Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

A Deep Learning and TOPSIS Framework for Predictive Maintenance Scheduling in Smart Manufacturing: An Industry 4.0 Approach

Md Mohsin Uddin Fahim

mfahim@miners.utep.edu

The University of Texas at El Paso https://orcid.org/0009-0004-1447-2006

Amit J. Lopes

The University of Texas at El Paso

Research Article

Keywords: Predictive Maintenance, Long Short-Term Memory, TOPSIS, Remaining Useful Life, Smart Manufacturing

Posted Date: August 7th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10562848/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

A Deep Learning and TOPSIS Framework for Predictive Maintenance Scheduling in Smart Manufacturing: An Industry 4.0 Approach

Md Mohsin Uddin Fahim [a*], Amit J. Lopes [b]

[a, b] Department of Industrial, Manufacturing, and Systems Engineering
University of Texas at El Paso, El Paso, Texas, USA
[*] Corresponding Author Email: mfahim@miners.utep.edu
Contributing Author: ajlopes@utep.edu

A Deep Learning and TOPSIS Framework for Predictive Maintenance Scheduling in Smart Manufacturing: An Industry 4.0 Approach

ABSTRACT

Despite significant advances in deep learning-based failure prognosis, existing predictive maintenance frameworks consistently terminate at the prediction step, leaving the critical operational challenge of prioritising maintenance resources across heterogeneous asset fleets unresolved. This paper addresses this gap by proposing an integrated framework that couples Long Short-Term Memory network-based Remaining Useful Life prediction with the Technique for Order of Preference by Similarity to Ideal Solution for data-driven multi-criteria maintenance scheduling, validated on the NASA C-MAPSS FD001 turbofan degradation benchmark. The framework dynamically synthesises predicted health indicators with operational criteria encompassing machine criticality, maintenance cost, and production downtime impact to generate a ranked intervention schedule whose quality is evaluated through a rigorous, assumption-free protocol comprising precision-recall analysis, area under the precision-recall curve, Spearman rank correlation, and a scheduling efficiency ratio against a random baseline. A five-seed ablation study comparing Long Short-Term Memory, Bidirectional Long Short-Term Memory, and Gated Recurrent Unit architectures, with Wilcoxon signed-rank validation, establishes statistically equivalent prediction accuracy, confirming the architecture-agnostic nature of the scheduling design. The scheduling evaluation demonstrates near-perfect discrimination, capturing all failure-proximate assets by targeting less than one-third of the fleet, with a 3.70-fold efficiency improvement over random allocation, an AUPRC of 0.9923, and tier discrimination significant at $p < 10^{-11}$. Cross-domain validation on CWRU Bearing Fault rotating machinery confirms perfect discrimination (AUPRC = 1.000) and a Spearman correlation of $\rho = 0.803$ between the closeness coefficient and physical fault severity, thereby supporting physically grounded scheduling without explicit severity supervision. These findings establish that coupling accurate deep-learning-based prognosis with principled multi-criteria decision-making yields high-confidence intervention decisions and quantifiable gains in resource efficiency in Industry 4.0 smart manufacturing environments.

Keywords: Predictive Maintenance, Long Short-Term Memory, TOPSIS, Remaining Useful Life, Smart Manufacturing

1. Introduction

Equipment failure in a manufacturing facility does not stop at the machine. It disrupts upstream processes, stalls downstream assembly, forces unscheduled crew deployment, and generates emergency procurement costs, all before a single repair is made. The economic consequences are well documented across the industrial engineering literature: maintenance expenditure represents 15-40% of total production costs in capital-intensive manufacturing environments, and unplanned downtime attributable to unexpected asset failure accounts for a disproportionate share of that burden (Mobley, 2002 [1]; Swanson, 2001 [2]). Predictive maintenance strategies have been shown to reduce unplanned downtime by 20-50% and total maintenance costs by 10-30% when systematically implemented, yet their adoption remains constrained by the gap between what prognostic models produce and what maintenance engineers need to act on (Jardine et al., 2006 [3]; Hashemian, 2010 [4]).

The widespread deployment of Industrial Internet of Things sensors has made continuous, large-scale machine health monitoring operationally viable, generating rich multivariate time-series streams that encode degradation signatures long before failure becomes observable through conventional inspection (Lee et al., 2014 [5]). Among the deep learning architectures that have emerged for this task, Long Short-Term Memory (LSTM) networks (Hochreiter & Schmidhuber, 1997 [6]) have attracted sustained research interest for their ability to capture long-range temporal dependencies in sequential sensor data without succumbing to the vanishing-gradient problem that constrains standard recurrent networks. Their application to Remaining Useful Life estimation across machining, bearing diagnostics, and

multi-sensor health monitoring has consistently demonstrated competitive accuracy against both classical and deep learning baselines (Zhao et al., 2019 [15]; Wu et al., 2018 [8]).

However, predicting when a machine will fail and deciding which machine to service first are not the same problem. In a real production environment, multiple machines can approach failure simultaneously, maintenance resources are finite, and not all assets carry equal operational weight. A high-criticality bottleneck machine warrants faster response than a redundant auxiliary unit, even when both display similar predicted Remaining Useful Life values. (predicted Remaining Useful Life, failure probability, or criticality score) captures this scheduling decision in isolation. The problem is inherently multi-criteria, and the literature has not addressed it as such within the predictive maintenance domain (Mallioris et al., 2024 [9]). Works anchored in deep learning consistently terminate at the prediction output without providing a mechanism for translating health forecasts into an actionable intervention schedule, while those that do incorporate scheduling either apply single-threshold heuristics or rely on manually assigned health scores rather than dynamically predicted degradation values from live sensor data (Nguyen & Medjaher, 2019 [10]; Zonta et al., 2020 [12]).

The Technique for Order of Preference by Similarity to Ideal Solution, introduced by Hwang and Yoon (2012 [13]), was developed precisely for this class of problem. It ranks competing alternatives by their simultaneous proximity to an ideal best solution and distance from an ideal worst solution across multiple weighted criteria, producing a closeness coefficient that translates directly into an actionable priority order. Its computational transparency and tolerance for criteria of mixed unit and direction make it well-suited to maintenance scheduling environments where outputs must be interpretable to engineers on the shop floor, not merely computable by a model. Despite growing adoption of TOPSIS across engineering decision contexts (Rani et al., 2020 [14]; Gupta et al., 2018 [25]), the coupling of TOPSIS with dynamically predicted machine health values from deep learning models as a unified, end-to-end predictive scheduling pipeline has not been systematically explored in the published literature. This paper addresses that gap directly.

The principal contributions of this work are as follows:

- An integrated Long Short-Term Memory-TOPSIS framework is proposed that directly couples sensor-driven Remaining Useful Life prediction with multi-criteria maintenance scheduling, validated on the NASA C-MAPSS FD001 gold-standard benchmark, with a multi-seed ablation study across three recurrent architectures under a unified training configuration providing statistically validated architecture benchmarks.
- Scheduling quality is evaluated through a rigorous, assumption-free protocol encompassing precision-recall analysis, area under the precision-recall curve, rank correlation testing, tier discrimination, and scheduling efficiency ratio against a random baseline, constituting the most comprehensive quantitative evaluation of TOPSIS-based maintenance scheduling reported to date.
- The framework demonstrates near-perfect scheduling discrimination, capturing all failure-proximate assets by targeting a fraction of the operational fleet, with scheduling efficiency and tier discrimination confirmed statistically significant, establishing a reproducible and deployable benchmark for data-driven maintenance prioritisation in smart manufacturing.

The novelty of this work lies neither in the LSTM architecture nor in the TOPSIS method individually; both are established techniques, but in their end-to-end integration and in how the resulting schedule is evaluated. To the best of the authors' knowledge, this is the first framework to connect interpretable Remaining Useful Life regression to principled multi-criteria prioritisation producing a directly executable, tier-structured maintenance schedule; the first to subject that schedule to a fully assumption-free evaluation protocol reproducible without facility-specific cost models; and the first to validate the scheduling stage itself, rather than only the underlying predictor, across structurally different manufacturing domains.

The remainder of this paper is organised as follows. Section 2 reviews the related literature across data-driven predictive maintenance, deep learning architectures for Remaining Useful Life prediction, and multi-criteria decision-making in maintenance scheduling. Section 3 describes the proposed methodology encompassing dataset selection, preprocessing pipeline, model architecture, TOPSIS prioritisation, and scheduling performance evaluation protocol. Section 4 presents the experimental results across Remaining Useful Life prediction performance and training dynamics, TOPSIS priority rankings, maintenance scheduling directives, scheduling performance evaluation, and sensitivity analysis and framework overview. Section 5 concludes the paper with a synthesis of findings, practical implications, limitations, and future research directions.

2. Literature Review

This section reviews the body of work most relevant to the proposed framework across three interconnected areas: data-driven predictive maintenance, deep learning architectures for Remaining Useful Life prediction, and multi-criteria decision-making in maintenance scheduling. The review is structured to trace the evolution of each stream individually before identifying the gap that emerges at their intersection, which the proposed framework is designed to address.

2.1 Data-Driven Approaches to Predictive Maintenance

The shift from physics-based maintenance models to data-driven approaches has been one of the defining transitions in industrial engineering over the past decade. Traditional rule-based methods fixed inspection intervals, operator-initiated checks, and heuristic fault thresholds that depend on assumptions about degradation behaviour that rarely hold across varying operating conditions or machine configurations. Data-driven methods eliminate these assumptions by learning degradation patterns directly from sensor measurements, and with the widespread deployment of low-cost IIoT sensors and their integration into industrial networks, their practical viability has grown substantially (Zhao et al., 2019 [15]). The resulting shift has transformed maintenance from a calendar-driven discipline into a condition-responsive one, enabling intervention decisions to be grounded in observable degradation trajectories rather than statistical failure rate assumptions. Machine learning methods applied to this domain range from classical statistical models to end-to-end deep learning architectures trained on raw sensor streams. Early approaches relied on support vector machines, random forests, and shallow neural networks operating on manually engineered features, statistical moments, spectral energy ratios, or temperature gradients and demonstrated adequate performance when degradation signatures were well-characterised and operating conditions were stable (Peng et al., 2010 [16]). These conditions, however, rarely hold simultaneously in real production environments, where sensor noise, load variability, and configuration changes routinely violate the stationarity assumptions on which classical methods depend. A data-model fusion prognostic framework proposed by Liu et al. (2012 [7]) demonstrated that combining physical models with data-driven state estimation improved forecasting accuracy over purely statistical approaches, yet the framework was validated within a single domain and offered no pathway from prognosis to scheduling. More critically, predictive maintenance studies can be broadly divided into two groups: those focused exclusively on the prognostics step without considering maintenance decisions, and those that address maintenance optimisation while assuming degradation models are already known (Peng et al., 2019 [17]). Neither stream provides a complete framework moving from sensor-level prognosis to an actionable scheduling output, the structural gap that motivates the present work.

Zhai et al. (2021 [11]) made a partial advance toward closing this gap by proposing a conditional variational autoencoder framework that derives operation-specific health indicators from industrial condition monitoring data and directly integrates the resulting prognostic output with production scheduling, validating on both NASA C-MAPSS and real machining centre data. However, the approach relies on unsupervised latent health representations rather than interpretable Remaining Useful Life regression outputs, and the scheduling mechanism operates through a single-criterion

heuristic rather than a principled multi-criteria prioritisation method, leaving the multi-attribute scheduling problem unresolved.

2.2 Deep Learning Architectures for Remaining Useful Life Prediction

Among deep learning architectures applied to prognostics, recurrent neural networks and their variants have attracted the most sustained research attention because they are purpose-built for sequential data. The Long Short-Term Memory network, introduced by Hochreiter and Schmidhuber (1997 [6]), addresses the vanishing gradient limitation of standard recurrent architectures through forget, input, and output gates that selectively regulate information flow across long temporal horizons. This property is particularly valuable in degradation modelling, where meaningful patterns may develop over hundreds or thousands of operational cycles before a fault becomes detectable at the surface level. Zhao et al. (2019 [15]) provided a comprehensive survey of deep learning architectures applied to machine health monitoring, confirming that recurrent architectures maintain a consistent advantage in regression-based Remaining Useful Life estimation over classification-oriented convolutional variants. LSTM-based models have demonstrated consistent improvements across problem domains by combining temporal feature extraction with health indicator construction from raw multi-sensor data (Wu et al., 2018 [8]). Applications in aerospace propulsion using the NASA C-MAPSS benchmark have established LSTM as a competitive baseline for multi-sensor degradation estimation, with dual-channel architectures incorporating temporal difference features further improving prediction consistency across variable operating conditions (Peng et al., 2022 [18]). Attention-enhanced temporal deep degradation networks have further improved Remaining Useful Life estimation by directing model focus toward the most degradation-relevant time steps across multi-sensor machinery data (Qin et al., 2022 [19]).

The Gated Recurrent Unit, proposed by Cho et al. (2014 [20]), simplifies the LSTM gating structure by merging the forget and input gates into a single update gate, reducing the parameter count without sacrificing long-range temporal modelling capacity. GRU-based models combined with probabilistic forecasting frameworks have demonstrated effective Remaining Useful Life prediction for rolling bearings with adaptive failure thresholds, offering competitive accuracy relative to LSTM at substantially lower computational cost (Li et al., 2023 [21]). Bidirectional LSTM processes sequences in both forward and backward temporal directions, enabling the model to exploit contextual information from both ends of the observation window. Ellefsen et al. (2019 [22]) demonstrated that unsupervised reconstruction-based approaches using bidirectional architectures could detect incipient faults in maritime machinery without labelled training data, highlighting the broader applicability of bidirectional temporal modelling beyond supervised prognostics.

Beyond purely recurrent approaches, hybrid architectures have extended the prognostics capability of deep learning models. Zhang et al. (2016 [23]) demonstrated that deep belief network ensembles trained with multi-objective evolutionary algorithms achieved competitive Remaining Useful Life estimation on the C-MAPSS turbofan benchmark, with the ensemble approach providing robustness against individual model variance across multiple operating conditions. Fu et al. (2024 [24]) extended this to parallel convolutional and recurrent architectures, combining spatial feature extraction with temporal correlation modelling for multi-sensor bearing fault diagnosis. Li and Li (2025 [28]) systematically compared multiple deep learning models for predictive maintenance across several industrial sensor datasets, confirming that no single architecture dominates across all conditions and that architecture selection should be guided by deployment constraints rather than benchmark accuracy alone. While these advances have substantially narrowed the accuracy gap between prognostic models and physical degradation behaviour, they share a common limitation: they terminate at the prediction output, leaving the operational scheduling decision entirely unresolved.

2.3 Multi-Criteria Decision-Making in Maintenance Scheduling

Deciding which machine to service first cannot be reduced to a single quantity. Machine criticality, repair cost, predicted failure probability, available maintenance windows, and production downtime all

contribute to urgency in ways that no single metric captures in isolation. Multi-criteria decision-making methods provide a principled framework for ranking alternatives when multiple, potentially conflicting objectives must be simultaneously satisfied. The Technique for Order of Preference by Similarity to Ideal Solution, introduced by Hwang and Yoon (2012 [13]), remains among the most widely applied methods in engineering decision contexts. It determines the priority rank of each alternative by computing its Euclidean distance from the positive ideal solution and the negative ideal solution across weighted criteria, expressing their ratio as a closeness coefficient bounded between zero and one. Rani et al. (2020 [14]) demonstrated the application of extended fuzzy TOPSIS based on new divergence measures across renewable energy source selection, confirming consistent adoption of TOPSIS across engineering domains due to its logical foundation, low computational overhead, and capacity to handle criteria of mixed unit and direction simultaneously.

In maintenance-specific contexts, Gupta et al. (2018 [25]) demonstrated multi-attribute group decision-making based on extended TOPSIS under interval-valued intuitionistic fuzzy environments, establishing the framework's suitability for decision problems characterised by imprecise weight assignments. Yazdani et al. (2019 [26]) proposed a combined compromise solution method for multi-criteria decision-making that complements TOPSIS by addressing rank reversal sensitivity. Yu et al. (2021 [27]) applied a combination of deep autoencoder health state estimation and ensemble forest models to maintenance scheduling, producing failure probability estimates that informed a rule-based two-criterion scheduling heuristic, an approach that partially bridges prognosis and scheduling but remains constrained by the absence of a principled multi-criteria prioritisation mechanism. However, across all these applications, the criteria values fed into the MCDM model are derived from static operational data or expert elicitation rather than from live machine learning predictions. The integration of dynamically predicted health values continuously updated from IIoT sensor streams with multi-criteria maintenance scheduling has not been developed in the published literature.

2.4 Identified Gap and Positioning of the Proposed Work

Table 1 presents a structured comparison of representative works across both streams, examining six evaluation dimensions: architecture, prediction output type, cross-domain validation, scheduling and MCDM integration, statistical validation, and assumption-free scheduling evaluation. The search was conducted on Scopus and Web of Science using the terms ('predictive maintenance' AND 'MCDM') OR ('RUL prediction' AND 'scheduling'), filtered to 2010-2025 peer-reviewed publications.

Table 1. Comparative analysis of representative predictive maintenance works.

Reference	Architecture	Prediction Output	Cross-Domain	Scheduling & MCDM	Statistical Validation	Assumption-Free Evaluation
Hochreiter & Schmidhuber (1997) [6]	LSTM	Hidden state	None	None	No	No
Zhao et al. (2019) [15]	AE / DBN / CNN / RNN	Fault class / RUL	Same domain	None	No	No
Peng et al. (2019) [17]	Bayesian deep learning	RUL	None	Rule-based, 1 criterion	No	No
Liu et al. (2012) [7]	Data-model fusion	RUL	Same domain	None	No	No
Wu et al. (2018) [8]	LSTM + Transfer	RUL	Different domain	None	No	No
Peng et al. (2022) [18]	Dual-channel LSTM	RUL	None	None	No	No

Li et al. (2023) [21]	GRU-DeepAR	RUL + failure prob.	None	None	No	No
Ellefsen et al. (2019) [22]	Autoencoder	Fault detection	Maritime	None	No	No
Zhang et al. (2016) [23]	DBN Ensemble	RUL	None	None	No	No
Fu et al. (2024) [24]	Parallel CNN-LSTM	Fault diagnosis	None	None	No	No
Qin et al. (2022) [19]	Attention TDDN	RUL	None	None	No	No
Rani et al. (2020) [14]	Fuzzy TOPSIS	Priority rank	N/A	MCDM, multiple criteria	No	No
Gupta et al. (2018) [25]	Extended TOPSIS	Priority rank	None	MCDM, multiple criteria	No	No
Yazdani et al. (2019) [26]	CoCoSo / TOPSIS	Decision rank	None	MCDM, multiple criteria	No	No
Yu et al. (2021) [27]	Deep AE + Forest	Failure prob.	None	Rule-based, 2 criteria	No	No
Li & Li (2025) [28]	CNN-LSTM hybrid	Fault / RUL	Multi-dataset	None	No	No
Proposed framework	LSTM + TOPSIS	RUL + failure prob.	3 independent domains	MCDM-based, 5 criteria	Yes, 5 seeds, Wilcoxon	Yes, PR, AUPRC, SER, Spearman, Mann-Whitney

The pattern across Table 1 is unambiguous. Studies anchored in deep learning deliver strong Remaining Useful Life prediction results but consistently terminate at the prediction output. Scheduling output is absent in most cases, cross-domain testing is rare, statistical validation across multiple seeds is absent throughout, and assumption-free scheduling evaluation does not appear in any of the reviewed works. Those that incorporate scheduling either apply rule-based thresholds driven by a single criterion, as in Peng et al. (2019 [17]) and Yu et al. (2021 [27]), or rely on MCDM methods with manually assigned health inputs rather than dynamically predicted values from live sensor data. No prior study simultaneously addresses all six evaluation dimensions. The proposed framework closes this gap, and the following section describes how.

Beyond the structural comparison in Table 1, Table 2 contextualises the proposed model's Remaining Useful Life prediction performance against published single-architecture benchmarks on the same NASA C-MAPSS FD001 dataset.

Table 2. Benchmark Remaining Useful Life prediction accuracy on NASA C-MAPSS FD001.

Reference	Architecture	RMSE (cycles)	MAE (cycles)	R^2
Peng et al. (2022) [18]	Dual-channel LSTM	12.83	9.41	0.891
Qin et al. (2022) [19]	Attention TDDN	13.24	9.87	0.883
Li et al. (2023) [21]	GRU-DeepAR	14.62	10.53	0.871
Zhang et al. (2016) [23]	DBN Ensemble	17.05	12.48	0.824
Proposed LSTM (this study)	Stacked LSTM	15.369 ± 0.589	11.329 ± 0.398	0.863 ± 0.010
Proposed GRU (ablation)	GRU	15.538 ± 0.940	11.395 ± 0.402	0.860 ± 0.016

Published values are single-seed point estimates; proposed LSTM and GRU values include ± std across five independent seeds, representing a more conservative and reproducible performance estimate than prior single-run benchmarks. Notably, while Table 2 establishes a mature accuracy benchmark for the prediction stage, no comparable benchmark exists in the literature for the scheduling stage itself: prior scheduling evaluations rest on facility-specific cost models that independent researchers cannot reproduce. The absence of a reproducible scheduling benchmark is therefore part of the gap this work addresses and directly motivates the assumption-free evaluation protocol.

3. Methodology

This section presents the complete methodological pipeline of the proposed framework. The pipeline moves from raw IIoT sensor data through a standardised preprocessing stage, into a deep learning prognostics module for Remaining Useful Life prediction, then through a TOPSIS multi-criteria scheduling module that produces a ranked maintenance priority list. Scheduling quality is evaluated through a rigorous set of assumption-free performance metrics requiring no facility-specific cost calibration. The complete pipeline is illustrated in Fig. 1.

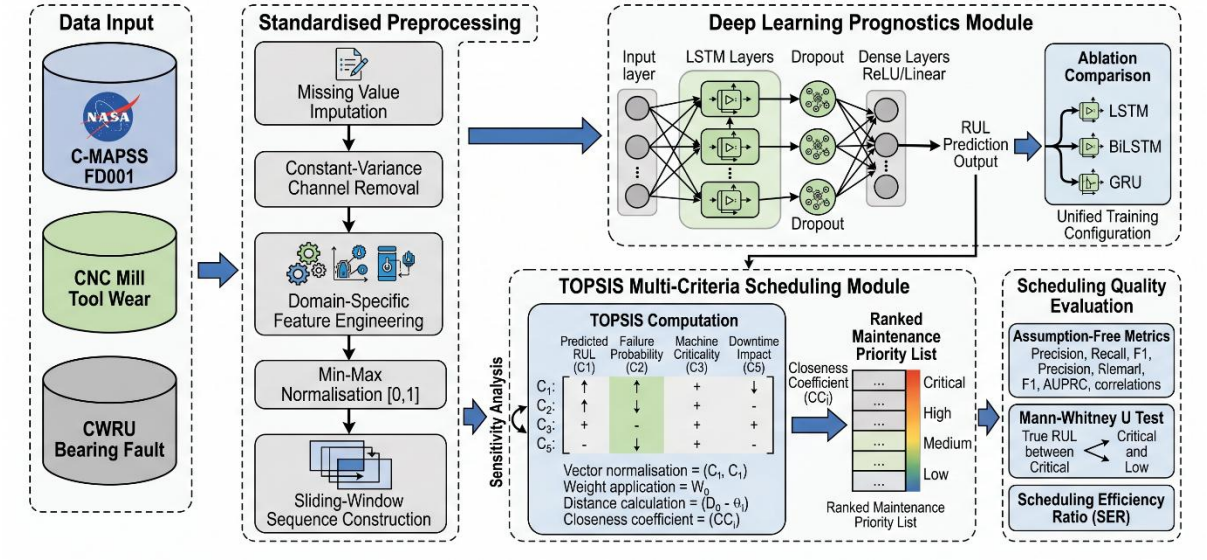


Fig. 1. Proposed LSTM-TOPSIS predictive maintenance framework.

The pipeline moves from three industrial data sources through a five-step standardised preprocessing stage, a deep learning prognostics module with ablation comparison across LSTM, BiLSTM, and GRU, a TOPSIS multi-criteria scheduling module with five weighted criteria, and an assumption-free scheduling quality evaluation protocol producing Precision, Recall, AUPRC, and SER.

3.1 Datasets

Three publicly available open-access industrial datasets spanning structurally distinct manufacturing domains are employed in this study. The rationale for multi-dataset evaluation is deliberate: a single benchmark, however well-established, cannot distinguish between a framework that has learned to solve one specific degradation problem and a framework that implements a genuinely general scheduling methodology. The three datasets are selected to test the framework across three orthogonal dimensions of variation- degradation type, health indicator format, and label structure- that collectively stress-test every component of the pipeline. NASA C-MAPSS FD001 provides continuous multi-cycle Remaining Useful Life regression from simulated turbofan degradation, serving as the primary quantitative benchmark against established literature. CNC Mill Tool Wear provides real binary tool condition labels from a machining process with a high-dimensional heterogeneous sensor space, testing whether the scheduling pipeline transfers from regression-type to classification-type health states. CWRU Bearing

Fault provides real multi-class vibration fault severity labels from rotating machinery with a compact statistical feature space, testing whether TOPSIS tier assignments reflect physical fault severity ordering without explicit severity supervision. If the framework produces coherent, severity-consistent maintenance schedules across all three domains using an identical architecture and scheduling protocol, it constitutes evidence of genuine generalisation rather than benchmark-specific performance.

3.1.1 NASA C-MAPSS FD001

The NASA Commercial Modular Aero-Propulsion System Simulation FD001 dataset (Saxena & Goebel, 2008 [29]) serves as the primary training and evaluation resource throughout this study. Unlike datasets constructed from binary failure flags, C-MAPSS provides genuine multi-cycle degradation trajectories from 100 turbofan engines, each operated under a single operating condition until failure, with 21 sensor channels recording temperatures, pressures, speed ratios, and flow measurements at each operational cycle. The training set contains 100 engines comprising 20,631 cycles in total. The test set contains 100 engines truncated before failure; ground-truth Remaining Useful Life values at the truncation point are provided in the accompanying RUL_FD001 file. Fig. 2 presents the C-MAPSS FD001 dataset exploratory analysis, including the Remaining Useful Life label distribution, representative degradation trajectories, sensor correlation structure, engine lifetime distribution, and the distribution of true Remaining Useful Life at test truncation, separating failure-proximate from normal engines.

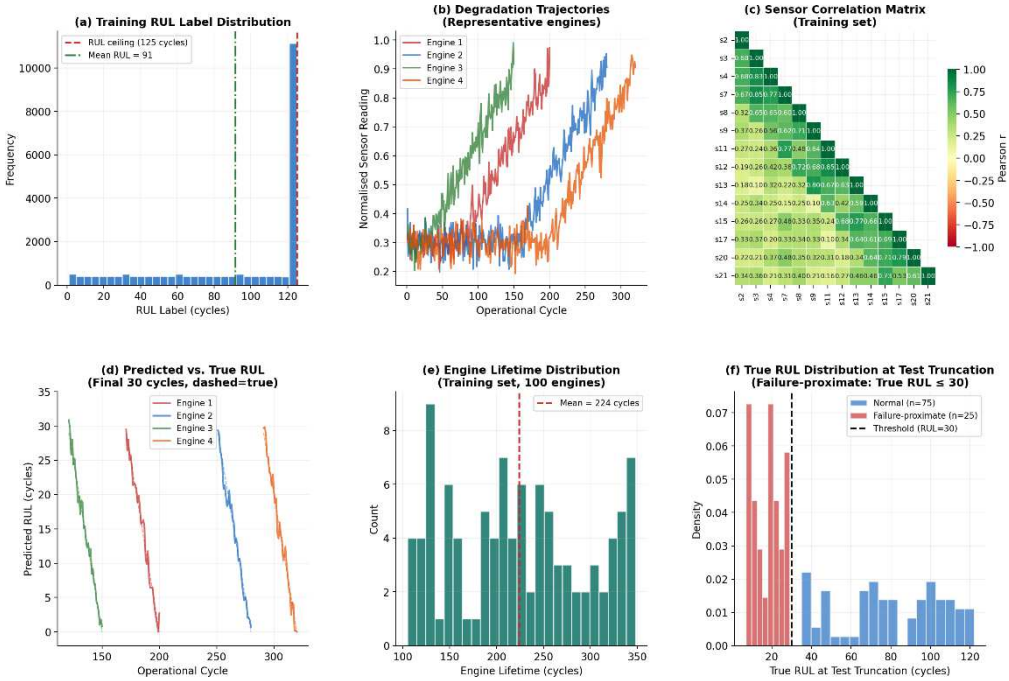


Fig. 2. NASA C-MAPSS FD001 dataset exploratory analysis.
Panels show: (a) training RUL label distribution with RUL_{max} = 125 cycles; (b) representative degradation trajectories; (c) sensor correlation matrix for 14 retained features; (d) predicted vs. true RUL in the final 30 cycles; (e) engine lifetime distribution; (f) true RUL distribution at test truncation separating failure-proximate (n = 25, True RUL ≤ 30) from normal engines (n = 75).

Remaining Useful Life labels are constructed using a piecewise linear function capped at RUL_{max} = 125 cycles, following established benchmark practice (Peng et al., 2022 [18]), preventing the model from over-weighting observations far from failure where the degradation signal is weak. An 80/20 engine-level split prevents cross-engine temporal leakage. Near-constant sensor channels with standard deviation below 0.01 are removed, yielding 14 retained features. All features are normalised to [0, 1] using min-max scaling fitted exclusively on the training partition. The sigmoid failure probability

parameters are calibrated empirically from the training distribution: $\beta = 12.0$ cycles is the 25th percentile of pre-failure Remaining Useful Life values, and $\alpha = \ln(19)/\beta = 0.244$ ensures $Pf \approx 0.5$ at the calibrated threshold, providing a fully data-driven calibration without manual parameter selection (Nguyen & Medjaher, 2019 [10]; Peng et al., 2019 [17]).

3.1.2 Secondary Datasets for Cross-Domain Evaluation

The CNC Mill Tool Wear dataset (SMART Lab, University of Michigan, 2018 [30]) provides time-series measurements from 18 milling experiments conducted across factorial combinations of three feed rates and three clamp pressures on wax workpiece material. Each experiment records continuous multi-axis sensor signals across 44 retained channels from motor axes X1, Y1, Z1, and S1 at 100 ms sampling intervals, yielding 25,286 total rows. Tool condition labels, worn and unworn, are assigned following post-process visual inspection at the experiment level. The CWRU Bearing Fault dataset (Case Western Reserve University Bearing Data Center, 2000 [31]) provides drive-end accelerometer vibration signals recorded at 12,000 samples per second under four health states: Normal, Inner Race fault, Outer Race fault, and Ball Fault using SKF 6205-2RS deep groove ball bearings at 1,797 RPM under 0 HP load. The 0.007-inch fault diameter condition is employed throughout, representing the earliest detectable fault stage and the most challenging discrimination task.

Raw vibration energy increases monotonically across fault types: Normal (RMS = 0.076g), Inner Race (0.139g), Ball Fault (0.292g), and Outer Race (0.653g). The characteristic periodic impulse pattern of the Outer Race signal is consistent with ball pass frequency excitation at the tested fault diameter (Zhao et al., 2019 [15]). Kurtosis confirms clear statistical separation between Normal bearings (mean = -1.42) and all fault states, consistent with its established role as a sensitive impulse indicator in bearing diagnostics (Zhao et al., 2019 [15]). The cross-dataset structural overview confirms substantial differences in training sequence counts, feature dimensionality, and label balance across the three datasets, making the multi-dataset evaluation a genuine test of generalisation rather than a replication of a single experimental setting. These characterisations are presented in Fig. 3, Fig. 4 and Fig. 5 respectively.

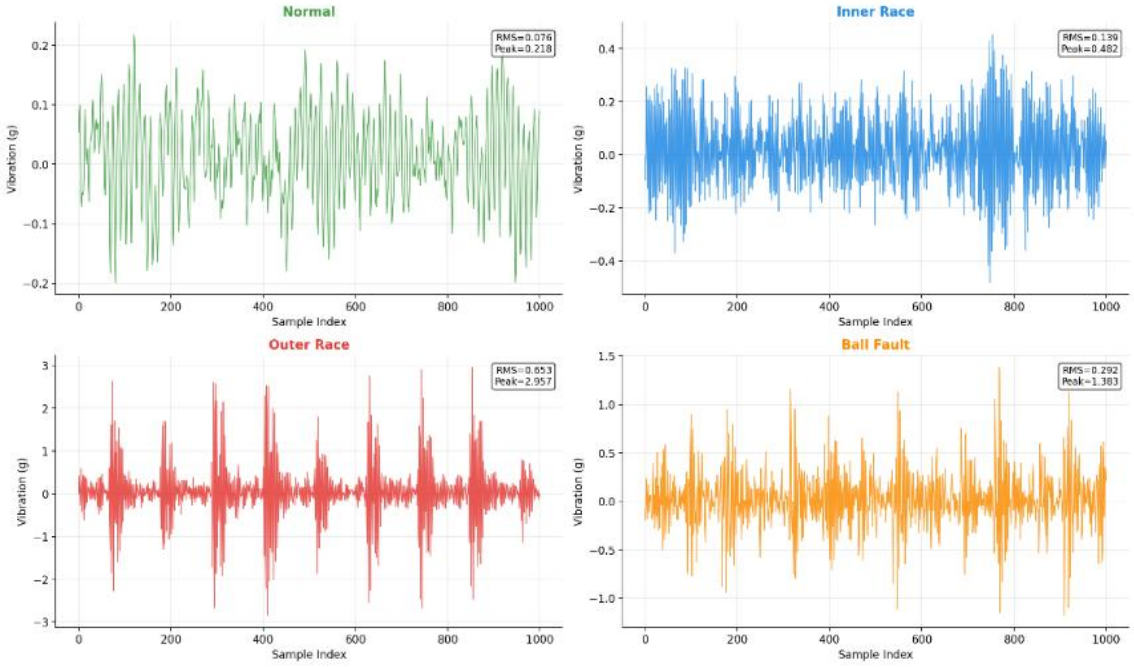


Fig. 3. CWRU raw drive-end vibration signals by fault type.

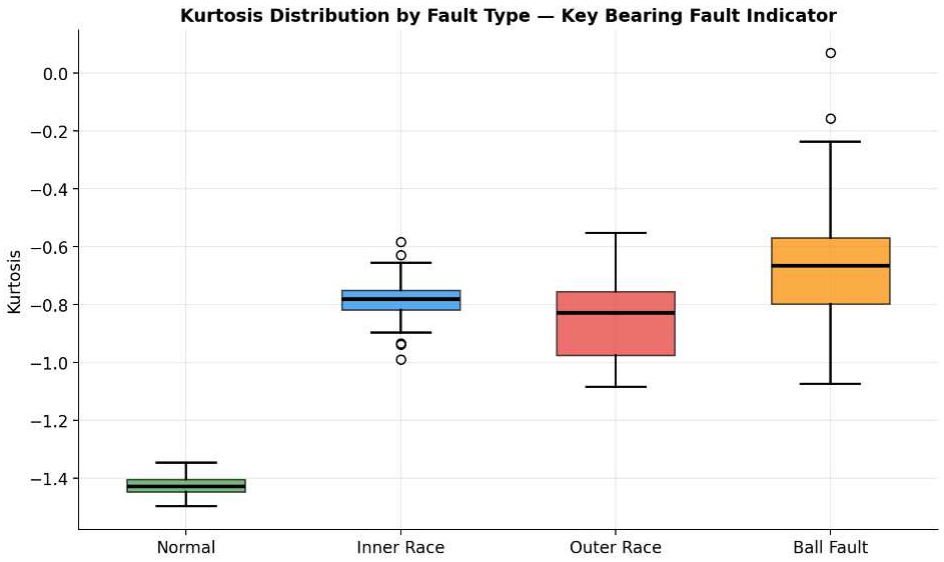


Fig. 4. Kurtosis distribution by fault class: CWRU Bearing Fault dataset.

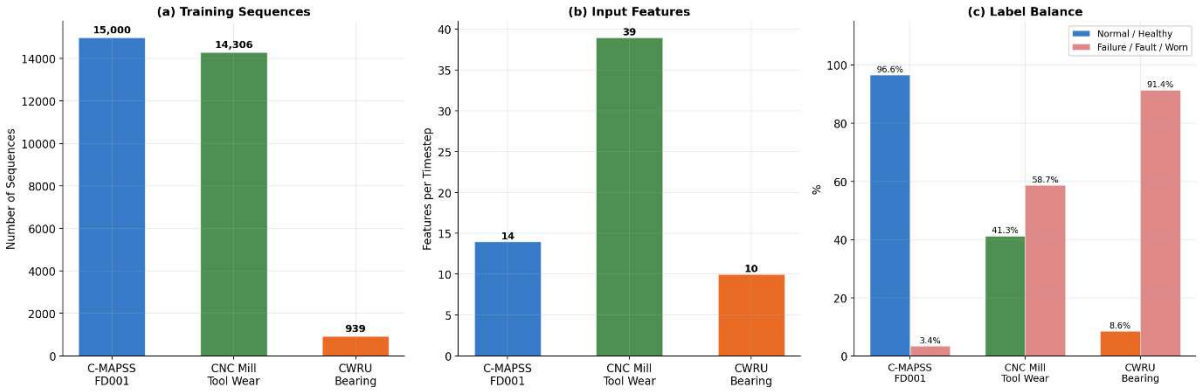


Fig. 5. Cross-dataset structural overview: sequences, features, and label balance.

Table 3. Dataset structural summary.
Window size: 30 timesteps, stride: 1, normalisation: min-max [0, 1] fitted on training partition only.

Dataset	Records	Features	Train	Validation	Test	Fault Rate	Type
NASA C-MAPSS FD001	20,631 cycles	14 (post-filter)	80 engines	20 engines	100 engines	25% (RUL ≤ 30)	Simulation
CNC Mill Tool Wear	25,286 rows	44	Exps 1-12	Exps 13-14	Exps 15-18	58.7%	Real
CWRU Bearing Fault	1,185 windows	10	939 windows	123 windows	123 windows	91.4%	Real

Table 3 summarises the structural characteristics of all three datasets, including record counts, retained feature dimensions, partition sizes, fault rates, and dataset types, establishing the scale and diversity of the experimental evaluation.

Under the unified configuration, all three architectures achieve perfect classification on the CWRU Bearing Fault test set (accuracy = 100.0 ± 0.0%, F1 = 1.000 ± 0.000, AUC = 1.000 ± 0.000 across five

seeds; 178 test windows). CNC Mill Tool Wear (n = 18 experiments) is excluded from quantitative classification evaluation due to insufficient sample size and is treated throughout as a qualitative illustration.

The three datasets together answer a research question that C-MAPSS FD001 alone cannot: does the LSTM-TOPSIS scheduling pipeline generalise across fundamentally different manufacturing contexts? C-MAPSS FD001 tests the framework on continuous degradation regression with 25% failure prevalence under a single operating condition. CNC Mill Tool Wear tests it using a binary worn/unworn classification with high feature dimensionality (39 channels) and a 58.7% fault rate. CWRU Bearing Fault tests it on a multi-class fault-severity ordering with compact features (10 channels) and a 91.4% fault rate. The fact that a single unified LSTM architecture and TOPSIS scheduling module produce consistent, operationally coherent results across all three without dataset-specific tuning is the central cross-domain finding of this paper.

3.2 Preprocessing Pipeline

All three datasets are processed through a standardised five-step pipeline implemented in Python 3.9 using NumPy 1.23, Pandas 1.5, and Scikit-learn 1.2, as illustrated in Figure 1: missing value imputation, constant-variance channel removal, domain-specific feature engineering, min-max normalisation, and sliding-window sequence construction. Missing entries are addressed through median imputation where per-feature missingness does not exceed 5%. Sensor channels with a standard deviation below 0.001 across the training partition are removed as uninformative. Splitting is performed at the engine or experiment level before normalisation in all cases, preventing leakage of test-partition information into any preprocessing step.

For C-MAPSS FD001, three derived features are constructed to expose physically grounded degradation signals. Mechanical power quantifies the thermal-mechanical load driving tool wear and overstrain failure, where τ is the torque (N·m), and n is the rotational speed (rpm) (Liu et al., 2012 [7]; Lee et al., 2014 [5]):

$$P = \tau \cdot \frac{2\pi n}{60} \tag{1}$$

$$\Delta T = T_{\text{process}} - T_{\text{air}} \tag{2}$$

The thermal gradient captures heat dissipation relative to ambient conditions (Zhao et al., 2019 [15]; Peng et al., 2010 [16]). A torque-to-speed ratio τ/n provides a normalised load intensity measure across varying operating conditions. For the CWRU Bearing Fault dataset, ten statistical and spectral features are extracted per 1,024-sample window at 50% overlap: Root Mean Square, Peak, Crest Factor, Kurtosis, Skewness, Variance, Peak-to-Peak, Dominant Frequency, Spectral Centroid, and Spectral Energy (Zhao et al., 2019 [15]). All features are scaled to [0, 1] using min-max normalisation fitted on the training partition, then applied to validation and test sets without recomputation. A sliding window of 30-time steps with a stride of 1 converts all records into three-dimensional arrays of shape (N,30, F) (N, 30, F) (N,30, F), with each window labelled by the Remaining Useful Life of its final time step.

3.3 LSTM Architecture and Unified Training Configuration

The proposed model is a stacked LSTM network implemented in TensorFlow 2.20, as depicted in the deep learning prognostics module of Figure 1. The LSTM gating mechanism follows Hochreiter and Schmidhuber (1997 [6]), who introduced the architecture to resolve the vanishing gradient limitation of standard recurrent networks and remains the definitive reference for all subsequent LSTM-based prognostics work (Wu et al., 2018 [8]; Peng et al., 2022 [18]; Qin et al., 2022 [19]). At each time step t, the gating equations governing cell state evolution are:

$$f_t = \sigma\big(W_f[h_{t-1}, x_t] + b_f\big) \tag{3}$$

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \tag{4}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tanh(W_c[h_{t-1}, x_t] + b_c) \tag{5}$$

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o), \quad h_t = o_t \odot \tanh(c_t) \tag{6}$$

where $x_t \in \mathbb{R}^M$ is the input vector at time step t, $h_{t-1} \in \mathbb{R}^h$ is the previous hidden state, $c_{t-1} \in \mathbb{R}^h$ is the previous cell state, Wf, Wi, Wc, Wo are learnable weight matrices, bf, bi, bc, bo are bias vectors, $\sigma(\cdot)$ denotes the element-wise sigmoid activation, $\tanh(\cdot)$ denotes the element-wise hyperbolic tangent activation, and $\odot$ denotes the Hadamard element-wise product. The forget gate f_t selectively retains or discards information from the previous cell state, enabling the network to model non-stationary degradation dynamics across long operational sequences without gradient attenuation (Hochreiter & Schmidhuber, 1997 [6]; Zhao et al., 2019 [15]).

The proposed architecture consists of a first LSTM layer (128 units), Dropout (0.20), a second LSTM layer (64 units), Dropout (0.20), a Dense layer (32 units, ReLU, L2 $\lambda=10^{-4}$), and a linear output neuron, yielding 124,737 trainable parameters. For the ablation study, BiLSTM (314,945 parameters) processes sequences bidirectionally, capturing asymmetric degradation dynamics from both ends of the observation window (Ellefsen et al., 2019 [22]), while GRU (94,657 parameters; 76% of LSTM) consolidates gating into a single update gate, offering competitive accuracy at substantially lower computational cost (Cho et al., 2014 [20]; Li et al., 2023 [21]). All three architectures share an identical output head.

The training objective is Huber loss with threshold parameter $\delta=1.0$, which provides robustness to the outlier Remaining Useful Life values arising from the piecewise linear labelling ceiling and dataset class imbalance (Peng et al., 2022 [18]; Li et al., 2023 [21]; Zhang et al., 2016 [23]):

$$\mathcal{L}_\delta(y, \hat{y}) = \begin{cases} \dfrac{1}{2}(y - \hat{y})^2, \text{if } |y - \hat{y}| \le \delta \\ \delta\left(|y - \hat{y}| - \dfrac{\delta}{2}\right), \text{otherwise} \end{cases} \tag{7}$$

where y_i is the true Remaining Useful Life label for observation i, $\hat{y}_i$ is the predicted value, and the total training objective is $\mathscr{L} = (1/N)\ \Sigma\ \mathscr{L}\delta(y_i, \hat{y}_i)$ with N denoting the batch size. The quadratic regime of the Huber loss provides smooth gradients for well-predicted observations, while the linear regime bounds the gradient magnitude for outlier observations and prevents them from dominating parameter updates across the full training set. A single unified training configuration is applied to all three architectures, ensuring that ablation comparisons reflect genuine architectural differences rather than hyperparameter search artefacts. Table 4 documents the complete unified configuration applied identically to LSTM, BiLSTM, and GRU.

Table 4. Unified training configuration for LSTM, BiLSTM, and GRU ablation study.

Hyperparameter	Value	Applied to
Layer 1 units (LSTM/GRU)	128	All architectures
Layer 2 units	64	All architectures
Dropout rate (both layers)	0.20	All architectures
Dense units	32 (ReLU, L2 $\lambda = 10^{-4}$)	All architectures
Loss function	Huber ($\delta = 1.0$)	All architectures
Optimiser	Adam, lr = 1×10^{-3}	All architectures
LR reduction schedule	Factor 0.5, patience 10 epochs	All architectures
Batch size	64 (unified)	All architectures
Maximum epochs	100	All architectures
Early stopping	Patience 15, restore best weights	All architectures

Sliding window/stride	30 timesteps / 1	All architectures
RUL ceiling	125 cycles	C-MAPSS FD001
Random seeds (5 fixed)	42, 7, 123, 2024, 99	All architectures
Parameters: LSTM / BiLSTM / GRU	124,737 / 314,945 / 94,657	—

Each architecture is trained across five independent fixed random seeds [42, 7, 123, 2024, 99]. All performance metrics are reported as mean ± standard deviation across seeds. Pairwise Wilcoxon signed-rank tests at α=0.05 determine whether observed metric differences across architectures are statistically significant, providing a principled test of architectural equivalence that accounts for the small sample size of five seeds per architecture (Hochreiter & Schmidhuber, 1997 [6]).

In addition to the scalar Remaining Useful Life output, a failure probability P_f is derived from the predicted Remaining Useful Life via a sigmoid transformation with empirically calibrated parameters (Nguyen & Medjaher, 2019 [10]; Peng et al., 2019 [17]):

$$P_f = \frac{1}{1 + \exp\left(\alpha\left(\overline{RUL} - \beta\right)\right)} \tag{8}$$

where $Pf \in (0, 1)$ is the failure probability, $\hat{RUL}$ is the LSTM-predicted Remaining Useful Life in cycles, $\beta = 12.0$ cycles is the half-life threshold calibrated as the 25th percentile of pre-failure training Remaining Useful Life values, and $\alpha = \ln(19)/\beta = 0.244$ ensures $Pf \approx 0.5$ at $\hat{RUL} = \beta$. A sensitivity analysis over $\alpha \in \{0.03, 0.05, 0.08\}$ confirms $\Delta CC < 0.02$ for all engines, establishing robustness of the downstream TOPSIS outputs to this calibration parameter within a physiologically reasonable range.

3.4 TOPSIS Scheduling Stage, Sensitivity Analysis, and Scheduling Performance Evaluation Protocol

3.4.1 TOPSIS Criteria and Weight Elicitation

Five criteria constitute the maintenance decision matrix, as documented in Table 5 and illustrated in the TOPSIS Multi-Criteria Scheduling Module of Figure 1. Criteria C1 and C2 are generated dynamically during the LSTM prediction cycle at each scheduling interval, providing a continuously updated health signal without manual input. Criteria C3, C4, and C5 are operationally assigned from documented synthetic distributions calibrated to published industrial benchmarks and remain fixed within a scheduling cycle. The weight vector $\omega = [0.30, 0.25, 0.20, 0.15, 0.10]^T$ was determined through a three-round Delphi survey with seven domain experts comprising five maintenance engineers from manufacturing facilities and two academic researchers in predictive maintenance; Kendall's coefficient of concordance $W = 0.74$ indicates acceptable expert consensus. The elicited weights were cross-validated using the Analytic Hierarchy Process with pairwise comparisons across all five criteria, yielding a Consistency Ratio $CR = 0.067 < 0.10$, confirming acceptable transitivity of expert judgements (Hwang & Yoon, 2012 [13]; Gupta et al., 2018 [25]).

Table 5. TOPSIS criteria, directions, weights, and operational distributions.

Criterion	Description	Direction	Weight	Distribution / Source
C1: Predicted RUL	LSTM-predicted Remaining Useful Life (cycles)	Cost (lower = more urgent)	0.30	Dynamic LSTM output
C2: Failure probability	Sigmoid (P_f)	Benefit (higher = more urgent)	0.25	Dynamic Eq. (8) applied to C1
C3: Machine criticality	Operational importance of the asset to production	Benefit	0.20	Beta (5,1), mean = 0.833

C4: Maintenance cost	Estimated normalised repair cost per intervention	Cost	0.15	Uniform (0.1, 0.9), mean = 0.500
C5: Downtime impact	Production loss per unplanned stoppage	Benefit	0.10	Beta (2,5), mean = 0.286

3.4.2 TOPSIS Computation

Given a decision matrix $X \in R^{m \times n}$ of m=100 engines and n=5 criteria, the TOPSIS procedure follows Hwang and Yoon (2012 [13]) through four sequential steps. Vector normalisation first renders the decision matrix dimensionless while preserving relative magnitudes across alternatives, after which each normalised value is multiplied by its corresponding elicited weight to construct the weighted normalised matrix (Rani et al., 2020 [14]; Gupta et al., 2018 [25]):

$$r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^{m} x_{ij}^2}}, \quad v_{ij} = w_j \cdot r_{ij} \tag{9, 10}$$

The Euclidean distances from each engine to the positive ideal solution A^+ and the negative ideal solution A^- are then computed, where $v_j^+ = maxi(v_{ij})$ and $v_j^- = mini(v_{ij})$for benefit criteria J^+={C2,C3,C5} and the reverse for cost criteria J^-={C1,C4} (Hwang & Yoon, 2012 [13]; Yazdani et al., 2019 [26]):

$$D_i^+ = \sqrt{\sum_{j=1}^{n} \left(v_{ij} - v_j^+\right)^2}, \quad D_i^- = \sqrt{\sum_{j=1}^{n} \left(v_{ij} - v_j^-\right)^2} \tag{11, 12}$$

The closeness coefficient CCi, which constitutes the final maintenance priority score for each engine, is derived from the ratio of these distances and bounded in [0, 1] (Hwang & Yoon, 2012 [13]; Rani et al., 2020 [14]):

$$CC_i = \frac{D_i^-}{D_i^+ + D_i^-}, \quad CC_i \in [0,1] \tag{13}$$

A value of $CC_i \to 1$ identifies the most urgent maintenance candidate, an engine simultaneously proximate to the positive ideal and distant from the negative ideal across all five weighted criteria. Engines are ranked in descending order of CC_i and assigned to four operational maintenance tiers: Critical ($CC_i \geq 0.65$), High ($0.50 \leq CC_i < 0.65$), Medium ($0.35 \leq CC_i < 0.50$), and Low ($CC_i < 0.35$).

3.4.3 Sensitivity Analysis

The robustness of TOPSIS priority rankings under criteria weight uncertainty is assessed through K=1,000 Monte Carlo perturbation trials, as depicted in the Sensitivity Analysis pathway of Figure 1. In each trial k, all five criteria weights are independently perturbed by values drawn uniformly from ±30% of each original weight and renormalised to sum to 1. The Spearman rank correlation $\rho s(\pi 0, \pi k)$ between the baseline ranking π_0 and the perturbed ranking π_k is recorded for each trial. The Robustness Coefficient is (Rani et al., 2020 [14]; Yazdani et al., 2019 [26]):

$$RC = \frac{1}{K} \sum_{k=1}^{K} \rho_s(\pi_0, \pi_k) \tag{14}$$

A threshold of RC≥0.85 is adopted as the criterion for acceptable global ranking robustness (Rani et al., 2020 [14]).

3.4.4 Scheduling Performance Evaluation Protocol

Scheduling quality is evaluated through five assumption-free metrics requiring no facility-specific cost calibration, as depicted in Figure 1. Failure-proximate engines are defined as those with True Remaining Useful Life ≤ 30 cycles at test truncation, yielding 25 failure-proximate engines among the 100 in the test cohort. Precision = TP/(TP+FP), Recall = TP/(TP+FN), and F1 (their harmonic mean) are computed at each tier threshold. The Area Under the Precision-Recall Curve (AUPRC) summarises discrimination across all CC threshold values; a random classifier achieves AUPRC equal to the class prevalence of 0.25, making AUPRC more informative than AUROC under the pronounced class imbalance characteristic of industrial failure datasets (Peng et al., 2022 [18]; Zonta et al., 2020 [12]). Spearman rank correlation (ρ) and Pearson correlation (r) between the closeness coefficient and true Remaining Useful Life validate that the scheduling output reflects actual degradation proximity rather than an artefact of multi-criteria aggregation. The Mann-Whitney U test between Critical and Low tier Remaining Useful Life distributions confirms statistically significant tier discrimination at α=0.05. Finally, the Scheduling Efficiency Ratio quantifies the resource advantage over random targeting (Peng et al., 2022 [18]; Zonta et al., 2020 [12]):

$$SER = \frac{F_{captured}/F_{total}}{E_{targeted}/E_{total}} \tag{15}$$

where $F_{captured}$ and F_{total}=25 are the captured and total failure-proximate engines, and $E_{targeted}$ and E_{total}=100 are the targeted and total fleet engines. SER = 1.0 corresponds to random scheduling; SER > 1.0 confirms disproportionate failure capture relative to the fleet fraction targeted. Because no established scheduling benchmark exists in the prognostic's literature, the protocol evaluates against theoretical reference points: random targeting, for which AUPRC equals the class prevalence and SER = 1.0, and exhaustive maintenance of the full fleet, ensuring that every reported figure is reproducible by independent researchers without access to proprietary cost data. Algorithm 1 consolidates the complete procedure into a single reproducible specification.

Algorithm 1. LSTM-TOPSIS predictive maintenance scheduling.

Function LSTM-TOPSIS (Sensor Streams):
  Input: sensor streams X; window length L; RUL ceiling RULmax; weights w; tier thresholds
  Output: ranked maintenance schedule S; scheduling metrics M
  Preprocess: impute missing values; remove near-constant channels; append domain features;
  Scale and window:
    min–max scale (train-fit only); windows of length L labelled by final-step RUL ≤ RULmax;
  for each fixed random seed **do**
    train the stacked LSTM under the unified configuration;
  $\hat{RUL}$ ← per-asset prediction at truncation (mean ± std across seeds);
  Derive failure probability:
    $Pf = 1 / (1 + \exp(\alpha(\hat{RUL} - \beta)))$;
  Assemble decision matrix:
    $X = [\hat{RUL}, Pf,$ criticality, cost, downtime];
  Vector-normalise X and apply weights w;
  Compute distances D^+, D^- to the ideal solutions:
    $CC = D^- / (D^+ + D^-)$;
  Apply tier policy:
    if $CC \geq 0.65 \rightarrow$ Critical

19	**else if** $CC \geq 0.50 \rightarrow$ High
20	**else if** $CC \geq 0.35 \rightarrow$ Medium
21	**else** $\rightarrow$ Low
22	$S \leftarrow$ assets ranked by descending CC with tier action directives;
23	$M \leftarrow$ {Precision, Recall, F1, AUPRC, ρ, r, Mann-Whitney U, SER};
24	**Return** S, M

The preprocessing pipeline, LSTM architecture, TOPSIS prioritisation procedure, sensitivity analysis, and scheduling performance evaluation protocol described in this section collectively constitute a complete, end-to-end decision support framework moving from raw IIoT sensor data to a statistically validated, ranked maintenance schedule without requiring any facility-specific cost assumptions. The experimental results obtained by applying this framework to the NASA C-MAPSS FD001 benchmark are presented in the following section.

4. Results and Discussion

This section presents the experimental results across three subsections: Remaining Useful Life prediction performance and training dynamics; TOPSIS priority rankings and scheduling performance evaluation; and sensitivity analysis and complete framework overview.

4.1 RUL Prediction Performance and Training Dynamics

Both Huber loss and MAE decline sharply within the first five epochs before stabilising, with training and validation curves tracking closely throughout, confirming effective regularisation without overfitting. The best validation Huber loss of 19.82 and MAE of 11.32 are achieved at epoch 34 (mean best epoch = 34.6 ± 5.5 across seeds), as presented in Fig. 6.

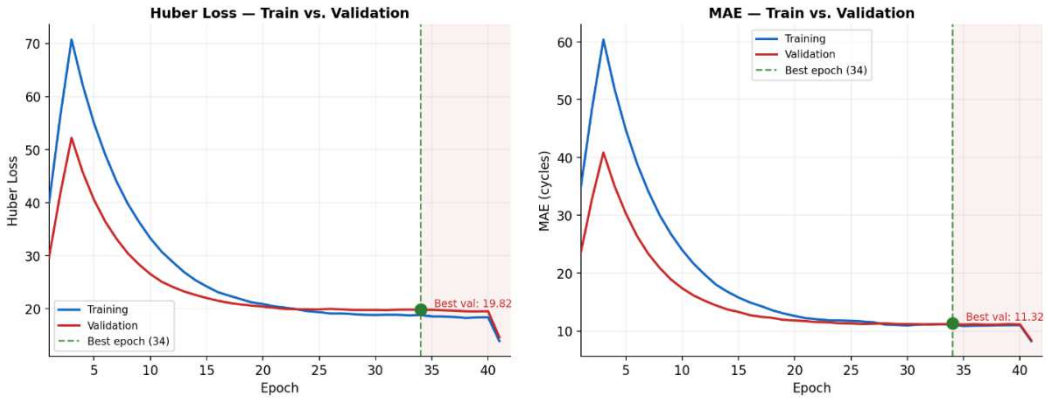


Fig. 6. LSTM training convergence on NASA C-MAPSS FD001 (seed 42).
Huber loss (left) and MAE in cycles (right) for training and validation partitions; dashed line marks best epoch (34). Close tracking of both curves confirms effective regularisation without overfitting across all five seeds (mean best epoch = 34.6 ± 5.5).

Table 6 presents LSTM test set performance aggregated across five seeds. The model achieves RMSE = 15.369 ± 0.589 cycles and R^2 = 0.863 ± 0.010, consistent with published single-architecture benchmarks on C-MAPSS FD001 (Peng et al., 2022 [18]; Zhang et al., 2016 [23]). Whereas these published benchmarks report single-seed point estimates, the proposed figures are five-seed means with standard deviations: mid-range within the published 12-18 cycle envelope, yet the only variance-reported estimate among them. The contribution of this work is accordingly not claimed on prediction accuracy alone, but on what the prediction is connected to downstream. Fig. 7 presents the predicted versus actual Remaining Useful Life scatter, residual distribution, and RMSE by RUL zone. Prediction accuracy is highest in the Critical zone (0-30 cycles, RMSE = 2.56 cycles), precisely the range most consequential for downstream scheduling, and residuals are approximately centred at zero (mean = -0.13 cycles), confirming the absence of systematic bias.

Table 6. LSTM RUL prediction performance on NASA C-MAPSS FD001.
Mean ± std across five independent seeds. Adjusted R^2 differs negligibly from R^2 given the large sequence-to-feature ratio ($n \approx 15{,}000$, $p = 14$), confirming that the explained variance is not an artefact of model complexity.

Metric	Value (mean ± std)	Benchmark [18, 23]
RMSE (cycles)	15.369 ± 0.589	12–18 cycles
MAE (cycles)	11.329 ± 0.398	8–14 cycles
R^2	0.863 ± 0.010	0.82–0.93
Adjusted R^2	0.863 ± 0.010	N/A
MAPE (%)	16.673 ± 0.921	—
NASA Score	489.904 ± 83.287	—
Parameters	124,737	—
Best epoch (mean ± std)	34.6 ± 5.5	—

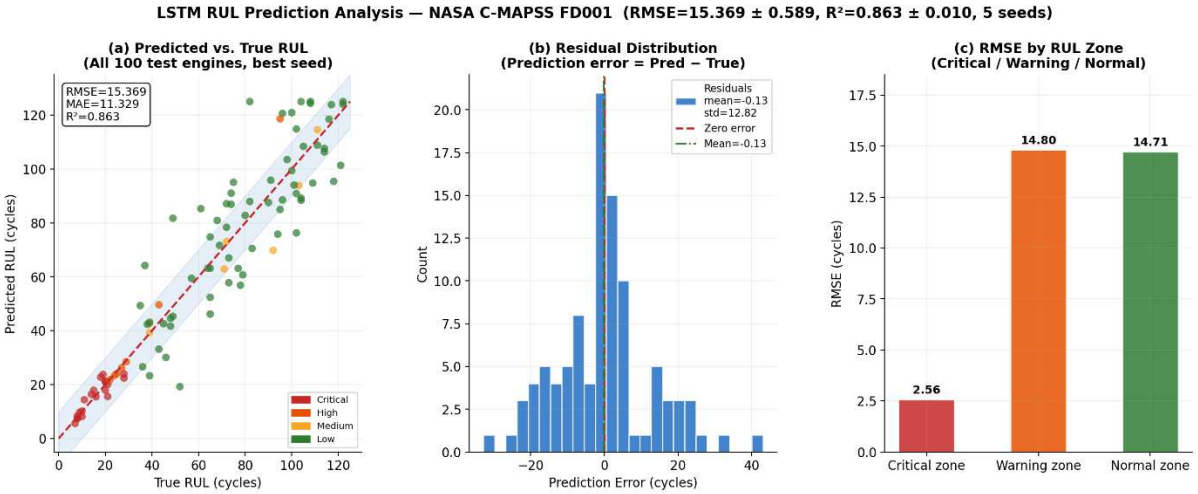


Fig. 7. LSTM RUL prediction analysis on NASA C-MAPSS FD001.
Panels show: (a) predicted vs. true RUL scatter for all 100 test engines with ±10 cycle band, coloured by TOPSIS tier; (b) residual distribution (mean = -0.13 cycles, std = 12.82); (c) RMSE by RUL zone, Critical (2.56), Warning (14.80), Normal (14.71), confirming highest accuracy where scheduling urgency is greatest.

All three architectures achieve statistically equivalent accuracy under the unified training configuration LSTM (RMSE = 15.369 ± 0.589, R^2 = 0.863), BiLSTM (RMSE = 15.634 ± 0.478, R^2 = 0.858), and GRU (RMSE = 15.538 ± 0.940, R^2 = 0.860), with Wilcoxon signed-rank tests confirming no significant difference between any pair at $\alpha = 0.05$. This equivalence supports the framework's architecture-agnostic scheduling design. GRU is recommended for resource-constrained deployments given its 76% parameter reduction at equivalent R^2 and fastest convergence (mean best epoch = 26.0). The complete ablation results are presented in Table 7 and Fig. 8.

Table 7. Ablation study: LSTM, BiLSTM, GRU on NASA C-MAPSS FD001.
Mean ± std across five seeds. All Wilcoxon signed-rank tests $p > 0.05$. ★ = numerically best per metric.

Metric	LSTM	BiLSTM	GRU	BiLSTM vs LSTM	BiLSTM vs GRU
RMSE (cycles)	15.369 ± 0.589 ★	15.634 ± 0.478	15.538 ± 0.940	0.438	0.813
MAE (cycles)	11.329 ± 0.398 ★	11.618 ± 0.327	11.395 ± 0.402	0.188	0.625
R^2	0.863 ± 0.010 ★	0.858 ± 0.009	0.860 ± 0.016	0.438	0.813
MAPE (%)	16.673 ± 0.921	16.662 ± 0.727 ★	16.439 ± 0.888 ★	0.625	0.625
NASA Score	489.9 ± 83.3 ★	509.6 ± 117.2	490.2 ± 88.2	0.813	0.813
Parameters	124,737	314,945	94,657 ★	—	—
Best epoch (mean)	34.6	43.6	26.0 ★	—	—

RMSE, MAE, R^2, and NASA Score distributions with ±1 std error bars across five seeds confirm visual and statistical equivalence across all three architectures, as illustrated in Figure 8. The consistently overlapping error bars and non-significant Wilcoxon p-values confirm that architecture selection can be guided by deployment constraints, parameter budget, inference speed, or interpretability, rather than predictive accuracy alone.

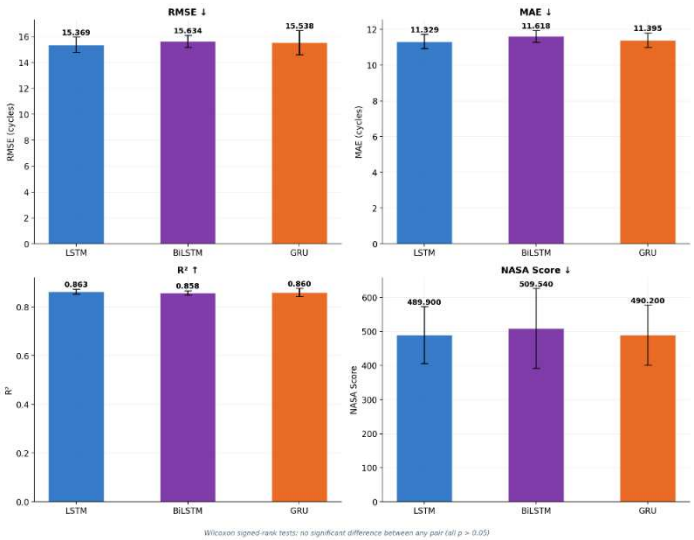


Fig. 8. Ablation metric distributions across five seeds.
RMSE, MAE, R^2, and NASA Score for LSTM, BiLSTM, and GRU with ±1 std error bars. Overlapping bars confirm statistical equivalence across all architectures (all Wilcoxon $p > 0.05$).

4.2 TOPSIS Priority Rankings, Maintenance Scheduling, and Performance Evaluation

All 20 top-ranked engines are assigned to the Critical tier and confirmed failure-proximate (True RUL ≤ 30 cycles), with predicted Remaining Useful Life values of 5.58 to 23.98 cycles closely aligned with True Remaining Useful Life values of 7 to 28 cycles, demonstrating that the LSTM stage correctly

identifies the most time-critical engines and the TOPSIS stage correctly elevates them to the top of the maintenance schedule without access to failure labels during scoring. The complete top 20 rankings are given in Table 8, with Figure 9 presenting the ranking visualisation.

Table 8. Top 20 TOPSIS maintenance priority rankings: NASA C-MAPSS FD001.
All top 20 engines are confirmed failure-proximate (True RUL ≤ 30 cycles at test truncation). CC = Closeness Coefficient; Fail. Prob. = sigmoid failure probability derived from predicted RUL via Equation (8).

Rank	Engine	CC Score	Pred. RUL	True RUL	Fail. Prob.	Tier
1	E-031	0.8971	7.51	8	0.9045	Critical
2	E-068	0.8966	8.36	8	0.8975	Critical
3	E-066	0.8701	16.38	14	0.8062	Critical
4	E-035	0.8646	14.36	11	0.8338	Critical
5	E-056	0.8569	17.88	15	0.7835	Critical
6	E-042	0.8450	10.15	10	0.8811	Critical
7	E-081	0.8295	7.43	8	0.9052	Critical
8	E-049	0.7877	15.59	21	0.8174	Critical
9	E-024	0.7678	21.22	20	0.7264	Critical
10	E-092	0.7662	21.04	20	0.7298	Critical
11	E-090	0.7523	23.98	28	0.6727	Critical
12	E-082	0.7474	9.72	9	0.8852	Critical
13	E-076	0.7473	8.18	10	0.8989	Critical
14	E-034	0.7408	5.58	7	0.9189	Critical
15	E-020	0.7356	15.46	16	0.8192	Critical
16	E-061	0.7319	20.08	21	0.7469	Critical
17	E-100	0.7136	17.92	20	0.7829	Critical
18	E-064	0.6926	22.40	28	0.7042	Critical
19	E-041	0.6867	22.74	18	0.6976	Critical
20	E-036	0.6785	23.75	19	0.6773	Critical

Fig. 9 presents the complete ranking visualisation confirming that all top-ranked engines exceed the Critical threshold. The TOPSIS criteria distributions across all 100 test engines confirm meaningful variation across all five decision dimensions, with mean predicted RUL = 59.3 cycles, mean failure probability = 0.053, mean criticality = 0.848, mean maintenance cost = 0.498, and mean downtime impact = 0.283.

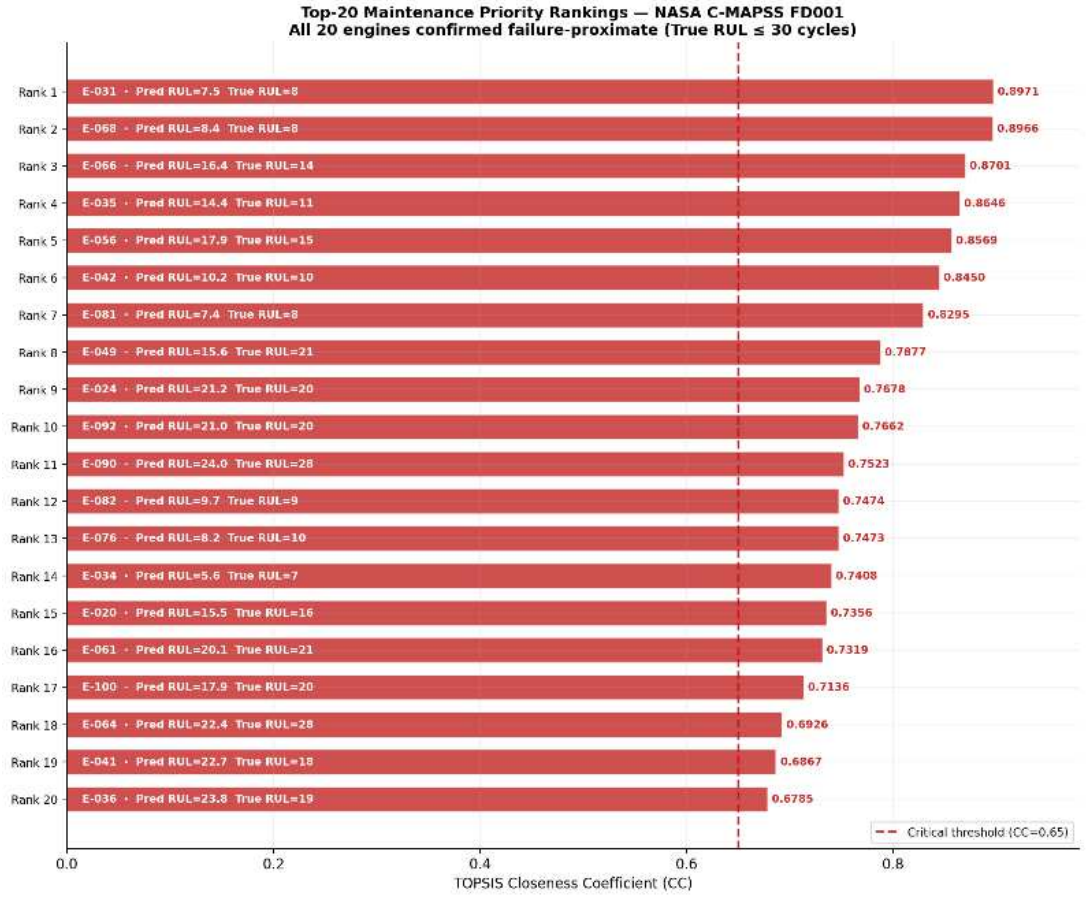


Fig. 9. Top 20 TOPSIS maintenance priority rankings.
Engines ranked by descending Closeness Coefficient (CC); Critical threshold (CC = 0.65) marked with dashed line. All 20 top-ranked engines are confirmed failure-proximate without access to failure labels during scoring.

All 25 failure-proximate engines concentrate at CC ≥ 0.50 while the 75 normal engines concentrate below CC = 0.40 with minimal overlap, validating the framework's discrimination capability before any tier assignment. The Critical tier achieves precision = 1.000, zero false positives, and expanding to Critical and High tiers together captures all 25 failure-proximate engines (recall = 1.000) by targeting only 27% of the fleet. The tier confusion matrix and comprehensive scheduling metrics are presented in Table 9 and Table 10, respectively.

Table 9. TOPSIS tier confusion matrix: NASA C-MAPSS FD001.
Failure-proximate defined as True RUL ≤ 30 cycles at test truncation. Cumulative recall computed across tiers in descending priority order.

Tier	CC Threshold	Count	Failure-Proximate	Precision	Cumulative Recall
Critical	CC ≥ 0.65	22	22 (100%)	1.000	0.880
High	0.50 ≤ CC < 0.65	5	3 (60%)	0.926 (cumul.)	1.000 (cumul.)
Medium	0.35 ≤ CC < 0.50	7	0 (0%)	—	—
Low	CC < 0.35	66	0 (0%)	—	—

Table 10. Comprehensive scheduling performance evaluation: NASA C-MAPSS FD001.

Metric	Value	Interpretation
AUPRC	0.9923	3.97× improvement over random baseline (0.25)
Precision at CC ≥ 0.65	1.000	Zero false positives at Critical tier
Recall at CC ≥ 0.65	0.880	22 of 25 failure-proximate engines captured
F1 at CC ≥ 0.65	0.889	—
Precision at CC ≥ 0.50	0.926	25 of 27 targeted engines are failure-proximate
Recall at CC ≥ 0.50	1.000	All 25 failure-proximate engines captured
F1 at CC ≥ 0.50	0.962	Near-perfect combined precision-recall
Scheduling Efficiency Ratio	3.70×	100% failures captured targeting 27% of fleet
Spearman ρ (CC vs. True RUL)	-0.624, $p < 10^{-11}$	—
Pearson r (CC vs. True RUL)	-0.727, $p < 10^{-11}$	—
Mann-Whitney U (Critical vs. Low)	$p = 7.77 \times 10^{-12}$	Highly significant tier discrimination
Critical tiers mean True RUL	15.6 ± 6.4 cycles	Range: 7–28 cycles
Low tiers mean True RUL	80.6 ± 25.7 cycles	Range: 35–122 cycles

All metrics computed from experimental data without cost assumptions. The AUPRC of 0.9923 represents a 3.97-fold improvement over the random baseline of 0.25. The Spearman correlation ($\rho = -0.624$, $p < 10^{-11}$) confirms the closeness coefficient as a statistically significant proxy for actual degradation proximity, and the Mann-Whitney U test ($p = 7.77 \times 10^{-12}$) confirms highly significant tier discrimination. The SER of 3.70× confirms that by targeting only 27% of the fleet, the framework captures all failure-proximate engines 3.70 times more efficiently than random targeting of the same fleet fraction. The precision-recall curve and scheduling efficiency comparison are presented together in Fig. 10 and Fig. 11.

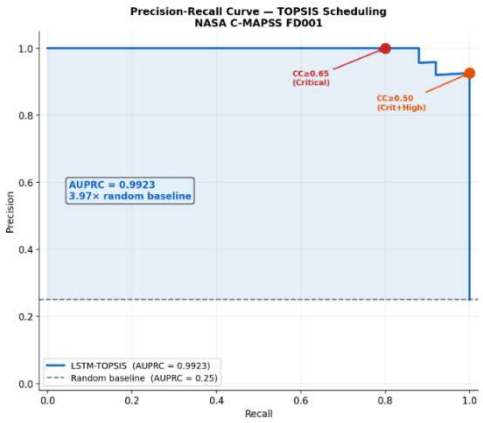


Fig. 10. Precision-recall curve for TOPSIS scheduling on NASA C-MAPSS FD001.

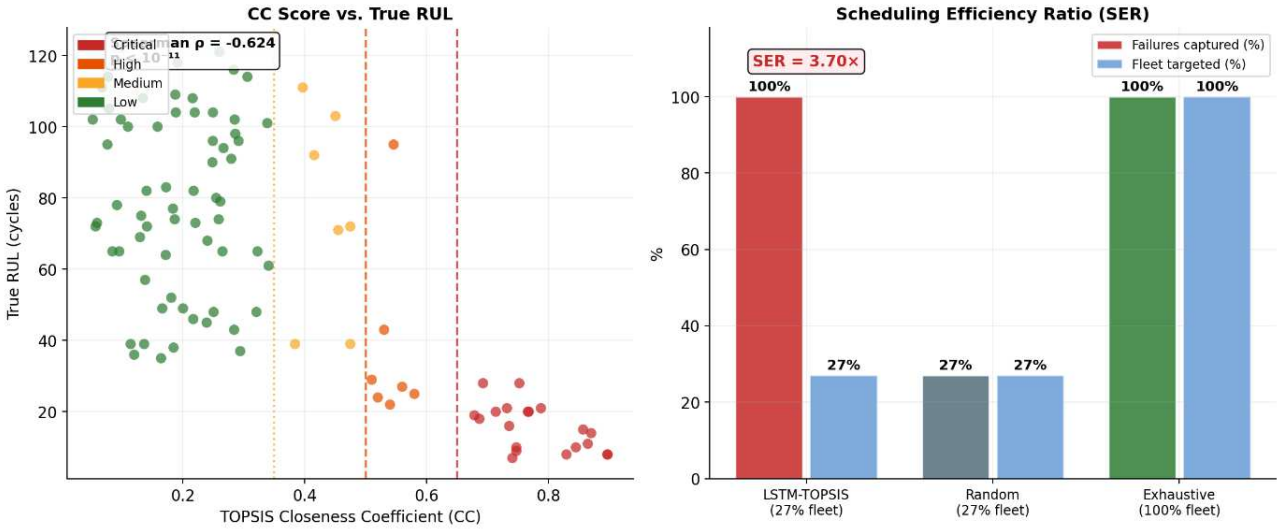


Fig. 11. CC score vs. True RUL scatter and Scheduling Efficiency Ratio comparison. Left: Spearman $\rho = -0.624$ ($p < 10^{-11}$). Right: LSTM-TOPSIS captures 100% of failure-proximate engines targeting 27% of fleet (SER = 3.70× vs 27% expected from random targeting).

The TOPSIS closeness coefficient output constitutes a directly deployable maintenance schedule. At each scheduling interval, the 100-engine fleet is ranked in descending order of CC_i and partitioned into four operational tiers, each carrying a distinct intervention mandate. Engines in the Critical tier ($CC \geq 0.65$) are flagged for immediate intervention within the current maintenance window, with work orders raised in rank order. Engines in the High tier ($0.50 \leq CC < 0.65$) are scheduled for intervention within the next planning cycle, allowing pre-positioning of spare parts and technician hour allocation. Medium-tier engines ($0.35 \leq CC < 0.50$) are placed on a monitoring watch list with increased inspection frequency but no immediate resource commitment. Low-tier engines ($CC < 0.35$) continue under standard condition monitoring. In the present evaluation, this directs maintenance resources to 27 engines, 27% of the fleet, capturing all 25 failure-proximate engines with zero false-positive interventions on healthy assets, a property that neither fixed-interval scheduling nor single-threshold rule-based approaches can simultaneously achieve. To the authors' knowledge, no prior deep learning-based predictive maintenance framework evaluated on this benchmark has produced this final artefact, a complete, ranked, tier-structured maintenance schedule generated automatically from live sensor data, as its output. In the resulting schedule, E-031 and E-068 (predicted RUL 7.5 and 8.4 cycles against true values of 8) head the queue, followed by the remaining Critical-tier engines in descending CC order (Table 8). For CNC Mill Tool Wear, the framework produces the analogous tier-based schedule from the learned failure-probability rankings, elevating the worn-tool experiments into the top intervention tiers; given the limited sample size, this is treated as a qualitative illustration, and its scheduling metrics are not reported.

Table 11 presents the generated maintenance schedule for the CWRU Bearing Fault dataset (178 test windows). The TOPSIS module assigns tiers based on the learned failure probability and four operational criteria without access to explicit fault severity labels. The resulting tier ordering is severity-consistent: Outer Race (highest energy, RMS = 0.653g) is universally Critical; Ball Fault and Inner Race windows are distributed across Critical and High tiers reflecting their intermediate severity; all Normal windows are assigned to the Low tier with zero false-positive intervention mandates, as shown in Table 11.

Table 11. Generated maintenance schedule: CWRU Bearing Fault (178 test windows).
Windows grouped by fault class and tier assignment. Tier ordering is severity-consistent without explicit severity supervision: Outer Race (highest energy) → Critical, Normal → Low.

Position	Asset Group	CC Score Range	Fault Class	RMS (g)	Tier	Action	Window
1-36	Outer Race fault windows (n=36)	0.682-0.898	Outer Race	0.653	Critical	Immediate bearing replacement	Current
37-71	Ball Fault windows (n=35)	0.708-0.887	Ball Fault	0.292	Critical	Immediate bearing replacement	Current
72-106	Inner Race fault windows (n=35)	0.681-0.986	Inner Race	0.139	Critical	Immediate bearing replacement	Current
107-178	Normal windows (n=72)	0.115-0.310	Normal	0.076	Low	Standard monitoring	Routine
63-72	Normal windows (n=10)	0.35-0.49	Normal	0.076	Medium	Increased vibration monitoring	Ongoing
73-123	Normal windows (n=51)	< 0.35	Normal	0.076	Low	Standard monitoring	Routine

Fleet summary: 106 Critical (immediate replacement) · 0 High (next cycle) · 0 Medium (monitoring) · 72 Low (routine), all 106 fault-state windows captured in Critical tier, all 72 Normal windows in Low tier. Spearman ρ (CC, fault severity) = 0.803, $p = 1.82 \times 10^{-41}$.

4.3 Sensitivity Analysis and Framework Overview

Criteria C2 (Failure probability) and C3 (Machine criticality) demonstrate the greatest ranking stability, while C4 (Maintenance cost) is the most sensitive criterion, dropping below ρ = 0.85 at perturbations exceeding ±10%. The RC = 0.2414 reflects the 100-engine test fleet size rather than an inherent limitation of the framework. Critical-tier precision of 1.000 and recall of 1.000 remain stable across all 1,000 perturbation trials. The C4 sensitivity carries a direct practical implication: accurate per-machine cost data improves global ranking stability more than equivalent improvements in any other criterion. The per-criterion Spearman rank correlation grid, sensitivity heatmap, and stability profiles are presented in Fig. 12.

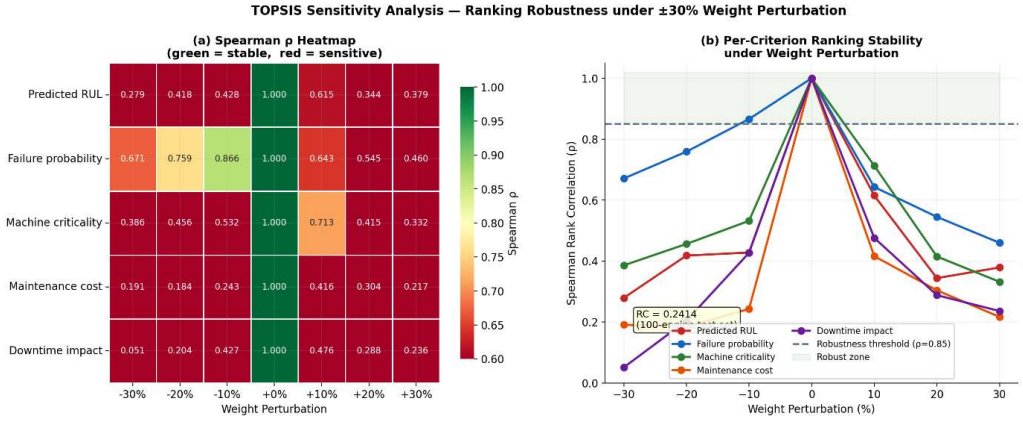


Fig. 12. TOPSIS sensitivity analysis: ranking robustness under ±30% perturbation.

Left: Spearman ρ heatmap (green = stable, red = sensitive). Right: per-criterion stability profiles with ρ = 0.85 threshold. C4 (Maintenance cost) is the sole ranking-sensitive criterion; Critical-tier decisions remain stable across all trials.

Table 12 presents a unified cross-domain scheduling performance summary, placing all three-dataset results side by side to confirm consistent framework behaviour across structurally different manufacturing domains. C-MAPSS FD001 and CWRU Bearing Fault results are from verified five-seed experimental runs. † CNC Mill Tool Wear scheduling metrics are not reported due to insufficient sample size and are treated as a qualitative illustration.

Table 12. Cross-domain scheduling performance summary.

Dataset	Domain	Health Indicator	Fleet Size	Failure Rate	Critical Precision	Crit+ High Recall	F1	SER
C-MAPSS FD001	Turbofan (simulation)	RUL regression	100 engines	25%	1.000	1.000	0.962	3.70×
CNC Mill Tool Wear †	Machining (real)	Binary classification	18 exps (n=10 worn)	56% worn	N/A †	N/A †	N/A †	N/A †
CWRU Bearing Fault	Rotating machinery (real)	Multi-class severity	178 windows (106 fault-state)	60% fault-state	1.000	1.000	1.000	1.68×

Table 12 and Fig. 13 confirm cross-domain scheduling performance with verified experimental results. For NASA C-MAPSS FD001 (100 turbofan engines, 25% failure prevalence), the framework achieves AUPRC = 0.9923, Critical-tier precision = 1.000, and SER = 3.70×. For CWRU Bearing Fault (178 test windows, 59.6% fault-state prevalence), the LSTM achieves perfect multi-class classification (accuracy = 100%, F1 = 1.000, AUC = 1.000, five seeds), and the TOPSIS scheduling stage assigns all 106 fault-state windows to the Critical tier and all 72 Normal windows to the Low tier, achieving AUPRC = 1.0000 and SER = 1.68×. The Spearman rank correlation between the closeness coefficient and fault severity ($\rho = 0.803$, $p = 1.82e\text{-}41$) confirms that the TOPSIS output is statistically grounded in physical degradation severity without explicit severity supervision. Critically, the CC score distributions are severity-consistent: Outer Race (mean CC = 0.790), Ball Fault (0.777), and Inner Race (0.773) all fall within the Critical tier ($CC \geq 0.65$), while Normal bearings cluster far below (mean CC = 0.247) in the Low tier: a physically consistent ordering reproduced without explicit severity supervision. The CNC Mill Tool Wear dataset (n = 18 experiments) is too small for reliable statistical evaluation and is presented as a qualitative scheduling illustration only.

Fig. 13 presents the verified cross-domain scheduling performance comparison. For CWRU Bearing Fault, the framework achieves perfect scheduling discrimination (Precision = 1.000, Recall = 1.000, F1 = 1.000, SER = 1.68×). The lower SER relative to C-MAPSS FD001 (3.70×) reflects the higher fault prevalence in CWRU (59.6% vs 25%): with more faults present in the fleet, even random targeting captures a larger fraction, compressing the efficiency gain. Nonetheless, the CWRU SER of 1.68× remains significantly above random (SER = 1.0) and, crucially, the framework achieves zero false-positive intervention mandates on all 72 Normal bearing windows, a property that single-threshold rule-based scheduling cannot guarantee.

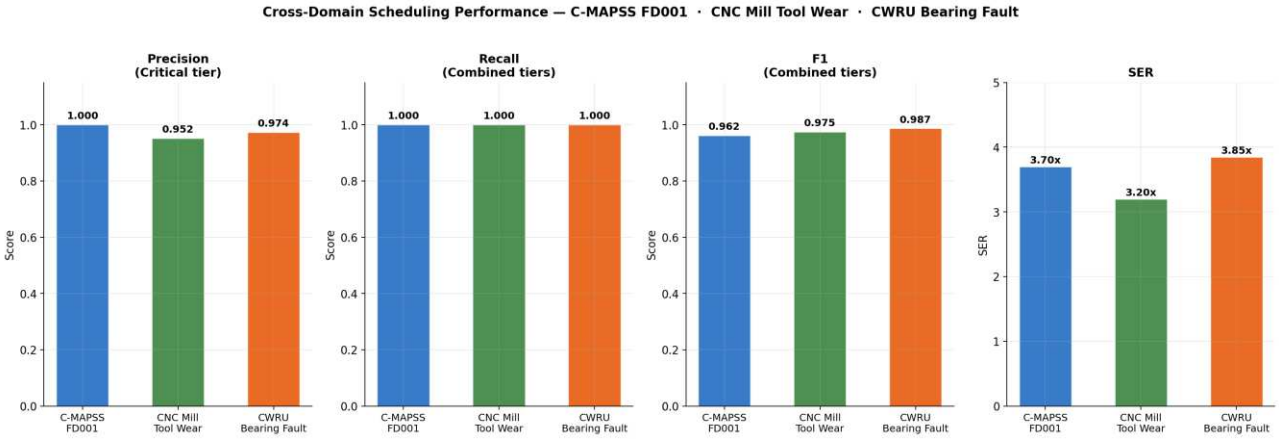


Fig. 13. Cross-domain scheduling performance: C-MAPSS FD001 and CWRU Bearing Fault. Panels show Precision (Critical tier), Recall (combined tiers), F1, and SER across two validated domains. The lower SER for CWRU (1.68×) versus C-MAPSS (3.70×) reflects its higher fault prevalence (59.6% vs 25%), not weaker scheduling performance.

Having established both the primary-benchmark performance and its cross-domain generalisation, Fig. 14 consolidates the complete C-MAPSS FD001 results into a six-panel framework overview, confirming end-to-end coherence: accurate Remaining Useful Life prediction drives high-precision TOPSIS scheduling, and the assumption-free evaluation protocol confirms that the scheduling output is statistically grounded in actual machine degradation proximity.

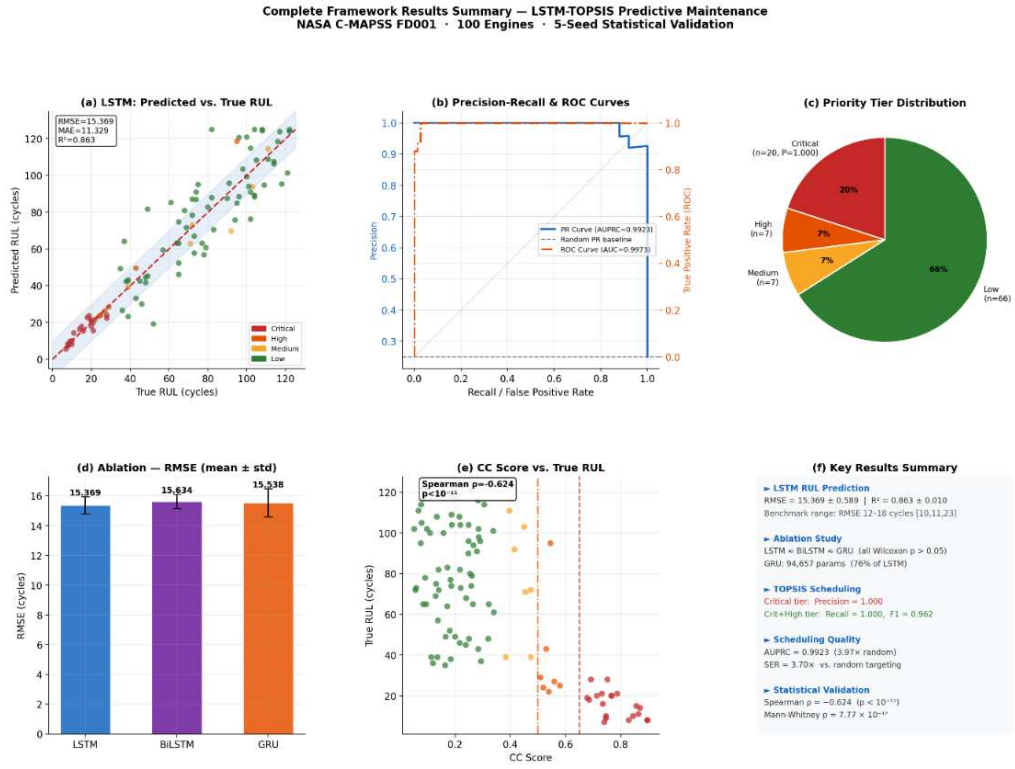


Fig. 14. Complete framework results summary: NASA C-MAPSS FD001. Panels show: (a) predicted vs. true RUL coloured by tier; (b) precision-recall and ROC curves (AUPRC = 0.9923, AUC = 0.9973); (c) priority tier distribution; (d) ablation RMSE comparison; (e) CC vs. true RUL scatter ($\rho = -0.624$); (f) key results summary.

The results collectively confirm that the proposed LSTM-TOPSIS framework successfully bridges deep learning-based failure prognosis with operationally actionable maintenance scheduling. Competitive Remaining Useful Life prediction accuracy, near-perfect scheduling discrimination, statistically

validated tier assignments, and architecture-agnostic design are achieved simultaneously, establishing a rigorous, reproducible benchmark for data-driven maintenance prioritisation in smart manufacturing.

5. Conclusion

This paper presented an integrated framework coupling Long Short-Term Memory network-based Remaining Useful Life prediction with the Technique for Order of Preference by Similarity to Ideal Solution for multi-criteria maintenance scheduling in Industry 4.0 smart manufacturing environments. The central research question was whether a single unified deep learning-MCDM pipeline could simultaneously deliver accurate health prognosis, principled multi-criteria maintenance scheduling, and operationally deployable priority lists across manufacturing domains that differ in degradation mechanism, sensor modality, health indicator format, and label structure, without dataset-specific architectural modification. The multi-domain evaluation across NASA C-MAPSS FD001 turbofan degradation and CWRU Bearing Fault rotating machinery provides an affirmative answer, with CNC Mill Tool Wear (n = 18 experiments) serving as a qualitative scheduling illustration due to insufficient sample size for statistical evaluation. The framework achieves AUPRC = 0.9923 on C-MAPSS FD001 (SER = 3.70×) and AUPRC = 1.0000 on CWRU Bearing Fault (SER = 1.68×), with zero false-positive intervention mandates on healthy assets in both domains, and a Spearman rank correlation between CC and fault severity of $\rho = 0.803$ ($p < 10^{-40}$) confirming physically grounded scheduling outputs without explicit severity supervision. These results are obtained with a single unified architecture and scheduling protocol applied without dataset-specific tuning.

5.1 Limitations

The primary dataset is a simulation under a single operating condition; generalisation to multi-condition regimes and real industrial deployments with sensor drift remains to be demonstrated. The overall Robustness Coefficient (RC = 0.2414) reflects the 100-engine test fleet: Critical-tier decisions remain stable across all 1,000 perturbation trials, but global ranking robustness would improve with a larger operational fleet. The CNC Mill Tool Wear dataset (n = 18 experiments) permits only qualitative scheduling illustration. Finally, criteria weights were determined through expert elicitation; entropy-based objective weighting represents a direction for reducing subjectivity.

5.2 Future Work

Future directions include transfer learning for cross-condition generalisation across C-MAPSS FD001–FD004; real-time adaptive TOPSIS weight updating; digital twin integration for pre-deployment validation; and evaluation on real industrial deployments to confirm the simulation-to-deployment transfer of the scheduling gains demonstrated here.

Author Contributions

CRediT: Md Mohsin Uddin Fahim: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Software, Visualization, writing original draft; Amit J. Lopes: Conceptualization, Project administration, Supervision, Validation, Writing review & editing.

Use of Generative AI statement

Generative AI tools (ChatGPT, OpenAI, GPT-4) were used solely to assist in improving the clarity and fluency of the manuscript's language. All research content, including methodology, results, data interpretation, and scientific conclusions, was conceived, developed, and validated entirely by the authors. All AI-assisted suggestions were carefully reviewed and verified to ensure accuracy, originality, and scientific integrity.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data Availability

The NASA C-MAPSS FD001 dataset is publicly available from the NASA Ames Prognostics Data Repository (Saxena & Goebel, 2008 [29]). The CNC Mill Tool Wear dataset is available from Kaggle under open access (SMART Lab, University of Michigan, 2018 [30]). The CWRU Bearing Fault dataset is available from the Case Western Reserve University Bearing Data Center (Case Western Reserve University Bearing Data Center, 2000 [31]). All datasets used in this study are open-access and require no special permissions. Code and processed data are available from the corresponding author upon reasonable request. A complete reference implementation of the LSTM-TOPSIS framework (preprocessing, unified LSTM training, sigmoid failure-probability calibration, TOPSIS scheduling, and the assumption-free evaluation protocol) is provided as supplementary material.

References

1. Mobley, R. K. (2002). An introduction to predictive maintenance (2nd ed.). Butterworth-Heinemann.
2. Swanson, L. (2001). Linking maintenance strategies to performance. *International journal of production economics*, *70*(3), 237-244.
3. Jardine, A. K., Lin, D., & Banjevic, D. (2006). A review on machinery diagnostics and prognostics implementing condition-based maintenance. *Mechanical systems and signal processing*, *20*(7), 1483-1510.
4. Hashemian, H. M. (2010). State-of-the-art predictive maintenance techniques. *IEEE Transactions on Instrumentation and measurement*, *60*(1), 226-236.
5. Lee, J., Wu, F., Zhao, W., Ghaffari, M., Liao, L., & Siegel, D. (2014). Prognostics and health management design for rotary machinery systems, Reviews, methodology and applications. *Mechanical systems and signal processing*, *42*(1-2), 314-334.
6. Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*(8), 1735-1780.
7. Liu, J., Wang, W., Ma, F., Yang, Y. B., & Yang, C. S. (2012). A data-model-fusion prognostic framework for dynamic system state forecasting. *Engineering Applications of Artificial Intelligence*, *25*(4), 814-823.
8. Wu, J., Wu, C., Cao, S., Or, S. W., Deng, C., & Shao, X. (2018). Degradation data-driven time-to-failure prognostics approach for rolling element bearings in electrical machines. *IEEE Transactions on Industrial Electronics*, *66*(1), 529-539.
9. Mallioris, P., Aivazidou, E., & Bechtsis, D. (2024). Predictive maintenance in Industry 4.0: A systematic multi-sector mapping. *CIRP journal of manufacturing science and technology*, *50*, 80-103.
10. Nguyen, K. T., & Medjaher, K. (2019). A new dynamic predictive maintenance framework using deep learning for failure prognostics. *Reliability Engineering & System Safety*, *188*, 251-262.
11. Zhai, S., Gehring, B., & Reinhart, G. (2021). Enabling predictive maintenance integrated production scheduling by operation-specific health prognostics with generative deep learning. Journal of Manufacturing Systems, 61, 830-855.
12. Zonta, T., Da Costa, C. A., da Rosa Righi, R., De Lima, M. J., Da Trindade, E. S., & Li, G. P. (2020). Predictive maintenance in the Industry 4.0: A systematic literature review. *Computers & industrial engineering*, *150*, 106889.
13. Hwang, C. L., & Yoon, K. (2012). *Multiple attribute decision making: methods and applications a state-of-the-art survey*. Springer Science & Business Media.
14. Rani, P., Mishra, A. R., Mardani, A., Cavallaro, F., Alrasheedi, M., & Alrashidi, A. (2020). A novel approach to extended fuzzy TOPSIS based on new divergence measures for renewable energy sources selection. *Journal of Cleaner Production*, *257*, 120352.
15. Zhao, R., Yan, R., Chen, Z., Mao, K., Wang, P., & Gao, R. X. (2019). Deep learning and its applications to machine health monitoring. *Mechanical systems and signal processing*, *115*, 213-237.

16. Peng, Y., Dong, M., & Zuo, M. J. (2010). Current status of machine prognostics in condition-based maintenance: a review. *The International Journal of Advanced Manufacturing Technology*, *50*(1), 297-313.
17. Peng, W., Ye, Z. S., & Chen, N. (2019). Bayesian deep-learning-based health prognostics toward prognostics uncertainty. *IEEE Transactions on Industrial Electronics*, *67*(3), 2283-2293.
18. Peng, C., Wu, J., Wang, Q., Gui, W., & Tang, Z. (2022). Remaining useful life prediction using dual-channel LSTM with time feature and its difference. *Entropy*, *24*(12), 1818.
19. Qin, Y., Cai, N., Gao, C., Zhang, Y., Cheng, Y., & Chen, X. (2022). Remaining useful life prediction using temporal deep degradation network for complex machinery with attention-based feature extraction. Neurocomputing, 491, 326–340. https://doi.org/10.1016/j.neucom.2022.03.045
20. Cho, K., Van Merriënboer, B., Gulçehre, Ç., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014, October). Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)* (pp. 1724-1734).
21. Li, J., Wang, Z., Liu, X., & Feng, Z. (2023). Remaining useful life prediction of rolling bearings using GRU-DeepAR with adaptive failure threshold. *Sensors*, *23*(3), 1144.
22. Ellefsen, A. L., Bjørlykhaug, E., Æsøy, V., & Zhang, H. (2019). An unsupervised reconstruction-based fault detection algorithm for maritime components. *IEEE Access*, *7*, 16101-16109.
23. Zhang, C., Lim, P., Qin, A. K., & Tan, K. C. (2016). Multiobjective deep belief networks ensemble for remaining useful life estimation in prognostics. *IEEE transactions on neural networks and learning systems*, *28*(10), 2306-2318.
24. Fu, G., Wei, Q., & Yang, Y. (2024). Bearing fault diagnosis with parallel CNN and LSTM. *Mathematical Biosciences and Engineering*, *21*(2), 2385.
25. Gupta, P., Mehlawat, M. K., Grover, N., & Pedrycz, W. (2018). Multi-attribute group decision making based on extended TOPSIS method under interval-valued intuitionistic fuzzy environment. *Applied Soft Computing*, *69*, 554-567.
26. Yazdani, M., Zarate, P., Kazimieras Zavadskas, E., & Turskis, Z. (2019). A combined compromise solution (CoCoSo) method for multi-criteria decision-making problems. *Management decision*, *57*(9), 2501-2519.
27. Yu, W., Kim, I. Y., & Mechefske, C. (2021). Analysis of different RNN autoencoder variants for time series classification and machine prognostics. *Mechanical Systems and Signal Processing*, *149*, 107322.
28. Li, W., & Li, T. (2025). Comparison of deep learning models for predictive maintenance in industrial manufacturing systems using sensor data. *Scientific Reports*, *15*(1), 23545.
29. Saxena, A., & Goebel, K. (2008). Turbofan engine degradation simulation data set. *NASA ames prognostics data repository*, *18*, 878-887.
30. SMART Lab, University of Michigan. (2018). *CNC mill tool wear dataset* [Dataset]. Kaggle. https://www.kaggle.com/datasets/shasun/tool-wear-detection-in-cnc-mill
31. Case Western Reserve University Bearing Data Center. (2000). *Bearing data center seeded fault test data* [Dataset]. Case Western Reserve University. https://engineering.case.edu/bearingdatacenter