A Transformer Approach for Remaining Useful Life Prediction and Fault Diagnosis of Mechanical Equipment

A Transformer Approach for Remaining Useful Life Prediction and Fault Diagnosis of Mechanical Equipment

IOP Conf. Series: Materials Science and Engineering **1342** (2026) 012059 doi:10.1088/1757-899X/1342/1/012059

A Transformer Approach for Remaining Useful Life Prediction and Fault Diagnosis of Mechanical Equipment

Yuchen Liu[1], Siyuan Chen[1*] and Ebru Turanoglu Bekar[1]

[1] Department of Mechanical Engineering, Chalmers University of Technology, Gothenburg, Sweden

[*]E-mail: siyuan.chen@chalmers.se

Abstract. Accurately estimating remaining useful life (RUL) and performing timely fault diagnosis are critical for ensuring the reliability and safety of industrial equipment. However, these tasks are often treated separately, reducing the effectiveness of maintenance decisions and limiting the use of sensor data. This study extends the Transformer architecture to simultaneously perform RUL prediction and multi-label fault classification within the same framework. Through a shared encoder using self-attention to capture temporal and cross-sensor dependencies, the model learns representations that capture machine health evolution and fault characteristics. Two output heads handle RUL regression and fault classification, while uncertainty-based loss weighting automatically balances the training of both tasks. The model is evaluated on the Microsoft Azure predictive maintenance dataset containing multi-sensor readings from 100 industrial machines. Results show that the proposed approach outperforms machine-learning and deep-learning baselines on both tasks, achieving high accuracy and strong robustness across test machines. By jointly predicting fault types and RUL within a single framework, the proposed method enhances maintenance planning, enables earlier and more reliable interventions, and provides a scalable solution for intelligent monitoring in smart manufacturing environments.

1. Introduction

With the advent of Industry 5.0, the manufacturing sector is becoming increasingly interconnected, data-driven, and autonomous [1]. Among various digitalization strategies, predictive maintenance (PdM) has gained particular importance in ensuring the reliability, safety, and efficiency of production assets [2]. By estimating the remaining useful life (RUL) of machinery, PdM allows maintenance actions to be planned proactively, thereby minimizing downtime, extending equipment lifespan, and reducing operational costs. In addition, fault diagnosis plays a crucial role in identifying the specific causes of system degradation, enabling targeted interventions, and improving resource allocation across industrial operations.

In industrial settings, PdM systems collect multi-sensor data such as vibration and temperature and rely on traditional signal-based or statistical approaches to trigger maintenance actions [3]. In academia, fault diagnosis and RUL estimation are typically addressed using feature-based machine-learning models or deep sequence architectures. Reviews on data-driven PdM

highlight the growing adoption of hybrid and deep methods, while reporting persistent limitations in generalization and interpretability [4, 5].

However, these solutions treat RUL estimation and fault diagnosis separately, leaving opportunities untapped in shared representations and cross-task learning. Models trained on single machines or fixed operating regimes often fail to generalize new assets or conditions. In addition, interpretation and uncertainty quantification are rarely addressed, which limits trust and deployment in industrial contexts. Recent advances in Transformer architectures provide a promising solution, since self-attention can capture long-range temporal dependencies and cross-sensor relations [6]. But their use in a unified RUL and fault diagnosis framework remains limited, motivating our proposed multi-task approach.

This study aims to address these challenges by proposing a Transformer-based multi-task framework evaluated on the Microsoft Azure predictive maintenance dataset [7]. The framework learns shared representations from multi sensor time series data to jointly perform RUL prediction and fault classification, while uncertainty weighted training ensures balanced optimization between tasks. Its performance is evaluated using both accuracy and calibration metrics, and compared against representative baselines including long short-term memory networks (LSTMs), convolutional neural networks (CNNs), region-based convolutional neural networks (RCNNs) and extreme gradient boosting (XGBoost).

The remainder of this paper is organized as follows. Section 2 reviews the related work on RUL prediction and fault diagnosis. Section 3 details the proposed Transformer-based multi-task methodology. Section 4 presents the experimental setup and results, followed by an in-depth discussion in Section 5. Finally, Section 6 concludes the paper and outlines future research.

2. Related Work

PdM has been extensively studied as a strategy to detect faults and schedule maintenance actions to minimize downtime and operational risk. Two key tasks in this context are RUL estimation and fault diagnosis, indicating when to maintain and what to repair [8, 9]. Early PdM approaches relied on survival, reliability, and signal-processing methods, which were effective under stationary conditions but struggled with nonlinear degradation dynamics and sensor variability [10].

With the rise of data-driven methods, machine learning (ML) extended PdM capabilities by employing ensemble algorithms such as XGBoost and support vector machines for fault detection and health-state classification [10]. Deep neural networks subsequently enabled end-to-end modeling of degradation dynamics from raw sensor data. CNNs capture local temporal-frequency features, LSTMs long-term dependencies, and RCNNs combine both [11].

Attention-based methods alleviate these issues by capturing long-term patterns and cross-sensor interactions. Channel–spatial attention networks have improved predictive health management [12]. Studies explored dual-aspect attention for turbofan engines [13], multivariate Transformer frameworks [14], multi-head attention for battery degradation [15], and conditional variational Transformers for bearings [16]. However, most of these approaches focus on RUL and treat diagnosis separately.

Recent research integrates RUL and fault diagnosis in unified architectures. Branched networks first diagnose then estimate RUL [17], multi-task recurrent models combine event prediction and lifetime estimation [18], attention-guided graph learning shares features across tasks [19], and Bayesian formulations incorporate uncertainty [20]. A recent review highlights a shift toward attention-based solutions but notes issues in calibration, generalization, and

evaluation protocols [8], motivating unified frameworks capable of jointly modeling fault progression and lifetime, as pursued in this work.

3. Methodology

The proposed framework, shown in Figure 1, is a Transformer-based approach that integrates RUL estimation and fault classification within a single pipeline. The framework consists of data preprocessing, model training, and evaluation. Raw multi-sensor signals from the Microsoft Azure Predictive Maintenance dataset are first converted into fixed-length windows with RUL and fault labels. These inputs are then encoded by a shared Transformer, whose outputs feed into two task-specific heads for regression and classification. Training adopts an uncertainty-aware loss balancing strategy, while task-specific metrics are used for evaluation.

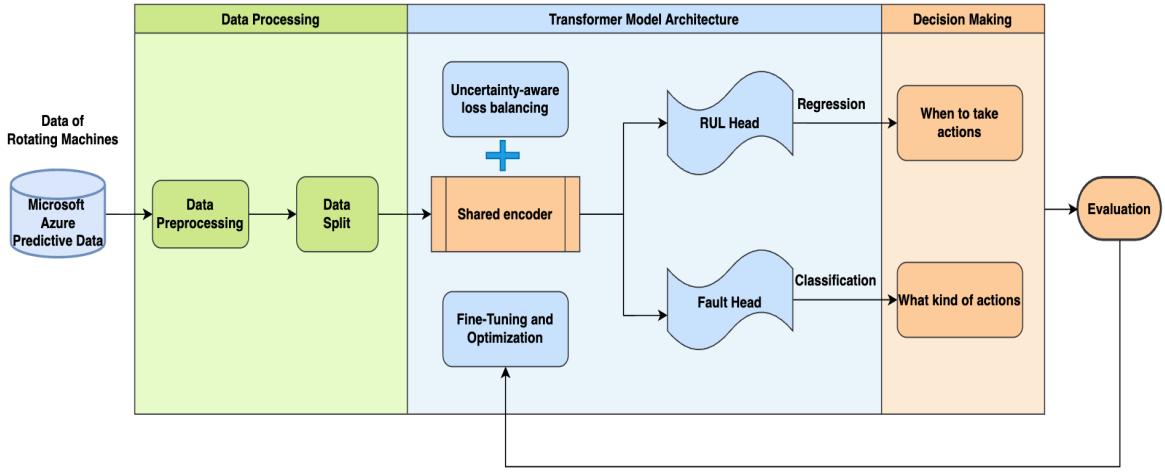

Figure 1. Architecture of the Transformer-based framework for RUL estimation and fault classification.

3.1 Data Preprocessing

Raw sensor streams are transformed into inputs suitable for model training through a preprocessing pipeline, as illustrated in Figure 2.

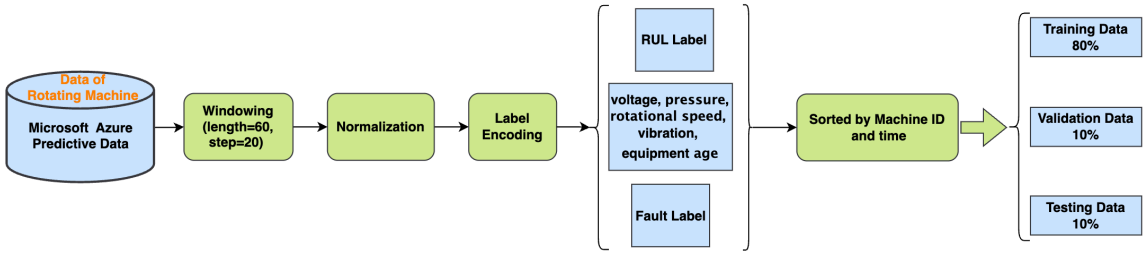

Figure 2. The pipeline for data preprocessing.

3.1.1 Data Transformation and Encoding

The preprocessing pipeline converts raw multi-sensor readings into structured inputs suitable for training and evaluation. First, each sensor sequence is segmented into overlapping windows of 60 time steps with a stride of 20, allowing the model to capture both short-term variations and

long-term degradation patterns. Next, all continuous features are normalized to zero mean and unit variance to ensure comparability across sensors and improve training stability. Finally, categorical fault indicators are transformed into one-hot encoded vectors to enable multi-label classification within the learning framework.

3.1.2 RUL Label Generation

The RUL at each time step is defined as the time difference between the current observation and the next recorded failure event, as illustrated in Figure 3. To avoid unreliable labels, data recorded before the first observed failure of each machine are excluded. For each subsequent failure event, timestamps are sorted and duplicates removed. Then, within the interval between two consecutive failures, each sample is assigned an RUL value equal to the remaining time until the next failure. Formally, for machine m with a sequence of failure times $\{t_1, t_2, \ldots, t_k\}$, the RUL of a record observed at time t (where $t_{i-1} < t \leq t_i$) is:

$$RUL(t) = t_i - t \tag{1}$$

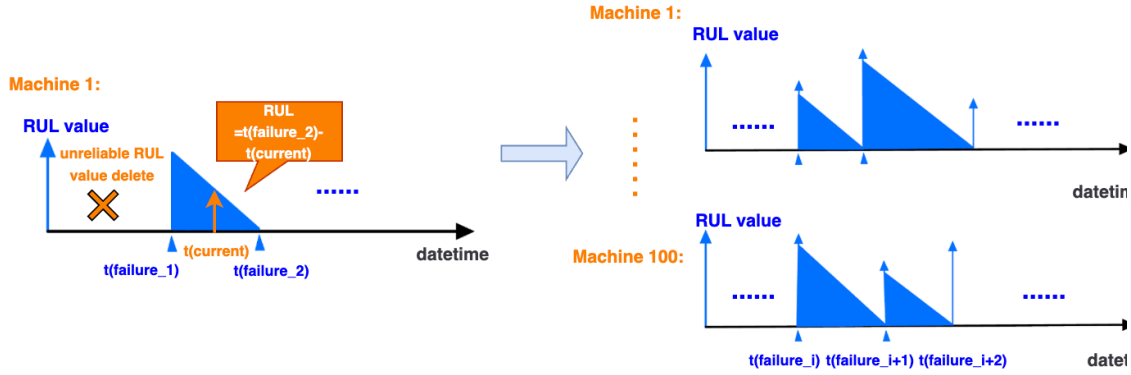

Figure 3. RUL labels generation from failure events. Each sample within a failure cycle is assigned an RUL value representing the remaining time until the next failure.

This procedure ensures that RUL values decrease monotonically toward zero as the failure point approaches, while data prior to the first failure are discarded as unreliable.

3.1.3 Data Splitting and Preparation

To ensure balanced and reliable evaluation, the processed dataset of approximately 130,000 samples was split into training, validation, and test sets following an 80%–10%–10% ratio. This strategy is commonly adopted in ML research and validated in multiple high-impact PdM studies [21]. The training subset provides sufficient data diversity for learning degradation patterns, while the validation subset is used for hyperparameter tuning and early stopping to prevent overfitting. The test subset remains unseen during training, enabling unbiased performance assessment and fair model comparison.

All subsets maintain consistent feature scaling and temporal ordering to preserve the causal structure of sensor measurements. This preprocessing step ensures that the input tensors exhibit uniform scale, dimension, and statistical distribution across different machines, which effectively reduces data bias and facilitates stable convergence during model optimization.

3.2 Model Framework

The proposed framework showed in Figure 4 employs a Transformer architecture to simultaneously address RUL estimation and fault diagnosis.

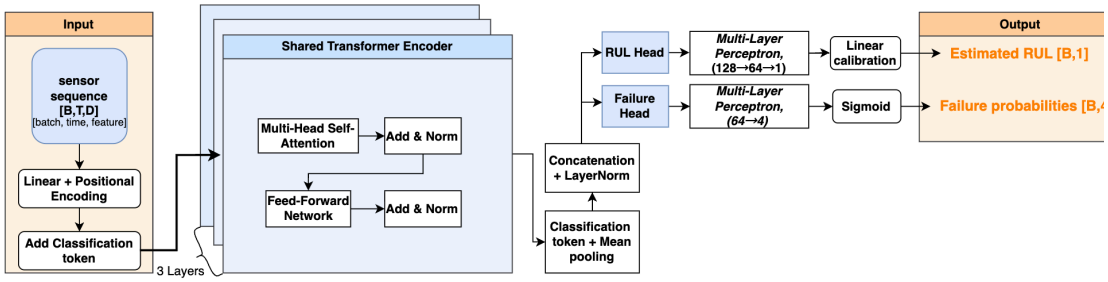

Figure 4. Transformer-based multi-task framework integrating a shared encoder with dual heads for simultaneous RUL estimation and fault classification.

We adopted a shared Transformer encoder pre-trained on large-scale time-series data to provide generalizable feature representations for multivariate sensor signals. Each windowed sample $x \in R^{T \times D}$, where T denotes the temporal length and D denotes the number of sensor channels, is aligned to the encoder's pre-training context length without information leakage. The inputs are linearly projected to the embedding dimension and augmented with positional information consistent with the pre-trained configuration. The encoder produces a sequence of hidden states, and when a classification token is present, its embedding is concatenated with the mean representation of valid tokens. A LayerNorm operation is then applied to form the shared latent representation for both downstream tasks.

The encoder is composed of multiple attention and feedforward layers, each followed by residual connections and normalization. Sharing the encoder allows the two downstream tasks to leverage common temporal patterns and cross-sensor relationships learned through the attention mechanism.

From the shared representation, two task heads are applied as summarized in Table 1.

Table 1. Architecture of task-specific heads.

Component	RUL Regression Head	Fault Classification Head
Hidden layers	Two fully connected layers (64 units each)	Two fully connected layers (64 units each)
Activation function	GELU	ReLU
Dropout rate	0.2	0.3
Output layer	Linear calibration layer (identity initialization)	Fully connected layer with four logits
Output type	Single continuous value (predicted RUL)	Fault probabilities for four categories

3.3 Training Strategy

For fault classification, we employed binary cross-entropy with logits, which is named BCEWithLogitsLoss. This objective directly models independent binary probabilities for each label, ensuring stable convergence even under class imbalance.

For RUL prediction, a hybrid robust loss is designed to combine absolute and relative error components, while emphasizing boundary regions near the beginning and end of the equipment life. The formulation is based on the SmoothL1 loss [22], which behaves quadratically for small errors and linearly for large ones. This property makes it less sensitive to outliers than L2 loss while maintaining smoother gradients than L1 loss, thereby stabilizing training under noisy and skewed lifetime distributions where extreme or small target values often dominate learning. The loss is defined as:

$$L_{\text{rul}} = (1-\lambda)\,\text{SmoothL1}(\hat{y}, y) \;+\; \lambda\,\text{SmoothL1}\left(\frac{\hat{y}-y}{\max(y,\varepsilon)}, 0\right). \tag{2}$$

where L_{rul} denote the regression loss, $\hat{y}$ and y are the predicted and actual RUL values. And λ balances absolute and relative error terms, while ε avoids division by zero for small target values.

To balance tasks without manual tuning, we followed uncertainty-based weighting [23] with learnable log-variances $\log\sigma_{\text{rul}}^2$ and $\log\sigma_{\text{fail}}^2$:

$$\mathcal{L} = \exp\left(-\log\sigma_{\text{rul}}^2\right)\, L_{\text{rul}} + \exp\left(-\log\sigma_{\text{fail}}^2\right)\, L_{\text{fail}} + \log\sigma_{\text{rul}}^2 + \log\sigma_{\text{fail}}^2. \tag{3}$$

where L_{fail} the multi-label classification loss. This formulation originates from probabilistic multi-task learning, where each task's loss is scaled by its predicted uncertainty. Using the log–exp form ensures that the weights remain positive and differentiable, allowing the model to learn both the optimal loss balance and the underlying task uncertainties during training. As a result, the framework adapts dynamically across operating conditions and reduces the need for manual hyperparameter tuning.

This scheme dynamically adjusts task weights during training and has shown robust performance across different operating conditions. To reduce overfitting and accelerate training while benefiting from the pre-trained backbone, training follows a three-stage schedule summarized in Table 2.

Table 2. Multi-Stage Training Schedule.

Stage	Epochs	Training Configuration	Description
1	3	Encoder frozen; only task heads updated	Focuses on stabilizing the task-specific layers while preserving pre-trained feature representations.
2	12	Last 2 Transformer blocks unfrozen	Gradually fine-tunes higher-level encoder representations to adapt to domain-specific degradation patterns.
3	25	Last 4 Transformer blocks trainable	Fully adapts the encoder to task characteristics, enabling joint optimization of low- and high-level features.

The model is trained using an adaptive gradient-based optimizer with regularization named AdamW and a learning rate scheduling mechanism named ReduceLROnPlateau scheduler that automatically reduces the rate when validation performance plateaus. Each batch contains 32 time windows, and dropout is applied in both output heads to prevent overfitting. Early stopping monitors validation loss to retain the best-performing model while avoiding unnecessary training.

3.4 Evaluation Matrix

Model performance is evaluated using quantitative metrics tailored to each task. For RUL prediction, the Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) are employed to assess the deviation between predicted and actual lifetime values. MAE measures the average prediction error, while RMSE penalizes larger deviations more strongly, providing a stricter indication of robustness under abrupt degradation conditions [24]. Lower values of both metrics indicate higher precision and stability of lifetime estimation. In addition, RUL trajectories are visualized for representative machines to illustrate the temporal alignment between predictions and actual degradation trends, offering intuitive validation of the model's predictive accuracy.

For fault classification, Accuracy, F1-score, and Sensitivity are used to evaluate overall correctness, balance between precision and recall, and the ability to detect true fault events, respectively [25]. High values across these metrics demonstrate that the model can reliably identify multiple fault types even under class imbalance, ensuring consistent and practical diagnostic performance.

4. Results

4.1 RUL prediction

A representative trajectory of true and predicted RUL for Machine 17, randomly selected from the test set, is shown in Figure 5. The estimated RULs exhibit a consistent downward trend toward failure, closely matching the ground truth after each reset cycle. The predicted RUL values align closely with the true RUL at almost every time step, showing minor deviations near abrupt failure transitions. This close correspondence confirms the model's ability to learn temporal degradation patterns and maintain stable prediction accuracy over time.

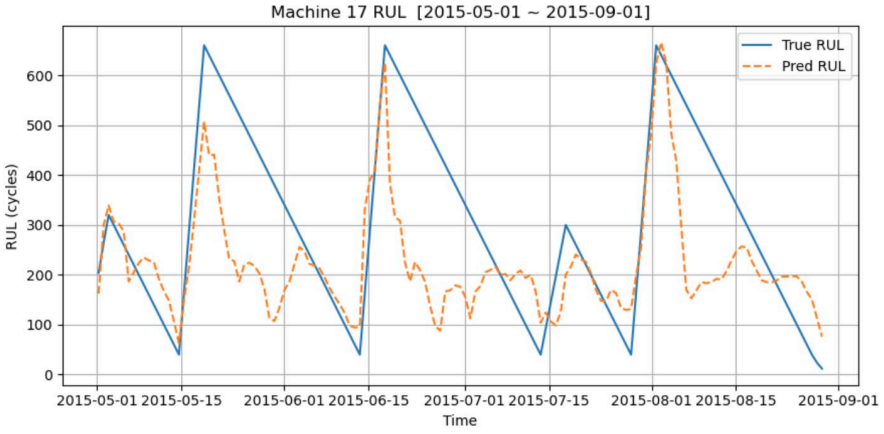

Figure 5. Predicted and true RUL trajectories for Machine 17, showing close alignment and accurate tracking of degradation trends.

Table 3 presents the RUL prediction results, comparing the proposed Transformer model with representative baselines, including XGBoost, CNN, LSTM, and RCNN, all trained under identical preprocessing and data split settings. These baselines, which have been widely adopted in prior studies and briefly reviewed in Section 2, provide a representative benchmark for evaluating the proposed method. The Transformer model achieves the lowest MAE of 0.17 and RMSE of 0.19, surpassing both traditional ML and deep learning baselines. The improved accuracy highlights its practical value for PdM, as reducing even small errors in RUL estimation can

significantly enhance the timing and efficiency of maintenance decisions in industrial environments.

Table 3. RUL prediction performance comparison.

Model	MAE	RMSE
XGBoost baseline	0.21	0.24
CNN baseline	0.21	0.27
LSTM baseline	0.22	0.28
RCNN baseline	0.22	0.29
Transformer	0.15	0.17

4.2 Fault classification

To further assess the model's diagnostic capability, the classification accuracy and sensitivity across training epochs are illustrated in Figure 6. The results show that the proposed Transformer-based framework rapidly converges within the first few epochs, achieving accuracy levels above 0.98 and steadily improving sensitivity thereafter. The slight oscillations observed in the sensitivity curve are mainly caused by fluctuations in the recall of minority fault classes, as the model periodically rebalances its attention between dominant and rare fault patterns during optimization. This behavior is typical in imbalanced multi-label classification and indicates adaptive feature refinement rather than instability in training. This trend overall suggests that the model quickly learns dominant fault patterns while gradually enhancing its ability to detect minority and subtle fault cases.

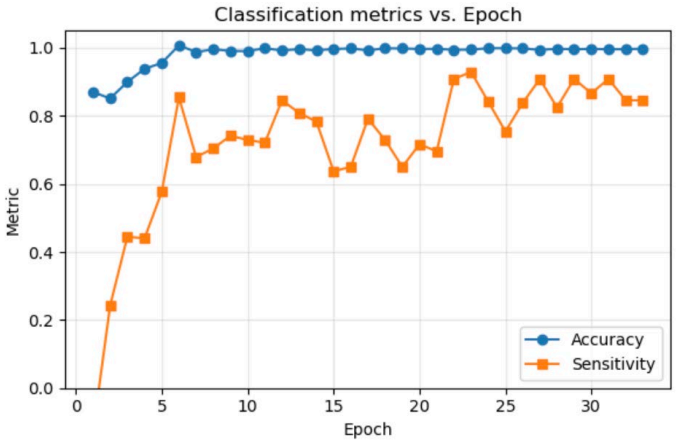

Figure 6. Classification accuracy and sensitivity over training epochs, illustrating fast convergence and mild sensitivity oscillations due to class imbalance.

The confusion matrix presented in Figure 7 provides a detailed view of per-class prediction performance. Most fault types are correctly identified, with diagonal dominance exceeding 0.85 across all classes. Only minor misclassifications occur between similar Fail2 and Fail3, which share overlapping sensor characteristics. These results confirm that the shared Transformer

encoder captures discriminative temporal–spatial dependencies effectively, even under class imbalance conditions.

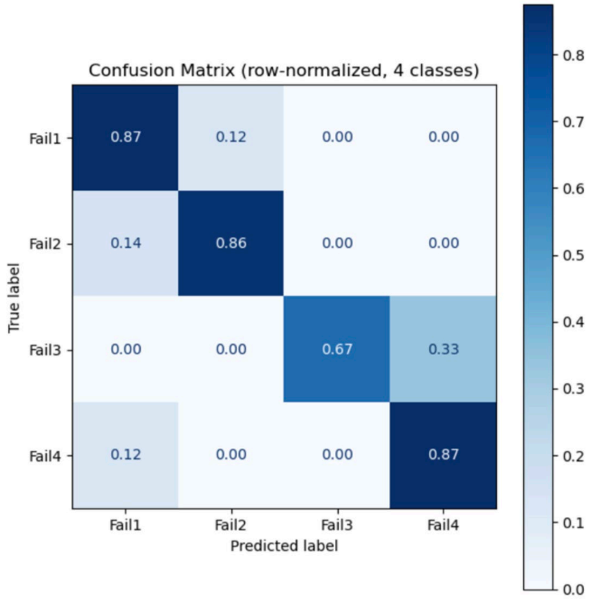

Figure 7. Confusion matrix of fault classification results, demonstrating high per-class accuracy with minor overlap between similar fault modes.

Finally, Table 4 compares baseline models. While overall accuracy remains above 95 % for all methods, the Transformer achieves the highest F1-score at 0.73, substantially improving the balance between precision and recall. This result confirms that joint feature learning across RUL and fault tasks strengthens the model's discriminative power, improving generalization under imbalanced fault distributions

Table 4. Fault classification performance comparison.

Model	Accuracy	F1-score
XGBoost baseline	0.98	0.47
CNN baseline	0.99	0.67
LSTM baseline	0.98	0.41
RCNN baseline	0.98	0.44
Transformer	0.99	0.74

5. Discussion

5.1 Overall performance

The proposed Transformer based framework achieves consistent and significant improvements over both classical and deep learning baselines in the dual tasks of RUL prediction and fault classification. Compared with XGBoost, CNN, LSTM, and RCNN, it obtains the lowest RUL errors and the highest F1-score for classification. These results confirm that the Transformer effectively

captures long-range temporal dependencies and cross-sensor relationships that conventional sequence or convolutional networks often miss. Like the findings in prior PdM studies using CNNs and LSTMs [45], our model benefits from data-driven feature learning, but its attention mechanism further extends modeling capacity beyond local or short-term dynamics.

In contrast to earlier statistical and reliability models [11,13], which assumed stationary degradation, the Transformer captures nonlinear degradation patterns directly from raw multi-sensor data. Even small improvements in RUL estimation accuracy are practically meaningful, as they allow maintenance to be scheduled closer to the actual end of life, reducing downtime and cost. The superior F1-score also reflects balanced precision and recall, indicating that the framework maintains high sensitivity to minority and subtle fault modes, which is a limitation frequently reported in traditional fault classifiers [13]. These findings demonstrate that a global attention mechanism yields measurable operational benefits in safety-critical industrial environments.

5.2 Joint learning and attention mechanism

A key advantage of the proposed approach lies in its joint learning strategy, which enables RUL estimation and fault diagnosis to share a single Transformer encoder. This unified representation allows the model to simultaneously learn when and what aspects of degradation, overcoming the separation of tasks seen in prior works [14–17]. In earlier attention-based PdM studies, attention was primarily used for RUL estimation, with diagnostic classification treated independently. Our shared-encoder design bridges this gap, ensuring that fault-related temporal cues directly inform lifetime prediction.

The self-attention mechanism dynamically allocates weights to critical time steps and sensor channels, highlighting subtle degradation precursors while maintaining awareness of long-term behavior. This capability contrasts with recurrent models [19] that capture sequential trends but overlook cross-sensor dependencies. Empirically, the shared encoder prevents redundant feature learning and promotes mutual enhancement between tasks, which aligns with the idea of cross-task feature sharing suggested in recent multi-task attention studies [3,18]. Together, these results validate that integrating RUL and fault diagnosis within one architecture provides richer temporal understanding and more stable feature learning.

5.3 Uncertainty weighting and training stability

The uncertainty-based loss weighting mechanism further strengthens multi-task optimization. Unlike traditional multi-task approaches that rely on manual tuning [18,19], our model automatically learns task-specific uncertainties, stabilizing convergence and improving robustness under noisy labels and class imbalance. This adaptive weighting directly addresses one of the open challenges identified in recent reviews [11], where multi-objective optimization was prone to overfitting toward one dominant task. During training, mild oscillations in task weights reflect adaptive rebalancing as the model allocates more focus to underrepresented fault categories while maintaining regression accuracy.

The fine-tuning schedule, which gradually unfreezes Transformer layers, preserves pretrained temporal representations while adapting to degradation-specific patterns. Together, these mechanisms yield improved generalization compared with static joint-learning frameworks reported in earlier PdM studies [20]. The overall stability of convergence confirms that uncertainty-driven adaptation provides a principled way to balance heterogeneous learning objectives without sacrificing interpretability or computational efficiency.

5.4 Limitations and future work

Despite its strong performance, several limitations remain. The Microsoft Azure dataset includes relatively few failure events and short monitoring durations, restricting the model's ability to represent long-term degradation dynamics. While uncertainty weighting improves stability, calibration metrics such as Expected Calibration Error were not assessed, which are crucial for confidence-aware maintenance decisions. Furthermore, domain shifts between machines or operating regimes were not explicitly addressed, suggesting that retraining or adaptation may be required for deployment in heterogeneous industrial environments.

Future research will focus on four directions. First, extending datasets to cover longer operation periods, higher-frequency signals, and more balanced fault distributions will allow learning of rare degradation patterns. Second, domain adaptation and test-time calibration techniques will be explored to enhance transferability to unseen assets. Finally, introducing survival-based RUL heads or graph based diagnostic modules may enhance interpretability and capture fault interrelations more explicitly.

6. Conclusion

This paper proposed a unified Transformer-based framework for PdM that jointly performs RUL estimation and fault classification. The approach contributes to a shared self-attention encoder that captures long-term temporal and cross-sensor dependencies, together with an uncertainty-aware loss weighting strategy that adaptively balances regression and classification tasks. Experiments on the mechanical dataset demonstrated superior performance over classical and deep learning baselines in both RUL prediction accuracy and F1-score of fault classification, confirming the benefits of joint feature learning. The framework provides a scalable and transferable solution for intelligent maintenance, enabling earlier and more reliable interventions in smart manufacturing environments. Future work will extend this framework with domain adaptation and calibration modules to improve confidence estimation and generalization to diverse industrial conditions.

Acknowledgement

The work was carried out within Chalmers' Area of Advance Production whose support is greatly acknowledged. The study was supported by the Vinnova - Sweden's innovation agency under grant number 2025-01110 (Advanced AI Architectures for Integrated and Enhanced Manufacturing Operations, AIMOps project). The computation was enabled by resources provided by the National Academic Infrastructure for Supercomputing in Sweden (NAISS), partially funded by the Swedish Research Council through grant agreement no. 2022-06725.

References

[1] Bokrantz J, Skoogh A, Berlin C, Wuest T, Stahre J. Smart maintenance: a research agenda for industrial maintenance management. Int J Prod Econ. 2020;224:107547. doi:10.1016/j.ijpe.2019.107547.
[2] Bokrantz J, Skoogh A, Berlin C, Stahre J. Maintenance in digitalised manufacturing: Delphi-based scenarios for 2030. Int J Prod Econ. 2017;191:154–169. doi:10.1016/j.ijpe.2017.06.010.
[3] Chen S, Bekar ET, Bokrantz J, Skoogh A. AI-enhanced digital twins in maintenance: systematic review, industrial challenges, and bridging research–practice gaps. J Manuf Syst. 2024;82:678–699. doi:10.1016/j.jmsy.2024.06.005.
[4] Ji D, Wang C, Li J, Dong H, others. A review: data driven-based fault diagnosis and RUL prediction of petroleum machinery and equipment. Syst Sci Control Eng. 2021;9:724–747. doi:10.1080/21642583.2021.1992684.

[5] Lekidis A, et al. Predictive maintenance framework for fault detection in remote terminal units: a digital twin approach. Forecasting. 2024;6(2):14. doi:10.3390/forecasting6020014.
[6] Bampoula X, Nikolakis N, Alexopoulos K. Condition monitoring and predictive maintenance of assets in manufacturing using LSTM-autoencoders and transformer encoders. Sensors (Basel). 2024;24(10):3215. doi:10.3390/s24103215.
[7] Biswas A. Microsoft Azure Predictive Maintenance Dataset [Internet]. Kaggle; 2018 [cited 2025-10-12], https://www.kaggle.com/datasets/arnabbiswas1/microsoft-azure-predictive-maintenance/data
[8] Martins AM, Silva J, Henriques R. Condition monitoring and predictive maintenance in industrial equipment: an NLP-assisted review of signal processing, hybrid models, and implementation challenges. Appl Sci (Basel). 2025;15(10):5465. doi:10.3390/app15105465.
[9] Nunes L, Martins R, Silva A. Challenges in predictive maintenance—A review. J Manuf Syst. 2023;68:193–205. doi:10.1016/j.jmsy.2022.10.006.
[10] da Rocha Faísca Moreira M. Data-driven predictive maintenance for component life-cycle extension [dissertation on the Internet]. Porto (PT): University of Porto; 2024 [cited 2025-10-12]. Available from: https://www.proquest.com/dissertations-theses/data-driven-predictive-maintenance-component-life/docview/3224609484/se-2.
[11] Chen S, Bandaru S, Marti S, Bekar ET, Skoogh A. Comparison of unsupervised image anomaly detection models for sheet metal glue lines. Eng Appl Artif Intell. 2024;153:110740. doi:10.1016/j.engappai.2024.110740.
[12] Liu C-L, Su H-C. Temporal learning in predictive health management using channel-spatial attention-based deep neural networks. Adv Eng Inform. 2024;62:102604. doi:10.1016/j.aei.2024.102604.
[13] Zhang Z, Song W, Li Q. Dual aspect self-attention based on transformer for remaining useful life prediction. arXiv [preprint]. 2021 [revised 2022 Apr 20; cited 2025-10-12]. arXiv:2106.15842.
[14] Ogunfowora O, Najjaran H. A transformer-based framework for multivariate time series: a remaining useful life prediction use case. arXiv [preprint] 2023. arXiv:2308.09884.
[15] Wu R, Zhang Z, Liu T. TransRUL: A transformer-based multihead attention model for battery remaining useful life prediction. Energies. 2024;17(16):3976. doi:10.3390/en17163976.
[16] Zhou Y, Lin H, Xu X. Conditional variational transformer for bearing remaining useful life prediction. Adv Eng Inform. 2024;58:102375. doi:10.1016/j.aei.2023.102375.
[17] Li Z, Li Y, Yue X, Zio E, Wu J. A deep branched network for failure mode diagnostics and remaining useful life prediction. IEEE Trans Instrum Meas. 2022;71(4):1–11. doi:10.1109/TIM.2022.3195280.
[18] Aggarwal K, Atan O, Farahat AK, Zhang C, Ristovski K, Gupta C. Two birds with one network: unifying failure event prediction and time-to-failure modeling. arXiv [preprint]. 2018. arXiv:1812.07142.
[19] Qi J, Chen Z, Kong Y, Qin W, Qin Y. Attention-guided graph isomorphism learning: a multi-task framework for fault diagnosis and remaining useful life prediction. Reliab Eng Syst Saf. 2025;263:111209. doi:10.1016/j.ress.2025.111209.
[20] Aghaee Dabaghan Fard S, Kim M, Deep A, Lee J. Bayesian joint model of multi-sensor and failure event data for multi-mode failure prediction. arXiv [preprint]. 2025. arXiv:2506.17036.
[21] Sivakumar M, Bibi A, Sivakumar S, et al. Trade-off between training and testing ratio in machine learning for medical image processing. BMC Med Res Methodol. 2024;24:112. doi:10.1186/s12874-024-02027-1.
[22] Girshick R. Fast R-CNN. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV); 2015 Dec 7–13; Santiago, Chile. p. 1440–1448. doi:10.1109/ICCV.2015.169.
[23] Kendall A, Gal Y, Cipolla R. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2018 Jun 18–22; Salt Lake City (UT). p. 7482–7491. doi:10.1109/CVPR.2018.00781.
[24] Ritter A, Muñoz-Carpena R. Performance evaluation of hydrological models: statistical significance for reducing subjectivity in goodness-of-fit assessments. J Hydrol. 2013;480:33–45. doi:10.1016/j.jhydrol.2012.12.004.
[25] Pliego Marugán A, Peco Chacón AM, García Márquez FP. Reliability analysis of detecting false alarms that employ neural networks: a real case study on wind turbines. Reliab Eng Syst Saf. 2019;191:106574. doi:10.1016/j.ress.2019.106574.