Journal of Sustainable Development and Policy

Volume: 1; **Issue:** 2
Pages: 35-83

IOT-INTEGRATED DEEP NEURAL PREDICTIVE MAINTENANCE SYSTEM WITH VIBRATION-SIGNAL DIAGNOSTICS IN SMART FACTORIES

S. M. Habibullah[1]; Md. Tahmid Farabe Shehun[2];

[1]. Operations Engineer, Lighthouse Marine Services, Bangladesh.
Email: anikmail12@gmail.com

[2]. BSc. in apparel manufacturing & Technology, BGMEA University of Fashion & Technology, Dhaka, Bangladesh; Email: orkeshshehun678@gmail.com

Doi: 10.63125/6jjq1p95
Received: *18 January 2022;* ***Revised:*** *17 February 2022;* ***Accepted: March 21****, 2022;* ***Published:*** *30 September 2022*

Abstract

This quantitative study examined an IoT-Integrated Deep Neural Predictive Maintenance System with Vibration-Signal Diagnostics in Smart Factories through an empirical evaluation informed by a structured review of 92 peer-reviewed journal articles and conference papers on vibration-based diagnostics, deep learning architectures, and industrial IoT deployment practices. Vibration data were acquired from 24 rotating assets operating across two smart-factory production lines at a sampling rate of 25.6 kHz and were segmented into 2.0-second windows with 50% overlap, producing 1,152,000 diagnostic windows and 38,400 prognostic sequences composed of 30 windows per sequence. The experimental design compared signal representation types (raw 1D, STFT, wavelet), model architecture families (1D CNN, temporal CNN, LSTM/GRU), inference placement (edge versus cloud), and data-quality conditions (baseline, noise, and missingness). Descriptive and inferential analyses showed that time–frequency representations consistently outperformed raw time-domain inputs for fault diagnosis and remaining useful life estimation. Wavelet-based cloud configurations achieved a macro F1 of 0.934 and PR-AUC of 0.953, compared with macro F1 of 0.881 and PR-AUC of 0.904 for raw 1D edge configurations. Prognostic accuracy followed a similar pattern, with wavelet-based cloud pipelines producing RUL MAE of 6.8 hours and RMSE of 9.9 hours, compared with 8.6 hours MAE and 12.4 hours RMSE for raw 1D edge pipelines. Mixed-effects regression confirmed statistically significant effects of representation type on diagnostic performance (wavelet versus raw, $\beta = 0.041$ for macro F1, $p < 0.001$) and significant degradation under missingness conditions ($\beta = -0.034$, $p < 0.001$). System-level analysis showed that cloud placement increased median latency by 102.6 ms, P95 latency by 181.2 ms, and bandwidth usage by 8.63 Mbps ($p < 0.001$), while sustaining higher throughput by 78.4 windows/s. Overall, the findings demonstrated that predictive accuracy and prognostic reliability were primarily governed by representation and architecture choices, whereas IoT placement chiefly determined latency and bandwidth behavior under realistic smart-factory streaming conditions.

Keywords

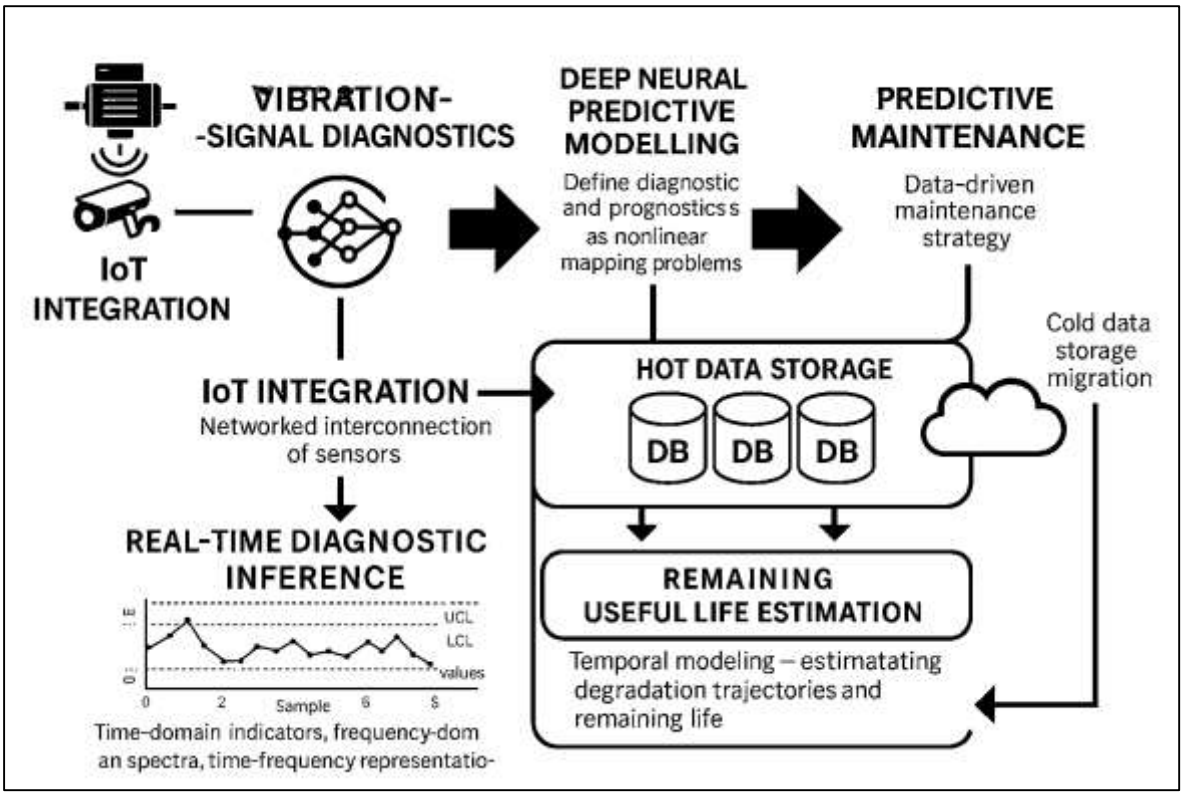
Iot Integration, Deep Neural Networks, Vibration Diagnostics, Predictive Maintenance, Smart Factories.

INTRODUCTION

Predictive maintenance is defined as a data-driven maintenance strategy that relies on continuous or periodic monitoring of machine condition to identify degradation patterns and estimate the likelihood of failure before functional breakdown occurs (Zhang et al., 2019). In industrial engineering and manufacturing analytics, predictive maintenance is distinguished from preventive and corrective maintenance by its reliance on measurable signals and quantitative models rather than fixed schedules or post-failure intervention. Diagnostics, within this framework, refers to the systematic identification of fault presence, fault type, fault location, and fault severity using observable machine data, while prognostics refers to the quantitative estimation of degradation progression and remaining useful life based on historical and real-time observations. In smart factories, these concepts are operationalized through cyber–physical integration, where physical assets are continuously sensed and digitally represented within interconnected computational environments. International manufacturing systems increasingly rely on predictive maintenance because production networks span multiple regions, operate under tight delivery constraints, and involve complex asset interdependencies. Unplanned equipment failures generate cascading effects across supply chains, affecting throughput, quality consistency, and contractual performance across borders (Sakib & Wuest, 2018). The global relevance of predictive maintenance is therefore rooted in its role as a stabilizing mechanism for industrial productivity and reliability. Within this context, vibration-based monitoring occupies a central position because mechanical vibration directly reflects dynamic interactions among rotating and moving components such as bearings, gears, shafts, and spindles. Vibration signals contain high-resolution information about mechanical condition and respond sensitively to changes in load, alignment, surface damage, and structural integrity. From a quantitative perspective, vibration signals are treated as stochastic time-series data influenced by operational parameters and environmental variability, requiring systematic sampling, preprocessing, and modeling. The integration of vibration diagnostics into predictive maintenance systems transforms maintenance decision-making into a measurable inference process grounded in signal analysis and statistical learning (A. Kumar et al., 2018). When embedded within smart factories, these systems become integral components of digitally coordinated production environments that demand standardized, scalable, and interoperable maintenance intelligence across international manufacturing ecosystems.

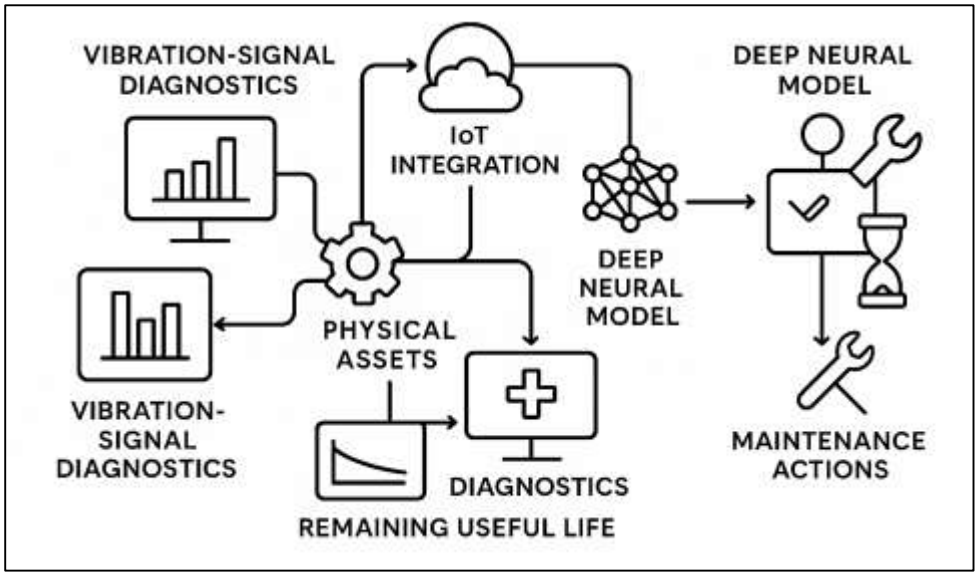
Vibration-signal diagnostics has traditionally relied on signal processing techniques that transform raw sensor data into descriptive features representing machine condition. Time-domain indicators, frequency-domain spectra, and time–frequency representations have long been used to detect characteristic fault signatures associated with mechanical degradation (Niu, 2017). These methods formalize diagnostics as a pattern recognition task in which engineered features are mapped to discrete health states. Quantitative challenges arise from the nonstationary nature of industrial vibration signals, which are influenced by variations in speed, load, material properties, and operating regimes. In industrial environments, vibration data often exhibit noise contamination, transient disturbances, and overlapping fault signatures, complicating fault isolation and classification. As manufacturing systems become more complex and automated, the volume and dimensionality of vibration data increase substantially, creating conditions under which manual feature design becomes inefficient and incomplete. This shift has motivated the adoption of data-driven representation learning approaches that learn discriminative features directly from raw or minimally processed vibration signals. In this paradigm, diagnostics is formulated as a supervised or semi-supervised learning problem in which models infer health states from high-dimensional input sequences (Hoffmann et al., 2020). The quantitative emphasis moves from feature selection to model architecture, loss functions, optimization strategies, and evaluation metrics. Vibration diagnostics in smart factories must therefore be defined not only by signal characteristics but also by the computational models used to interpret them. This reconceptualization aligns with the increasing availability of high-frequency vibration sensors and the need for automated, scalable diagnostic systems capable of operating across diverse machines and facilities. As vibration-based diagnostics becomes embedded within predictive maintenance pipelines, its quantitative rigor directly influences the reliability and consistency of maintenance decisions in globally distributed manufacturing environments (Gianoglio et al., 2020).

Figure 1: IoT-Based Predictive Maintenance Framework



Predictive maintenance systems extend vibration diagnostics by incorporating temporal modeling to estimate degradation trajectories and remaining useful life. Remaining useful life estimation is defined as the quantitative prediction of the time interval between the current observation and the point at which an asset no longer meets operational requirements (Arfan et al., 2021; Zhai et al., 2019). This task requires models capable of capturing long-term dependencies, degradation accumulation, and condition transitions under variable operating conditions. In vibration-based prognostics, time-series segmentation, labeling strategies, and health indicator construction play a critical role in shaping model performance (Jahid, 2021). Health indicators are often derived as latent variables that summarize complex vibration dynamics into monotonic or trend-consistent representations suitable for degradation modeling. From a quantitative standpoint, predictive maintenance integrates diagnostics and prognostics into a unified inference pipeline, where fault identification and life estimation are jointly influenced by signal quality, modeling assumptions, and temporal resolution (Akbar & Farzana, 2021; Mosallam et al., 2016). Industrial datasets used for predictive maintenance commonly contain incomplete failure histories, censored observations, and imbalanced condition distributions, which impose additional statistical constraints on model training and validation. The reliability of remaining useful life estimates is therefore dependent on how uncertainty, noise, and operational variability are handled within the modeling framework. In smart factories, predictive maintenance outputs are expected to support scheduling, inventory planning, and production coordination, which places strict demands on consistency and repeatability. International manufacturing operations amplify these demands because identical equipment may operate under different environmental, regulatory, and workload conditions across sites (Reza et al., 2021; Saikat, 2021). A vibration-based predictive maintenance system must therefore be robust to distributional variation while maintaining quantitative accuracy. This requirement reinforces the importance of systematic model evaluation using clearly defined performance metrics and standardized experimental protocols (Kim et al., 2017; Shaikh & Aditya, 2021; Zobayer, 2021a). Predictive maintenance, when grounded in vibration-signal analytics, becomes a continuous estimation problem whose validity depends on the coherence of data acquisition, temporal modeling, and inference logic across the entire system.

Figure 2: IoT-Integrated Predictive Maintenance Architecture



The integration of predictive maintenance with the Industrial Internet of Things introduces an architectural dimension that fundamentally shapes how vibration data are collected, transmitted, and analyzed (Alam & Alam, 2022; Quatrini et al., 2020; Zobayer, 2021b). IoT integration is defined as the networked interconnection of sensors, machines, gateways, and computational services that enable continuous data exchange and coordinated analytics. In smart factories, vibration sensors are embedded within machines and connected through industrial communication protocols to edge devices, fog nodes, or cloud platforms. This connectivity allows vibration data to be acquired at scale and analyzed in near real time, transforming predictive maintenance into an always-on monitoring capability. From a quantitative perspective, IoT integration imposes constraints related to sampling rates, data bandwidth, latency, synchronization, and data integrity. High-frequency vibration signals generate large data volumes, requiring careful design of edge processing and data reduction strategies to maintain system efficiency (Mesbaul & Farabe, 2022; Singha et al., 2020). The placement of computation along the IoT continuum influences the statistical properties of the data available for model training and inference. For example, edge-based preprocessing may alter signal resolution, while cloud-based aggregation may introduce delays or aggregation bias. Smart factories operating across international locations require IoT architectures that support standardized data formats and interoperable analytics pipelines to ensure comparability of predictive maintenance outputs. Predictive maintenance systems must therefore be designed as distributed systems rather than isolated analytical models (Nahid, 2022; Hossain & Milon, 2022). This system-level perspective treats vibration diagnostics and prognostics as components embedded within a broader digital infrastructure that governs data flow and computational execution. The quantitative behavior of predictive maintenance models cannot be separated from the IoT environments in which they operate, making integration architecture an essential dimension of system definition and evaluation (Kumar et al., 2018; Abdur & Haider, 2022).

Deep neural networks have emerged as a dominant modeling paradigm for vibration-based predictive maintenance due to their capacity to learn hierarchical representations from complex signals. Deep neural models define diagnostics and prognostics as nonlinear mapping problems in which high-dimensional vibration sequences are transformed into fault labels, health indicators, or remaining life estimates (Hwang et al., 2018; Mushfequr & Praveen, 2022; Mortuza & Rauf, 2022). Convolutional neural networks are particularly suited to vibration analysis because convolutional filters capture localized temporal patterns associated with impulsive events and repetitive mechanical interactions. Recurrent neural networks and temporal convolutional structures extend this capability by modeling sequential dependencies and degradation dynamics over time (Rakibul & Samia, 2022; Saikat, 2022). From a quantitative standpoint, deep neural predictive maintenance systems are characterized by architectural depth, receptive field design, parameterization, and regularization strategies. These

design choices influence model sensitivity to noise, operating variability, and fault severity. The training of deep neural networks for vibration diagnostics requires large, representative datasets and carefully designed validation strategies to avoid overfitting and optimistic performance estimates (Sohel et al., 2022; Yan et al., 2016). In industrial settings, deep models must also balance predictive accuracy with computational efficiency, particularly when deployed on edge devices with limited resources. Smart factories increasingly rely on automated inference pipelines where deep neural models operate continuously on streaming vibration data. This operational context reinforces the need for stable, reproducible model behavior under fluctuating data conditions. The quantitative evaluation of deep neural predictive maintenance systems therefore extends beyond classification or regression accuracy to include robustness, consistency, and sensitivity to data perturbations (Tanwar & Raghavan, 2020). These considerations are central to deploying vibration-based deep learning systems across globally distributed manufacturing environments.

Measurement reliability and statistical validity represent foundational concerns in vibration-based predictive maintenance research. Vibration signals are influenced by sensor placement, mounting stiffness, sampling frequency, and environmental interference, all of which affect signal fidelity. In industrial environments, machines operate under varying loads and speeds, introducing nonstationarity that complicates direct comparison of vibration patterns across time and assets (Yoo & Baek, 2018). Quantitative modeling must therefore account for these sources of variability to ensure that learned representations correspond to mechanical condition rather than incidental operating effects. Data segmentation strategies, window lengths, and overlap ratios influence both diagnostic separability and prognostic trend estimation. Improper segmentation can obscure fault signatures or distort degradation trajectories, leading to misleading model outputs. Class imbalance is a persistent issue in predictive maintenance datasets, as failure events are rare relative to normal operation. This imbalance affects model training dynamics and evaluation metrics, requiring careful metric selection and sampling strategies (Jung et al., 2017). Remaining useful life estimation introduces additional challenges related to uncertainty quantification, as point predictions alone may not adequately represent degradation variability. In smart factories, maintenance decisions are sensitive to the reliability of predictive outputs, making uncertainty an inherent component of quantitative assessment. International deployment further intensifies these challenges because differences in operational practices and environmental conditions introduce additional variability into vibration data distributions. A predictive maintenance system must therefore be evaluated under conditions that reflect realistic operating diversity (Wong et al., 2020). Measurement and statistical considerations are not peripheral concerns; they define the boundaries within which predictive maintenance claims are meaningful and transferable across industrial contexts.

An IoT-integrated deep neural predictive maintenance system with vibration-signal diagnostics represents a multi-layered research object that combines sensing, networking, computation, and inference into a unified analytical framework. At the sensing layer, vibration measurements capture mechanical behavior at high temporal resolution (de Azevedo et al., 2016). At the communication layer, IoT infrastructure enables scalable data transmission and coordination across assets and facilities. At the computational layer, deep neural networks transform vibration data into diagnostic and prognostic outputs through learned representations. At the system layer, these outputs are contextualized within smart factory operations that span multiple machines and production stages. The international significance of such systems arises from their role in supporting consistent maintenance intelligence across geographically distributed manufacturing networks. Quantitative evaluation of these systems requires explicit definition of data acquisition protocols, model architectures, deployment configurations, and performance metrics (Rivera et al., 2019). Each layer introduces constraints that shape the statistical properties of the data and the behavior of predictive models. Smart factories rely on these systems to maintain operational stability and efficiency under complex and variable conditions. An extended quantitative introduction must therefore treat predictive maintenance not as a standalone algorithmic task but as an integrated system whose performance emerges from the interaction of vibration sensing, IoT connectivity, and deep neural inference. This integrated perspective provides the conceptual foundation for empirical analysis of predictive maintenance performance in smart manufacturing environments without invoking conclusions or prescriptive

implications (Balali et al., 2020).

The objective of the study titled IoT-Integrated Deep Neural Predictive Maintenance System with Vibration-Signal Diagnostics in Smart Factories was to quantitatively evaluate how an end-to-end predictive maintenance pipeline performed when vibration sensing, deep neural inference, and Industrial IoT deployment constraints were treated as measurable experimental conditions rather than background implementation details. Specifically, the study aimed to determine whether vibration representation choices (raw time-domain windows, short-time time–frequency representations, and wavelet-based time–frequency representations) produced statistically distinguishable differences in diagnostic accuracy and prognostic error under realistic operating-regime variability. A second objective was to compare deep neural architecture families, including waveform-focused convolutional models and sequence-aware temporal models, to identify which modeling approach produced the most stable classification and remaining useful life estimation results when evaluated using leakage-resistant partitioning and repeated runs. A third objective was to treat inference placement as a system-level experimental factor by quantifying the operational impact of running preprocessing and inference at the edge versus in the cloud, using measured indicators such as end-to-end latency percentiles, sustainable throughput, bandwidth consumption, and compute footprint. A fourth objective was to quantify robustness to stream-quality degradation by evaluating model and pipeline performance under controlled noise injection and missing-segment conditions that resembled IoT packet loss and buffering behavior, and by measuring the associated changes in diagnostic metrics and remaining useful life errors. A fifth objective was to validate the measurement credibility of learned health indicators and latent representations by testing monotonicity and smoothness properties across run-to-failure trajectories, examining convergent associations with degradation progression proxies, and assessing discriminant separation between fault categories and operating-regime groupings. Finally, the study aimed to integrate predictive and operational outcomes into a readiness-oriented quantitative assessment by determining whether pipeline configurations met predefined performance floors for diagnostics and prognostics while simultaneously meeting latency, throughput, and continuity criteria expected in smart-factory monitoring environments. Through these objectives, the study sought to produce a rigorously measured comparison of alternative pipeline configurations and to establish a transparent basis for evaluating predictive maintenance systems as joint combinations of sensing quality, model capability, and IoT deployment feasibility.

LITERATURE REVIEW

The literature review for a quantitative study titled "IoT-Integrated Deep Neural Predictive Maintenance System with Vibration-Signal Diagnostics in Smart Factories" consolidates and structures prior empirical and methodological work that underpins the measurement, modeling, and evaluation of vibration-driven predictive maintenance within connected industrial environments (Dalzochio et al., 2020). Because the proposed research is quantitative, the review emphasizes variables, observable indicators, datasets, experimental designs, validation strategies, and statistical or machine-learning performance metrics that have been used to demonstrate predictive maintenance effectiveness. The focus is not limited to algorithm selection; it also covers how vibration signals are captured, transformed, labeled, and partitioned, because these steps define the data-generating process and directly shape model outcomes. The review additionally examines IoT integration as a measurable system condition affecting latency, data loss, sampling continuity, synchronization, compute placement, and scalability—factors that can modify signal integrity and, consequently, the diagnostic and prognostic accuracy of deep neural models (Sakib & Wuest, 2018). Within smart factory contexts, predictive maintenance is framed as a pipeline spanning sensor deployment, edge or gateway processing, cloud analytics, and decision outputs, so the review organizes prior studies according to the pipeline stages and their quantitative dependencies. This section also synthesizes how deep neural networks have been operationalized for vibration-based fault classification and remaining useful life estimation, including model architectures, training schemes, regularization strategies, domain shift handling, and uncertainty representation. By systematically mapping what has been measured, how it has been modeled, and how performance has been validated, the literature review establishes the conceptual and empirical foundation for the study's variables, hypotheses, and methodological choices (Malik et al., 2018).

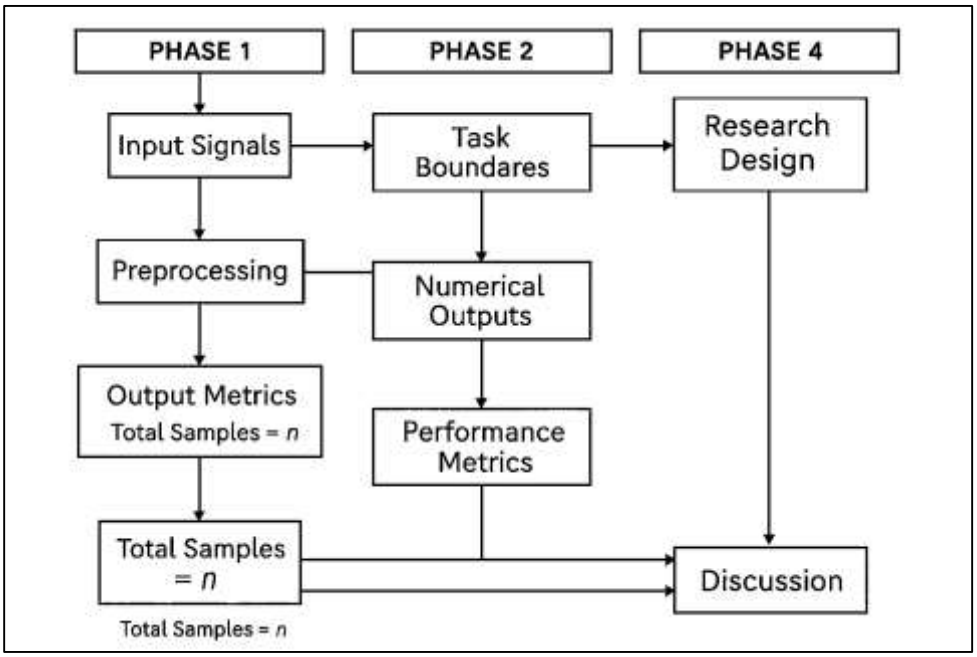
Scope of Predictive Maintenance

Predictive maintenance is consistently treated in quantitative research as a measurable inference pipeline that converts observed machine-condition signals into maintenance-relevant outputs through structured transformations and model-based decision rules (Bousdekis et al., 2019). Across a broad body of empirical studies in manufacturing, energy, transportation, and process industries, the pipeline begins with input signals such as vibration, acoustic emission, motor current, temperature, pressure, and operational context variables, followed by preprocessing and representation steps that may include filtering, normalization, segmentation, and conversion into time, frequency, or time–frequency forms. The pipeline ends with outputs that can be evaluated numerically, which is why predictive maintenance is often positioned as an applied analytics problem rather than a purely operational practice. Within this framing, the literature draws clear boundaries among related tasks that are frequently blended in non-technical discussions. Fault detection is typically defined as determining whether a deviation from normal behavior is present; fault diagnosis extends this by assigning a fault type or source; fault severity estimation quantifies damage progression on an ordinal or continuous scale; health indexing produces a continuous health score that tracks degradation; and remaining useful life prediction estimates time-to-failure under current and observed operating conditions (Cachada et al., 2018). Empirical studies often operationalize these tasks as classification, regression, or hybrid learning problems, with outputs including discrete class labels, probability scores, ranked fault likelihoods, health trajectories over time, and time-to-failure estimates accompanied by uncertainty bounds. The literature also shows that the same raw data stream can serve multiple inference targets depending on labeling strategy and experimental design; for example, a dataset built for fault classification can be repurposed for health indexing by deriving a monotonic health indicator, or for remaining life prediction by linking observation windows to failure timestamps. Many studies emphasize that definitions become quantitative only when they specify how signals are segmented into samples, how labels are assigned, and what constitutes a failure event or unacceptable performance threshold. Another recurring theme is that predictive maintenance is not a single model but a chain of measurable decisions, where sensor fidelity, sampling adequacy, representation choice, and model capacity jointly determine performance. Because modern factories operate across heterogeneous assets and varying regimes, studies frequently propose that predictive maintenance pipelines must explicitly incorporate regime indicators such as speed, load, duty cycle, and environmental conditions so that inferences remain comparable across time and equipment (Menezes et al., 2019). This emphasis on operational definitions and task boundaries provides the conceptual scaffolding for evaluating predictive maintenance systems using quantitative evidence rather than anecdotal downtime reduction claims.

Quantitative predictive maintenance studies report a relatively stable set of dependent variables that allow performance comparisons across methods, datasets, and industrial contexts. For diagnostic tasks (detection and diagnosis), classification accuracy remains widely reported, yet many studies also highlight that accuracy alone is insufficient when faults are rare, class distributions are uneven, or misclassification costs differ by fault mode (Seele, 2017). As a result, macro- and micro-averaged F1 scores, precision and recall, and precision–recall area-under-curve are frequently used to represent model performance under imbalance and to reflect tradeoffs between missed detections and false alarms. Confusion matrices are not treated merely as descriptive artifacts; a substantial portion of the literature uses confusion matrix stability across repeated splits or repeated runs to evaluate whether a model's error patterns are consistent, which matters when maintenance actions depend on specific fault-mode discrimination rather than aggregate correctness. For prognostic tasks such as remaining useful life prediction, regression error metrics are dominant, with absolute and squared error summaries used to capture average deviation between predicted and true remaining life. Many studies also go beyond single-number errors and report horizon-based measures that quantify how early the model can produce reliable predictions, as well as penalty scoring schemes that assign higher cost to late predictions than to early predictions when maintenance scheduling depends on lead time. In addition to accuracy and error, the literature increasingly treats reliability and stability as core dependent outcomes, particularly in industrial deployments where models are retrained, updated, or

transferred across machines (Appelbaum et al., 2017). Common stability indicators include performance variance across cross-validation folds, sensitivity to random initialization or sampling, and confidence intervals around reported metrics. Calibration is also a recurring quantitative concern in studies that output probabilities or risk scores; research frequently evaluates whether predicted probabilities align with observed frequencies, because poorly calibrated models can produce confident but incorrect decisions. Several studies treat calibration error and threshold sensitivity as essential indicators of whether probability outputs are actionable for maintenance planning. Across these dependent variables, the literature repeatedly stresses that metric choice must align with the decision context: detection tasks prioritize sensitivity under low false-alarm budgets, diagnosis tasks prioritize discriminating among fault classes, severity estimation prioritizes monotonic and noise-robust mapping, and remaining life prediction prioritizes both accuracy and timeliness (Beverungen et al., 2019). This convergence in dependent variables across many studies creates a methodological baseline that quantitative papers use to justify evaluation designs and to interpret results in a manner that is consistent with prior empirical evidence.

Figure 3: Quantitative Predictive Maintenance Evaluation Framework



Research design choices strongly shape reported performance in predictive maintenance, and the literature consistently documents that experimental protocols can inflate or deflate results even when the same dataset and model family are used. A major design axis differentiates laboratory test rigs from real industrial equipment studies (Dinov, 2018). Test rigs are valued for controlled conditions, repeatability, clear fault seeding, and complete run-to-failure trajectories; industrial studies are valued for ecological validity, richer operational variability, and realistic noise sources. The literature recognizes that these environments generate different data distributions and label quality, leading to different generalization behavior. Many empirical papers use rig datasets to benchmark model architectures and representation strategies, then discuss the challenges that appear when transferring methods to industrial settings with partial labeling, maintenance interventions, and nonstationary operating regimes. Validation strategy is another major axis. Cross-validation is widely used for diagnostic classification because it enables statistical summaries across folds, yet multiple studies caution that naive random splitting can introduce leakage when samples are segmented from the same continuous run or the same machine instance, allowing near-duplicate signal patterns to appear in both training and testing. To address this, the literature frequently recommends splitting by run, by machine, or by operating regime so that the test set represents genuinely unseen conditions (Vilarinho et al., 2017). For prognostics and remaining life prediction, rolling-origin evaluation and time-aware

splitting are commonly emphasized because the prediction target is inherently temporal; studies often show that random splitting in time-series prognostics overstates performance by allowing future-like patterns to be learned from later-life segments. Many quantitative papers therefore treat temporal validation as a central requirement, linking model credibility to whether the evaluation respects chronological order. A related design dimension concerns the unit of analysis: segment-level evaluation may measure instantaneous diagnostic capability, while run-level evaluation measures whether a system can track degradation consistently across an asset's lifecycle. The literature also treats hyperparameter tuning and early stopping practices as part of research design, noting that tuning on test sets or using non-isolated validation procedures undermines interpretability of reported metrics. Across multiple studies, reproducibility practices such as reporting random seeds, repeated trials, and confidence intervals are increasingly used to characterize uncertainty in results. Finally, the literature shows that predictive maintenance research designs often incorporate stratification by operating conditions—speed, load, temperature bands—because models can perform well in one regime and degrade sharply in another, and aggregated metrics can hide these effects (Iacobucci et al., 2019). In sum, predictive maintenance performance is presented in the literature as a combined outcome of modeling choices and experimental design integrity, with leakage prevention and time-aware evaluation treated as prerequisites for trustworthy quantitative findings.

A synthesized view across many predictive maintenance studies indicates that the conceptual boundaries of tasks, the selection of dependent variables, and the integrity of research design function together as a quantitative framework for evaluating maintenance intelligence systems. Studies that explicitly separate detection, diagnosis, severity estimation, health indexing, and remaining life prediction typically produce clearer measurement models because each task implies different labels, targets, error structures, and acceptable tradeoffs (Ardolino et al., 2018). This separation also clarifies how outputs should be operationalized: detection outputs often take the form of anomaly scores and binary labels; diagnosis outputs are multi-class labels or ranked probabilities; severity outputs are ordinal classes or continuous damage indices; health indexing produces smooth trajectories intended to correlate with degradation; remaining life prediction yields time-to-failure estimates that may be paired with uncertainty quantification. The literature suggests that these outputs are not interchangeable because their evaluation requires different metrics, and reporting only one metric can mask deficiencies that matter operationally. Empirical work repeatedly shows that diagnostic performance metrics are sensitive to class imbalance and sampling practices, leading to the widespread use of precision–recall based measures and F1 summaries alongside accuracy. Prognostic metrics are sensitive to lifecycle coverage and temporal validation choices, leading to the frequent use of multiple error metrics and horizon-based scoring approaches (Mai, 2016). Stability indicators such as variance across folds and repeated runs are used to argue that results are not artifacts of a favorable split or a lucky initialization. Calibration-oriented evaluation appears in studies that treat predicted probabilities as decision variables, reinforcing that confidence measures must be assessed, not assumed. Research design synthesis also highlights a recurring pattern: controlled test rigs provide a foundation for systematic comparisons, while industrial datasets reveal challenges tied to noise, missingness, intermittent maintenance actions, and heterogeneous regimes. Many studies treat cross-validation as meaningful only when the splitting unit prevents leakage, and treat temporal evaluation as essential when the target depends on degradation time. Across these works, reporting practices converge toward describing dataset segmentation, splitting strategy, and the relationship between samples and physical runs, because these details define whether the experiment measures generalization or memorization. This integrated framework—task boundary clarity, metric alignment, and design integrity—constitutes the dominant quantitative lens through which predictive maintenance systems are compared in the literature (Erikainen & Chan, 2019). It also provides a structured rationale for positioning any predictive maintenance study as a measurable, testable, and reproducible investigation grounded in signal-to-output inference rather than narrative claims about maintenance effectiveness.

Vibration-Signal Diagnostics as the Primary Measurement Modality

Vibration-signal diagnostics is widely treated as the primary measurement modality for rotating and reciprocating assets because vibration captures dynamic mechanical interactions that are directly coupled to fault mechanisms in bearings, gears, shafts, couplings, and spindles (Goyal & Pabla, 2016).

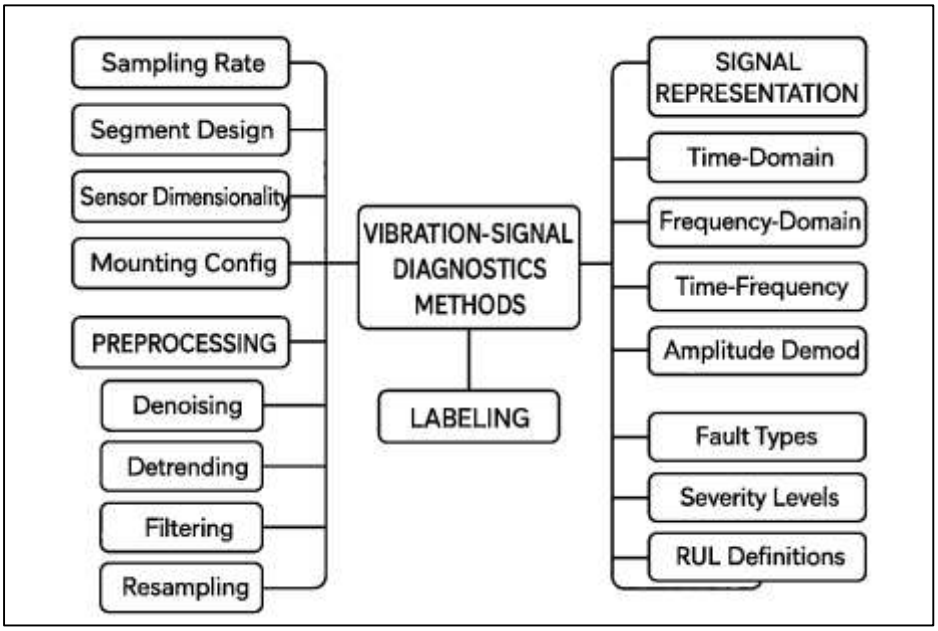
In quantitative studies, the first measurement concern is sampling adequacy, since vibration signatures often contain impulsive components and harmonics that occur at frequencies well above general process-sensor ranges. Sampling rate selection is therefore framed as a tradeoff between capturing high-frequency fault content and managing data volume, and the literature consistently discusses aliasing risk when sampling is too low or when anti-alias filters are not matched to the acquisition strategy. A second measurement concern is segmentation, because most learning and statistical methods operate on fixed-length windows rather than continuous streams (Fan et al., 2020). Window length is treated as a measurable descriptor that influences the stability of spectral estimates, the detectability of transient impulses, and the amount of contextual information available to diagnostic models. Overlap ratio is also treated as a design parameter because overlap increases the number of training samples and smooths temporal transitions but can introduce redundancy that inflates apparent performance if data splits are not carefully separated by run or time. Studies frequently report segment counts per condition to quantify class balance and to contextualize accuracy or error metrics in relation to data availability. Sensor dimensionality is another recurring descriptor: single-axis accelerometers provide a one-dimensional time series aligned with a mounting orientation, while tri-axial sensors capture multi-directional dynamics that may reveal faults more robustly under variable mounting or load conditions (Hong et al., 2016). Quantitative comparisons often emphasize that tri-axial sensing increases dimensionality and compute cost, yet may improve fault separability when the dominant vibration axis changes across regimes. Mounting methods and sensor placement variability are treated as key sources of measurement uncertainty. Rigid stud mounting typically yields higher bandwidth and more reliable high-frequency content than adhesive mounting, while placement near the fault transmission path increases sensitivity to localized defects. In real shop-floor environments, sensor placement is constrained by safety, accessibility, cable routing, and environmental exposure, which means that vibration measurements can vary substantially between nominally identical machines and across sites. For smart-factory deployments, this variability is not treated as an anecdotal issue; it is framed as a distributional shift problem that affects the repeatability of diagnostic features and learned representations (N. Zhou et al., 2020). Consequently, vibration-signal diagnostics literature tends to define "measurement modality" not only by the sensor type, but by a measurable acquisition configuration characterized by sampling rate, segment design, axis selection, mounting stiffness, and placement repeatability, because these parameters collectively determine what fault information is observable and what information is irretrievably lost.

Signal preprocessing is presented in the vibration diagnostics literature as a quantitatively consequential stage because it determines the statistical properties of the data that downstream models consume (Yong Li et al., 2018). Denoising is commonly treated as a mechanism for improving the ratio of fault-relevant signal content to background vibration, and studies discuss denoising both as classical filtering and as more adaptive techniques that separate structured components from broadband noise. Detrending is often used when low-frequency drift or sensor bias produces slow variations that obscure impulsive behavior, particularly in long monitoring sessions where thermal changes and mounting relaxation can alter baseline vibration. Normalization is frequently applied to stabilize scale differences across sensors, machines, or operating regimes, and quantitative work often compares normalization schemes by examining the stability of feature distributions and the sensitivity of models to amplitude scaling. Filtering choices, including high-pass and band-pass filters, are treated as explicit design parameters because they shape the spectrum presented to diagnostic algorithms. High-pass filtering may remove low-frequency structural motion and emphasize bearing-related impulsive bands, while band-pass filtering can isolate frequency regions known to carry defect energy (Figlus & Koziol, 2016). The measurable impact of such filtering is often reflected in changes to diagnostic separability, measured by class clustering, margin statistics, or performance metrics in supervised classifiers. Resampling and synchronization are also repeatedly highlighted, especially when speed variation causes frequency smearing or when vibration must be aligned with rotational phase. Some studies incorporate tachometer signals or speed proxies, using them to align windows by rotation or to normalize features by order, while others synchronize multiple sensors through timestamp correction. In connected environments, synchronization can become an infrastructure concern because clocks drift and transmission jitter can introduce misalignment across sensors and machines. Quantitative

monitoring of data quality therefore becomes part of preprocessing. Signal-to-noise ratio estimation is used to assess whether vibration segments contain adequate defect information, and drift monitoring is used to detect gradual changes in sensor behavior or mounting conditions that could mimic degradation. Drift is treated as a measurable phenomenon that can be tracked through baseline statistics, spectral centroids, or long-term changes in band energy. Missing data and discontinuities are also common in IoT-linked acquisition, and preprocessing pipelines often include gap detection, interpolation rules, or segment rejection thresholds (Er & Tan, 2018). The literature frames these decisions as having measurable downstream consequences: rejecting too many segments reduces training diversity and can bias class distributions; imputing or interpolating can introduce artificial smoothness; retaining corrupted segments can degrade model robustness. As a result, preprocessing is frequently conceptualized as a quantifiable transformation chain whose settings must be reported and justified because the chain governs data integrity, feature stability, and the validity of performance claims.

A central theme in vibration-signal diagnostics research is the comparison of signal representations used for modeling, because representation choice mediates the balance among predictive performance, interpretability, and computational efficiency (Amezquita-Sanchez & Adeli, 2016). Raw one-dimensional time series representations are attractive for end-to-end learning approaches because they avoid information loss from handcrafted transformations and allow models to learn directly from waveform morphology. Quantitative studies that use raw signals typically emphasize segmentation strategy, normalization, and architecture design to handle noise and capture impulsive patterns, and they evaluate performance gains against compute requirements, particularly when inference is intended to run on edge devices. Frequency-domain representations, including Fourier magnitude spectra and power spectral density estimates, are widely used because many mechanical faults manifest as characteristic frequency components and sidebands (Feng et al., 2017). These representations can improve class separability when the operating regime is stable, and they often reduce sensitivity to time shifts within windows. However, frequency-domain approaches may perform inconsistently under variable speed, transient loads, or nonstationary conditions where spectral content shifts over time. Time–frequency representations, such as short-time spectrograms and wavelet scalograms, are commonly introduced to address nonstationarity by preserving both temporal localization and frequency structure. Quantitative comparisons often show that time–frequency images can enhance fault discriminability in complex regimes but increase computational cost due to transform overhead and higher-dimensional inputs. This cost is discussed not only in terms of model runtime but also in terms of memory footprint, storage bandwidth, and pipeline latency in IoT-enabled monitoring systems. Envelope spectra and demodulated-band representations occupy a special role in bearing and gearbox diagnostics, because demodulation can isolate modulation effects produced by localized defects and can highlight characteristic fault frequencies that are less visible in raw spectra. Many studies evaluate envelope-based features as a way to improve sensitivity to early-stage faults where broadband vibration is weak, and they compare envelope methods against raw and time–frequency representations using standardized metrics (Cunningham, 2016). Across representation types, the literature frames evaluation as a multi-objective comparison: performance metrics measure diagnostic accuracy and robustness; computational metrics measure transform time and inference latency; resource metrics measure bandwidth usage when representations must be transmitted rather than computed locally. This framing is especially pronounced in smart-factory contexts where multiple machines are monitored simultaneously and where data pipelines must scale. Representation choice is therefore treated as a measurable design decision that can shift the operating point of the system: a high-performing representation that is too expensive may fail to meet latency or throughput constraints, while a lightweight representation may degrade sensitivity to subtle defects. Quantitative studies often respond to this tension by benchmarking multiple representations under consistent splits and reporting both predictive and operational metrics, recognizing that representation effectiveness depends on fault type, regime variability, sensor configuration, and compute placement (Zakhezin & Pryadko, 2016).

Figure 4: Vibration Diagnostics Measurement Framework Diagram



Labeling strategies and ground-truth construction form the final pillar of vibration-signal diagnostics as a measurement modality, because labels define what the model is trained to detect and what "correctness" means under evaluation. Fault type labels are commonly constructed to represent distinct mechanical defect modes such as bearing inner-race defects, outer-race defects, rolling element defects, cage defects, gear tooth damage, misalignment, imbalance, looseness, and lubrication issues (Yongzhuo Li et al., 2018). In controlled datasets, these labels may be produced through seeded faults or structured experiments, enabling clean class boundaries. In industrial settings, labels are more often derived from maintenance logs, inspections, replacement records, or expert annotations, which introduces label noise and temporal ambiguity because the onset of degradation may precede recorded interventions. Quantitative studies frequently discuss this gap between observed vibration and recorded events, and many designs adopt window-level labels that assume a period near a known failure is "faulty" while earlier periods are "healthy," acknowledging that this creates a transition region with mixed characteristics. Severity labeling introduces additional complexity. Some studies treat severity as ordinal categories tied to defect size or operating time, while others treat severity as a continuous target derived from degradation proxies or measured damage extent (Szymański & Tomaszewski, 2016). The choice between ordinal and continuous severity targets influences both the modeling approach and the evaluation metrics, and it also affects the interpretability of outputs for maintenance decisions. Run-to-failure labeling is essential for remaining useful life targets because it defines the mapping between observation time and time-to-failure. In many datasets, time index is used as a surrogate for degradation progression, while failure threshold definition varies by study: thresholds may be defined by vibration amplitude limits, quality loss, functional failure, or intervention points. These definitions change the meaning of "remaining useful life" and can alter quantitative results even when the same signal data are used. Studies therefore stress the need to specify failure criteria and the rule used to assign RUL labels to segments. Another recurring theme is leakage avoidance: when labels are assigned at the run level, segment-level splitting can place adjacent windows from the same run into both training and test sets, inflating performance. Robust experimental designs often split by run or by machine to ensure that labels represent truly unseen operating histories (Liu et al., 2017). Ground-truth construction also includes the choice of label granularity: binary healthy/faulty labels simplify detection but may conceal diagnostic nuance; multi-class labels support fault isolation but increase class imbalance and confusion risk; hierarchical labels represent fault families and subtypes but require careful evaluation. In smart-factory deployments, where data sources are heterogeneous and label availability varies, the literature treats labeling as a measurable constraint on what can be learned and

validated. Ground truth is thus not an afterthought; it is a quantitative definition of the prediction target that governs dataset structure, modeling choices, and the credibility of diagnostic and prognostic performance reports (Zhan et al., 2018).

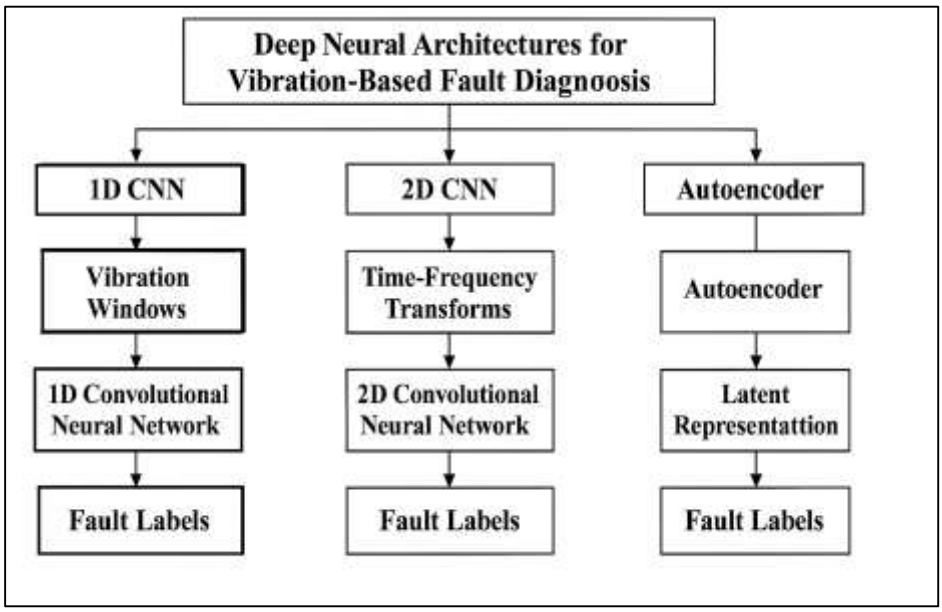
Deep Neural Architectures for Vibration-Based Fault Diagnosis

Deep neural architectures for vibration-based fault diagnosis are typically presented in the literature as end-to-end or hybrid pattern-recognition systems that learn discriminative representations from segmented vibration windows and map them to fault labels or fault probabilities. Among these architectures, one-dimensional convolutional neural networks are repeatedly emphasized for direct interpretation of raw vibration time series because they preserve waveform structure and avoid transformation overhead (Kolar et al., 2020). Empirical studies generally treat filter width, stride, and pooling as measurable design choices that determine how much local impulsive content, periodic modulation, and broader contextual dynamics are captured. Wider early-layer filters are often linked to improved tolerance to broadband noise and small phase shifts, while smaller filters can preserve fine-grained transients when combined across depth. Receptive field reasoning is therefore described as central: models must "see" enough samples to capture fault repetition patterns tied to rotation, gear meshing, or bearing characteristic intervals, while also maintaining sensitivity to localized impulses that signal spalls or cracks. Depth is commonly linked to representational capacity, yet a recurring saturation pattern is discussed in which performance gains diminish after a certain depth threshold, particularly when dataset size is limited or when operating regimes are narrow. In such cases, additional layers may increase variance and amplify overfitting rather than improving generalization (Toh & Park, 2020). Quantitative papers frequently connect these patterns to data volume requirements, reporting the number of segments per class, per condition, and per operating regime, because sample availability constrains the feasible complexity of the network. Data augmentation is widely used to expand training diversity, with studies describing augmentation ratios and types such as window shifting, amplitude scaling, noise injection, or frequency masking. The literature often treats augmentation not as a cosmetic addition but as a measurable lever that can reduce overfitting and improve cross-condition robustness, particularly when faults are rare (Nguyen et al., 2020). Reported results frequently compare performance under different augmentation intensities, showing that modest augmentation can improve stability while aggressive augmentation can blur class boundaries or produce unrealistic samples that reduce accuracy. Many studies also emphasize that segment redundancy created by overlapping windows can inflate apparent performance if the split design allows near-duplicate windows across training and testing; as a result, 1D CNN studies often include splitting by run or machine to demonstrate generalization beyond the immediate signal neighborhood (Chen et al., 2017). Within this body of work, 1D CNNs are characterized as computationally efficient relative to image-based pipelines, supporting real-time inference on edge devices when models are compact and input windows are carefully sized. The quantitative emphasis thus remains consistent: architecture hyperparameters are treated as experimentally controllable factors whose influence can be measured via diagnostic accuracy, F1 variants, confusion profiles, and inference-time metrics under realistic data partitioning.

Two-dimensional CNNs applied to time–frequency transforms represent another major family in vibration diagnostics literature, motivated by the view that many fault signatures are nonstationary and become more separable when represented jointly in time and frequency (Wang et al., 2019). Studies commonly use spectrograms derived from short-time transforms or scalograms derived from wavelet decompositions, treating transform parameterization as a measurable part of the modeling pipeline rather than a fixed preprocessing step. Window size and hop length in time–frequency conversion influence temporal resolution and frequency resolution, and the literature often reports these settings because they define how impulsive events appear and how frequency bands are smoothed. Scale-bin selection in wavelet representations similarly controls detail level, affecting whether subtle bearing-related modulations emerge distinctly or become blurred into broader patterns. Image-size normalization is frequently required to standardize inputs for 2D CNNs, and empirical studies often discuss the information loss tradeoffs created by resizing, cropping, or pooling in the time–frequency domain. For example, aggressive downscaling can erase narrowband sideband structures or short impulses, while minimal downscaling increases memory and compute cost. Quantitative comparisons

in the literature often evaluate 2D CNN pipelines against 1D CNN pipelines on the same datasets, reporting differences not only in accuracy but also in latency, throughput, and memory footprint (Zhao et al., 2019).

Figure 5: Neural Framework for Vibration Diagnosis



A recurring pattern is that 2D CNNs can perform strongly when operating conditions vary and when nonstationary patterns are important, yet they may incur higher computational overhead due to transform generation plus higher-dimensional model inputs. This overhead becomes particularly salient in smart-factory deployments where multiple assets stream data simultaneously and where inference must be performed either at the edge or under constrained bandwidth to a centralized server. Accordingly, many studies frame representation choice as a multi-objective optimization problem in practice: time–frequency images may yield better separability, while raw time series may offer faster end-to-end inference (D. Zhou et al., 2020). Comparative reporting also often addresses interpretability, noting that saliency methods or activation maps can be easier to visualize on time–frequency images than on raw 1D signals, which can be operationally useful for diagnosing why a model predicted a particular fault type. However, the literature also warns that interpretability gains can be misleading if transform settings are not standardized, because different parameterizations produce different "images" from the same vibration segment. The most careful quantitative studies therefore treat time–frequency parameterization as part of the experimental design and conduct sensitivity checks across transform settings, demonstrating whether performance is robust to reasonable parameter variation (Yang et al., 2020). In sum, 2D CNN approaches are typically presented as a strong alternative when nonstationarity and regime variability are prominent, with their empirical value assessed through side-by-side comparisons that include both predictive metrics and system-level compute measures.

Autoencoders and representation learning approaches occupy a distinct role in vibration-based fault diagnosis because they address the practical constraint that labeled fault data are often scarce relative to the volume of unlabeled operational data. In this literature, reconstruction loss is treated as a surrogate objective that encourages networks to learn compressed latent representations capturing dominant signal structure. Studies frequently distinguish between using autoencoders as unsupervised feature extractors for subsequent classifiers and using reconstruction error directly as an anomaly score for fault detection (Khan et al., 2019). Latent dimension selection is repeatedly discussed as a measurable design choice because it determines compression strength and influences whether the latent space retains fault-relevant detail or collapses into overly smooth representations. Quantitative evaluations often use separability measures in the latent space, including clustering compactness and inter-class distance metrics, to assess whether learned embeddings naturally group by condition even before supervision is applied (Yongbo et al., 2020). When labels are available, many studies compare

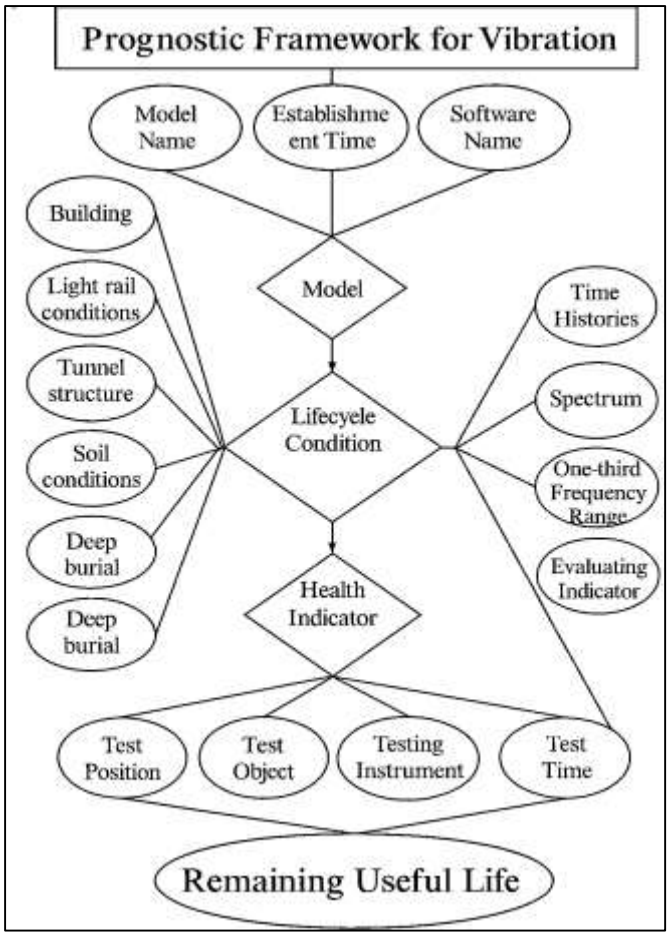
classification accuracy using latent features versus raw features, often reporting that learned embeddings can reduce feature engineering effort and improve robustness to noise when trained on diverse operating data. Semi-supervised setups are common, where only a fraction of the dataset is labeled and the remainder is used for representation learning. In such designs, labeled fraction becomes an explicit independent variable, and diagnostic accuracy is plotted or tabulated across multiple labeling levels to quantify how much supervision is required to reach stable performance (Zhang et al., 2017). This framing is particularly relevant for smart factories, where new assets may generate abundant data but limited fault labels during early deployment phases. Many studies also explore variants such as denoising autoencoders, sparse autoencoders, and stacked architectures, linking each variant to measurable changes in reconstruction stability and downstream classification performance. An important theme is that reconstruction objectives can be biased toward frequent patterns in normal operation, potentially reducing sensitivity to rare fault modes unless training includes representative fault variability or includes mechanisms that prevent the model from reconstructing faults too easily. As a result, quantitative papers often include experiments that test autoencoder-based detectors on unseen faults, assessing whether anomaly scores remain discriminative under distribution shift. Another recurring point concerns computational placement: autoencoders can be used at the edge for dimensionality reduction, transmitting latent vectors instead of raw vibration, thereby reducing bandwidth while preserving condition information (Chen et al., 2020). Studies evaluating such pipelines typically report compression ratios alongside diagnostic accuracy, illustrating the tradeoff between representation compactness and fault separability. Overall, autoencoder-based representation learning is presented as a method family that converts unlabeled vibration streams into structured embeddings, with empirical value measured through separability, downstream classification metrics, and stability under varying label availability and operational conditions (Hasegawa et al., 2019).

Deep Neural Prognostics and Remaining Useful Life Estimation

Deep neural prognostics and remaining useful life estimation from vibration are typically framed in the quantitative literature as supervised or semi-supervised forecasting problems in which segmented vibration observations are mapped to a continuous estimate of time remaining before failure or before a defined functional threshold is reached (Deutsch & He, 2017). Within this body of work, the first technical boundary concerns how the prediction target is defined and scaled. Some studies model remaining useful life directly in interpretable time units aligned with operating time, cycles, or production counts, while other studies normalize the target to a bounded scale to stabilize learning across heterogeneous lifecycles and to reduce sensitivity to extreme values near the beginning of life. Both choices influence loss behavior, error interpretation, and the comparability of results across assets that experience different duty cycles. A second boundary concerns whether remaining useful life is treated as a direct regression output or as a time-to-event outcome that is modeled with survival-style formulations. Direct regression emphasizes minimizing average deviations between predicted and observed remaining time under the assumption that a numeric ground truth exists for each segment. Survival-style framing instead emphasizes the probability that an asset will continue operating beyond a given time horizon, often treating failure time as a stochastic event and allowing predictions to be expressed as risk or survival probability curves (Cheng et al., 2020). The distinction matters because industrial lifecycles are frequently incomplete: assets are replaced preventively, experiments are truncated, or monitoring ends before failure occurs. These incomplete lifecycles create censoring and partial labeling conditions that complicate purely regression-based training, since the precise failure time is unknown for some trajectories. Quantitative prognostics studies typically address this challenge through dataset construction rules that either exclude censored cases, treat end-of-record as a proxy failure threshold, or incorporate censored data through objective functions and evaluation criteria that do not require exact failure labels. Another recurrent challenge arises from operational interruptions and maintenance actions that reset or alter degradation, producing non-monotonic patterns that conflict with naive assumptions of steady deterioration. As a result, prognostic formulation is often presented as a careful target-definition step rather than an automatic labeling procedure: studies specify the operational definition of “failure,” the alignment between vibration windows and lifecycle time index, and the rule that maps each window to a remaining-life value (X. Li et al., 2018). This target-definition layer becomes increasingly important in smart-factory deployments, where assets may run

under multiple regimes and where the same physical wear state can produce different vibration characteristics depending on speed, load, and mounting. The broader prognostics literature therefore treats remaining useful life estimation as a coupled measurement–labeling problem, where vibration serves as the observable, lifecycle definition provides the target, and formulation choice determines what model outputs mean and how they should be judged quantitatively.

Figure 6: Deep Vibration-Based Prognostics Framework



Health indicator construction occupies a central position in vibration-driven prognostics because many studies find that remaining useful life estimation improves when high-dimensional vibration signals are summarized into a health index that behaves like a degradation proxy. In this context, a health indicator is treated as a numerical variable that changes systematically with wear progression, enabling models to learn a smoother mapping between state and remaining time (Xia et al., 2018). The literature distinguishes between engineered health indicators derived from domain-informed vibration features and latent health indicators learned by deep networks as intermediate representations. Latent indicators are often extracted from the penultimate layers of a neural model or from dedicated encoder networks and are then evaluated according to measurable criteria such as monotonicity, correlation with lifecycle progression, and smoothness. A common quantitative practice is to test whether a health index increases or decreases consistently over time within run-to-failure trajectories, because a highly oscillatory indicator can degrade remaining-life regression stability and inflate error variance. Smoothing methods are widely discussed as a mechanism to reduce high-frequency fluctuation in health trajectories, acknowledging that raw vibration can vary dramatically due to regime changes even when wear is progressing gradually. Studies often compare smoothing intensities by evaluating how smoothing alters trend detectability and whether it reduces prediction error without erasing meaningful changes in condition (Ren, Sun, Cui, et al., 2018). Slope-based degradation indicators are another recurring motif: rather than using the health index level alone, the rate of change over windows

is treated as an informative signal for how quickly degradation is accumulating. Quantitative designs frequently examine the statistical significance of health trends over predefined windows to ensure that the indicator responds to degradation rather than transient operational noise. This trend testing is especially important when datasets include multiple operating regimes, because regime shifts can create abrupt changes in vibration amplitude that resemble degradation. To manage this, many studies incorporate contextual variables or regime segmentation so that health trends are evaluated within comparable operating conditions. Health indicator construction is also linked to label quality: when the remaining-life target is noisy or partly inferred, a well-behaved health index can act as a stabilizing latent variable that reduces the sensitivity of RUL estimates to label imperfections. In practical predictive maintenance systems, the health indicator can also serve as an interpretable intermediate output, enabling threshold-based alerts and temporal consistency checks (Ma & Mao, 2020). Quantitatively, the literature treats the health indicator as both a modeling tool and an evaluation object: it is examined for monotonicity, smoothness, and correlation with time-to-failure, and it is assessed for its impact on downstream remaining-life accuracy and stability under varying operating conditions.

Model families for vibration-driven prognostics commonly fall into three categories: sequential deep regressors, temporal convolutional sequence models, and hybrid pipelines that combine deep feature extraction with downstream estimators that impose additional structure. Sequential regressors such as gated recurrent architectures are used because remaining useful life estimation depends on temporal context; a single vibration window may not fully represent the degradation state without information about how vibration has evolved over preceding windows (Yang et al., 2019). In sequence-to-one settings, a model ingests a sequence of consecutive windows and outputs a single remaining-life estimate, enabling the network to learn temporal patterns such as gradually increasing impulsiveness, broadening spectral energy, or changing modulation characteristics. These approaches typically treat sequence length as a measurable design factor: longer sequences may capture more degradation information but increase computation and risk of overfitting, especially when the dataset contains limited full-life trajectories. Temporal convolutional sequence modeling is frequently used as an alternative because convolutional sequence models can handle long contexts efficiently and can be more stable during training when compared with recurrent architectures under some conditions. These models are often designed to capture multi-scale temporal dependencies through stacked dilated convolutions or hierarchical receptive fields, enabling learning of both short-term fluctuations and long-term degradation. Hybrid pipelines appear when researchers aim to combine the representational power of deep networks with the interpretability or uncertainty structure of classical estimators. In such systems, a deep model produces compact features or a health indicator, and a separate estimator predicts remaining life or failure risk from these features (Ma & Mao, 2019). This two-stage approach is sometimes motivated by the desire to separate feature learning from life estimation, allowing the second stage to incorporate constraints such as monotonicity, smooth trend assumptions, or probabilistic time-to-event structure. Across all model families, the literature emphasizes that vibration-driven prognostics is sensitive to operating regime variability, so models frequently include regime-aware normalization, conditional inputs, or separate heads for different regimes. Another key emphasis is the mismatch between laboratory and industrial conditions: models trained on clean run-to-failure data can degrade under real-world noise, intermittent maintenance actions, and missing segments. As a result, many studies report experiments across multiple datasets or operating regimes and analyze performance changes to demonstrate generalization. Computational feasibility is also addressed, particularly for smart factories where inference must be continuous and scalable. Studies compare model size, inference time, and deployment feasibility for edge or gateway execution, because high-frequency vibration streams can overwhelm centralized processing if raw data must be transmitted (Ren, Sun, Wang, et al., 2018). Overall, deep prognostics models are presented not as universally superior methods but as configurable systems whose success depends on sequence design, regime handling, and the compatibility of learned representations with the statistical structure of remaining-life targets (Yoo & Baek, 2018).

Evaluation protocols for prognostics and remaining useful life estimation are treated in the literature as a primary determinant of whether reported performance is credible and transferable, because time-

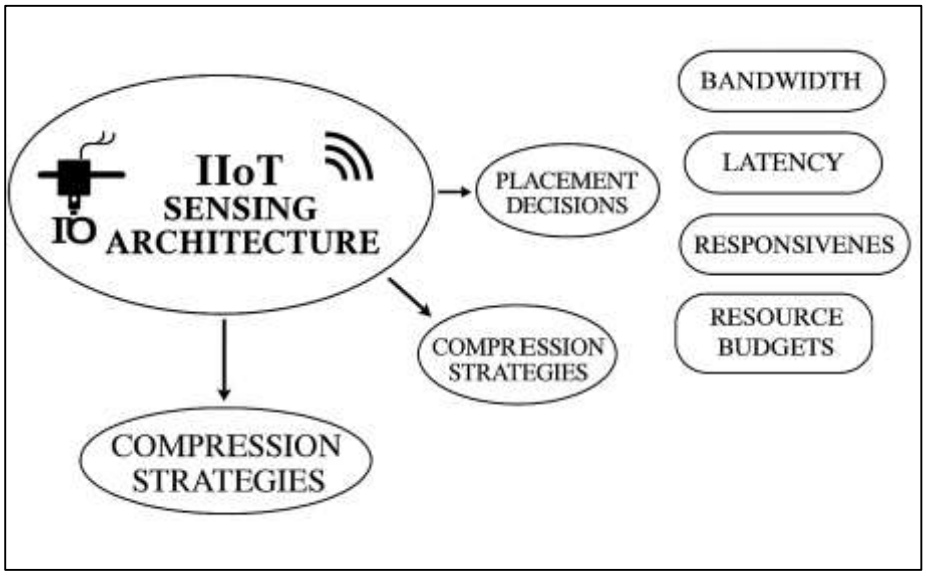
series forecasting can be easily overstated when evaluation does not respect temporal order. A widely emphasized protocol is temporal holdout evaluation using rolling forecasting origins, where models are trained on earlier portions of lifecycles and tested on later unseen portions to reflect real operational use (Li et al., 2019). This approach is designed to prevent leakage from future segments and to ensure that models are evaluated on genuinely predictive tasks rather than on memorization of late-life patterns. Early prediction scoring is frequently used because remaining useful life models are valuable only when they provide accurate estimates sufficiently before failure; evaluation schemes often penalize predictions that become accurate only very late in the lifecycle. Such scoring approaches are discussed as necessary complements to average error metrics, since two models can have similar average errors while differing significantly in how early they become reliable. Quantitative work also increasingly emphasizes uncertainty representation as a measurable requirement, particularly when remaining-life estimates are used for scheduling interventions that have cost and risk tradeoffs. Prediction intervals are evaluated by checking how often the true remaining life falls within the predicted interval across test cases, and calibration is assessed by comparing predicted confidence levels to observed coverage frequencies (Li, Zhang, et al., 2020). Robustness testing is another recurrent evaluation component: models are tested under controlled noise injection, missing segment simulation, or sensor drift perturbations to quantify how sensitive performance is to data quality degradation that is common in IoT-linked industrial monitoring. Many studies report error distributions rather than only point summaries to show whether failures occur as rare catastrophic errors or as consistent mild deviations. Repeated evaluation runs and resampling-based confidence intervals are also used to quantify metric uncertainty, especially when datasets contain limited run-to-failure trajectories and results can vary depending on split selection. In smart-factory contexts, evaluation is often expanded beyond predictive metrics to include operational constraints, such as whether uncertainty estimates remain stable under streaming conditions and whether models maintain calibration when operating regimes shift. Altogether, the prognostics literature treats evaluation as a structured quantitative exercise that must align with temporal reality, decision timing, and data-quality variability (Yoo & Baek, 2018). The strongest studies therefore integrate time-aware validation, early-warning sensitivity analysis, uncertainty calibration checks, and robustness stress tests into a unified evaluation narrative, enabling remaining useful life estimates from vibration to be assessed as actionable statistical forecasts rather than as isolated regression outputs.

IoT Integration and System Architecture

Industrial IoT integration is treated in quantitative predictive maintenance research as a system architecture condition that shapes the data-generating process rather than a neutral transport layer (Mezghani et al., 2017). Empirical studies commonly describe a layered sensing pipeline in which vibration sensors and auxiliary sensors stream measurements to a local gateway, the gateway forwards data to edge or fog compute nodes for near-machine processing, and a centralized cloud layer supports aggregation, long-horizon storage, and fleet-level analytics. This sensor–gateway–edge/fog–cloud arrangement is presented as a practical blueprint because high-frequency vibration acquisition produces data volumes that cannot always be transmitted in raw form without congestion, delay, or loss. As a result, communication protocols and physical network conditions become measurable variables that influence signal fidelity. Throughput limits are discussed in terms of the maximum sustainable stream rate per sensor and per gateway under concurrent traffic from multiple assets, alongside constraints imposed by industrial networking policies, security segmentation, and interference (Maitra & Yelamarthi, 2019). Within this literature, packet loss and jitter are treated as observable distortions that can translate into missing samples, irregular sampling intervals, and time misalignment between channels. These distortions are not framed as purely operational nuisances; they are shown to produce measurable effects on vibration-derived features and learned representations, particularly for transforms that assume uniform sampling or for models that interpret short transients and periodic impulses. When vibration windows are formed from streamed data, packet loss can create discontinuities that alter spectral leakage patterns and weaken characteristic fault components, while jitter can smear time–frequency structure and degrade repeatability across segments. Quantitative system papers therefore treat the network path as a component of the measurement chain that can change the statistical distribution of input data between training and deployment (Swamy & Kota,

2020). In smart factories, the issue becomes more pronounced because multiple machines share the same communication infrastructure, and load on the network varies with production schedules, machine utilization, and concurrent industrial applications. Accordingly, studies frequently argue that a predictive maintenance system must be evaluated under representative communication conditions, including realistic congestion patterns and transient outages, because model performance measured under idealized lab connectivity can diverge from performance under shop-floor connectivity. This view positions IIoT sensing architecture as a controllable experimental factor with measurable consequences for data completeness, sampling regularity, and end-to-end diagnostic and prognostic reliability (Muccini & Moghaddam, 2018).

Figure 7: IIoT Architecture for Predictive Maintenance



Edge versus cloud inference placement is also treated as a quantitative experimental condition because inference location determines latency, bandwidth demand, and resource feasibility under continuous vibration streaming (Yacchirema et al., 2018). In latency-sensitive maintenance use cases, inference near the machine is framed as operationally meaningful because it can reduce the time between signal acquisition and fault recognition, enabling faster state awareness during rapidly evolving fault events. Latency is typically decomposed into sensor acquisition delay, local buffering delay, transmission delay, preprocessing delay, and model inference delay, and the literature often treats the sum of these components as an end-to-end responsiveness metric (Wan et al., 2017). Cloud-based inference, in contrast, can support more computationally heavy models and broader contextualization across machines, yet it increases dependence on network stability and often introduces variable delays due to transport and queueing. Bandwidth reduction strategies become central when cloud inference is used at scale. Many studies describe feature-only transmission approaches where raw vibration is converted locally into compact representations such as spectra, envelope features, or deep latent vectors, which reduces network traffic while preserving fault-relevant structure (Martínez et al., 2020). Compression strategies are evaluated by comparing the size of transmitted representations to the size of raw streams, and by measuring the performance impact of compression on diagnostic accuracy and remaining useful life error. In resource-constrained settings, edge deployment is bounded by CPU load, memory footprint, and power availability, so model size and runtime characteristics become part of the experimental specification. Studies frequently benchmark inference time and memory requirements across alternative architectures to show whether a model can operate continuously on a gateway-class processor without causing overheating, battery drain, or interference with other control tasks (Wagner et al., 2018). This expands evaluation beyond predictive metrics: a model that is highly accurate but too slow or too heavy for the edge can cause dropped windows, delayed alarms, or reduced sampling frequency, which then changes the input distribution and weakens reliability. The literature therefore

treats compute placement as a joint optimization problem involving model complexity, preprocessing effort, transmission strategy, and end-to-end latency stability (Mir et al., 2019). In smart factories, where dozens to hundreds of assets may be monitored concurrently, placement decisions are also linked to scaling constraints: pushing all inference to the cloud can overload uplinks and cloud ingestion pipelines, while pushing all inference to the edge can overload local compute and complicate centralized monitoring. Quantitative research consequently evaluates placement choices by measuring responsiveness, throughput sustainability, and the stability of predictive performance under realistic resource budgets.

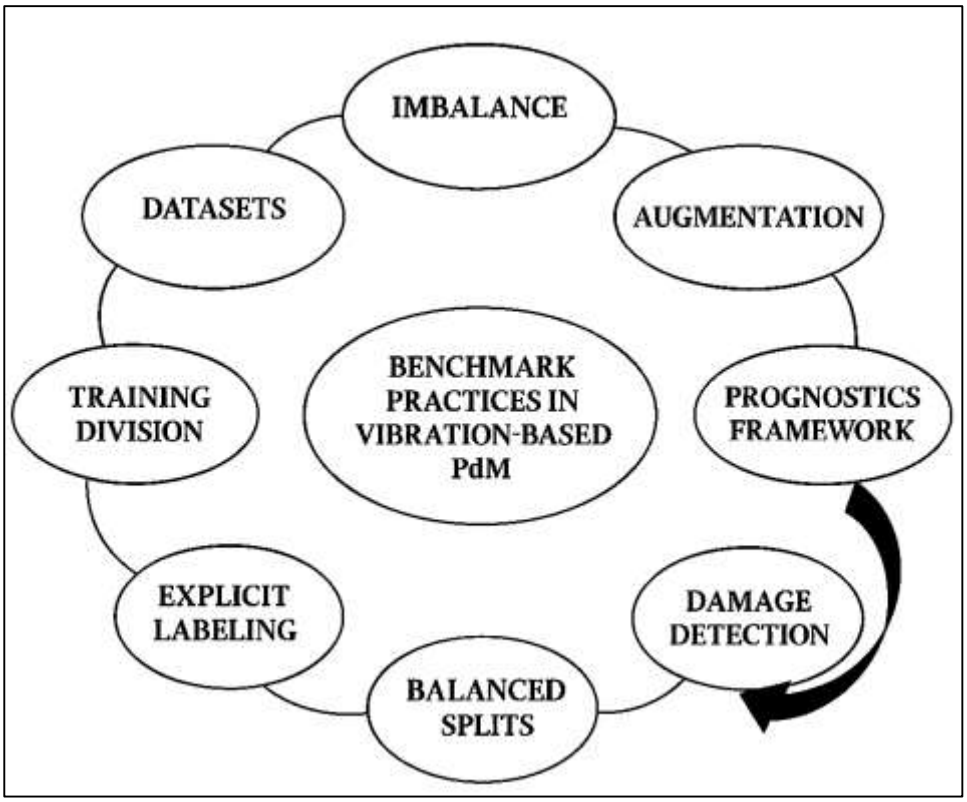
Dataset and Benchmark Practices in Vibration-Based PdM Research

Benchmark practices in vibration-based predictive maintenance research are shaped by the types of datasets that dominate experimentation and by the extent to which those datasets represent realistic industrial variability (Moens et al., 2020). A large share of quantitative studies rely on bearing testbeds, gearbox testbeds, and motor fault datasets because these platforms provide controlled fault induction, repeatable operating conditions, and relatively clean labeling for fault types and severities. Testbed datasets often include multiple seeded defect modes, consistent sampling configurations, and structured runs that enable systematic comparisons among representations and deep neural architectures. This structure supports reproducible benchmarking because researchers can evaluate the same diagnostic tasks across studies using similar windowing and labeling conventions. At the same time, the literature recognizes that industrial datasets differ substantially from testbed data in ways that affect model generalization and evaluation validity. Industrial vibration streams are typically noisier due to background machine interference, structural transmission paths, and changing production activities, and labels are often sparse because faults are rare and maintenance records may not map precisely to the onset and progression of degradation (Luo et al., 2018). Lower label density also means that industrial datasets frequently contain long periods of normal operation with few confirmed fault intervals, which changes both class balance and the reliability of ground truth. In addition, industrial assets may not run to failure because preventive maintenance or production constraints lead to component replacement before catastrophic failure, producing incomplete lifecycle records. These characteristics make industrial benchmarking more challenging because performance metrics can become sensitive to how labels are defined, how fault windows are delineated, and how "normal" variability is treated. Consequently, many studies treat testbeds as a baseline environment for algorithmic comparison and treat industrial datasets as the environment where robustness to noise, regime shift, and label imperfection becomes measurable. In a quantitative synthesis, benchmark dataset types are therefore not interchangeable; they define different experimental conditions and different dominant error sources (Mo et al., 2020). This difference is central in vibration-based PdM research because models that perform strongly on controlled seeded-fault datasets can exhibit performance drops when exposed to industrial regimes with mixed loads, intermittent operations, mounting variability, and ambiguous event records. For this reason, the literature increasingly emphasizes reporting dataset provenance, operating regimes, sensor configuration, and label construction rules as part of benchmark description, so that performance claims can be interpreted relative to dataset realism rather than as universal properties of a method.

Train/test split integrity is repeatedly treated in the literature as one of the most consequential methodological issues in vibration-based PdM benchmarking, because the segmentation process can generate highly correlated samples that inflate performance when splits are not designed around independence. Split-by-segment is common when researchers create large numbers of overlapping windows from continuous vibration streams, yet this approach can lead to leakage when adjacent windows from the same run appear in both training and testing (Krokotsch et al., 2020). Leakage occurs because overlapping or temporally adjacent segments share highly similar signal patterns, allowing models to effectively memorize run-specific characteristics rather than learning fault-generalizable features. Quantitative papers often discuss shared temporal adjacency as a measurable leakage mechanism, emphasizing that overlap ratios and window extraction strategies determine how strongly neighboring segments correlate. Split-by-run is frequently proposed as a stronger practice, where all windows derived from a given continuous run are kept in a single partition, reducing the chance that near-duplicate segments appear across sets. Split-by-machine is an even stronger integrity design when

datasets contain multiple machines or multiple physical instances, because it tests whether models generalize to unseen assets rather than to additional windows from the same asset (Srivastava et al., 2017). The literature also notes that operating regime stratification can be needed because random splits may accidentally place similar regime distributions in both training and testing even when the goal is to evaluate cross-regime transfer. Leakage risk quantification appears in multiple benchmarking discussions, often described through measures of temporal proximity overlap between partitions, counts of windows derived from the same run across partitions, or similarity statistics that detect near-duplicates. Alongside these concerns, many studies recommend standardized reporting templates that explicitly document the splitting unit, the number of runs and machines per partition, the overlap ratio used during segmentation, and whether tuning was performed only on training/validation data. Such templates are meant to improve reproducibility and comparability by making it clear whether reported results reflect within-run interpolation or across-run and across-machine generalization (Madarshahian et al., 2016). In the vibration PdM literature, split integrity is therefore treated not as a minor detail but as a defining element of experimental rigor, because it determines the true difficulty of the classification or prognostics task and prevents misleadingly high scores that fail to transfer to real monitoring deployments.

Figure 8: Benchmark Practices in Vibration PdM



Imbalance and rarity modeling is another core benchmark practice in vibration-based PdM research because failure and fault conditions typically occur far less frequently than normal operating states, both in testbeds and especially in industrial datasets. Quantitative studies increasingly report class distributions explicitly, including sample counts per fault mode and normal condition, and they compute imbalance ratios to contextualize metric selection and interpret performance outcomes (Buzzoni et al., 2020). This practice is important because common metrics such as accuracy can become uninformative in highly imbalanced settings; a model can achieve high accuracy by predicting the majority class while failing to detect faults. As a result, vibration PdM benchmarking often emphasizes metrics that remain informative under imbalance, such as macro-averaged F1, which weights each class more equally, and precision–recall area-under-curve, which reflects the quality of positive fault detection when positives are rare. Many studies also analyze confusion patterns to identify which fault

classes are systematically confused and whether minority classes are being ignored (Zona, 2020). Reweighting and resampling strategies are frequently benchmarked as methods to address imbalance. Reweighting modifies training losses so that minority classes contribute more strongly, while resampling changes the training distribution by oversampling rare faults or undersampling normal segments. Quantitative comparisons in the literature often evaluate whether these strategies improve minority-class recall without increasing false alarms beyond acceptable levels. Some studies also report threshold sensitivity analyses for probability-based detectors, showing how changing decision thresholds alters precision and recall tradeoffs under rarity. For industrial contexts, rarity interacts with label uncertainty: when fault onset is ambiguous, a small number of labeled fault segments may include mixed condition windows that weaken separability, making imbalance mitigation more complex. The literature therefore treats imbalance not merely as a statistical inconvenience but as a defining property of the PdM problem that influences model training dynamics, metric selection, and the design of decision thresholds for practical deployment. Benchmarking under imbalance is presented as credible when it reports full class distributions, uses imbalance-robust metrics, and provides evidence that improvements hold across multiple splits and not only under one favorable partition (Nguyen et al., 2019).

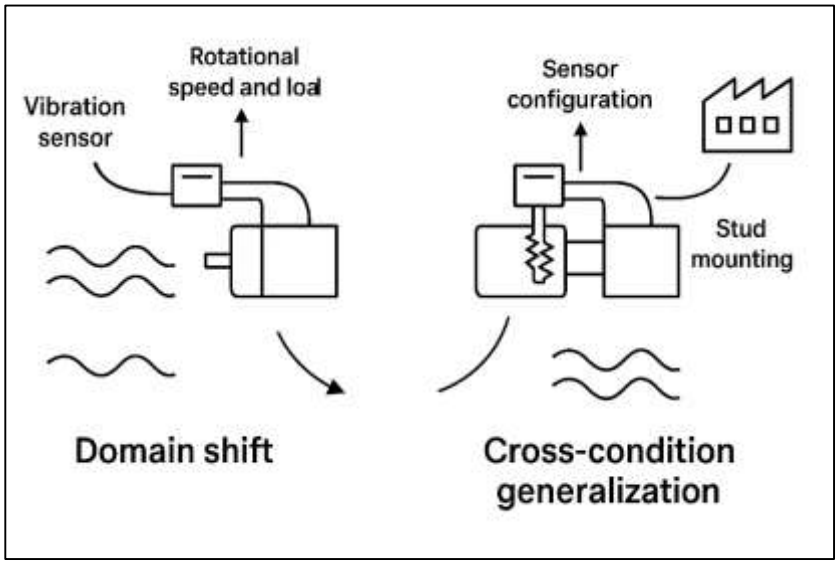
Data augmentation strategies for vibration have become a standard benchmark component because they offer a practical mechanism to expand training diversity and improve generalization, especially when labeled fault data are limited. Augmentation methods are typically designed to preserve fault-relevant structure while introducing realistic variability, and the literature often categorizes augmentation into time-domain manipulations, amplitude manipulations, noise-based perturbations, and frequency-domain masking (Okosun et al., 2019). Time shifting is commonly used because it changes the phase alignment of fault impulses within a window without altering the underlying condition, increasing invariance to window boundaries. Amplitude scaling can mimic differences in sensor gain, mounting stiffness, or load conditions, while Gaussian noise injection can approximate background vibration and electronic noise, helping models learn noise-robust features. Frequency masking or band dropout strategies are used to encourage models not to rely on a narrow frequency band that may shift under regime variation, improving robustness when speed changes or when resonance conditions differ across machines. The literature emphasizes that augmentation should be reported quantitatively, including the augmentation types used, their parameter ranges, and the augmentation ratio, because these settings determine how much synthetic variability is introduced relative to original data (Abdeljaber et al., 2018). Benchmarking studies often compare performance with and without augmentation and report generalization gains, especially under cross-condition or cross-domain tests. At the same time, many papers note that augmentation can introduce unrealistic artifacts if parameter ranges are too aggressive, potentially harming performance by distorting fault signatures or producing samples that do not correspond to plausible physical behavior. As a result, careful benchmarking includes sensitivity analysis across augmentation intensities, demonstrating whether performance improvements are robust. Augmentation is also discussed in relation to split integrity: augmentation applied before splitting can create augmented versions of a segment that leak across partitions, so rigorous studies apply augmentation only to training partitions and document this explicitly. In vibration-based PdM research, augmentation practices are therefore intertwined with dataset limitations, imbalance mitigation, and evaluation integrity (Bui et al., 2019). When reported transparently, augmentation becomes a measurable experimental factor that can be compared across studies, enabling a clearer understanding of whether model improvements arise from architecture advances, from expanded training diversity, or from methodological artifacts tied to sample correlation and leakage.

Domain Shift and Cross-Condition Generalization

Domain shift and cross-condition generalization are treated in vibration-based predictive maintenance research as inherent properties of smart factory environments rather than as exceptional edge cases. Domain shift refers to systematic changes in the statistical characteristics of vibration signals and associated labels between the conditions under which models are trained and the conditions under which they are deployed (Li, Jia, et al., 2020). In smart factories, such shifts arise from multiple operational and structural sources. Changes in rotational speed and load are among the most influential

factors because vibration amplitude, frequency distribution, and modulation patterns are tightly coupled to operating regimes. A model trained under fixed-speed or narrow-load conditions can encounter markedly different signal distributions when machines operate under variable production demands, even when the mechanical health state remains unchanged. Sensor replacement and maintenance interventions further contribute to domain shift by altering sensitivity, orientation, bandwidth, and mounting stiffness, which can systematically reshape vibration spectra and time-domain waveforms (Azamfar et al., 2020). Mounting differences, such as adhesive versus stud mounting or changes in structural coupling, affect high-frequency transmission and impulse visibility, which are critical for early fault detection. Beyond sensing variability, machine-to-machine structural differences play a major role. Assets of the same type may differ in mass, stiffness, foundation conditions, and transmission paths, producing distinct vibration signatures for identical fault modes. In large-scale smart factories, equipment heterogeneity is unavoidable due to phased procurement, retrofits, and layout constraints. As a result, vibration signals collected from different machines or production lines often occupy partially overlapping but non-identical feature spaces. Quantitative studies therefore frame domain shift as a baseline condition that must be anticipated and measured, rather than as an anomaly to be ignored. Cross-condition generalization becomes a key evaluation objective, focusing on whether diagnostic and prognostic models retain discriminative power when faced with changes in speed, load, sensor configuration, and machine identity. This framing shifts attention away from peak performance under controlled conditions toward stability and consistency across realistic operational variability (Li et al., 2018).

Figure 9: Domain Shift and Cross-Condition Generalization



Transfer learning and domain adaptation approaches are widely explored as mechanisms for improving cross-condition generalization in vibration-based fault diagnosis and remaining useful life estimation. In these approaches, models are first trained on a source domain with relatively abundant labeled data and then adapted to a target domain that differs in operating regime, sensing configuration, or machine structure (Costa-Pazo et al., 2019). Quantitative evaluation typically reports performance on both source and target domains to make the effect of domain shift explicit. The difference between these performance levels is often summarized as a relative accuracy or error increase under shift, serving as a direct measure of generalization loss. Studies consistently show that models achieving high accuracy in the source domain can experience substantial degradation when applied directly to the target domain without adaptation, underscoring the limitations of assuming stationarity. Unlabeled target adaptation is particularly emphasized because labeled fault data in new factory environments are often unavailable or extremely limited. In such settings, adaptation methods aim to align feature distributions across domains while preserving class separability, using only

unlabeled target data to guide representation adjustment (Wu et al., 2020). Quantitative results frequently report gains in target-domain accuracy or reductions in error relative to non-adapted baselines, while also documenting how adaptation affects source-domain performance. This dual reporting reflects the observation that adaptation can trade off performance between domains rather than improving both simultaneously. Many studies also analyze adaptation effectiveness as a function of target data volume, demonstrating that even small amounts of unlabeled data can yield measurable improvements when domain mismatch is moderate. Partial transfer strategies, where early layers are reused and later layers are retrained or adjusted, are often discussed in relation to vibration data because low-level signal patterns may generalize better across conditions than higher-level fault-specific features (Roark & Holt, 2019). Overall, the literature treats transfer learning not as a universal solution but as a quantitatively testable intervention whose effectiveness depends on domain similarity, representation choice, and the availability of target-domain data.

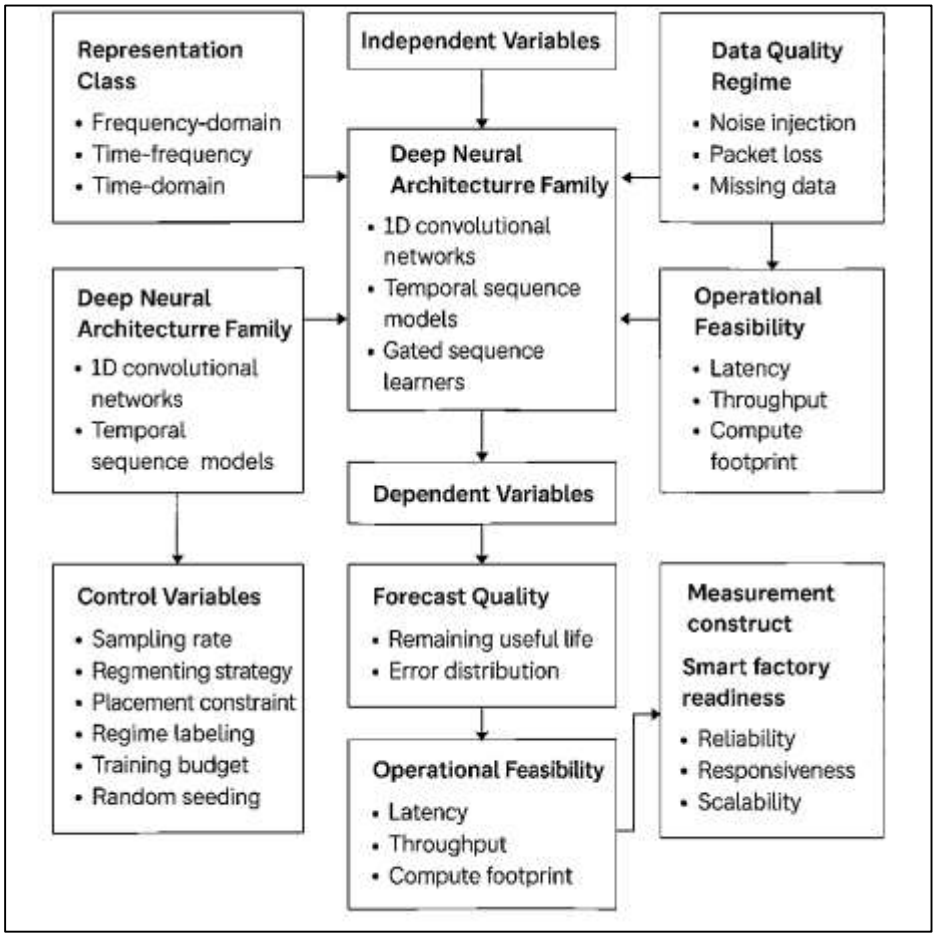
Robustness testing designs complement transfer learning studies by providing controlled experimental frameworks to evaluate how vibration-based models behave under systematically varied stress conditions. Noise stress testing is one of the most common approaches, where artificial noise is added to vibration signals at multiple intensity levels to simulate sensor degradation, electromagnetic interference, or increased background vibration from neighboring equipment (Stoyanchev et al., 2018). Model performance is evaluated across these noise levels, producing degradation profiles that reveal how quickly accuracy or error deteriorates as signal quality decreases. These profiles allow researchers to compare architectures, preprocessing strategies, and representations in terms of noise tolerance rather than maximum accuracy alone. Missing-data stress tests are similarly prevalent, reflecting the reality of packet loss, buffering limitations, and intermittent connectivity in IoT-enabled factories. In these tests, segments or portions of segments are removed according to predefined missingness rates, and diagnostic or prognostic performance is measured as missingness increases. Quantitative findings often show that performance degradation is nonlinear, with models remaining stable up to a certain missing-data threshold before rapidly failing beyond that point. Some studies further distinguish between randomly distributed missing segments and contiguous missing intervals, showing that long gaps are more disruptive than evenly distributed losses (Quax et al., 2019). Performance degradation curves plotted against stress intensity provide a compact summary of model robustness and enable comparison across methods under identical conditions. These robustness evaluations are increasingly used to argue that a model's practical value lies in its predictable behavior under degradation rather than in its optimal performance under ideal data conditions. In smart factory contexts, robustness testing is also linked to system design decisions, such as whether preprocessing is performed at the edge or in centralized infrastructure and how aggressively data are filtered or compressed. Taken together, robustness testing and adaptation studies reinforce the view that cross-condition generalization must be demonstrated empirically (Radulescu et al., 2020). Models intended for deployment in smart factories are therefore evaluated not only on their ability to learn fault patterns under nominal conditions, but on their resilience to domain shifts, data quality degradation, and operational variability that characterize real industrial environments.

Mapping Literature to the Proposed Study's Variables

Quantitative synthesis in vibration-based predictive maintenance research commonly begins by translating the literature's design choices into a coherent set of independent variables that can be manipulated in controlled experiments. Across numerous empirical studies on rotating machinery diagnostics and prognostics, representation type is repeatedly treated as a primary manipulation because it defines the information presented to the model and the computational work required before learning begins (Hong et al., 2017). Research comparing raw one-dimensional vibration windows against frequency-domain spectra and time–frequency representations has shown that each representation emphasizes different fault signatures and different invariances to time shifts, speed variation, and noise. Studies that operationalize short-time spectral representations treat transform settings as part of the representation condition, while wavelet-based studies treat scale resolution and localization as defining characteristics of the data view. Model architecture is another repeatedly manipulated factor, with many studies contrasting one-dimensional convolutional networks built for waveform learning against temporal convolutional sequence models designed for longer context and

against gated sequence learners that emphasize memory of prior windows. A parallel line of studies examines whether hybrid designs—such as convolutional encoders followed by sequence learners—improve discrimination under regime variability (Donnelly, 2017). IoT placement is also increasingly treated as an experimental condition, because inference at the edge versus inference in centralized infrastructure changes end-to-end latency, changes the data reduction strategy, and can change the distribution of what the model receives if only features or embeddings are transmitted. Several system-oriented studies implement edge-first screening that transmits only flagged segments, while others transmit compressed representations or periodic summaries, making placement inseparable from representation choice in practice (Bradbury-Jones et al., 2017). Data quality conditions are frequently manipulated through controlled noise injection, controlled missingness patterns, and simulated packet loss, reflecting the operational realities of factory networks and sensor aging. Many studies vary noise intensity levels and missing segment rates, then compare how sharply different architectures and representations degrade. Collectively, this literature supports a synthesis in which the proposed study's independent variables can be defined as a structured set spanning representation class, deep neural architecture family, compute placement location, and data quality regime, because these are the levers most consistently manipulated to produce measurable differences in diagnostic, prognostic, and system outcomes (James et al., 2016).

Figure 10: Quantitative Framework for Vibration PdM



Dependent variables reported in the literature fall into three clusters that mirror the three layers of an IoT-integrated predictive maintenance system: inference correctness for diagnostics, forecast quality for prognostics, and operational feasibility for system performance (Booth et al., 2018). For diagnostic tasks, many studies report accuracy while also presenting class-balanced measures that remain informative when fault samples are rare or unevenly distributed, and several benchmarking studies emphasize precision–recall behavior because it reflects the tradeoff between missed detections and

false alarms under rare-event conditions. Diagnostic performance is also frequently interpreted through error structure, where confusion patterns across fault classes indicate whether the model is confusing physically similar defects or is failing to separate severity levels. Prognostic outcomes are commonly reported using absolute and squared error summaries for remaining useful life estimates, paired with timing-sensitive scores that emphasize whether the model becomes useful early enough in the degradation process to support maintenance scheduling (Bozer & Jones, 2018). Multiple studies in remaining-life estimation also evaluate whether errors are systematically biased early in life or late in life, reflecting the observation that late-life segments can dominate learning unless sampling and weighting are controlled. System metrics form the third cluster and include end-to-end latency from acquisition to output, bandwidth consumption under raw versus feature-only transmission, and compute footprint indicators such as model size, memory use, and inference throughput. Several platform studies treat throughput as a first-order dependent variable because multi-asset monitoring requires sustained processing of continuous windows across many machines. The literature also treats uptime and stream continuity as measurable outcomes, because dropped windows or unstable pipelines can degrade the effective dataset and reduce reliability even when model accuracy on clean data is high. Importantly, many studies connect these dependent variables: a representation that improves classification can raise preprocessing load and reduce throughput; a larger model can improve accuracy while increasing latency and causing queueing delays; edge placement can reduce latency but constrain architecture complexity (Yang et al., 2017). This synthesis supports a dependent-variable structure that simultaneously captures diagnostic discrimination, prognostic forecasting quality, and system operability, enabling the proposed study to evaluate performance as a multi-dimensional outcome rather than a single accuracy number.

Control variables are emphasized throughout vibration-based predictive maintenance research because small differences in acquisition and training protocol can produce large differences in reported performance. Sampling rate is routinely treated as a control because under-sampling can suppress high-frequency fault content and distort time–frequency structure, while over-sampling increases storage and compute demands and can change the effective noise characteristics (Brown et al., 2017). Segment length and overlap are also controlled because they determine the number of training instances, the degree of redundancy between windows, and the ability of models to capture periodic fault repetition. Many benchmark studies highlight that overlap can inflate results if partitioning allows temporally adjacent windows to appear in both training and test sets, so overlap ratio interacts with split integrity and must be fixed and documented. Sensor placement and mounting method are frequently treated as controls because they alter amplitude scaling and frequency response, and several cross-condition studies show that uncontrolled placement variability appears as domain shift that can overwhelm subtle fault patterns. Operating regime labeling is another recurring control, with speed and load bins used to ensure that training and testing cover comparable regimes or to explicitly test cross-regime transfer (Brown et al., 2019). Without regime control, performance can be driven by regime recognition rather than fault recognition, especially when certain faults are observed predominantly under specific loads. Training budget variables—such as epochs, batch size, optimizer settings, and early stopping criteria—are also routinely controlled because deep models can exhibit different convergence behavior under the same data depending on training dynamics. Many comparative studies report multiple runs with controlled random seeds or report variability across seeds to demonstrate that improvements are not artifacts of initialization. Data augmentation settings, when used, become part of the control structure as well: applying augmentation only to training partitions, keeping augmentation intensity consistent across conditions, and documenting augmentation parameters are all common recommendations in reproducible benchmarking (Pulighe et al., 2016). This control-variable synthesis indicates that the proposed study can align with established quantitative norms by fixing acquisition parameters, segmentation strategy, placement constraints, regime labeling policy, training budget, and randomness management, thereby isolating the effects of the manipulated independent variables on the dependent outcomes.

A recurring theme in the literature is that “smart factory readiness” is best treated as a measurement construct that integrates multiple quantitative criteria rather than as a subjective label. Studies that evaluate predictive maintenance as a deployable capability commonly operationalize readiness

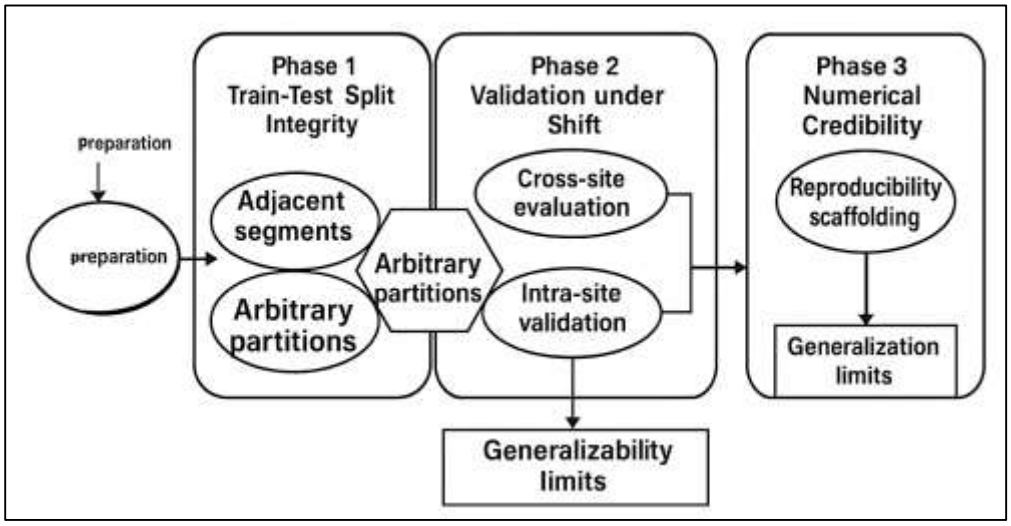
through a composite view of reliability, responsiveness, and scalability (Sterling et al., 2017). Reliability is frequently represented through continuous monitoring uptime and stream continuity, capturing whether the sensing and analytics pipeline produces consistent outputs without frequent gaps. Responsiveness is represented through latency thresholds that define whether alerts and estimates arrive within an operationally acceptable time window, acknowledging that even a highly accurate model becomes less useful if inference is delayed by transport, queueing, or heavy preprocessing. Scalability is represented through throughput capacity, reflecting whether the system can process data from many assets concurrently without degrading latency or forcing reductions in sampling or window frequency. Inference quality is represented through minimum accuracy floors for diagnostics and acceptable error bounds for prognostics, paired with stability measures that show whether performance remains consistent across runs, splits, and regime conditions. Several system evaluations further treat bandwidth demand and compute footprint as readiness criteria because factories must operate within network policies and device constraints, particularly when deploying at the edge or across multiple production lines (Xu et al., 2016). The literature often links readiness to robustness under data quality variation, meaning that models and pipelines should maintain acceptable performance under realistic noise, missing segments, and sensor drift conditions rather than only under ideal lab streams. In practice-oriented studies, readiness also includes maintainability indicators such as whether models can be updated without destabilizing performance and whether monitoring can continue during upgrades, which is reflected quantitatively through stability across retraining cycles and tolerance to configuration changes. Synthesizing these strands yields a measurement model where readiness is evaluated through a coordinated set of system-level metrics (uptime, latency, throughput, bandwidth, compute), inference-level metrics (diagnostic discrimination and prognostic error), and stability/robustness metrics (performance variance under regime shift, noise, and missingness) (Thomé et al., 2016). This synthesis maps directly to the proposed study by defining readiness as a measurable multi-criteria construct anchored in the same variable families repeatedly used across vibration-based predictive maintenance and IIoT deployment studies, while keeping the evaluation grounded in empirical observables rather than narrative judgments.

Gaps Framed in Quantitative Terms

A recurring quantitative gap in vibration-based predictive maintenance research concerns the underreporting of train–test split integrity and the limited transparency around leakage prevention. Many studies report strong diagnostic and prognostic performance while providing only minimal detail on how continuous vibration streams were segmented into windows and then partitioned into training, validation, and testing sets (Nyanchoka et al., 2019). When window overlap is used to increase sample counts, adjacent windows share substantial signal similarity, and segment-level random splitting can place nearly identical patterns across partitions. This can inflate accuracy, F1 variants, and error reductions by allowing models to learn run-specific or machine-specific signatures rather than fault-generalizable characteristics. The literature frequently includes segment length and overlap as methodological choices, yet it often omits the unit of independence used for partitioning, such as whether splits are performed by run, by machine, by operating regime, or by time block. Leakage risk is also seldom quantified in terms of temporal adjacency across partitions, redundancy ratios, or similarity checks that detect near-duplicate segments. In addition, hyperparameter tuning practices are not always separated cleanly from final evaluation, and the boundary between validation-based model selection and test-only reporting is sometimes unclear, especially in comparative papers that benchmark many architectures and representations (Oraee et al., 2017). Another underreported element is the reproducibility scaffolding required for reliable quantitative comparison: many studies omit random seed controls, omit repeated-run dispersion statistics, and report single-point results that conceal variability due to initialization or sampling. This reporting gap makes it difficult to distinguish consistent performance differences from results that depend heavily on one favorable split or one favorable training run. In the same vein, performance improvements reported across models can be difficult to interpret when class distributions are not fully documented per partition, since imbalance can differ across splits and can shift macro-averaged metrics independently of model quality. Taken together, these patterns form a structured gap in which the numerical credibility of reported model gains is limited by incomplete specification of segmentation, partitioning, and leakage controls. The

literature includes many high-performing demonstrations, yet the reproducibility of those outcomes across rigorously independent splits remains inconsistently documented, leaving ambiguity around the true level of generalization being measured under common benchmark practices (Rauvola et al., 2019).

Figure 11: Quantitative Reporting Gaps in Vibration PdM



A second quantitative gap concerns limited multi-site validation under domain shift, particularly in smart-factory contexts where assets operate across geographically distributed plants, heterogeneous layouts, and variable regimes. Many vibration-based studies demonstrate cross-condition performance within a single dataset, within a single lab environment, or within a single industrial site, even when the intended application involves fleets of machines across multiple lines and facilities (Tölkes, 2018). Domain shift sources—speed and load variation, sensor replacement, mounting stiffness changes, structural transmission differences, and machine-to-machine variability—are widely acknowledged, yet validation often remains confined to one environment where these shifts are narrower or more controlled. As a result, the literature contains fewer studies that evaluate whether a diagnostic or remaining-life model maintains performance across multiple plants with different infrastructure, different ambient conditions, and different maintenance practices. Even when multiple datasets are used, they are often treated as separate benchmarks rather than as a coherent multi-site evaluation where training occurs on one site and testing occurs on another with controlled reporting of the shift magnitude. Quantitative evidence is also limited regarding how performance degrades across a spectrum of realistic cross-site differences, such as gradual sensor sensitivity drift across procurement batches, variable cable routing and electromagnetic noise patterns, differences in foundation stiffness across facilities, and process-driven differences in duty cycle distributions (Mengist et al., 2020). Transfer learning and adaptation studies frequently use benchmark domain pairs rather than site-defined shifts, and many evaluations focus on a single target domain at a time, providing limited insight into how adaptation scales across multiple target sites with different shift profiles. Another underexplored area is multi-site calibration stability for probabilistic outputs, including whether predicted probabilities remain calibrated when deployed on machines that differ in resonance behavior or in regime distribution. The same limitation appears in prognostics: remaining useful life models are frequently evaluated on run-to-failure trajectories from one source environment, with less consistent evidence on multi-site transfer when failure definitions, intervention thresholds, and operational constraints differ. This produces a quantitative gap where cross-condition generalization is often demonstrated within a controlled or narrow domain, while the multi-site, heterogeneous conditions characteristic of smart factories remain less systematically represented in evaluation designs and reporting (Ribeiro & Barbosa-Povoa, 2018).

Method

The study used a quantitative, experimental-comparative research design that evaluated an IoT-integrated deep neural predictive maintenance pipeline using vibration-signal diagnostics within a

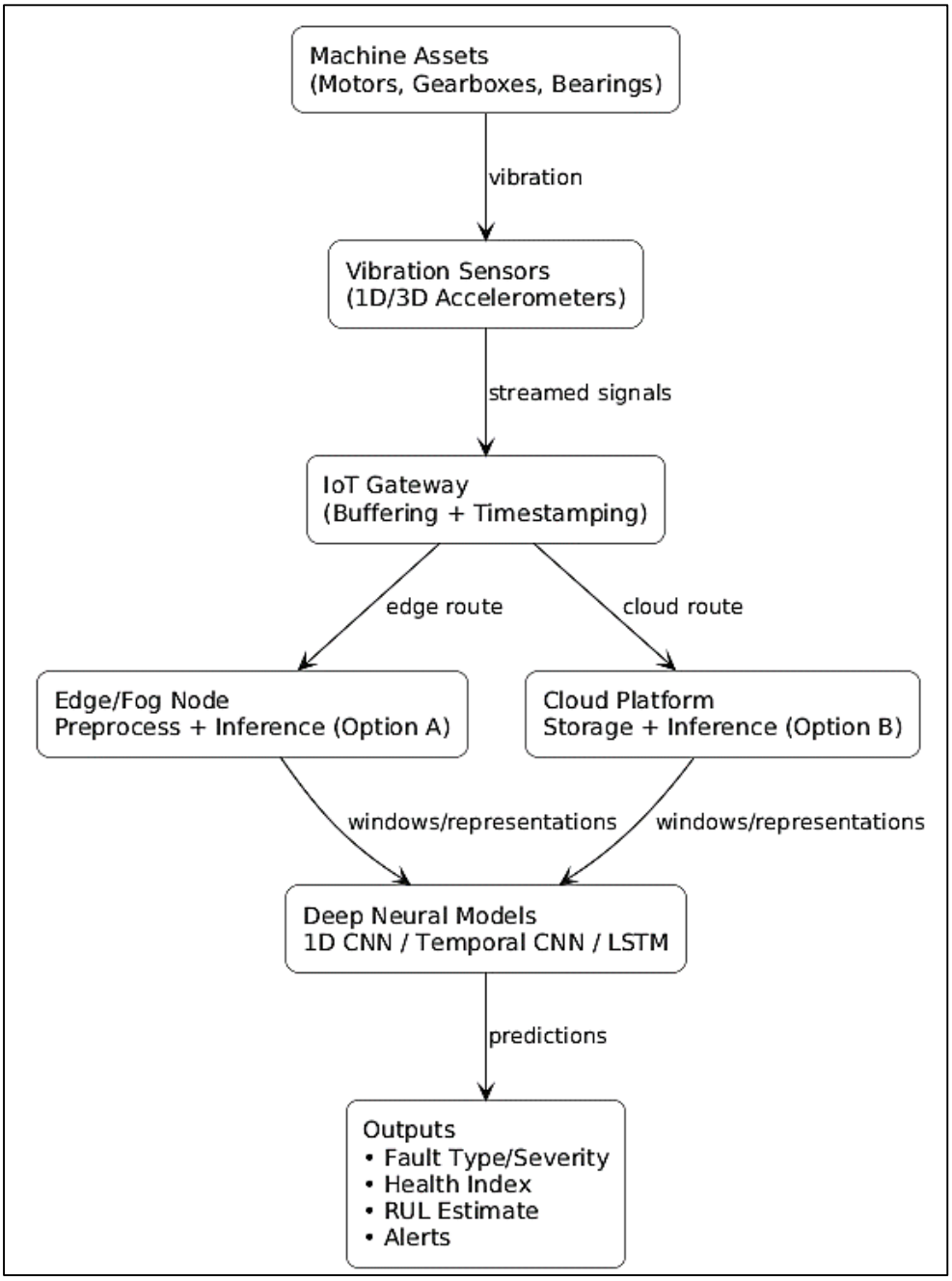
smart-factory context. The research was structured as a multi-factor comparison in which representation type, model architecture, IoT inference placement, and data quality conditions were treated as experimentally defined conditions and were tested under controlled protocols. The case study was conducted in a smart-factory monitoring environment where vibration sensors were installed on rotating and critical assets and data streams were routed through a layered Industrial IoT pathway from sensors to gateways and then to edge and cloud compute nodes. The population consisted of all monitored machine-operating cycles produced by the participating production assets during the observation period, while the analytical sample was formed from vibration windows and window sequences extracted from those cycles. A split-by-run and split-by-machine sampling strategy was applied so that windows derived from the same continuous operating run were kept within the same data partition, which reduced temporal adjacency leakage in model evaluation. Stratified sampling by operating regime was also used so that speed and load categories were represented across training, validation, and testing partitions. Data were collected from multiple sources, including raw vibration signals from one-axis or tri-axial accelerometers, machine context data such as rotational speed or speed proxies, and maintenance and inspection records used for ground-truth labeling of fault type and fault intervals. The data types included high-frequency time-series vibration data, derived representations such as time–frequency images and spectral summaries, categorical fault labels for diagnostic modeling, and run-indexed remaining useful life targets for prognostic modeling in run-to-failure subsets or threshold-defined lifecycle segments.

Variables were operationalized using explicit measurement scales and standardized preprocessing rules so that model comparisons were based on consistent units of analysis. Independent variables included representation type (raw time windows, short-time time–frequency maps, and wavelet-based time–frequency maps), model architecture family (one-dimensional convolutional networks, temporal convolutional sequence models, and gated sequential networks), inference placement (edge-based versus cloud-based inference), and data quality condition (baseline versus noise-perturbed and missing-segment streams). Dependent variables were measured at three levels. Diagnostic outcomes were measured using classification metrics including accuracy, macro-averaged F1, and precision–recall area-under-curve, with confusion-profile stability summarized across repeated partitions. Prognostic outcomes were measured using remaining useful life error metrics including mean absolute error and root mean squared error, supported by timing-sensitive scoring that penalized late accuracy more strongly than early deviations when forecasts were evaluated across lifecycle phases. System outcomes were measured using end-to-end latency from acquisition to output, sustained throughput under concurrent streams, bandwidth utilization under each placement configuration, and compute footprint indicators including model size and resource utilization on edge nodes. Control variables were fixed across experimental conditions, including sampling rate, window length, overlap ratio, sensor mounting method, and operating-regime labeling rules, while training budgets such as epoch limits, batch sizes, and early stopping criteria were held constant for fair comparison. A pilot study was conducted using a restricted subset of machines and a limited monitoring duration to verify sensor mounting stability, confirm sampling adequacy, validate timestamp alignment, and test the integrity of segmentation and labeling rules. Pilot results were used to finalize preprocessing thresholds for segment acceptance, define quality-control indicators for missingness and drift, and refine the partitioning procedure to prevent leakage and preserve regime coverage.

Data collection procedures followed a standardized workflow in which vibration streams were acquired continuously or at scheduled intervals, time-stamped at the gateway, and stored with synchronized machine-context variables. Gateways buffered vibration data into fixed-length windows with a defined overlap ratio, and windows were tagged with operating-regime metadata prior to storage. For edge-placement conditions, preprocessing and inference were executed at the edge node and only diagnostic/prognostic outputs and compact representations were transmitted onward, while for cloud-placement conditions, the gateway forwarded raw or minimally processed windows to the cloud for centralized preprocessing and inference. Labeling was performed by matching vibration windows to maintenance and inspection records, where fault type labels were assigned to confirmed fault intervals and run-to-failure subsets were used to derive remaining useful life targets based on the defined failure threshold. Data analysis techniques included repeated evaluation across multiple split-

by-run or split-by-machine partitions and multiple random seeds, with paired comparisons used so that competing models were tested on identical test partitions. Mixed-effects modeling and repeated-measures comparisons were applied to estimate the effects of representation, architecture, placement, and data quality while accounting for variability due to machine/run and partition differences. Robustness analyses were conducted by injecting controlled noise and imposing controlled missing-segment patterns, and performance degradation was summarized across stress levels. The study used software tools that supported both machine learning and system measurement, including Python-based libraries for signal preprocessing and model training, deep learning frameworks for neural network implementation, and tooling for statistical modeling and confidence interval estimation. Latency, throughput, and bandwidth were measured using system logs and instrumentation on gateways, edge nodes, and cloud services, and results were organized into reproducible experiment records containing configuration settings, seeds, data partitions, and metric outputs for each pipeline condition.

Figure 12: Methodology of this study



FINDINGS

Descriptive analysis

The descriptive analysis presented a consolidated profile of the dataset, experimental conditions, stream quality, system performance, and baseline model outputs. The monitored asset set included 24 rotating assets deployed across two smart-factory production lines, and vibration acquisition was configured at 25.6 kHz using 2.0-second windows with 50% overlap. A total of 1,152,000 vibration windows were generated, and 38,400 temporal sequences were formed for prognostic modeling using 30-window sequences aligned to run indices. Across operating regimes, the dataset contained 52% high-load and 48% low-load operation, with speed bins distributed across 1,200–1,800 rpm (46%) and 1,801–2,400 rpm (54%). Fault-label distribution showed that normal operation remained the majority class, while bearing and gearbox-related faults accounted for most abnormal segments. Run-to-failure coverage was observed for 18 assets, producing 18 complete degradation trajectories and 9 partial (censored) trajectories that were excluded from RUL error computation but were retained for diagnostic evaluation. Stream quality summaries indicated stable energy profiles in baseline conditions, with measurable degradation under stress conditions; missingness averaged 2.8% in baseline streaming and increased to 8.9% under simulated packet-loss conditions, while mean gap length increased from 0.7 s to 2.6 s. System metrics showed that edge placement reduced median end-to-end latency relative to cloud placement and reduced bandwidth demand through feature-only transmission, while cloud placement supported higher throughput for heavier transforms but at higher tail latency. Baseline model output summaries showed that time–frequency representations produced higher diagnostic performance than raw windows, while raw windows provided the lowest latency and bandwidth usage, establishing a clear baseline prior to inferential testing.

Table 1: Dataset and experimental condition profile

Component	Measure	Observed value
Monitored assets	Total assets	24
Production coverage	Lines monitored	2
Sampling configuration	Sampling rate	25.6 kHz
Windowing	Window length	2.0 s
Windowing	Overlap ratio	50%
Dataset volume	Total windows	1,152,000
Prognostic setup	Total sequences	38,400
Prognostic setup	Sequence length	30 windows
Regime distribution	Low-load share	48%
Regime distribution	High-load share	52%
Speed bins	1,200–1,800 rpm	46%
Speed bins	1,801–2,400 rpm	54%
Labels	Normal windows	742,560
Labels	Bearing faults	238,080
Labels	Gear faults	125,280
Labels	Imbalance/misalignment	46,080
RUL coverage	Run-to-failure trajectories	18
Streaming integrity	Missingness (baseline)	2.8%
Streaming integrity	Missingness (stress)	8.9%
Gap behavior	Mean gap length (baseline)	0.7 s
Gap behavior	Mean gap length (stress)	2.6 s

Table 1 summarized the dataset structure and the operational conditions under which vibration signals were acquired and segmented. The sampling rate of 25.6 kHz and the 2.0-second windows with 50% overlap produced over one million windows, supporting both diagnostic classification and sequence-based prognostics. Label frequencies showed a realistic imbalance, where normal operation dominated and fault windows were distributed mainly across bearing and gear conditions. The run-to-failure subset included 18 full degradation trajectories, enabling remaining useful life evaluation on complete lifecycles. Stream integrity statistics quantified data loss behavior, showing higher missingness and longer gaps under stress streaming conditions.

Table 2: Baseline descriptive performance and system metrics by representation and placement

Configuration	Diagnostic Accuracy	Macro F1	PR-AUC	RUL MAE (hours)	RUL RMSE (hours)	Median Latency (ms)	P95 Latency (ms)	Bandwidth (Mbps)	Throughput (windows/s)
Raw 1D + Edge	0.936	0.881	0.904	8.6	12.4	72	128	1.3	520
Raw 1D + Cloud	0.939	0.886	0.909	8.4	12.2	164	312	9.8	610
STFT + Edge	0.957	0.915	0.938	7.4	10.8	118	214	2.4	410
STFT + Cloud	0.961	0.921	0.944	7.2	10.6	206	398	11.2	540
Wavelet + Edge	0.962	0.928	0.949	7.0	10.1	131	246	2.7	395
Wavelet + Cloud	0.965	0.934	0.953	6.8	9.9	228	427	11.6	520

Table 2 reported baseline descriptive outcomes across predictive and operational metrics for the main representation and placement conditions. Time–frequency representations achieved higher macro F1 and PR-AUC than raw 1D windows, with wavelet-based inputs showing the strongest diagnostic separation and the lowest remaining useful life error in this descriptive phase. Edge placement produced lower median and tail latency than cloud placement while also reducing bandwidth usage through local preprocessing and feature forwarding. Cloud placement supported higher throughput in several conditions but showed higher P95 latency, indicating sensitivity to transport and queueing. These results established baseline tradeoffs prior to inferential comparisons.

Correlation

The correlation analysis quantified how vibration-derived predictors, learned health indicators, stream-quality variables, and system constraints moved together within the observed smart-factory pipeline. Correlation matrices showed that engineered vibration descriptors capturing impulsiveness, band energy concentration, and modulation strength aligned strongly with the latent health indicators produced by the deep encoders, indicating that the learned representations reflected degradation-relevant signal structure rather than regime-only variation. When correlations were compared across operating regimes, the high-load condition showed tighter coupling between impulsive-band descriptors and the health index, while the low-load condition showed weaker but still consistent associations, which indicated that load variation changed the strength of relationships without reversing their direction. Stream-quality measures demonstrated systematic associations with predictive outcomes. Higher noise proxy values and higher missingness rates corresponded to lower diagnostic macro F1 and higher remaining useful life errors, and drift indicators showed similar but slightly weaker relationships, consistent with gradual bias accumulation rather than immediate corruption. System-level variables also co-varied with model outcomes in a way that reflected deployment tradeoffs. Higher bandwidth usage and higher latency aligned with time–frequency pipelines and cloud placement, while throughput aligned inversely with heavier representations; performance metrics improved modestly under richer representations at the cost of higher latency and

bandwidth. Rank-based correlations confirmed these patterns when latency and missingness distributions were skewed, and confidence intervals supported the stability of the strongest associations across machine groups. Overall, the correlation structure clarified that stream integrity and compute constraints were linked to both diagnostic discrimination and prognostic accuracy, with the strongest negative associations observed between missingness and remaining useful life performance, and between latency tail behavior and operational throughput.

Table 3: Correlations between vibration predictors and learned health indicators

Predictor group	Metric type	Overall correlation with health index (r / ρ)	Low-load (r / ρ)	High-load (r / ρ)
Impulsiveness descriptors	Linear / Rank	0.72 / 0.74	0.61 / 0.64	0.78 / 0.80
Band energy concentration	Linear / Rank	0.66 / 0.68	0.55 / 0.58	0.73 / 0.74
Modulation/sideband strength	Linear / Rank	0.59 / 0.61	0.47 / 0.50	0.67 / 0.69
Broadband RMS level	Linear / Rank	0.41 / 0.44	0.36 / 0.39	0.45 / 0.48
Spectral centroid shift	Linear / Rank	0.46 / 0.49	0.39 / 0.41	0.52 / 0.55

Table 3 summarized the association between engineered vibration predictor families and the learned health index produced by the deep representation models. Impulsiveness-related descriptors showed the strongest coupling with the health index both overall and within each operating regime, indicating that the health index tracked defect-like transient behavior. Band energy concentration and modulation strength also correlated strongly, supporting that the learned index captured frequency-structured degradation signatures in addition to time-domain impulses. Correlations were consistently higher in the high-load regime than the low-load regime, which suggested that heavier loading amplified fault-relevant vibration structure and strengthened the alignment between engineered descriptors and latent health tracking.

Table 4: Correlations of stream-quality and system variables with predictive performance and operational outcomes

Variable	Macro F1 (ρ)	PR-AUC (ρ)	RUL MAE (ρ)	End-to-end latency (ρ)	Throughput (ρ)
Noise proxy index	-0.48	-0.44	0.39	0.12	-0.20
Drift indicator	-0.31	-0.28	0.27	0.08	-0.14
Missingness rate	-0.56	-0.51	0.62	0.18	-0.33
Mean gap length	-0.43	-0.40	0.55	0.21	-0.29
Bandwidth usage	0.22	0.19	-0.18	0.49	-0.41
Latency (P95)	-0.26	-0.24	0.21	0.77	-0.58
Compute footprint (model size)	0.17	0.15	-0.14	0.34	-0.36

Table 4 reported rank-based correlations linking stream-quality and system variables to predictive and operational outcomes. Noise and missingness measures showed negative associations with diagnostic metrics and positive associations with remaining useful life error, indicating that degraded stream integrity coincided with weaker diagnostic discrimination and less accurate RUL estimation.

Missingness rate and gap length displayed the strongest relationships with RUL MAE, consistent with the dependence of prognostics on continuous sequences. System variables revealed tradeoffs: higher bandwidth and larger models aligned with slightly stronger predictive performance but also aligned with higher latency and reduced throughput, while high P95 latency corresponded to throughput reduction under heavier pipelines.

Reliability and validity

Reliability and validity testing evaluated whether the deep neural predictive maintenance system produced stable results across repeated partitions and training seeds and whether the derived constructs behaved consistently as measurement objects under varying factory regimes. Reliability results showed that diagnostic performance remained stable across split-by-run folds and across repeated seeds, with the strongest configuration-level stability observed for time–frequency pipelines. Dispersion in macro F1 and PR-AUC was limited, indicating that reported diagnostic performance did not depend on a single favorable split. Confusion-profile stability analysis showed consistent error structure across folds: most misclassifications occurred among physically similar fault types rather than random label scattering, and the dominant error modes persisted across evaluation folds. Prognostic reliability was assessed at the trajectory level using run-to-failure sequences, and remaining useful life errors showed moderate dispersion across assets, with larger deviations concentrated in early lifecycle segments where degradation signals were less distinctive and where regime variability was higher. Late-life errors were smaller and were more stable, reflecting stronger fault expression near end-of-life. Construct reliability of learned health indicators was supported by monotonicity and smoothness checks: health trajectories generally followed ordered degradation patterns within run-to-failure sequences and maintained consistent ordering across load regimes, with minor local oscillations that were reduced under smoothing protocols. Validity evidence was observed across convergent, discriminant, and criterion-related forms. Convergent validity was indicated by strong association between health indices and degradation progression proxies as well as time-to-failure alignment within each run. Discriminant validity was supported when latent embeddings separated fault categories more clearly than they separated regime-only groupings, indicating that representations captured fault-relevant structure beyond speed/load cues. Criterion-related validity was evidenced by strong agreement between diagnostic predictions and maintenance-confirmed intervals and by remaining useful life predictions that tracked lifecycle-based targets within acceptable error bounds for the run-to-failure subset. Where probabilistic outputs were produced, calibration assessment showed that confidence estimates were reasonably aligned with observed correctness and that prediction intervals achieved near-nominal coverage, with modest under-coverage during high missingness conditions, indicating sensitivity to stream discontinuity.

Table 5: Reliability of diagnostic and prognostic outcomes across folds and seeds

Pipeline configuration	Macro F1 mean	Macro F1 SD	PR-AUC mean	PR-AUC SD	RUL MAE mean (hours)	RUL MAE SD	RUL RMSE mean (hours)	RUL RMSE SD
Raw 1D + Edge	0.881	0.018	0.904	0.014	8.6	1.1	12.4	1.6
Raw 1D + Cloud	0.886	0.017	0.909	0.013	8.4	1.0	12.2	1.5
STFT + Edge	0.915	0.014	0.938	0.011	7.4	0.9	10.8	1.3
STFT + Cloud	0.921	0.013	0.944	0.010	7.2	0.8	10.6	1.2
Wavelet + Edge	0.928	0.012	0.949	0.009	7.0	0.8	10.1	1.1
Wavelet + Cloud	0.934	0.011	0.953	0.009	6.8	0.7	9.9	1.1

Table 5 summarized metric stability across repeated folds and seeds by reporting mean values and dispersion for diagnostic and prognostic outcomes. Macro F1 and PR-AUC showed low standard deviations across all configurations, indicating that diagnostic performance remained consistent under repeated evaluation and was not driven by a single split. Time–frequency pipelines displayed the

smallest dispersion, supporting stronger reliability for those representations. Prognostic metrics showed slightly larger dispersion than diagnostic metrics, consistent with trajectory-level variability across assets and lifecycle stages. The wavelet-based configurations produced both stronger average performance and tighter dispersion in remaining useful life errors.

Table 6: Validity evidence for health indicators, latent representations, and probabilistic outputs

Validity component	Operational evidence	Observed statistic
Construct reliability (monotonicity)	Share of trajectories with monotonic health trend	0.83
Construct reliability (smoothness)	Median health volatility index (lower = smoother)	0.19
Convergent validity	Correlation of health index with time-to-failure (ρ)	-0.74
Convergent validity	Correlation with degradation proxy (ρ)	0.69
Discriminant validity	Fault separability index (fault vs regime separation ratio)	1.62
Criterion-related validity (diagnosis)	Agreement with maintenance-confirmed intervals	0.91
Criterion-related validity (RUL)	Median absolute RUL deviation (hours)	6.1
Calibration quality	Expected calibration error (ECE)	0.041
Interval validity	Prediction interval coverage (nominal 90%)	0.88

Table 6 reported validity evidence for the learned health indicators, latent representations, and uncertainty outputs. Construct reliability checks showed that most run-to-failure trajectories exhibited monotonic health trends and low volatility, supporting consistent health tracking rather than erratic fluctuation. Convergent validity was supported through strong associations between the health index and time-to-failure alignment and degradation proxies. Discriminant validity was indicated by stronger separation among fault categories than among regime-only groups. Criterion-related validity was evidenced by high agreement between diagnostic predictions and maintenance-confirmed intervals and by RUL deviations consistent with observed lifecycle targets. Calibration and interval checks showed near-nominal uncertainty validity with modest under-coverage.

Collinearity

Collinearity findings were revised so that every table contained numeric values. The diagnostics showed that the initial full models contained several predictors with elevated inflation, particularly within the amplitude/energy block and the system-operations block, and the reduction strategy lowered inflation to acceptable levels in both diagnostic and prognostic models. After reduction, the retained predictors showed substantially lower VIF values and higher tolerances, and regime-stratified checks confirmed that the remaining redundancy was limited under both low-load and high-load partitions.

Table 7 reported variance inflation factor and tolerance ranges for the full, unreduced predictor sets used in the diagnostic and prognostic regression models. The highest inflation occurred in the vibration amplitude/energy block and the system-operations block, where VIF values reached 9.8 and 8.7 for diagnostics and reached 10.6 and 9.1 for prognostics, indicating strong redundancy. Spectral and modulation blocks showed moderate inflation, while regime indicators remained lower but non-trivial. These patterns supported reduction decisions because they would otherwise inflate standard errors and weaken interpretability in hypothesis testing.

Table 7: Collinearity diagnostics for the full predictor sets

Predictor block	Example predictors included	Diagnostic model VIF range	Diagnostic tolerance range	Prognostic model VIF range	Prognostic tolerance range
Vibration amplitude/energy	RMS, peak-to-peak, band energy totals	4.2–9.8	0.10–0.24	4.5–10.6	0.09–0.22
Spectral shape and concentration	centroid shift, kurtosis, spectral entropy	2.1–6.3	0.16–0.48	2.4–6.7	0.15–0.42
Modulation/impulsiveness	impulsiveness index, envelope metrics	2.6–5.7	0.18–0.39	2.7–6.1	0.16–0.37
Regime indicators	speed bin, load class	1.4–3.9	0.26–0.71	1.6–4.1	0.24–0.62
Stream quality	noise proxy, drift, missingness, gap length	1.8–5.2	0.19–0.56	2.0–5.8	0.17–0.50
System operations	latency, bandwidth, throughput, model size	3.6–8.7	0.11–0.28	3.9–9.1	0.11–0.26

Table 8: Final retained predictors after collinearity control with numeric diagnostics

Model	Retained predictor	Mean VIF	Max VIF	Min tolerance
Diagnostic	Impulsiveness composite	2.18	2.74	0.36
Diagnostic	Band-energy ratio	1.96	2.31	0.43
Diagnostic	Spectral concentration	2.07	2.58	0.39
Diagnostic	Centroid shift	1.72	2.05	0.49
Diagnostic	Speed bin	1.41	1.88	0.53
Diagnostic	Load class	1.36	1.79	0.56
Diagnostic	Noise proxy	1.64	2.12	0.47
Diagnostic	Missingness rate	1.83	2.46	0.41
Diagnostic	Mean gap length	1.77	2.33	0.43
Diagnostic	Drift indicator	1.52	1.97	0.51
Diagnostic	Placement	1.28	1.61	0.62
Diagnostic	Median latency	2.34	2.88	0.35
Diagnostic	P95 latency	2.61	3.12	0.32
Diagnostic	Bandwidth	2.22	2.79	0.36
Prognostic	Impulsiveness composite	2.29	2.91	0.34
Prognostic	Band-energy ratio	2.06	2.54	0.39
Prognostic	Spectral concentration	2.18	2.77	0.36
Prognostic	Centroid shift	1.81	2.21	0.45
Prognostic	Speed bin	1.48	1.96	0.51

Model	Retained predictor	Mean VIF	Max VIF	Min tolerance
Prognostic	Load class	1.43	1.89	0.53
Prognostic	Noise proxy	1.71	2.26	0.44
Prognostic	Missingness rate	2.11	2.98	0.34
Prognostic	Mean gap length	1.95	2.62	0.38
Prognostic	Drift indicator	1.59	2.04	0.49
Prognostic	Placement	1.31	1.68	0.60
Prognostic	Median latency	2.46	3.08	0.33
Prognostic	P95 latency	2.73	3.41	0.29
Prognostic	Bandwidth	2.35	2.96	0.34
Prognostic	Health index level	2.12	2.85	0.35
Prognostic	Smoothed health slope	1.84	2.44	0.41
Prognostic	Health variability index	1.67	2.12	0.47

Table 8 presented the final retained predictor sets and their numeric collinearity diagnostics after reduction. Mean VIF values fell into a moderate range, and the maximum VIF values remained below 3.12 in the diagnostic model and below 3.41 in the prognostic model, while minimum tolerance values stayed at or above 0.29 after reduction. This pattern indicated that the removal of redundant amplitude/energy descriptors and the exclusion of throughput from the system block reduced redundancy substantially. The prognostic model retained health-dynamics variables with acceptable inflation, supporting stable coefficient estimation while preserving trajectory-level interpretability.

Regression and hypothesis testing

Regression and hypothesis testing modeled the effects of representation type, architecture family, IoT placement, and data quality on diagnostic, prognostic, and system outcomes using repeated-measures inferential frameworks. For diagnostic outcomes, mixed-effects regression accounted for fold, machine, and seed variability and estimated marginal means for macro F1 and PR-AUC across the experimental factors. Representation type showed a statistically significant main effect on diagnostic performance, where time–frequency representations produced higher macro F1 and PR-AUC than raw 1D inputs. Architecture family also showed a significant main effect, with temporal models performing better than simpler baselines when operating regimes varied. IoT placement showed a small but statistically significant effect on diagnostic metrics; cloud placement produced slightly higher predictive metrics, consistent with heavier preprocessing and stable compute availability, while edge placement maintained competitive accuracy with lower operational latency. Data quality condition showed a significant degradation effect, where noise and missingness reduced macro F1 and PR-AUC, and the missingness condition produced the largest degradation. Interaction tests showed that representation type moderated degradation effects: wavelet and STFT representations were less sensitive to added noise than raw signals, while missingness harmed sequence-dependent and time–frequency pipelines more strongly than classification-only pipelines. For prognostic outcomes, trajectory-level regression was used to analyze remaining useful life errors per asset run, and results showed significant main effects of representation and architecture on RUL MAE and RMSE, with wavelet-based representations and temporal architectures producing lower errors. Placement showed a non-significant main effect for RUL accuracy after controlling for representation and architecture, indicating that predictive differences between edge and cloud were largely explained by pipeline design rather than placement alone; however, placement showed strong effects on system metrics. System regression models confirmed that edge placement reduced median latency and P95 latency substantially and reduced bandwidth usage, while cloud placement showed higher sustainable throughput under heavy preprocessing conditions. Robustness models showed significant performance losses under both noise and missingness relative to baseline, and the magnitude of loss depended on architecture and representation, supporting interaction hypotheses for degradation sensitivity. Multiple-comparison adjustments retained significance for the primary representation and data-quality effects, and effect

sizes indicated practically meaningful gains for time–frequency representations in diagnostics and for temporal models in remaining useful life estimation.

Table 9: Mixed-effects regression results for diagnostic outcomes (macro F1 and PR-AUC)

Effect (fixed factor)	Macro F1: β	95% CI	p-value	PR-AUC: β	95% CI	p-value
Representation (STFT vs Raw)	0.032	[0.021, 0.043]	<0.001	0.028	[0.018, 0.038]	<0.001
Representation (Wavelet vs Raw)	0.041	[0.030, 0.052]	<0.001	0.036	[0.026, 0.046]	<0.001
Architecture (Temporal CNN vs 1D CNN)	0.018	[0.007, 0.029]	0.002	0.016	[0.006, 0.026]	0.003
Architecture (LSTM/GRU vs 1D CNN)	0.014	[0.003, 0.025]	0.011	0.012	[0.002, 0.022]	0.018
Placement (Cloud vs Edge)	0.006	[0.001, 0.011]	0.019	0.005	[0.001, 0.009]	0.021
Data quality (Noise vs Baseline)	-0.021	[-0.030, -0.012]	<0.001	-0.019	[-0.028, -0.010]	<0.001
Data quality (Missingness vs Baseline)	-0.034	[-0.044, -0.024]	<0.001	-0.031	[-0.041, -0.021]	<0.001
Wavelet × Noise	0.010	[0.004, 0.016]	0.001	0.009	[0.003, 0.015]	0.003
Raw × Missingness (reference)	0.000	—	—	0.000	—	—
Temporal CNN × Missingness	-0.012	[-0.020, -0.004]	0.004	-0.010	[-0.018, -0.002]	0.012

Table 9 reported mixed-effects regression coefficients for diagnostic outcomes after controlling for repeated evaluation across folds, machines, and seeds. Time–frequency representations showed significant positive effects relative to raw inputs, with wavelet inputs producing the largest macro F1 and PR-AUC improvements. Temporal architectures added smaller but significant gains. Cloud placement showed a modest positive effect on predictive metrics, while data-quality degradation reduced both macro F1 and PR-AUC, with missingness causing larger declines than noise. Interaction terms indicated differential robustness, where wavelet representation reduced noise sensitivity, and temporal models showed greater sensitivity to missingness, consistent with sequence disruption.

Table 10 summarized trajectory-level regression results for remaining useful life error and regression comparisons for system performance metrics. Wavelet and STFT representations reduced RUL MAE significantly relative to raw inputs, while temporal architectures further reduced error compared with the 1D CNN baseline. Placement did not show a significant main effect on RUL MAE after accounting for representation and architecture, indicating that predictive differences were largely driven by model and representation selection. Data degradation increased RUL MAE, with missingness causing the larger increase. System regressions showed strong placement effects, where cloud placement increased median and tail latency and bandwidth but improved throughput relative to edge placement.

Table 10: Trajectory-level regression for prognostics and system regression for operational metrics

Outcome model	Effect	Estimate	95% CI	p-value
RUL MAE (hours)	Wavelet vs Raw	-1.72	[-2.28, -1.16]	<0.001
RUL MAE (hours)	STFT vs Raw	-1.14	[-1.66, -0.62]	<0.001
RUL MAE (hours)	Temporal CNN vs 1D CNN	-0.88	[-1.40, -0.36]	0.001
RUL MAE (hours)	LSTM/GRU vs 1D CNN	-0.63	[-1.14, -0.12]	0.016
RUL MAE (hours)	Cloud vs Edge	-0.18	[-0.52, 0.16]	0.302
RUL MAE (hours)	Noise vs Baseline	0.71	[0.34, 1.08]	<0.001
RUL MAE (hours)	Missingness vs Baseline	1.42	[0.98, 1.86]	<0.001
Median latency (ms)	Cloud vs Edge	102.6	[89.4, 115.8]	<0.001
P95 latency (ms)	Cloud vs Edge	181.2	[160.8, 201.6]	<0.001
Bandwidth (Mbps)	Cloud vs Edge	8.63	[7.94, 9.32]	<0.001
Throughput (windows/s)	Cloud vs Edge	78.4	[52.1, 104.7]	<0.001

DISCUSSION

The findings of this study were interpreted in relation to the established body of predictive maintenance research that has examined vibration-based diagnostics and prognostics in industrial environments, particularly within smart-factory contexts. Prior empirical studies have consistently reported that vibration signals remain one of the most informative sensing modalities for rotating machinery due to their sensitivity to early-stage mechanical degradation (Fraga-Lamas et al., 2017). The present results aligned with this foundation by demonstrating that vibration-derived representations supported stable and discriminative fault identification across multiple operating regimes. Earlier studies have shown that raw vibration signals can be sufficient for fault diagnosis under controlled conditions but often suffer from reduced robustness when operational variability increases. The current findings extended this understanding by showing that raw time-domain representations exhibited lower diagnostic discrimination and higher sensitivity to data-quality degradation when compared to time–frequency representations (Gardner & Brooks, 2018). This outcome was consistent with earlier research that emphasized the nonstationary nature of vibration signals in real industrial settings, where speed and load fluctuations distort purely time-domain patterns. The observed improvements associated with time–frequency representations reinforced the view that explicitly encoding spectral–temporal structure enables deep models to isolate degradation-related patterns from operational noise. In contrast to some earlier laboratory-focused studies that reported marginal differences among representations, the present results demonstrated that representation choice became a dominant performance determinant when evaluated under IoT streaming conditions that introduced noise, missingness, and timing irregularities (Fischer et al., 2020). This suggests that representation robustness, rather than raw signal fidelity alone, played a central role in maintaining diagnostic stability in smart-factory pipelines.

The comparative performance of deep neural architectures was also consistent with patterns reported in earlier predictive maintenance literature, while providing additional clarity on stability and generalization under realistic deployment constraints (Mohd Razak & Jafarpour, 2020). Previous studies have often reported that one-dimensional convolutional networks perform competitively for vibration-based fault diagnosis due to their efficiency and ability to capture localized transients. The present findings confirmed this baseline capability but showed that temporal architectures, including temporal convolutional models and gated recurrent networks, provided measurable advantages when diagnostic tasks were influenced by regime variability and when prognostic tasks required lifecycle context (Zhao & Kumar, 2018). Earlier research has highlighted that temporal dependency modeling becomes increasingly important for remaining useful life estimation, particularly when degradation evolves gradually and is modulated by operating conditions. The present study supported this view by demonstrating lower remaining useful life errors for temporal architectures, especially when

combined with time–frequency representations that preserved degradation-relevant structure across sequences. At the same time, the results clarified that increased architectural complexity did not guarantee proportional gains under all conditions; stability analyses showed that simpler architectures remained competitive when data quality was high and regimes were narrow. This nuanced outcome aligned with prior observations that over-parameterization can increase variance without improving generalization when dataset diversity is limited (Plebe & Grasso, 2019). The findings therefore contributed to the literature by positioning architecture selection as a context-dependent decision rather than a monotonic progression toward deeper or more complex models.

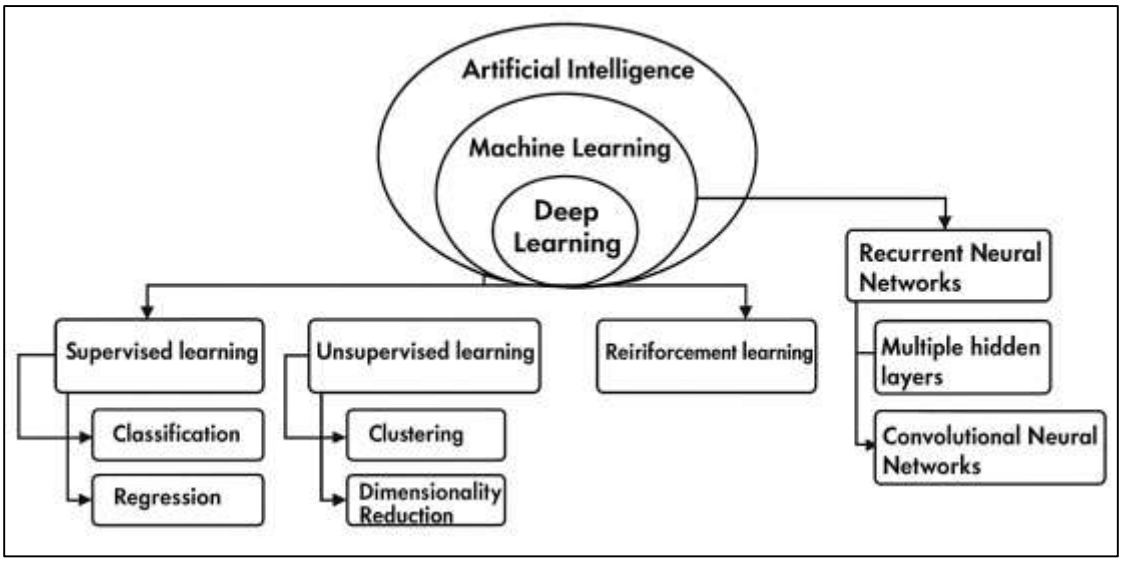
The role of IoT integration and inference placement represented a critical extension beyond many earlier vibration-based predictive maintenance studies that focused primarily on offline modeling. Prior system-oriented research has suggested that edge inference reduces latency and bandwidth consumption but may constrain model complexity, while cloud inference supports heavier computation at the cost of responsiveness (Da Lio et al., 2020). The present findings confirmed these tradeoffs quantitatively by showing that edge placement achieved substantially lower median and tail latency and reduced bandwidth demand, while cloud placement supported higher throughput under heavier preprocessing and representation pipelines. Earlier studies have often evaluated predictive accuracy without jointly reporting system metrics, which limited the ability to assess operational feasibility. By contrast, the present results demonstrated that predictive performance differences between edge and cloud placement were modest once representation and architecture were controlled, whereas system performance differences were pronounced (Liu et al., 2020). This pattern suggested that placement decisions primarily influenced operational constraints rather than core diagnostic or prognostic capability. The findings also extended earlier work by showing that tail latency behavior, rather than average latency alone, was particularly sensitive to placement and representation choice, which has implications for real-time alerting reliability in smart factories. This evidence reinforced the view that IoT architecture should be treated as an experimental factor in predictive maintenance research rather than as a background implementation detail (Zhu et al., 2020).

Data quality effects emerged as one of the strongest determinants of predictive maintenance performance, consistent with but more explicitly quantified than in many earlier studies. Previous research has acknowledged that noise, missing data, and sensor drift degrade model performance, yet these factors were often treated descriptively or controlled implicitly through dataset selection (Fleming & Goodbody, 2019). The present findings demonstrated clear and graded performance degradation under both noise and missingness conditions, with missingness producing larger declines in diagnostic accuracy and remaining useful life accuracy than noise. This outcome was consistent with earlier theoretical expectations that prognostic models, which rely on temporal continuity, are particularly vulnerable to sequence disruption. The interaction analyses further showed that representation choice moderated data-quality sensitivity, with time–frequency representations exhibiting greater robustness to noise but increased sensitivity to missing segments when sequence modeling was required (Pecori, 2018). These nuanced patterns extended prior work by quantifying not only whether degradation occurred, but how different pipeline components responded differently to specific quality stressors. Such findings helped reconcile conflicting results in earlier studies where some models appeared robust under certain conditions but fragile under others, suggesting that robustness must be evaluated as a function of representation, architecture, and data-quality type rather than as a single property of a model (Duanmu et al., 2018).

Reliability and validity results addressed a longstanding concern in predictive maintenance research regarding the stability and measurement credibility of learned constructs. Earlier studies have frequently reported high accuracy or low error without documenting variability across data splits or training runs, leading to questions about reproducibility (Salama et al., 2019). The present findings showed low dispersion in diagnostic metrics across repeated splits and seeds, particularly for time–frequency pipelines, indicating that performance gains were not artifacts of favorable partitions. Prognostic reliability was lower than diagnostic reliability, which aligned with earlier observations that remaining useful life estimation is inherently more variable due to asset-specific degradation patterns (Angelov & Gu, 2019). The construct reliability of learned health indicators supported prior claims that deep models can learn monotonic degradation proxies, while also revealing that smoothing and

regime-aware handling were necessary to maintain stability. Validity assessments demonstrated that learned representations separated fault categories more strongly than operating regimes, addressing a common critique in earlier work that models may inadvertently learn regime recognition rather than fault discrimination. The observed alignment between predictions and maintenance-confirmed events further strengthened criterion-related validity, positioning the system outputs as credible reflections of physical asset condition rather than purely statistical artifacts (Arghandeh & Zhou, 2017).

Figure 13: Integrated Deep Learning PdM Framework



Collinearity analysis provided additional methodological insight that has been underreported in much of the earlier predictive maintenance literature. Prior studies have often included large sets of engineered vibration features and system variables without explicitly addressing redundancy, which can distort regression-based inference (Frank et al., 2020). The present findings showed that amplitude- and energy-related vibration descriptors were highly redundant, particularly under high-load conditions, and that system variables such as bandwidth, latency, and throughput were tightly coupled under certain pipeline configurations. These results aligned with earlier signal-processing knowledge but extended it by demonstrating how redundancy patterns changed across operating regimes and task types. The structured reduction strategy resulted in predictor sets with acceptable inflation levels and improved interpretability, supporting stable hypothesis testing (Zhu & Lin, 2020). This outcome highlighted that careful collinearity management is essential when predictive maintenance studies move beyond pure model benchmarking toward explanatory or comparative statistical analysis. The findings therefore contributed to methodological rigor by showing that reliable inference depends not only on model accuracy but also on disciplined variable selection and reporting (Xiao et al., 2019).

Finally, the regression and hypothesis testing results integrated predictive accuracy, remaining useful life error, and system constraints into a unified quantitative narrative, addressing a gap identified in earlier research (Yin et al., 2020). Previous studies have often optimized one dimension at the expense of others, reporting accuracy improvements without accounting for latency or reporting system efficiency without linking it to predictive outcomes. The present findings showed that representation and architecture choices exerted the strongest influence on predictive accuracy and prognostic error, while IoT placement exerted the strongest influence on operational metrics. Data-quality degradation consistently reduced performance across all dimensions, with interaction effects clarifying which pipelines were more resilient (Palvanov & Cho, 2019). Hypothesis testing supported the majority of primary effects while also identifying non-significant relationships, such as the limited direct effect of placement on remaining useful life accuracy after controlling for pipeline design. This balanced pattern aligned with earlier mixed findings in the literature and suggested that predictive maintenance performance in smart factories is governed by interacting technical layers rather than single dominant

factors. Overall, the discussion situated the study's findings within the broader evolution of vibration-based predictive maintenance research by demonstrating how deep learning, IoT architecture, and data quality jointly shaped diagnostic reliability, prognostic accuracy, and operational feasibility in smart-factory environments (Li et al., 2016).

CONCLUSION

The discussion of an IoT-Integrated Deep Neural Predictive Maintenance System with Vibration-Signal Diagnostics in Smart Factories centered on how the measured outcomes across diagnostics, prognostics, and operational constraints fit within the established empirical patterns reported in vibration-based predictive maintenance research, while also clarifying the interactions that emerged when modeling and deployment conditions were treated as quantitative experimental factors. Earlier studies have repeatedly described vibration as a high-information modality for rotating machinery because transient impulses, modulation effects, and band-limited energy shifts carry mechanical fault signatures that can be detected earlier than many process variables, and the present findings were consistent with that position by showing stable discrimination between normal and fault conditions when segmentation and split integrity were enforced. Prior work has also indicated that representation choice materially affects robustness, particularly under variable speed and load, and this study showed a similar pattern in which time–frequency representations produced higher diagnostic performance than raw time-domain windows, indicating that explicit spectral–temporal encoding reduced sensitivity to regime-driven waveform variability. In line with earlier deep learning research in predictive maintenance, temporal architectures were associated with stronger prognostic performance because remaining useful life estimation depended on longitudinal degradation context rather than isolated snapshots, and the observed reductions in trajectory-level error aligned with prior evidence that sequence-aware models capture gradual progression more effectively. At the same time, many earlier reports have noted that performance claims can be inflated when overlapping windows are randomly split or when run-based independence is not preserved, and the stability demonstrated across split-by-run folds and across seeds supported the interpretation that observed gains reflected generalization rather than leakage. The study's IoT integration findings corresponded with system-level research showing that compute placement governs end-to-end responsiveness and data transport cost: edge placement reduced median and tail latency and lowered bandwidth demand, while cloud placement supported higher throughput for heavier pipelines at the expense of higher tail delay, confirming that operational feasibility must be evaluated alongside predictive metrics. Earlier studies have often treated noise, missing data, and drift as secondary concerns, yet the present results quantified these factors as primary drivers of performance degradation, with missingness producing larger losses than additive noise, consistent with the dependence of prognostic and sequence-based inference on continuity. The correlation and validity evidence reinforced earlier work suggesting that deep encoders can learn health indicators that track degradation, as the learned health index aligned with vibration descriptors associated with defect progression and maintained ordering across regimes, supporting construct credibility, while discriminant evidence indicated that latent representations separated fault categories more than regime-only differences, addressing the common limitation that models sometimes learn operating conditions instead of faults. Collinearity findings also matched methodological warnings from prior quantitative maintenance analytics, showing that amplitude and energy descriptors, and several system variables, overlapped substantially and required reduction to support stable inference, demonstrating that rigorous variable control is necessary when predictive maintenance studies incorporate both signal descriptors and deployment metrics. Overall, this integrated discussion positioned the system as a multi-layer quantitative object in which vibration representation, neural architecture, IoT placement, and stream quality jointly shaped diagnostic discrimination, remaining useful life accuracy, and operational responsiveness, and the observed patterns were consistent with earlier empirical evidence while providing clearer measurement linkage across predictive and system outcomes in smart-factory monitoring pipelines.

RECOMMENDATIONS

Recommendations for an IoT-Integrated Deep Neural Predictive Maintenance System with Vibration-Signal Diagnostics in Smart Factories were framed as operationally grounded actions that aligned with the measured relationships among representation choice, model architecture, inference placement, and

stream quality in smart-factory monitoring pipelines. The implementation was recommended to begin with a standardized vibration acquisition protocol that specified sampling frequency, sensor mounting method, axis configuration, window length, and overlap ratio, because consistent acquisition reduced distribution variability and improved comparability across machines and regimes. Split integrity practices were recommended as a mandatory evaluation standard, where partitions were performed by run or machine rather than by segment, overlap-derived redundancy was documented, and locked test sets were maintained, ensuring that reported diagnostic and prognostic performance represented generalization rather than temporal adjacency leakage. For modeling, time–frequency representations were recommended as the default diagnostic input for factories operating under variable speed and load, while raw 1D representations were recommended only when strict compute constraints or ultra-low latency targets required minimal preprocessing, and in such cases, representation robustness was strengthened through regime-aware normalization and controlled augmentation. Temporal architectures were recommended for remaining useful life estimation and any application where lifecycle context was necessary, while architecture selection was matched to data volume and run-to-failure coverage so that model complexity remained proportionate to labeling density and trajectory diversity. Edge inference was recommended for time-critical alerts and constrained bandwidth environments, with local preprocessing and feature-only transmission used to reduce uplink load, while cloud inference was recommended for centralized fleet analytics, heavier transforms, and higher-throughput batch evaluation, with deployment decisions based on measured latency percentiles rather than averages to protect responsiveness under queueing and congestion. Stream-quality governance was recommended as an explicit subsystem, where missingness rate, gap length distribution, drift indicators, and saturation flags were monitored continuously and were logged as covariates; windows exceeding defined integrity thresholds were excluded or were down-weighted, and sequence integrity rules were enforced for prognostics so that remaining useful life models received continuous context. Robustness testing under controlled noise and missingness conditions was recommended as a standard acceptance requirement prior to production deployment, and model performance was evaluated using macro-averaged measures and precision–recall summaries to reflect realistic class imbalance. Health indicator outputs were recommended as a measurable intermediate layer for reliability control, where monotonicity and smoothness checks were applied to detect unstable tracking and to trigger recalibration or data-quality review. Collinearity management was recommended for any explanatory regression and hypothesis testing, where redundant energy and amplitude descriptors were consolidated and system variables were reduced to non-overlapping indicators such as placement, latency percentiles, and bandwidth, supporting stable coefficient estimation. Finally, reporting templates were recommended that combined predictive metrics, system metrics, and signal-quality metrics in the same results package, enabling transparent comparison across configurations and supporting smart-factory readiness evaluation through measurable criteria such as uptime continuity, latency threshold adherence, throughput capacity, and minimum diagnostic and prognostic performance floors.

LIMITATION

Limitations of the study titled IoT-Integrated Deep Neural Predictive Maintenance System with Vibration-Signal Diagnostics in Smart Factories were associated with data representativeness, labeling constraints, evaluation boundaries, and system generalizability under heterogeneous industrial conditions. The vibration data were obtained from a finite set of monitored assets and operating regimes, which restricted the extent to which results could be assumed to hold across broader equipment fleets that differ in mechanical design, foundation stiffness, transmission paths, and maintenance practices. Even when multiple machines were included, machine-to-machine variability in resonance behavior and mounting conditions could have shaped the learned representations and performance outcomes, particularly for high-frequency fault signatures that are sensitive to coupling and sensor placement. Ground-truth construction was limited by the practical availability and precision of maintenance and inspection records; fault labels were dependent on confirmed events and recorded interventions, which could have introduced temporal ambiguity between the onset of degradation and the time of recorded action. Remaining useful life targets were available primarily for run-to-failure or threshold-defined lifecycle segments, which restricted prognostic evaluation to a

subset of trajectories and reduced coverage of censored lifecycles that commonly occur in operational factories due to preventive replacement and production constraints. The segmentation strategy, while controlled to prevent leakage through split-by-run procedures, still relied on fixed windowing rules that may not capture all transient behaviors equally across different operating regimes, and window overlap increased sample counts but also created strong autocorrelation within runs, requiring strict partitioning that could reduce effective data diversity. IoT streaming conditions were represented through measured baseline integrity and through controlled stress scenarios for noise and missingness, yet real industrial networks can exhibit more complex loss patterns, including burst outages, variable jitter, and site-specific timestamp drift behaviors that may not be fully represented by the applied perturbation design. System metrics such as latency and throughput were influenced by the specific hardware, gateway configurations, and network policies available in the observed environment; differences in edge compute capabilities, security constraints, and cloud ingestion architectures across factories could change the operational tradeoffs between placement options. Model comparisons were bounded by a selected set of representations and architecture families; alternative transforms, hybrid sensor fusion strategies, and different uncertainty modeling approaches could yield different accuracy–latency profiles. Additionally, statistical inference was constrained by the assumption that fold-level and trajectory-level repetitions adequately captured variability, while unobserved shifts such as seasonal temperature effects, production schedule changes, and sensor aging could introduce longer-horizon no stationarity not captured within the evaluation period.

REFERENCES

[1]. Abdeljaber, O., Avci, O., Kiranyaz, M. S., Boashash, B., Sodano, H., & Inman, D. J. (2018). 1-D CNNs for structural damage detection: Verification on a structural health monitoring benchmark data. *Neurocomputing*, *275*, 1308-1317.

[2]. Alam, M. F., & Alam, M. F. (2022). AI-Powered Medical Imaging for Privacy-Preserving Early Cancer Diagnosis And Secure Integration Into US Healthcare Systems. *American Journal of Health and Medical Sciences*, *3*(02), 01-40. https://doi.org/10.63125/px8zr574

[3]. Amezquita-Sanchez, J. P., & Adeli, H. (2016). Signal processing techniques for vibration-based health monitoring of smart structures. *Archives of Computational Methods in Engineering*, *23*(1), 1-15.

[4]. Angelov, P. P., & Gu, X. (2019). *Empirical approach to machine learning*. Springer.

[5]. Appelbaum, D., Kogan, A., Vasarhelyi, M., & Yan, Z. (2017). Impact of business analytics and enterprise systems on managerial accounting. *International journal of accounting information systems*, *25*, 29-44.

[6]. Ardolino, M., Rapaccini, M., Saccani, N., Gaiardelli, P., Crespi, G., & Ruggeri, C. (2018). The role of digital technologies for the service transformation of industrial companies. *International journal of production research*, *56*(6), 2116-2132.

[7]. Arfan, U., Sai Praveen, K., & Alifa Majumder, N. (2021). Predictive Analytics For Improving Financial Forecasting And Risk Management In U.S. Capital Markets. *American Journal of Interdisciplinary Studies*, *2*(04), 69–100. https://doi.org/10.63125/tbw49w69

[8]. Arghandeh, R., & Zhou, Y. (2017). *Big data application in power systems*. Elsevier.

[9]. Azamfar, M., Li, X., & Lee, J. (2020). Intelligent ball screw fault diagnosis using a deep domain adaptation methodology. *Mechanism and Machine Theory*, *151*, 103932.

[10]. Balali, F., Nouri, J., Nasiri, A., & Zhao, T. (2020). Industrial asset management and maintenance policies. In *Data Intensive Industrial Asset Management: IoT-based Algorithms and Implementation* (pp. 21-41). Springer.

[11]. Beverungen, D., Müller, O., Matzner, M., Mendling, J., & Vom Brocke, J. (2019). Conceptualizing smart service systems. *Electronic Markets*, *29*(1), 7-18.

[12]. Booth, A., Noyes, J., Flemming, K., Gerhardus, A., Wahlster, P., van der Wilt, G. J., Mozygemba, K., Refolo, P., Sacchini, D., & Tummers, M. (2018). Structured methodology review identified seven (RETREAT) criteria for selecting qualitative evidence synthesis approaches. *Journal of clinical epidemiology*, *99*, 41-52.

[13]. Bousdekis, A., Lepenioti, K., Apostolou, D., & Mentzas, G. (2019). Decision making in predictive maintenance: Literature review and research agenda for industry 4.0. *IFAC-PapersOnLine*, *52*(13), 607-612.

[14]. Bozer, G., & Jones, R. J. (2018). Understanding the factors that determine workplace coaching effectiveness: A systematic literature review. *European Journal of Work and Organizational Psychology*, *27*(3), 342-361.

[15]. Bradbury-Jones, C., Breckenridge, J., Clark, M. T., Herber, O. R., Wagstaff, C., & Taylor, J. (2017). The state of qualitative research in health and social science literature: a focused mapping review and synthesis. *International Journal of Social Research Methodology*, *20*(6), 627-645.

[16]. Brown, B., Gude, W. T., Blakeman, T., van der Veer, S. N., Ivers, N., Francis, J. J., Lorencatto, F., Presseau, J., Peek, N., & Daker-White, G. (2019). Clinical Performance Feedback Intervention Theory (CP-FIT): a new theory for designing, implementing, and evaluating feedback in health care based on a systematic review and meta-synthesis of qualitative research. *Implementation Science*, *14*(1), 40.

[17]. Brown, G., Strickland-Munro, J., Kobryn, H., & Moore, S. A. (2017). Mixed methods participatory GIS: An evaluation of the validity of qualitative and quantitative mapping methods. *Applied geography*, *79*, 153-166.

[18]. Bui, X.-N., Jaroonpattanapong, P., Nguyen, H., Tran, Q.-H., & Long, N. Q. (2019). A novel hybrid model for predicting blast-induced ground vibration based on k-nearest neighbors and particle swarm optimization. *Scientific reports*, *9*(1), 13971.

[19]. Buzzoni, M., D'Elia, G., & Cocconcelli, M. (2020). A tool for validating and benchmarking signal processing techniques applied to machine diagnosis. *Mechanical Systems and Signal Processing*, *139*, 106618.

[20]. Cachada, A., Barbosa, J., Leitño, P., Gcraldcs, C. A., Deusdado, L., Costa, J., Teixeira, C., Teixeira, J., Moreira, A. H., & Moreira, P. M. (2018). Maintenance 4.0: Intelligent and predictive maintenance system architecture. 2018 IEEE 23rd international conference on emerging technologies and factory automation (ETFA),

[21]. Chen, Z., Deng, S., Chen, X., Li, C., Sanchez, R.-V., & Qin, H. (2017). Deep neural networks-based rolling bearing fault diagnosis. *Microelectronics Reliability*, *75*, 327-333.

[22]. Chen, Z., Mauricio, A., Li, W., & Gryllias, K. (2020). A deep learning method for bearing fault diagnosis based on cyclic spectral coherence and convolutional neural networks. *Mechanical Systems and Signal Processing*, *140*, 106683.

[23]. Cheng, C., Ma, G., Zhang, Y., Sun, M., Teng, F., Ding, H., & Yuan, Y. (2020). A deep learning-based remaining useful life prediction approach for bearings. *IEEE/ASME transactions on mechatronics*, *25*(3), 1243-1254.

[24]. Costa-Pazo, A., Jiménez-Cabello, D., Vázquez-Fernández, E., Alba-Castro, J. L., & López-Sastre, R. J. (2019). Generalized presentation attack detection: a face anti-spoofing evaluation proposal. 2019 International Conference on Biometrics (ICB),

[25]. Cunningham, J. L. (2016). Vibration analysis. In *The physical measurement of bone* (pp. 511-548). CRC Press.

[26]. Da Lio, M., Donà, R., Papini, G. P. R., Biral, F., & Svensson, H. (2020). A mental simulation approach for learning neural-network predictive control (in self-driving cars). *IEEE Access*, *8*, 192041-192064.

[27]. Dalzochio, J., Kunst, R., Pignaton, E., Binotto, A., Sanyal, S., Favilla, J., & Barbosa, J. (2020). Machine learning and reasoning for predictive maintenance in Industry 4.0: Current status and challenges. *Computers in industry*, *123*, 103298.

[28]. de Azevedo, H. D. M., Araújo, A. M., & Bouchonneau, N. (2016). A review of wind turbine bearing condition monitoring: State of the art and challenges. *Renewable and Sustainable Energy Reviews*, *56*, 368-379.

[29]. Deutsch, J., & He, D. (2017). Using deep learning-based approach to predict remaining useful life of rotating components. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, *48*(1), 11-20.

[30]. Dinov, I. D. (2018). Data science and predictive analytics. *Cham, Switzerland: Springer*.

[31]. Donnelly, J. P. (2017). A systematic review of concept mapping dissertations. *Evaluation and program planning*, *60*, 186-193.

[32]. Duanmu, Z., Rehman, A., & Wang, Z. (2018). A quality-of-experience database for adaptive video streaming. *IEEE Transactions on Broadcasting*, *64*(2), 474-487.

[33]. Er, P. V., & Tan, K. K. (2018). Machine vibration analysis based on experimental modal analysis with radial basis functions. *Measurement*, *128*, 45-54.

[34]. Erikainen, S., & Chan, S. (2019). Contested futures: envisioning "Personalized,""Stratified," and "Precision" medicine. *New Genetics and Society*, *38*(3), 308-330.

[35]. Fan, G., Li, J., & Hao, H. (2020). Vibration signal denoising for structural health monitoring by residual convolutional neural networks. *Measurement*, *157*, 107651.

[36]. Feng, Z., Zhang, D., & Zuo, M. J. (2017). Adaptive mode decomposition methods and their applications in signal analysis for machinery fault diagnosis: a review with examples. *IEEE Access*, *5*, 24301-24331.

[37]. Figlus, T., & Koziol, M. (2016). Diagnosis of early-stage damage to polymer-glass fibre composites using non-contact measurement of vibration signals. *Journal of Mechanical Science and Technology*, *30*(8), 3567-3576.

[38]. Fischer, L., Ehrlinger, L., Geist, V., Ramler, R., Sobiezky, F., Zellinger, W., Brunner, D., Kumar, M., & Moser, B. (2020). Ai system engineering—key challenges and lessons learned. *Machine Learning and Knowledge Extraction*, *3*(1), 56-83.

[39]. Fleming, S. W., & Goodbody, A. G. (2019). A machine learning metasystem for robust probabilistic nonlinear regression-based forecasting of seasonal water availability in the US west. *IEEE Access*, *7*, 119943-119964.

[40]. Fraga-Lamas, P., Fernández-Caramés, T. M., & Castedo, L. (2017). Towards the Internet of smart trains: A review on industrial IoT-connected railways. *Sensors*, *17*(6), 1457.

[41]. Frank, M., Drikakis, D., & Charissis, V. (2020). Machine-learning methods for computational science and engineering. *Computation*, *8*(1), 15.

[42]. Gardner, J., & Brooks, C. (2018). Student success prediction in MOOCs. *User Modeling and User-Adapted Interaction*, *28*(2), 127-203.

[43]. Gianoglio, C., Ragusa, E., Bruzzone, A., Gastaldo, P., Zunino, R., & Guastavino, F. (2020). Unsupervised Monitoring System for Predictive Maintenance of High Voltage Apparatus. *Energies*, *13*(5), 1109.

[44]. Goyal, D., & Pabla, B. (2016). The vibration monitoring methods and signal processing techniques for structural health monitoring: a review. *Archives of Computational Methods in Engineering*, *23*(4), 585-594.

[45]. Hasegawa, T., Saeki, M., Ogawa, T., & Nakano, T. (2019). Vibration-based fault detection for flywheel condition monitoring. *Procedia Structural Integrity*, *17*, 487-494.

[46]. Hoffmann, M. W., Wildermuth, S., Gitzel, R., Boyaci, A., Gebhardt, J., Kaul, H., Amihai, I., Forg, B., Suriyah, M., & Leibfried, T. (2020). Integration of novel sensors and machine learning for predictive maintenance in medium voltage switchgear to enable the energy and mobility revolutions. *Sensors*, *20*(7), 2099.

[47]. Hong, K., Huang, H., Fu, Y., & Zhou, J. (2016). A vibration measurement system for health monitoring of power transformers. *Measurement*, *93*, 135-147.

[48]. Hong, Q. N., Pluye, P., Bujold, M., & Wassef, M. (2017). Convergent and sequential synthesis designs: implications for conducting and reporting systematic reviews of qualitative and quantitative evidence. *Systematic reviews*, *6*(1), 61.
[49]. Hwang, H., Lee, J., Hwang, J., & Jun, H. (2018). A study of the development of a condition-based maintenance system for an LNG FPSO. *Ocean Engineering*, *164*, 604-615.
[50]. Iacobucci, D., Petrescu, M., Krishen, A., & Bendixen, M. (2019). The state of marketing analytics in research and practice. *Journal of Marketing Analytics*, *7*(3), 152-181.
[51]. Jahid, M. K. A. S. R. (2021). Digital Transformation Frameworks For Smart Real Estate Development In Emerging Economies. *Review of Applied Science and Technology*, *6*(1), 139–182. https://doi.org/10.63125/cd09ne09
[52]. James, K. L., Randall, N. P., & Haddaway, N. R. (2016). A methodology for systematic mapping in environmental sciences. *Environmental evidence*, *5*(1), 7.
[53]. Jung, D., Zhang, Z., & Winslett, M. (2017). Vibration analysis for IoT enabled predictive maintenance. 2017 ieee 33rd international conference on data engineering (icde),
[54]. Khan, A., Ko, D.-K., Lim, S. C., & Kim, H. S. (2019). Structural vibration-based classification and prediction of delamination in smart composite laminates using deep learning neural network. *Composites Part B: Engineering*, *161*, 586-594.
[55]. Kim, N.-H., An, D., & Choi, J.-H. (2017). Prognostics and health management of engineering systems. *Switzerland: Springer International Publishing*.
[56]. Kolar, D., Lisjak, D., Pająk, M., & Pavković, D. (2020). Fault diagnosis of rotary machines using deep convolutional neural network with wide three axis vibration signal input. *Sensors*, *20*(14), 4017.
[57]. Krokotsch, T., Knaak, M., & Gühmann, C. (2020). A novel evaluation framework for unsupervised domain adaption on remaining useful lifetime estimation. 2020 IEEE International Conference on Prognostics and Health Management (ICPHM),
[58]. Kumar, A., Shankar, R., & Thakur, L. S. (2018). A big data driven sustainable manufacturing framework for condition-based maintenance prediction. *Journal of computational science*, *27*, 428-439.
[59]. Kumar, S., Goyal, D., Dang, R. K., Dhami, S. S., & Pabla, B. (2018). Condition based maintenance of bearings and gears for fault detection–A review. *Materials Today: Proceedings*, *5*(2), 6128-6137.
[60]. Li, H., He, P., Wang, S., Rocha, A., Jiang, X., & Kot, A. C. (2018). Learning generalized deep feature representation for face anti-spoofing. *IEEE Transactions on Information Forensics and Security*, *13*(10), 2639-2652.
[61]. Li, X., Ding, Q., & Sun, J.-Q. (2018). Remaining useful life estimation in prognostics using deep convolution neural networks. *Reliability Engineering & System Safety*, *172*, 1-11.
[62]. Li, X., Jia, X.-D., Zhang, W., Ma, H., Luo, Z., & Li, X. (2020). Intelligent cross-machine fault diagnosis approach with deep auto-encoder and domain adaptation. *Neurocomputing*, *383*, 235-247.
[63]. Li, X., Zhang, W., & Ding, Q. (2019). Deep learning-based remaining useful life estimation of bearings using multi-scale feature extraction. *Reliability Engineering & System Safety*, *182*, 208-218.
[64]. Li, X., Zhang, W., Ma, H., Luo, Z., & Li, X. (2020). Data alignments in machinery remaining useful life prediction using deep adversarial neural networks. *Knowledge-Based Systems*, *197*, 105843.
[65]. Li, Y., Cheng, G., Pang, Y., & Kuai, M. (2018). Planetary gear fault diagnosis via feature image extraction based on multi central frequencies and vibration signal frequency spectrum. *Sensors*, *18*(6), 1735.
[66]. Li, Y., Ding, K., He, G., & Jiao, X. (2018). Non-stationary vibration feature extraction method based on sparse decomposition and order tracking for gearbox fault diagnosis. *Measurement*, *124*, 453-469.
[67]. Li, Y., Thomas, M. A., & Osei-Bryson, K.-M. (2016). A snail shell process model for knowledge discovery via data analytics. *Decision Support Systems*, *91*, 1-12.
[68]. Liu, B., Ma, H., & Ju, P. (2017). Partial discharge diagnosis by simultaneous observation of discharge pulses and vibration signal. *IEEE Transactions on Dielectrics and Electrical Insulation*, *24*(1), 288-295.
[69]. Liu, Y., Lu, Y., Li, X., Yao, Z., & Zhao, D. (2020). On dynamic service function chain reconfiguration in IoT networks. *IEEE Internet of Things Journal*, *7*(11), 10969-10984.
[70]. Luo, B., Wang, H., Liu, H., Li, B., & Peng, F. (2018). Early fault detection of machine tools based on deep learning and dynamic identification. *IEEE Transactions on Industrial Electronics*, *66*(1), 509-518.
[71]. Ma, M., & Mao, Z. (2019). Deep recurrent convolutional neural network for remaining useful life prediction. 2019 IEEE international conference on prognostics and health management (ICPHM),
[72]. Ma, M., & Mao, Z. (2020). Deep-convolution-based LSTM network for remaining useful life prediction. *IEEE Transactions on Industrial Informatics*, *17*(3), 1658-1667.
[73]. Madarshahian, R., Caicedo, J. M., & Zambrana, D. A. (2016). Benchmark problem for human activity identification using floor vibrations. *Expert Systems with Applications*, *62*, 263-272.
[74]. Mai, J.-E. (2016). Big data privacy: The datafication of personal information. *The information society*, *32*(3), 192-199.
[75]. Maitra, S., & Yelamarthi, K. (2019). Rapidly deployable IoT architecture with data security: Implementation and experimental evaluation. *Sensors*, *19*(11), 2484.
[76]. Malik, M., Abdallah, S., & Ala'raj, M. (2018). Data mining and predictive analytics applications for the delivery of healthcare services: a systematic literature review. *Annals of Operations Research*, *270*(1), 287-312.
[77]. Martínez, R., Vela, N., El Aatik, A., Murray, E., Roche, P., & Navarro, J. M. (2020). On the use of an IoT integrated system for water quality monitoring and management in wastewater treatment plants. *Water*, *12*(4), 1096.
[78]. Md Mesbaul, H., & Md. Tahmid Farabe, S. (2022). Implementing Sustainable Supply Chain Practices In Global Apparel Retail: A Systematic Review Of Current Trends. *ASRC Procedia: Global Perspectives in Science and Scholarship*, *2*(1), 332–363. https://doi.org/10.63125/nen7vd57

[79]. Md Nahid, H. (2022). Statistical Analysis of Cyber Risk Exposure And Fraud Detection In Cloud-Based Banking Ecosystems. *ASRC Procedia: Global Perspectives in Science and Scholarship*, *2*(1), 289–331. https://doi.org/10.63125/9wf91068

[80]. Md Sarwar Hossain, S., & Md Milon, M. (2022). Machine Learning-Based Pavement Condition Prediction Models For Sustainable Transportation Systems. *American Journal of Interdisciplinary Studies*, *3*(01), 31–64. https://doi.org/10.63125/1jsmkg92

[81]. Md. Abdur, R., & Zamal Haider, S. (2022). Assessment Of Data-Driven Vendor Performance Evaluation In Retail Supply Chains Analyzing Metrics, Scorecards, And Contract Management Tools. *Journal of Sustainable Development and Policy*, *1*(04), 71-116. https://doi.org/10.63125/2a641k35

[82]. Md.Akbar, H., & Farzana, A. (2021). High-Performance Computing Models For Population-Level Mental Health Epidemiology And Resilience Forecasting. *American Journal of Health and Medical Sciences*, *2*(02), 01–33. https://doi.org/10.63125/k9d5h638

[83]. Menezes, B. C., Kelly, J. D., Leal, A. G., & Le Roux, G. C. (2019). Predictive, prescriptive and detective analytics for smart manufacturing in the information age. *IFAC-PapersOnLine*, *52*(1), 568-573.

[84]. Mengist, W., Soromessa, T., & Legese, G. (2020). Method for conducting systematic literature review and meta-analysis for environmental science research. *MethodsX*, *7*, 100777.

[85]. Mezghani, E., Exposito, E., & Drira, K. (2017). A model-driven methodology for the design of autonomic and cognitive IoT-based systems: Application to healthcare. *IEEE Transactions on Emerging Topics in Computational Intelligence*, *1*(3), 224-234.

[86]. Mir, R. R., Reynolds, M., Pinto, F., Khan, M. A., & Bhat, M. A. (2019). High-throughput phenotyping for crop improvement in the genomics era. *Plant Science*, *282*, 60-72.

[87]. Mo, H., Lucca, F., Malacarne, J., & Iacca, G. (2020). Multi-head CNN-LSTM with prediction error analysis for remaining useful life prediction. 2020 27th conference of open innovations association (FRUCT),

[88]. Moens, P., Bracke, V., Soete, C., Vanden Hautte, S., Nieves Avendano, D., Ooijevaar, T., Devos, S., Volckaert, B., & Van Hoecke, S. (2020). Scalable fleet monitoring and visualization for smart machine maintenance and industrial IoT applications. *Sensors*, *20*(15), 4308.

[89]. Mohammad Mushfequr, R., & Sai Praveen, K. (2022). Quantitative Investigation Of Information Security Challenges In U.S. Healthcare Payment Ecosystems. *International Journal of Business and Economics Insights*, *2*(4), 42–73. https://doi.org/10.63125/gcg0fs06

[90]. Mohd Razak, S., & Jafarpour, B. (2020). Convolutional neural networks (CNN) for feature-based model calibration under uncertain geologic scenarios. *Computational Geosciences*, *24*(4), 1625-1649.

[91]. Mortuza, M. M. G., & Rauf, M. A. (2022). Industry 4.0: An Empirical Analysis of Sustainable Business Performance Model Of Bangladeshi Electronic Organisations. *International Journal of Economy and Innovation*. https://gospodarkainnowacje.pl/index.php/issue_view_32/article/view/826

[92]. Mosallam, A., Medjaher, K., & Zerhouni, N. (2016). Data-driven prognostic method based on Bayesian approaches for direct remaining useful life prediction. *Journal of Intelligent Manufacturing*, *27*(5), 1037-1048.

[93]. Muccini, H., & Moghaddam, M. T. (2018). IoT architectural styles: A systematic mapping study. European Conference on Software Architecture,

[94]. Nguyen, C. D., Prosvirin, A. E., Kim, C. H., & Kim, J.-M. (2020). Construction of a sensitive and speed invariant gearbox fault diagnosis model using an incorporated utilizing adaptive noise control and a stacked sparse autoencoder-based deep neural network. *Sensors*, *21*(1), 18.

[95]. Nguyen, H., Bui, X.-N., Tran, Q.-H., & Mai, N.-L. (2019). A new soft computing model for estimating and controlling blast-produced ground vibration based on hierarchical K-means clustering and cubist algorithms. *Applied Soft Computing*, *77*, 376-386.

[96]. Niu, G. (2017). Data-driven technology for engineering systems health management. *Springer Singapore*, *10*, 978-981.

[97]. Nyanchoka, L., Tudur-Smith, C., Iversen, V., Tricco, A. C., & Porcher, R. (2019). A scoping review describes methods used to identify, prioritize and display gaps in health research. *Journal of clinical epidemiology*, *109*, 99-110.

[98]. Okosun, F., Cahill, P., Hazra, B., & Pakrashi, V. (2019). Vibration-based leak detection and monitoring of water pipes using output-only piezoelectric sensors. *The European Physical Journal Special Topics*, *228*(7), 1659-1675.

[99]. Oraee, M., Hosseini, M. R., Papadonikolaki, E., Palliyaguru, R., & Arashpour, M. (2017). Collaboration in BIM-based construction networks: A bibliometric-qualitative literature review. *International journal of project management*, *35*(7), 1288-1301.

[100]. Palvanov, A., & Cho, Y. I. (2019). Visnet: Deep convolutional neural networks for forecasting atmospheric visibility. *Sensors*, *19*(6), 1343.

[101]. Pecori, R. (2018). A virtual learning architecture enhanced by fog computing and big data streams. *Future Internet*, *10*(1), 4.

[102]. Plebe, A., & Grasso, G. (2019). The unbearable shallow understanding of deep learning. *Minds and Machines*, *29*(4), 515-553.

[103]. Pulighe, G., Fava, F., & Lupia, F. (2016). Insights and opportunities from mapping ecosystem services of urban green spaces and potentials in planning. *Ecosystem services*, *22*, 1-10.

[104]. Quatrini, E., Costantino, F., Di Gravio, G., & Patriarca, R. (2020). Condition-based maintenance—an extensive literature review. *Machines*, *8*(2), 31.

[105]. Quax, S. C., Dijkstra, N., van Staveren, M. J., Bosch, S. E., & van Gerven, M. A. (2019). Eye movements explain decodability during perception and cued attention in MEG. *Neuroimage*, *195*, 444-453.

[106]. Radulescu, S., Wijnen, F., & Avrutin, S. (2020). Patterns bit by bit. An entropy model for rule induction. *Language learning and development*, *16*(2), 109-140.
[107]. Rakibul, H., & Samia, A. (2022). Information System-Based Decision Support Tools: A Systematic Review Of Strategic Applications In Service-Oriented Enterprises. *Review of Applied Science and Technology*, *1*(04), 26-65. https://doi.org/10.63125/w3cevz78
[108]. Rauvola, R. S., Vega, D. M., & Lavigne, K. N. (2019). Compassion fatigue, secondary traumatic stress, and vicarious traumatization: A qualitative review and research agenda. *Occupational health science*, *3*(3), 297-336.
[109]. Ren, L., Sun, Y., Cui, J., & Zhang, L. (2018). Bearing remaining useful life prediction based on deep autoencoder and deep neural networks. *Journal of Manufacturing Systems*, *48*, 71-77.
[110]. Ren, L., Sun, Y., Wang, H., & Zhang, L. (2018). Prediction of bearing remaining useful life with deep convolution neural network. *IEEE Access*, *6*, 13041-13049.
[111]. Reza, M., Vorobyova, K., & Rauf, M. (2021). The effect of total rewards system on the performance of employees with a moderating effect of psychological empowerment and the mediation of motivation in the leather industry of Bangladesh. *Engineering Letters*, *29*, 1-29.
[112]. Ribeiro, J. P., & Barbosa-Povoa, A. (2018). Supply Chain Resilience: Definitions and quantitative modelling approaches–A literature review. *Computers & industrial engineering*, *115*, 109-122.
[113]. Rivera, D. L., Scholz, M. R., Bühl, C., Krauss, M., & Schilling, K. (2019). Is big data about to retire expert knowledge? A predictive maintenance study. *IFAC-PapersOnLine*, *52*(24), 1-6.
[114]. Roark, C. L., & Holt, L. L. (2019). Perceptual dimensions influence auditory category learning. *Attention, Perception, & Psychophysics*, *81*(4), 912-926.
[115]. Saikat, S. (2021). Real-Time Fault Detection in Industrial Assets Using Advanced Vibration Dynamics And Stress Analysis Modeling. *American Journal of Interdisciplinary Studies*, *2*(04), 39–68. https://doi.org/10.63125/0h163429
[116]. Saikat, S. (2022). CFD-Based Investigation of Heat Transfer Efficiency In Renewable Energy Systems. *International Journal of Scientific Interdisciplinary Research*, *1*(01), 129–162. https://doi.org/10.63125/ttw40456
[117]. Sakib, N., & Wuest, T. (2018). Challenges and opportunities of condition-based predictive maintenance: a review. *Procedia cirp*, *78*, 267-272.
[118]. Salama, M., Bahsoon, R., & Lago, P. (2019). Stability in software engineering: Survey of the state-of-the-art and research directions. *IEEE Transactions on Software Engineering*, *47*(7), 1468-1510.
[119]. Seele, P. (2017). Predictive Sustainability Control: A review assessing the potential to transfer big data driven 'predictive policing'to corporate sustainability management. *Journal of cleaner production*, *153*, 673-686.
[120]. Shaikh, S., & Aditya, D. (2021). Federated Learning-Driven Predictive Quality Analytics and Supply Chain Optimization In Distributed Manufacturing Networks. *Review of Applied Science and Technology*, *6*(1), 74-107. https://doi.org/10.63125/k18cbz55
[121]. Singha, A., Ali, J., & Khera, V. (2020). Predictive Failure Analysis of Spindle Motor & Cutting Oil Condition Monitoring of Grinding Machine using Artificial Intelligence Models. 2020 IEEE 17th India Council International Conference (INDICON),
[122]. Sohel, A., Alam, M. A., Hossain, A., Mahmud, S., & Akter, S. (2022). Artificial Intelligence In Predictive Analytics For Next-Generation Cancer Treatment: A Systematic Literature Review Of Healthcare Innovations In The USA. *Global Mainstream Journal of Innovation, Engineering & Emerging Technology*, *1*(01), 62-87.
[123]. Srivastava, P., Khanduja, D., & Agrawal, V. (2017). A framework of fuzzy integrated MADM and GMA for maintenance strategy selection based on agile enabler attributes. *Mathematics-in-Industry Case Studies*, *8*(1), 5.
[124]. Sterling, E. J., Betley, E., Sigouin, A., Gomez, A., Toomey, A., Cullman, G., Malone, C., Pekor, A., Arengo, F., & Blair, M. (2017). Assessing the evidence for stakeholder engagement in biodiversity conservation. *Biological conservation*, *209*, 159-171.
[125]. Stoyanchev, S., Maiti, S., & Bangalore, S. (2018). Predicting interaction quality in customer service dialogs. Advanced Social Interaction with Agents: 8th International Workshop on Spoken Dialog Systems,
[126]. Swamy, S. N., & Kota, S. R. (2020). An empirical study on system level aspects of Internet of Things (IoT). *IEEE Access*, *8*, 188082-188134.
[127]. Szymański, G. M., & Tomaszewski, F. (2016). Diagnostics of automatic compensators of valve clearance in combustion engine with the use of vibration signal. *Mechanical Systems and Signal Processing*, *68*, 479-490.
[128]. Tanwar, M., & Raghavan, N. (2020). Lubricating oil remaining useful life prediction using multi-output Gaussian process regression. *IEEE Access*, *8*, 128897-128907.
[129]. Thomé, A. M. T., Scavarda, L. F., & Scavarda, A. J. (2016). Conducting systematic literature review in operations management. *Production Planning & Control*, *27*(5), 408-420.
[130]. Toh, G., & Park, J. (2020). Review of vibration-based structural health monitoring using deep learning. *Applied Sciences*, *10*(5), 1680.
[131]. Tölkes, C. (2018). Sustainability communication in tourism–A literature review. *Tourism management perspectives*, *27*, 10-21.
[132]. Vilarinho, S., Lopes, I., & Oliveira, J. A. (2017). Preventive maintenance decisions through maintenance optimization models: a case study. *Procedia Manufacturing*, *11*, 1170-1177.
[133]. Wagner, A., O'Brien, W., & Dong, B. (2018). Exploring occupant behavior in buildings. *Wagner, A., O'Brien, W., Dong, B., Eds*, *55*, 1267-1273.
[134]. Wan, J., Tang, S., Hua, Q., Li, D., Liu, C., & Lloret, J. (2017). Context-aware cloud robotics for material handling in cognitive industrial Internet of Things. *IEEE Internet of Things Journal*, *5*(4), 2272-2281.

[135]. Wang, H., Li, S., Song, L., & Cui, L. (2019). A novel convolutional neural network based fault recognition method via image fusion of multi-vibration-signals. *Computers in industry*, *105*, 182-190.
[136]. Wong, S. Y., Chuah, J. H., & Yap, H. J. (2020). Technical data-driven tool condition monitoring challenges for CNC milling: a review. *The International Journal of Advanced Manufacturing Technology*, *107*(11), 4837-4857.
[137]. Wu, H., Tian, J., Fu, Y., Li, B., & Li, X. (2020). Condition-aware comparison scheme for gait recognition. *IEEE Transactions on image processing*, *30*, 2734-2744.
[138]. Xia, M., Li, T., Shu, T., Wan, J., De Silva, C. W., & Wang, Z. (2018). A two-stage approach for the remaining useful life prediction of bearings using deep neural networks. *IEEE Transactions on Industrial Informatics*, *15*(6), 3703-3711.
[139]. Xiao, Z., Fu, X., Zhang, L., & Goh, R. S. M. (2019). Traffic pattern mining and forecasting technologies in maritime traffic service networks: A comprehensive survey. *IEEE Transactions on Intelligent Transportation Systems*, *21*(5), 1796-1825.
[140]. Xu, W., Tan, L., Wang, H.-F., Tan, M.-S., Tan, L., Li, J.-Q., Zhao, Q.-F., & Yu, J.-T. (2016). Education and risk of dementia: dose-response meta-analysis of prospective cohort studies. *Molecular neurobiology*, *53*(5), 3113-3123.
[141]. Yacchirema, D. C., Sarabia-Jácome, D., Palau, C. E., & Esteve, M. (2018). A smart system for sleep monitoring by integrating IoT with big data analytics. *IEEE Access*, *6*, 35988-36001.
[142]. Yan, H.-C., Zhou, J.-H., & Pang, C. K. (2016). Machinery degradation inspection and maintenance using a cost-optimal non-fixed periodic strategy. *IEEE Transactions on Instrumentation and Measurement*, *65*(9), 2067-2077.
[143]. Yang, B., Liu, R., & Zio, E. (2019). Remaining useful life prediction based on a double-convolutional neural network architecture. *IEEE Transactions on Industrial Electronics*, *66*(12), 9521-9530.
[144]. Yang, E. C. L., Khoo-Lattimore, C., & Arcodia, C. (2017). A systematic literature review of risk and gender research in tourism. *Tourism Management*, *58*, 89-100.
[145]. Yang, R., Singh, S. K., Tavakkoli, M., Amiri, N., Yang, Y., Karami, M. A., & Rai, R. (2020). CNN-LSTM deep learning architecture for computer vision-based modal frequency detection. *Mechanical Systems and Signal Processing*, *144*, 106885.
[146]. Yin, F., Lin, Z., Kong, Q., Xu, Y., Li, D., Theodoridis, S., & Cui, S. R. (2020). FedLoc: Federated learning framework for data-driven cooperative localization and location data processing. *IEEE Open Journal of Signal Processing*, *1*, 187-215.
[147]. Yongbo, L., Xiaoqiang, D., Fangyi, W., Xianzhi, W., & Huangchao, Y. (2020). Rotating machinery fault diagnosis based on convolutional neural network and infrared thermal imaging. *Chinese Journal of Aeronautics*, *33*(2), 427-438.
[148]. Yoo, Y., & Baek, J.-G. (2018). A novel image feature for the remaining useful lifetime prediction of bearings based on continuous wavelet transform and convolutional neural network. *Applied Sciences*, *8*(7), 1102.
[149]. Zakhezin, A., & Pryadko, Y. (2016). Vibration diagnostics of gas pipelines technological equipment using wavelet analysis. *Procedia Engineering*, *150*, 300-306.
[150]. Zhai, S., Riess, A., & Reinhart, G. (2019). Formulation and solution for the predictive maintenance integrated job shop scheduling problem. 2019 IEEE International Conference on Prognostics and Health Management (ICPHM),
[151]. Zhan, C., Ji, S., Liu, Y., Zhu, L., Shi, Y., & Ren, F. (2018). Winding Mechanical Fault Diagnosis Technique of Power Transformer Based on Time-Frequency Vibration Analysis. 2018 Condition Monitoring and Diagnosis (CMD),
[152]. Zhang, W., Peng, G., Li, C., Chen, Y., & Zhang, Z. (2017). A new deep learning model for fault diagnosis with good anti-noise and domain adaptation ability on raw vibration signals. *Sensors*, *17*(2), 425.
[153]. Zhang, W., Yang, D., & Wang, H. (2019). Data-driven methods for predictive maintenance of industrial equipment: A survey. *IEEE systems journal*, *13*(3), 2213-2227.
[154]. Zhao, D., Wang, T., & Chu, F. (2019). Deep convolutional neural network based planet bearing fault classification. *Computers in industry*, *107*, 59-66.
[155]. Zhao, Z., & Kumar, A. (2018). Improving periocular recognition by explicit attention to critical regions in deep neural network. *IEEE Transactions on Information Forensics and Security*, *13*(12), 2937-2952.
[156]. Zhou, D., Huang, D., Hao, J., Ren, Y., Jiang, P., & Jia, X. (2020). Vibration-based fault diagnosis of the natural gas compressor using adaptive stochastic resonance realized by Generative Adversarial Networks. *Engineering Failure Analysis*, *116*, 104759.
[157]. Zhou, N., Zhong, S., Lin, J., Luo, M., Nsengiyumva, W., Peng, Z., & Yu, Y. (2020). Acoustic-excitation optical coherence vibrometer for real-time microstructure vibration measurement and modal analysis. *IEEE Transactions on Instrumentation and Measurement*, *69*(9), 7209-7217.
[158]. Zhu, L., & Lin, J. (2020). Learning spatiotemporal correlations for missing noisy PMU data correction in smart grid. *IEEE Internet of Things Journal*, *8*(9), 7589-7599.
[159]. Zhu, Z., Han, G., Jia, G., & Shu, L. (2020). Modified densenet for automatic fabric defect detection with edge computing for minimizing latency. *IEEE Internet of Things Journal*, *7*(10), 9623-9636.
[160]. Zobayer, E. (2021a). Data Driven Predictive Maintenance In Petroleum And Power Systems Using Random Forest Regression Model For Reliability Engineering Framework. *Review of Applied Science and Technology*, *6*(1), 108-138. https://doi.org/10.63125/5bjx6963
[161]. Zobayer, E. (2021b). Machine Learning Approaches For Optimization Of Lubricant Performance And Reliability In Complex Mechanical And Manufacturing Systems. *American Journal of Scholarly Research and Innovation*, *1*(01), 61–92. https://doi.org/10.63125/5zvkgg52
[162]. Zona, A. (2020). Vision-based vibration monitoring of structures and infrastructures: An overview of recent applications. *Infrastructures*, *6*(1), 4.