accompanying evaluation framework. Section V reports the results, and Section VI concludes the paper.

2. Related Works

PdM methods based on supervised learning, both traditional ML and deep learning (DL), have been widely applied to machine log data across many domains, including healthcare imaging devices Sipos et al. (2014); Patil et al. (2017); Wang et al. (2017); Gutschi et al. (2019); Huang et al. (2021). However, these approaches depend heavily on the availability of reliable, high-quality ground-truth labels. In settings where such labels are scarce or uncertain, their performance and generalizability may be limited.

Several DL–based log analysis frameworks (e.g., DeepLog Du et al. (2017), DeepCase Van Ede et al. (2022), LogBERT Guo et al. (2021), Drain He et al. (2017)) treat raw, unstructured system logs as natural language sequences. These models typically require log parsing and substantial preprocessing to transform free-text messages into structured formats before learning normal patterns or detecting anomalies. In contrast to these works, the log data used in our study are fundamentally different: domain experts have already mapped each log message to a structured numeric code representing a specific event type. As a result, our setting does not involve free-text parsing and instead relies on time-series of numeric event codes.

A broader survey Serradilla et al. (2022) summarizes semi-supervised DL methods for PdM, including variational autoencoders for detecting bearing faults Zhang et al. (2020) and mechanical failures Soete et al. (2024), generative adversarial networks for machine fault detection using vibration signals Bui et al. (2021), recurrent neural networks for estimating the RUL of ball bearings Kuo and Li (2020), and topology-based methods such as self-organizing maps Schwartz et al. (2020) and t-distributed stochastic neighbor embedding Talmoudi et al. (2021) for degradation classification.

3. Multi-Iteration Semi-Supervised Modeling

The proposed multi-iteration semi-supervised modeling framework is summarized in Figure 1.

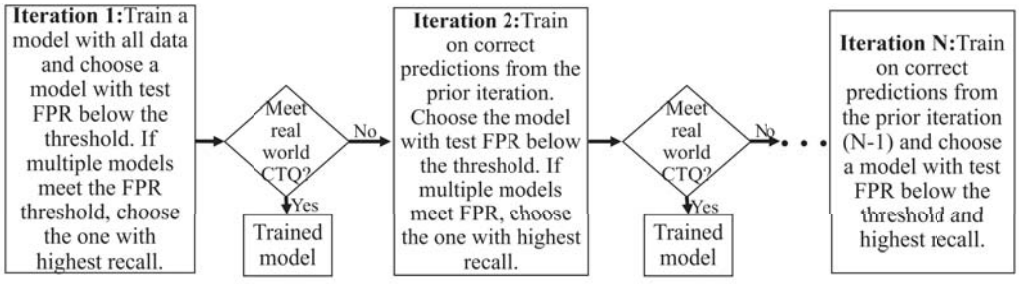

Fig. 1. Steps of multi-iteration semi-supervised modeling. FPR: false positive rate, CTQ: critical-to-quality.

The approach consists of one or more training cycles, each using a labeled dataset constructed from machine-operational logs and available ground-truth for a specific system, subsystem, or component (SSC) failure. Operational logs are collected daily, and labels (nominal or anomalous) are assigned using customer SR dates (described in Section 5 (data preparation)). Once labeled, a semi-supervised learning algorithm is used for model training.

In the first iteration, the model is trained on the full labeled dataset with the goal of maintaining the false positive rate (FPR) below a predefined threshold. Because service-derived ground truth is often unreliable, this initial iteration tends to capture signatures of the most dominant degradation mode, effectively learning patterns from the subset of failures that are most consistently represented. The trained model is then evaluated using real-world evaluation framework and performance metrics (Critical-to-Quality (CTQs)), as detailed in Section 4. Note that CTQs are measurable performance indicators that represent the essential outcomes a PdM solution must achieve to be considered successful. They translate business, service, and clinical expectations (e.g, system downtime) into quantifiable metrics (e.g., recall, precision, lead-times) that can be used to evaluate model performance.

In the second iteration, predictions from the first model are used as refined ground truth to train a new model. As before, the objective is to keep the FPR below the threshold and verify real-world performance. Additional iterations are introduced only if the FPR remains above the threshold or if CTQ performance criteria are not met. This iterative process can be extended to capture additional degradation modes. For example, in a CT collimator, the dominant degradation mode may involve the collimator blades and filters, while additional modes may arise from mechanical elements such as the collimator switches.

The FPR threshold is treated as an optimization parameter and may vary across degradation modes. It is selected by evaluating performance on validation data; empirically, a value of less than 0.004 provided strong results in our experiments. The maximum number of iterations (N) depends on available sample size. We observed that at least 30 SR instances are needed for statistically meaningful validation; thus, each iteration requires at least 30 model-predicted failure instances to proceed. Finally, when model performance does not improve across iterations, additional feature-engineering steps are necessary before continuing the semi-supervised training iterations.

4. Real-World Evaluation Framework

The real-world evaluation framework for model predictions is illustrated in Figure 2. In this framework, the horizontal axis represents calendar time, and the vertical axis indicates the functional status (system state) of the SSC, categorized as nominal, actionable, or failed. A nominal status means the SSC is operating as expected. An actionable status reflects anomalous behavior that suggests early degradation. From a PdM perspective, the actionable period is referred to as the predictive service window. A failed status indicates that the

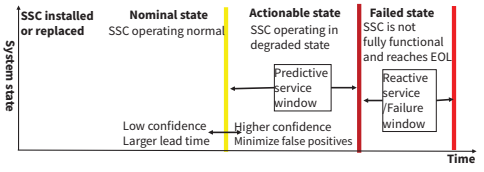

Fig. 2. The framework for the real world evaluation of the model predictions.

customer can no longer reliably use the SSC. Importantly, a reported failure does not always correspond to a complete hard-down event. Customer perception often influences when a failure is reported, so the "failed" period may span multiple days before the SSC reaches its true end-of-life (EOL). This interval, referred to as the reactive service or failure window, is a random variable that reflects the uncertainty involved in estimating the actual failure time.

From the model-output perspective, the objective of an ML model is to predict actionable states before the customer submits a SR, that is, to detect degradation within the predictive service window. As illustrated in Figure 2, prediction confidence generally rises as the system moves from nominal or actionable states toward failure, reflecting the increasing severity of degradation. Conversely, confidence decreases when attempting to predict degradation too early, such as when the system remains in a nominal state.

Note that this real-world evaluation process differs from standard ML evaluation fundamentally, which compares predicted vs. actual labels for every data point in the dataset and reports metrics (e.g., accuracy, recall, or F1-score). In contrast, the physics of failure in medical imaging components requires focusing on when the model first detects a deviation from normal behavior. Therefore, our real-world evaluation emphasizes the model's first prediction of an actionable state rather than classification accuracy of every data point. Specifically, once the model predicts first "Actionable" state for a given degradation mode, we compute several performance measures designed specifically for PdM evaluation as below:

- True positive (TP) = there is a customer SR related to a SSC of interest in the following $x + failure\ window$ days.
- False positive (FP) = there are no customer SRs related to SSC of interest in the following $x + failure\ window$ days.
- False negative (FN) = there is a customer SR but the model did not predict "Actionable" state before that SR.

Here, x denotes the SLA (service label agreement) period, typically 3–14 days depending SSCs, representing the expected time required to resolve the issue. Accordingly, using the results described above, we compute the evaluation metrics for the model predictions as follows.

- Recall (True positive Rate) = $\frac{TP}{TP+FN}$
- Precision = $\frac{TP}{TP+FP}$

We also evaluate prognostic performance by measuring the lead time, defined as the customer SR date minus the model's first predicted actionable date. This metric quantifies how effectively the model anticipates the RUL of a SSC.

5. Results

In this section, we report the results of our semi-supervised modeling approach, evaluated using the real-world evaluation framework for the collimator component of CT systems.

5.1. *Data preparation*

Our data consists of two sources: (1) machine data and (2) customer SR data. The machine data is derived from system logs that capture system dysfunctions, errors, warnings, and normal operational events (such as scan start/stop or system state changes) for SSCs. These logs are designed to support system behavior analysis, debugging, and root-cause investigation during failures or abnormal conditions. Because they span all major functionalities of an imaging system, the number of unique log messages is large. For example, for the product family considered in this study , we found approximately 3,000 unique log messages with corresponding numeric error codes.

On the other hand, The customer SR data captures issues and dysfunctions reported by users

Table 1. Target class labeling for each data point.

Distance from the failure event (in days)	Target class
Before failure event (SR) date: 0 to 21	Actionable
Before failure event (SR) date: 22 to 180	Nominal
After failure (SR) is corrected: (-60) to (-8)	Nominal

of imaging systems. Typical fields include: the date the issue was first observed, the customer's description of the issue, the actions taken by service engineers (including components or parts replaced), and the date when the issue was resolved and the system verified as functioning normally.

We merged these two datasets to construct a supervised dataset for ML model training. The merging logic is summarized in Table 1. Specifically, each time-series data point is labeled as either nominal or actionable based on observed failure dates. Data points occurring from 21 days prior to a component failure through the failure date were labeled as actionable. In contrast, data points occurring 180 to 22 days before the failure were labeled as nominal. After a failure was corrected, we labeled approximately two months of subsequent data as nominal as well. However, we excluded the first several days following the correction (e.g., the 7 days), since component-level parameters often require a short stabilization period before returning to nominal behavior.

The first step in our workflow is to prepare a semi-supervised dataset. Because customers report all issues they experience and service engineers document their corresponding actions, the resulting data is inherently noisy when developing models for a specific degradation mode or part-replacement pattern. To isolate the degradation mode of interest, we retained labeled data only for that particular failure type and relabeled all other failures as nominal. This yields a natural semi-supervised learning dataset, since labels are available only for a subset of the full dataset while the remaining samples are unlabeled with respect to the specific degradation mode.

5.2. *Feature creation and modeling*

Although the proposed framework is designed to operate effectively when ground-truth labels are uncertain or only partially available,

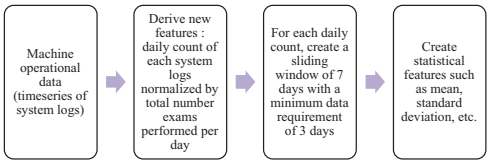

Fig. 3. The framework feature creation from raw machine syslog data.

we include below a concise description of the feature-engineering and modeling pipeline used in this paper for completeness.

- Feature creation: The feature-engineering workflow applied to the machine syslog data is summarized in Figure 3. We process raw time-series system logs generated during routine operation by counting each log type per day and normalizing by daily scan volume. From these normalized series, we form 7-day sliding windows (requiring at least 3 valid days) to capture short-term temporal patterns. For each window, we compute summary statistics that serve as model inputs. To reduce dimensionality and mitigate multicollinearity, we apply PCA and keep the smallest set of components explaining at least 80% of the variance.
- Modeling: We benchmarked multiple supervised models: tree-based methods (decision tree, Random Forest, XGBoost), logistic regression, support vector classifiers, and feed-forward neural networks. Across our experiments, Random Forest and XGBoost delivered superior performance in recall and precision while requiring less training time than the neural networks. Note that the framework is independent of any specific model class and other models can be used without modification.

5.3. *Standard and real-world evaluation*

We partitioned the dataset into training and testing sets based on calendar time. From the five years of available data (2020–2024), the period from 2020 to 2023 was used for training, while data from 2024 was reserved for testing. As described in the previous subsection, the log data were processed on a daily basis. Each day was then labeled as

either nominal or actionable according to its temporal distance from the SR open and close dates. This procedure resulted in 47,835 training data points and 13,647 testing data points. The class distribution for the nominal and actionable labels in each split is summarized in Table 2.

Table 2. Distribution of training and test dataset

Dataset	Nominal	Actionable
Training	46,780	1,055
Testing	13,256	391

In a standard evaluation, each predicted label is matched against its ground-truth label, yielding the confusion matrix summarized in Table 3.

Table 3. Confusion matrix on the test dataset

	Prediction class	
	Nominal	Actionable
Actual class		
Nominal	13,231	25
Actionable	360	31

The evaluation metrics associated with this setting, calculated using the ground-truth labels for all test dataset, are given as follows:

- Recall = 0.54, Precision = 0.76

On the other hand, applying the real-world evaluation methodology outlined in the section 4 yields the following TP, FP, and FN values, and the corresponding metrics on the test dataset.

- TP = 66, FN = 15, Recall = 0.82
- FP = 5, Precision = 0.93

We observe that the real-world evaluation yields substantial improvements in both recall and precision compared with the standard evaluation. As discussed in Section 4, the real-world framework assesses the model based on accurate predictions of customer-reported SSC failures (TP), missed detections (FN), and incorrect early predictions (FP). Specifically, TP and FP are determined by whether the model's first prediction of an actionable state correctly or incorrectly corresponds to a customer-initiated SR. FN represents cases in which the model fails to predict an actionable state despite a customer raising a SR.

Moreover, from the distribution of lead times (not shown due to space constraints), we observe a median lead time of 10 calendar days, with the 25th and 75th percentiles at 3 and 16 days, respectively. These results demonstrate that the proposed framework provides service engineers with a reasonable window to take preventive action, effectively supporting RUL estimation for the SSC. In addition, the 75th-percentile lead time of 16 days indicates that the model does not trigger premature failure predictions.

Finally, as discussed in the challenges outlined in Section 1, the onset of degradation is difficult to determine due to several factors, including reversible failures, external operating stresses, and differences in how customers or service engineers perceive and report issues, etc. As a result, the effective RUL threshold (i.e., the point at which degradation becomes detectable) varies across failures, and degradation signatures do not necessarily progress monotonically over time. The proposed real-world evaluation framework directly addresses this variability by accommodating the non-uniform and stress-dependent progression of failure modes, where external conditions may accelerate or slow the degradation process.

5.4. *Effectiveness of multi-iteration semi-supervised modeling*

In this subsection, we present the results from iterative applications of our semi-supervised modeling approach, demonstrating its effectiveness in learning degradation signatures for a specific failure mode. Table 4 reports the performance of models trained over two iterations for predicting replacements of a CT collimator component.

Across iterations, recall increases while the FPR decreases. In the first iteration, all SRs labeled as collimator replacements (based on customer and field-engineer reports) are used as positive examples. The resulting model correctly iden-

Table 4. Performance of collimator model trained for more than one iteration. FPR: false positive rate.

Iteration	test recall	test FPR
1	0.74	0.0019
2	0.96	0.0017

tifies 74% of such SRs in the test set. Additionally, 0.19% of nominal data points are incorrectly labeled as actionable, yielding an FPR of 0.19%. In the second iteration, the model is retrained using only those SRs that were correctly identified in the first iteration, allowing the model to better learn the true signature of dominant degradation. As a result, test-set recall increases substantially from 74% to 96%, and the FPR decreases to 0.17%.

Finally, obtaining reliable ground-truth labels with accurate degradation annotations is extremely challenging, as discussed in Section 1. In such settings, our multi-iterative semi-supervised modeling approach enables progressive refinement of the ground truth, helping to resolve ambiguities arising from multi-modal or non-specific labels. As demonstrated in our results, the iterative procedure systematically improves recall across iterations while maintaining low FPRs, allowing the model to better learns signatures specific to the dominant degradation mode of interest.

6. Conclusion

Predictive maintenance for medical imaging systems is constrained by sparse and noisy ground truth derived from SRs, where overlapping symptoms, reversible failures, and heterogeneous reporting behavior can obscure true degradation signatures. We introduced a multi-iteration semi-supervised framework that progressively refines noisy labels, isolates dominant degradation modes, and controls false positives through a tunable threshold. To better reflect deployment needs, we proposed an event-level evaluation that measures the first actionable prediction within a service window and reports lead-time statistics as an operational proxy for RUL. Experiments on CT collimator failures show improved early-warning performance, with event-level recall above 80%, precision above 90%, and a median lead time of 10 days. These results demonstrate that iterative label refinement combined with event-level scoring provides more reliable decision support than pointwise classification, especially when labels are uncertain or degradation is non-monotonic.

This study has limitations: results are demonstrated on a single component, may depend on temporal labeling choices, and iterative self-training can amplify dominant patterns. Additional safeguards, such as confidence-based filtering, expert review, or uncertainty-aware learning, may improve robustness. Future work includes extending the framework to more components, automating key thresholds, and exploring online learning for adapting operational conditions.

References

Bui, V., T. L. Pham, H. Nguyen, and Y. M. Jang (2021). Data augmentation using generative adversarial network for automatic machine fault detection based on vibration signals. *Applied Sciences 11*(5), 2166.

Du, M., F. Li, G. Zheng, and V. Srikumar (2017). Deeplog: Anomaly detection and diagnosis from system logs through deep learning. In *Proceedings of the 2017 ACM SIGSAC conference on computer and communications security*, pp. 1285–1298.

Guo, H., S. Yuan, and X. Wu (2021). Logbert: Log anomaly detection via bert. In *2021 international joint conference on neural networks (IJCNN)*, pp. 1–8. IEEE.

Gutschi, C., N. Furian, J. Suschnigg, D. Neubacher, and S. Voessner (2019). Log-based predictive maintenance in discrete parts manufacturing. *Procedia CIRP 79*, 528–533.

He, P., J. Zhu, Z. Zheng, and M. R. Lyu (2017). Drain: An online log parsing approach with fixed depth tree. In *2017 IEEE international conference on web services (ICWS)*, pp. 33–40. IEEE.

Huang, C., A. Deep, S. Zhou, and D. Veeramani (2021). A deep learning approach for predicting critical events using event logs. *Quality and Reliability Engineering International 37*(5), 2214–2234.

Kuo, R. and C. Li (2020). Predicting remaining

useful life of ball bearing using an independent recurrent neural network. In *Proceedings of the 2020 2nd International Conference on Management Science and Industrial Engineering*, pp. 237–241.

Manchadi, O., F.-E. Ben-Bouazza, and B. Jioudi (2023). Predictive maintenance in healthcare system: a survey. *IEEE Access 11*, 61313–61330.

Meddaoui, A., A. Hachmoud, and M. Hain (2024). Advanced ml for predictive maintenance: A case study on remaining useful life prediction and reliability enhancement. *The International Journal of Advanced Manufacturing Technology 132*, 323–335.

Patil, R. B., M. A. Patil, V. Ravi, and S. Naik (2017). Predictive modeling for corrective maintenance of imaging devices from machine logs. In *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 1676–1679. IEEE.

Schwartz, S., J. J. Montero Jimenez, M. Salaün, and R. Vingerhoeds (2020). A fault mode identification methodology based on self-organizing map. *Neural Computing and Applications 32*(17), 13405–13423.

Serradilla, O., E. Zugasti, J. Rodriguez, and U. Zurutuza (2022). Deep learning models for predictive maintenance: a survey, comparison, challenges and prospects. *Applied Intelligence 52*(10), 10934–10964.

Shah Ershad, M. H., S. Mohd Azrul Shazril, S. Mashohor, M. E. Amran, N. F. Hafiz, A. M. Ali, M. S. Naseri, and M. F. A. Rasid (2024). Assessment of iot-driven predictive maintenance strategies for computed tomography equipment: A machine learning approach. *IEEE Access*.

Sipos, R., D. Fradkin, F. Moerchen, and Z. Wang (2014). Log-based predictive maintenance. In *Proceedings of the 20th ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1867–1876.

Soete, C., M. Rademaker, and S. Van Hoecke (2024). A semi-supervised anomaly detection approach for detecting mechanical failures. *AI EDAM 38*, e16.

Talmoudi, S., T. Kanada, and Y. Hirata (2021). Tracking and visualizing signs of degradation for early failure prediction of rolling bearings. *Journal of Robotics and Mechatronics 33*(3), 629–642.

Tsallis, C., P. Papageorgas, D. Piromalis, and R. A. Munteanu (2025). Application-wise review of machine learning-based predictive maintenance: Trends, challenges, and future directions. *Applied Sciences 15*(9), 4898.

Van Ede, T., H. Aghakhani, N. Spahn, R. Bortolameotti, M. Cova, A. Continella, M. Van Steen, A. Peter, C. Kruegel, and G. Vigna (2022). Deepcase: Semi-supervised contextual analysis of security events. In *2022 IEEE Symposium on Security and Privacy (SP)*, pp. 522–539. IEEE.

Wang, J., C. Li, S. Han, S. Sarkar, and X. Zhou (2017). Predictive maintenance based on event-log analysis: A case study. *IBM Journal of Research and Development 61*(1), 11–121.

Zhang, S., F. Ye, B. Wang, and T. G. Habetler (2020). Semi-supervised bearing fault diagnosis and classification using variational autoencoder-based deep generative models. *IEEE Sensors Journal 21*(5), 6476–6486.

Zhang, W., D. Yang, and H. Wang (2019). Data-driven methods for predictive maintenance of industrial equipment: A survey. *IEEE systems journal 13*(3), 2213–2227.

Zhou, J., J. Yang, Q. Qian, and Y. Qin (2024). A comprehensive survey of machine remaining useful life prediction approaches based on pattern recognition: Taxonomy and challenges. *Measurement Science and Technology 35*(6), 062001.

Zhu, T., Y. Ran, X. Zhou, and Y. Wen (2019). A survey of predictive maintenance: Systems, purposes and approaches. *arXiv preprint arXiv:1912.07383*.

Zonta, T., C. A. Da Costa, R. da Rosa Righi, M. J. de Lima, E. S. Da Trindade, and G. P. Li (2020). Predictive maintenance in the industry 4.0: A systematic literature review. *Computers & industrial engineering 150*, 106889.