IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 12 February 2026, accepted 17 February 2026, date of publication 23 February 2026, date of current version 26 February 2026.

Digital Object Identifier 10.1109/ACCESS.2026.3667206

RESEARCH ARTICLE

Digital Twin Synthetic Dataset for Bearing Fault Diagnosis in Industrial Spindles

MOHSEN ZEYNIVAND[1], (Member, IEEE), AZADEH KERMANSARAVI[2], (Senior Member, IEEE), HANI VAHEDI[2,3], (Senior Member, IEEE), AND GIAMBATTISTA GRUOSSO[1], (Senior Member, IEEE)

[1]Department of Electronics and Informatics, Politecnico di Milano, 20133 Milan, Italy
[2]Electrical Sustainable Energy, Delft University of Technology, 2628CD Delft, The Netherlands
[3]College of Engineering and Energy, Abdullah Al Salem University, Khaldiya 72303, Kuwait

Corresponding author: Mohsen Zeynivand (mohsen.zeynivand@polimi.it)

This work was supported by Politecnico di Milano and by the MICS (Made in Italy – Circular and Sustainable) Extended Partnership, funded by the Next Generation European Union (EU) initiative under the Italian Piano Nazionale di Ripresa e Resilienza (PNRR) (Mission 4, Component 2, Investment 1.3; D.D. 1551.11-10-2022; Project PE00000004), Codice Unico di Progetto (CUP) D43C22003120001.

ABSTRACT This paper proposes a hybrid digital twin framework that couples a real-time physics-based digital twin model with a data-driven diagnostic layer implemented through cloud-based data acquisition and analysis. This framework generates synthetic datasets across multiple speed levels and fault severities for bearing fault detection and classification in industrial spindle systems, where real fault recordings are costly, risky, and difficult to reproduce. Once the system is validated, a two-stage classifier is trained and used for online fault detection and fault-type identification, whereas the deep-sequence model provides offline verification. To improve robustness, training data are enhanced with multi-domain feature enrichment and targeted data augmentation techniques that simulate measurement noise and small operating variations. The resulting models achieved strong performance under previously unseen operating conditions within the validated digital twin envelope. Overall, the proposed approach reduces the dependence on real fault experiments by enabling the risk-reduced development and evaluation of data-driven bearing fault diagnosis.

INDEX TERMS Ball bearings, data-driven modeling, digital twins, fault diagnosis, fourth industrial revolution, hardware-in-the-loop simulation, predictive maintenance, smart manufacturing, supervised learning, synthetic data.

I. INTRODUCTION

The transition toward Industry 4.0, which accelerates the adoption of predictive maintenance (PdM), where reliable fault detection (healthy vs. faulty operation) and fault-type classification (identifying the fault category) are central to ensuring system reliability, minimizing downtime, and optimizing resource utilization [1], [2]. However, in industrial practice, data-driven diagnosis remains constrained by the scarcity of labeled fault data [3], which are rare, expensive to reproduce, and often unsafe to induce on production assets [4]. Consequently, the available datasets are typically dominated by healthy operation, whereas incipient or uncommon fault modes are underrepresented, which degrades the robustness and generalization of supervised machine learning models [5], [6], [7].

This limitation is particularly pronounced in industrial machine tools (MTs), where reliability and availability are critical and unplanned stops are costly. Among the MT subsystems, rolling element bearings are the dominant degradation modes in spindle drives, and their early diagnosis is essential to prevent secondary damage and reduce downtime [8], [9]. Although modern learning-based methods have achieved strong performance in controlled studies, their effectiveness still depends on the representative training data across operating regimes (e.g., speed) and fault severities [10]. Such coverage is difficult to obtain and maintain in actual plants.

The associate editor coordinating the review of this manuscript and approving it for publication was Gerard-Andre Capolino.

Digital twins (DTs) [11], [12] offer a practical approach to mitigating this bottleneck by enabling controlled, repeatable experimentation on a virtual replica aligned with the physical asset [13], [14]. In manufacturing, DTs have been developed as structured architectures and services [15], [16] and are increasingly connected to PdM pipelines [17], [18].

Nevertheless, DT representations may deviate from real machine behavior because of modeling abstractions, unmodeled dynamics, and parameter uncertainty [19]. Moreover, DTs alone do not fully resolve dataset imbalances and limited variability, because they generate a finite set of modeled scenarios and may not capture all sources of operational variability and measurement uncertainty [20].

Accordingly, DTs should be viewed as controlled data generation and validation tools within specific machines and operating envelopes rather than as substitutes for real-world deployment [11], [12]. Therefore, in this work, we avoid claims of universal generalization across arbitrary machines and instead evaluate the robustness to previously unseen fault severities and operating conditions within a validated DT envelope.

For fault diagnosis, DT-assisted approaches have been explored through sim-to-real transfer and domain adaptation [21], [22], [23], DT parameter updating to reduce model–plant mismatch [19], and "no-fault-data" formulations that learn dynamics from healthy operations and synthesize fault-like signals for training [24], [25]. These directions collectively indicate that DTs can extend beyond monitoring models and serve as data generation and validation tools when real fault capture is limited.

Simultaneously, DT-driven diagnosis is hindered by two practical gaps when targeting deployment-oriented MT spindle systems. First, fidelity and timing: Many DT studies rely on offline simulations or do not explicitly enforce real-time constraints, which limits their ability to generate timing-consistent electromechanical signals that match the execution conditions of embedded/online diagnostics. Real-time hardware-in-the-loop (HIL) DTs address this by executing a physics-based model with deterministic timing [26], thereby enabling repeatable fault scenario replay under realistic sampling and control constraints. However, field mismatch remains a known issue that motivates validation and parameter alignment strategies [19]. Second, even when synthetic data are generated, many pipelines do not target systematic coverage across operating speeds and fault severities, nor do they combine DT-generated data with the explicit variability modeling (noise, small operating perturbations) required for robust classification in practice [20], [27]. Complementary approaches, such as physics-constrained cross-domain generation [28] and noise-robust feature unification for cross-speed transfer [29], address parts of this problem. However, they are typically not coupled to a validated real-time spindle DT executed in an HIL.

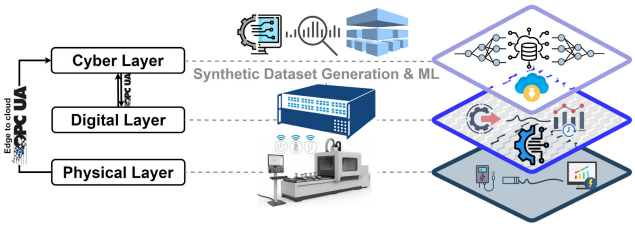

FIGURE 1. HyDT architecture and data flow across the physical, digital, and cyber layers used for real-time HIL execution, synthetic dataset generation, and online/offline diagnosis.

In the MT domain, DT efforts have primarily focused on commissioning, monitoring, or system-level integration rather than on producing machine learning (ML) ready-labeled datasets for supervised fault-type diagnosis. For example, a recent study investigated real-time DTs for spindle-related monitoring and validation [30], whereas related industrial research projects have emphasized DT-enabled measurement integration and sustainability in manufacturing contexts [4], [31]. In parallel, DT-assisted diagnosis has been demonstrated in rotating machinery and bearing-oriented studies using deep learning classifiers or representation learning [32], [33], and DT-enabled transfer strategies have been proposed to mitigate domain shifts between simulated and measured data [19], [22], [23].

However, the closest strands still leave limited evidence of an end-to-end workflow that executes a physics-based spindle DT under real-time HIL constraints, thus validating the DT against operational measurements, and using the validated DT as a disciplined generator of labeled datasets across multiple speeds and fault severities, integrated with a low-latency online diagnosis layer.

Beyond conventional ML and deep learning (DL) baselines, recent research on machinery fault diagnosis has proposed representation-learning architectures that explicitly target robustness to domain shifts, nonstationary regimes, and limited labels. For example, Deep Adversarial Capsule Networks combine capsule-based feature hierarchies with adversarial learning to encourage domain-invariant representations and improve generalization across operating conditions and compound fault scenarios [34]. Wavelet-capsule designs such as WavCapsNet integrate time-frequency multiresolution analysis with capsule routing to capture localized transients and multiscale fault signatures, improving diagnostic robustness under noise and changing regimes [35]. Transformer-based models have also been explored for long-range dependency modeling; in particular, exogenous-feature-aware attention mechanisms explicitly incorporate operating and context variables as conditioning information, which is conceptually aligned with nonstationary industrial monitoring applications [36], [37].

Although these advances strengthen data-driven feature learning, they remain fundamentally dependent on access to sufficiently diverse labeled data across speeds, loads,

and severities. Accordingly, the proposed framework is both model and complementary-oriented. It provides a physics-grounded mechanism to systematically generate labeled multi-condition datasets that can be used for training, stress testing, and benchmarking both lightweight edge models and state-of-the-art deep architectures. In this study, we adopted an OS-ELM-based online pipeline (for low-latency edge deployment) and a CNN-BiLSTM verifier (for offline high-capacity validation), noting that the same HyDT-generated dataset can directly support these advanced architectures.

This study proposes a Hybrid Digital Twin (HyDT) framework for industrial spindle systems (Fig. 1). Here, hybrid denotes system-level hybridization, and a real-time physics-based DT is coupled to a networked data-acquisition pipeline and data-driven diagnostic layer within a single closed workflow. HyDT is distinguished from a conventional DT-assisted ML setup by four minimal conditions: (i) real-time HIL execution to preserve the timing consistency required by online monitoring; (ii) validation of operational measurements to establish a credible operating envelope; (iii) systematic synthetic dataset generation across operating regimes and fault severities within the envelope; and (iv) integration with an online-capable, low-latency diagnostic pipeline evaluated under held-out operating conditions. DT is not a hybrid physics–ML plant model; rather, it provides physically grounded, envelope-consistent labeled signals, whereas ML models learn decision rules for fault detection and fault-type classification when real fault data are limited.

In contrast to DT-assisted diagnosis studies that primarily emphasize transfer/adaptation or DT parameter updating [19], [22], [23], and machine tool DT efforts focused on monitoring/commissioning [30], [31], this work centers on a validated real-time HIL spindle DT explicitly designed for systematic dataset generation across speeds and severities and for integration with deployment-oriented diagnosis. The main contributions of the proposed HyDT framework for bearing fault diagnosis in industrial spindle systems are as follows:

- The real-time HIL spindle DT is validated using operational measurements. A physics-based DT was executed on an OPAL-RT HIL platform and aligned with field data with bidirectional signal exchange via OPC-UA [38].
- Parameterized synthetic dataset design across regimes. Systematic generation of labeled signals across multiple operating speeds and fault severities within the validated DT envelope.
- Controlled variability modeling for robustness. Multi-domain feature enrichment is complemented by targeted augmentation to reflect measurement noise and small operating perturbations [27].
- Deployment-oriented diagnosis using online and offline models. A low-latency two-stage online architecture based on the OS-ELM for fault detection and fault-type classification [39] complemented by a high-capacity offline reference model (CNN-based) for verification [40].

In this study, fault diagnosis refers to the objective of detecting and identifying bearing-related faults. We distinguish between two supervised tasks: fault detection (binary classification: normal vs. fault) and fault-type classification (multi-class classification over the considered fault categories). Finally, the DT was validated mainly in healthy operations; faulty conditions were evaluated via pattern-based external validation using characteristic defect signatures. Therefore, the results reflect the robustness of the validated DT envelope under unseen operating conditions. The goal is to reduce the dependence on real fault data without eliminating it.

The remainder of this paper is organized as follows. Section II describes the real-time DT modeling and communication architectures. Section III presents the fault scenario modeling. Section IV describes the synthetic dataset generation procedure. Section V describes the diagnostic models and their evaluation. Section VI reveals the restrictions of the proposed method and the future trends of this study. Finally, Section VII concludes the paper.

II. REAL-TIME DT FRAMEWORK FOR THE MACHINE TOOL

This section describes the development, implementation, and validation of the proposed HyDT on an OPAL-RT HIL platform. The framework integrates real-time physical measurements with high-fidelity virtual modeling and cloud-based data processing, enabling the generation of synthetic datasets for the ML-based PdM. The overall architecture is shown in Fig. 1, which includes the physical layer, the digital layer (OPAL-RT HIL simulator), the cyber/application layer, and the communication layer, which is responsible for secure data exchange via OPC-UA. Real-time signals continuously flow between the physical machine and DT through edge-to-cloud communication, whereas the application layer handles data enrichment, feature extraction, and ML inference.

A. PHYSICAL ENTITY

DT models rely on heterogeneous streams of high-accuracy, high-frequency data to represent system states and parameters. In MTs, such data can originate from multiple sources [41]. Integrated sensors provide signals such as encoder readings [42], torque, and motor currents [43], whereas external sensors measure vibrations [44] and power consumption [45]. Additionally, production data (e.g., production speed, reliability, and part quality) and control system information (e.g., position and velocity deviations) can be used to evaluate tracking accuracy and overall system performance.

The physical system considered in this study is a high-performance industrial MT employed in precision manufacturing. It features multiple motion axes and a high-speed electrospindle, which play a critical role in the material removal processes. The machine operates under diverse load

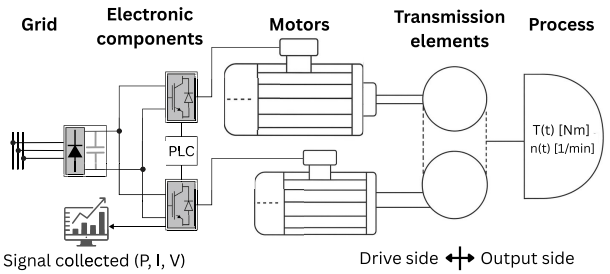

FIGURE 2. Measurement architecture for DT validation, showing acquisition points for spindle drive signals and the data collection chain.

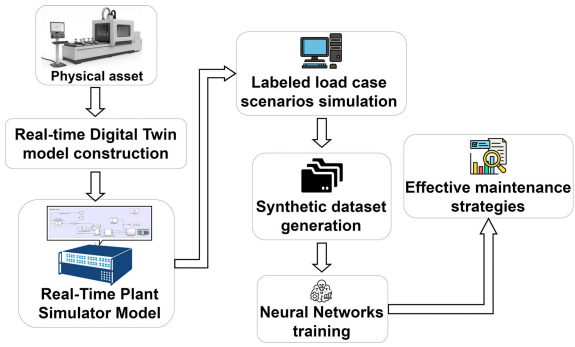

FIGURE 3. HyDT workflow for synthetic dataset generation and diagnosis (DT execution, fault injection, preprocessing, and supervised learning).

conditions and supports different modes, including idle runs, ramp-up sequences, and cutting operations.

To develop and validate the DT model, real-world measurements were acquired from the machine under nominal operating conditions [30], as illustrated in Fig. 2. The data acquisition campaign focused on capturing key electrical and mechanical signals from the main drive units, particularly the electrospindle motor. The following measurements were collected:

- **The three-phase currents** of the electrospindle motor (I_a, I_b, I_c) were recorded using current transducers.
- **Active power** consumption, measured with precision power analyzers at the drive input;
- **Velocity and estimated torque profiles** obtained from drive feedback signals.

All data were acquired using a high-speed system synchronized with the machine controller to ensure alignment between the command inputs and the measured responses. Each test scenario was executed under repeatable conditions with predefined motion or load profiles [30]. In total, 20 test sequences were collected across various operating regimes to validate the virtual model and subsequently generate a synthetic dataset.

B. VIRTUAL ENTITY IN OPAL-RT

Physics-based modeling leverages known physical laws and parameters to describe system behavior, but it relies on simplifications and assumptions and requires detailed knowledge of the system [46]. By contrast, data-driven modeling learns input–output mappings directly from data without prior system knowledge, often through ML or neural networks [3]. Its main limitation is the need for large and diverse datasets, which are difficult to obtain for fault detection when faults are relatively rare.

Given these challenges, the proposed HyDT framework, a system-level hybrid workflow, couples physics-based real-time DT with data-driven diagnostic layers. The plant DT model is physics-based. This coupling mitigates the limitations of each component while retaining their strengths (Fig. 3).

Following this principle, the virtual counterpart of the MT in this study was realized using a high-fidelity Simulink/Simscape model deployed on an OPAL-RT real-time HIL simulator, as illustrated in Fig. 4. The model replicates the drive system, including the electrospindle motor, inverter, and mechanical load, allowing dynamic simulations under both normal and faulty conditions. Furthermore, bearing faults were simulated using a custom MATLAB Simscape component that models torque disturbances as functions of fault severity and spindle speed. This enables the injection of time-varying vibration or torque, providing the flexibility to reproduce realistic degradation patterns.

The virtual system supports the injection of faults such as bearing defects, sudden torque changes, and motor imbalances. This enables systematic scenario testing without risking hardware damage. The main contribution of this study is the identification of different bearing fault types. Each test configuration can be triggered remotely through a Python interface that communicates with the OPAL-RT system, allowing efficient execution of multiple test scenarios and automated data collection.

C. MODEL VALIDATION

An accurate DT-based fault diagnosis requires high-fidelity virtual models of the core components and subsystems of the system to ensure that the DT faithfully reproduces the behavior of the physical machine [30]. To this end, as shown in Fig. 5, a validation process was conducted by comparing the outputs of the virtual entity with experimental measurements obtained from the MT under nominal operating conditions. For an MT, the relevant input variables typically include the motor torque, force, and current, whereas the outputs comprise the linear or rotary position, velocity, acceleration, and power consumption.

Real and simulated data were collected under identical operational profiles (e.g., spindle speed ramps and constant-load phases) to ensure fair, synchronized comparisons. The excitation patterns were designed to sufficiently cover the operating envelope of the system, thereby avoiding overfitting under a single condition.

In practice, the mechanical properties of MTs vary across stroke length and velocity ranges owing to changes in stiffness, friction, and wear. Similarly, the current and

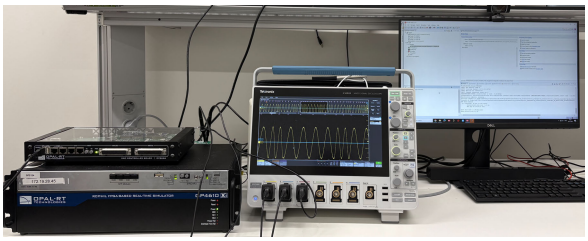

FIGURE 4. OPAL-RT real-time HIL setup used to execute the DT with deterministic timing and to interface with the cloud-based layer for automated scenario triggering and signal logging.

TABLE 1. DT validation metrics (MAE, NRMSE, and R^2) for I_A and P under normal and the pattern-based faulty check.

Condition	Signal	MAE	MAE% working	NRMSE% working	R^2
Normal	I_A	1.0648 A	5.66	13.62	0.927
	P	930 W	5.00	6.75	0.974
Faulty	I_A	0.18 A	4.29	5.5	0.96
	P	112 W	5.0	7.0	0.95

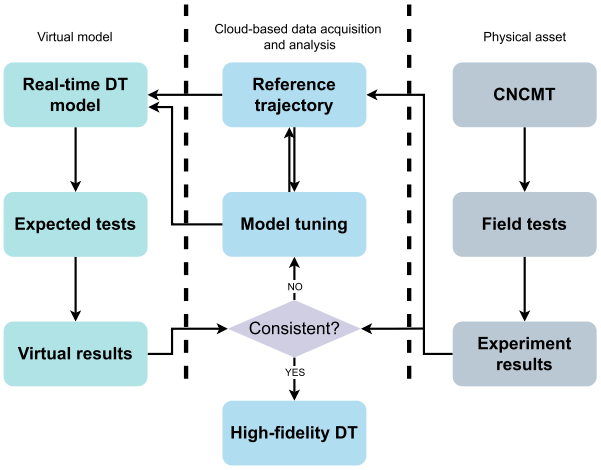

FIGURE 5. Model validation workflow used to align the proposed DT with the physical asset.

power signals exhibit nonlinearities under different operating conditions. To account for these variations, the model outputs from virtual sensors were compared with experimental measurements, and parameter optimization was performed using a genetic algorithm to minimize the prediction error. Once the error converged to a stable minimum, the validation process was considered complete [30]. This validation must be repeated periodically during the machine lifecycle to ensure that the DT remains aligned with the actual system.

For example, Fig. 6 compares the active filtered current (I_A) of the electrospindle during the ramp-up operation. The signals were sampled at 100 kHz over a 74-s interval, demonstrating close agreement between the physical and virtual systems within acceptable error bounds. The quantitative evaluation employed the mean absolute error (MAE), normalized root mean square error (NRMSE), and coefficient of determination (R^2), which are defined as

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N}\left|x_i - \hat{x}_i\right|, \tag{1}$$

$$\mathrm{NRMSE} = \frac{\sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(x_i - \hat{x}_i\right)^2}}{\bar{x}} \times 100, \tag{2}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{N}\left(x_i - \hat{x}_i\right)^2}{\sum_{i=1}^{N}\left(x_i - \bar{x}\right)^2}, \tag{3}$$

where x_i is the measured signal, $\hat{x}_i$ the simulated signal, and $\bar{x}$ the mean measured signal.

The validation results summarized in Table 1 demonstrate that the virtual platform reproduced the electromechanical dynamics of the physical system within acceptable error bounds.

For normal operating conditions, the DT was validated against the physical electrospindle (22.5 kW, 353.3 Hz), reproducing current dynamics with MAE of 1.06 A (5.66%), NRMSE of 13.62%, and $R^2 = 0.927$, while power was presented with MAE of 930 W (5.00%), NRMSE of 6.75%, and $R^2 = 0.974$.

However, direct fault experiments on the target electrospindle are not feasible owing to safety and operational constraints. Therefore, the faulty condition validation in this study is performed in a pattern-based (signature-level) manner rather than through a one-to-one comparison with the measured faulty traces from the same spindle.

Thus, the DT was reconfigured with the 2.24 kW induction motor parameters from [47] to enable a direct comparison with published experimental fault data. Bearing faults were injected using a physics-grounded formulation (4), whose excitation frequencies were computed online from the instantaneous shaft rotational frequency, consistent with the classical bearing fault theory, introducing disturbance torques at characteristic defect frequencies (ball pass frequency inner race: 272 Hz, ball pass frequency outer race: 179 Hz) to ensure physical plausibility.

As shown in Fig. 7, the envelope spectra preserve the diagnostic peak structure (defect frequency, sidebands, and harmonic components), indicating that the synthetic signals retain the physically expected fault signatures observed in the real-world data.

Time- and frequency-domain comparisons, including envelope spectrum analysis, verify that the envelope spectrum exhibits the expected defect-frequency peaks, harmonics, and modulation sidebands, and quantify similarity via peak-location error and spectral correlation. This procedure provides evidence that the injected faults generate physically meaningful diagnostic patterns; however, we explicitly acknowledge that it does not replace direct validation of the faulty condition on the target electrospindle. Accordingly, the reported diagnostic performance should be interpreted as robust within the validated DT operating envelope, and the final deployment would benefit from verification using a small amount of real fault data from the target machine.

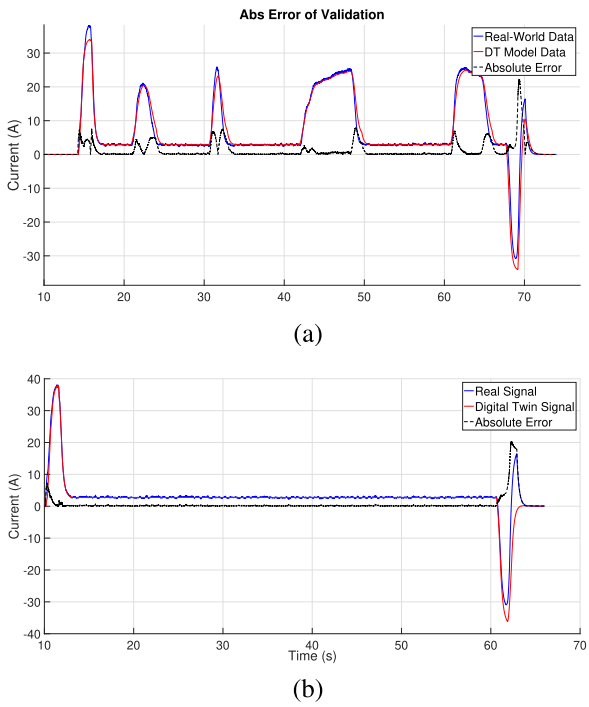

FIGURE 6. Measured vs. DT active current I_A under (a) load and (b) no-load conditions; each panel overlays real signal, DT output, and the absolute error.

These results confirm the capability of DT to replicate high-fidelity electromechanical behavior under both healthy and degraded operating conditions, establishing it as a reliable proxy for exploring operational scenarios and generating synthetic datasets for ML-based PdMs.

III. FAULT SCENARIO SIMULATION

A. BEARING FAULT TYPE

The fault scenarios explored in this study were exclusively focused on electrospindle bearing faults, which represent one of the most critical failure modes in high-speed machine tools [48]. Bearing degradation leads to mechanical imbalance, torque oscillations, and distortions in current waveforms, which often appear as characteristic frequency components in the stator currents and power signals [49].

In the context of this study, we target early-stage localized defects [3], such as ball-cage faults (*BCF*), which physically correspond to the cage-related fault component. Their characteristic frequency is the fundamental train frequency (*FTF*), ball-pass frequency outer race (*BPFO*), inner race (*BPFI*), and ball spin frequency (*BSF*) faults, which are typically associated with modulated current patterns and harmonics at known fault-related frequencies. To ensure robustness of the simulated datasets, each fault type was injected under different load and speed conditions.

The simulated fault symptoms include:

- Modulation of motor torque with a known fault frequency f_{fault}.

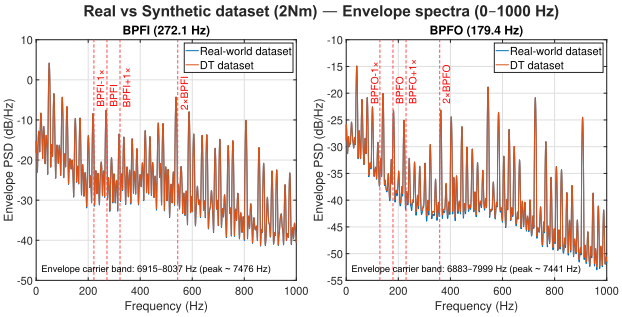

FIGURE 7. Envelope spectrum validation (measured vibration) for (left) *BPFI* and (right) *BPFO* at 2 *Nm*; Red dashed lines indicate, f_d, $f_d \pm f_r$, and $2f_d$ (with f_r).

- The envelope fluctuations and spectral energy spread in the current waveforms were consistent with early localized defect impacts.
- Disturbances in active power, reflecting the energy transfer irregularities induced by bearing faults.

B. FAULT INJECTION STRATEGY

Localized bearing defects generate quasi-periodic contact excitations whose repetition rate is governed by the bearing kinematics and scales with the instantaneous shaft rotational frequency $f_r(t)$. In rotating spindles, these excitations induce small speed fluctuations and torque ripple transmitted through the drivetrain, which in turn modulate the electromagnetic torque balance and the stator current/power via slip and electromechanical coupling. Here, bearing faults are injected via a "Bearing Custom Block" MATLAB Simscape component that applies the disturbance torque defined in (4) [48]:

$$\tau_{\text{fault}}(t) = \alpha\, \tau_{\text{nom}}\, E(t) \sum_{f \in \mathcal{F}} w_f \sin\!\big(2\pi f_f(t)\, t + \phi_f\big), \qquad (4)$$

where τ_{nom} is the nominal spindle load torque; $\alpha \in [0, 1]$ is a dimensionless fault-severity factor; $E(t)$ is an optional envelope modeling gradual degradation; w_f and ϕ_f are the corresponding weighting factors and initial phases, respectively $f_f(t)$ denotes the defect characteristic frequency computed online from the instantaneous shaft rotational frequency $f_r(t)$ (Hz), and $\mathcal{F} \in \{\text{BPFO, BPFI, BCF, BSF}\}$ is the set of modeled fault components.

Accordingly, the proposed strategy injects a speed-dependent disturbance torque at the characteristic defect frequencies, $\tau_{\text{fault}}(t)$ in (4), where $f_f(t)$ is computed online from $f_r(t)$ and the disturbance amplitude scales with the nominal load torque τ_{nom} through the severity factor α. This formulation targets the dominant low-to-mid frequency modulation cues that are typically exploited in electrical signal-based diagnosis, spectral lines near f_d and its harmonics, and modulation sidebands around $f_d \pm \kappa f_r$ arising from the amplitude modulation of the fault-related ripple. This disturbance torque model targets dominant modulation cues in current and power; the limitations are discussed in Section VI.

To systematically explore the operating variability, we exercised the model over a discrete grid of fault severity and speed levels. The fault severity factor was selected as $\alpha \in \{0.05, 0.10, 0.15, 0.20, 0.28, 0.30, 0.33\}$, corresponding to a disturbance torque amplitude $\tau_{\text{amp}} = \alpha\, \tau_{\text{nom}}$. The spindle speed was swept over $f_r \in \{0.50, 0.57, 0.62, 0.71, 0.85, 1.00\} \times f_{r,\text{ref}}$, with $f_{r,\text{ref}} = 353.3$ Hz.

For each fault type, severity, and speed combination, as well as for the three baseline healthy cases at the same speeds, DT records the stator phase current $I_a(t)$ and active power $P(t)$. These signals served as the basis for a synthetic dataset. An external trigger input enables repeatable on/off fault transients, and an optional vibration port allows additive broadband perturbations when an envelope or impact-like behavior is desired.

This implementation yields controlled, repeatable fault synthesis without risking the real machine, while preserving the correct causality (defect → torque ripple → electrical signature).

C. SIGNATURE-LEVEL VALIDATION

To provide a physically grounded check that the disturbance-torque formulation in (4) produces discriminative fault signatures in the electrical measurements, we performed a signature-level sensitivity analysis directly on the DT-generated $I_a(t)$. For each steady-state record, we computed the envelope spectrum of $I_a(t)$ and extracted the matched spectral peak within a narrow neighborhood of the injected defect line $f_d = \kappa f_r$, where the DT uses $\kappa = \{4.93, 3.07, 2.03, 0.38\}$ for *BPFI*, *BPFO*, *BSF*, and *BCF*, respectively. Figure 8 shows that the normalized peak location f_{peak}/f_r concentrates tightly around the corresponding κ values across the validated operating speeds and severity levels, confirming that DT injection yields consistent, fault modulation components rather than arbitrary spectral artifacts.

D. PREPROCESSING FOR SYNTHETIC DATASET

Each experiment was executed on the real-time DT of the spindle drive, in which the bearing fault model was embedded within the electromagnetic equations of the machine. Communication with the simulator was handled via an OPC-UA link, and a Python-based client running in the cloud transmitted the reference speed and torque commands, triggered the real-time execution on the OPAL-RT, and collected the resulting feedback signals. After every run, the cloud service automatically stores the stator phase current $I_a(t)$ and instantaneous active power $P(t)$, enabling the unattended batch generation of multiple fault scenarios.

Each simulation lasted 35 s and was sampled at 100 kHz to preserve high-frequency fault-related components. From each trace, we extracted a steady-state interval after the spindle reached the target speed and the injected torque disturbance was stabilized. Unless otherwise stated, the analysis window was $t \in [21, 25]$ s, whereas for a limited subset of runs, the bounds were adjusted (within a few seconds) to match the steady-state region identified from the measured signals.

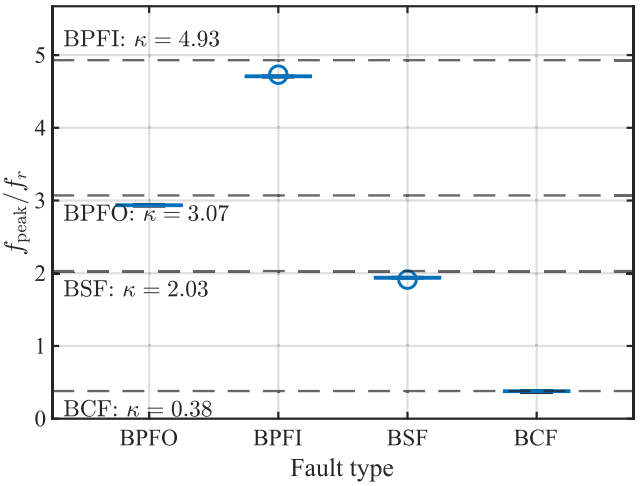

FIGURE 8. Signature-level sensitivity of the DT fault injection; normalized envelope-spectrum peak location f_{peak}/f_r, where f_{peak} is the maximum within a narrow band around the injected defect line $f_d = \kappa f_r$.

Because the number of samples within this interval varies with spindle speed, each window is segmented into different time steps. This normalization ensures that all operating conditions are represented with consistent temporal resolution while retaining the dominant dynamics and modulation patterns.

IV. SYNTHETIC DATASET GENERATION

A. DATA AUGMENTATION OF DT SIGNALS

The synthetic dataset generation process was designed to transform the continuous output of the DT into structured ML-ready samples that accurately represented the electrical and mechanical dynamics of the spindle system. Although DT produces realistic current and power waveforms under various conditions, these raw time series must be reformatted into short, consistent segments to enable supervised learning and temporal analysis [50].

For each DT operating condition, we extract one steady-state record of duration $T_{\text{rec}} = 4$ s comprising the phase-a stator current $I_a(t)$ and corresponding active power $P(t)$. Each record is then segmented into fixed-length windows of $T_w = 0.6$ s with a stride of $T_s = 0.1$ s. For learning-model compatibility and to bound the input dimensionality for online inference, each window is resampled to a fixed length of $N = 60$ samples per channel, yielding a compact 2×60 representation of $\{I_a, P\}$. This resampling step normalizes the input size across the operating points; it does not increase the physical spectral resolution, which is dictated by the window length T_w.

In addition to the time-domain representation, each window was characterized in the frequency domain. Specifically, the one-sided FFT magnitude spectrum is computed on the windowed signal at the native sampling rate and then compressed to a fixed-length descriptor of 60 bins

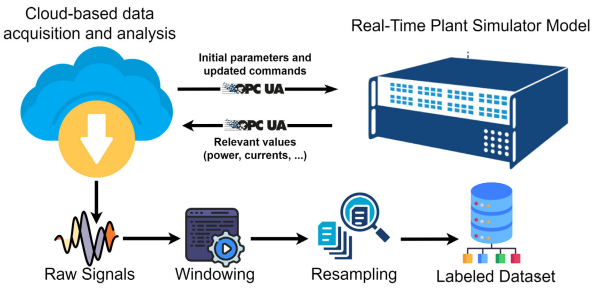

FIGURE 9. End-to-end acquisition and preprocessing pipeline for synthetic dataset construction; cloud client triggers OPAL-RT scenarios via OPC-UA, logs $I_a(t)$ and $P(t)$, windows, and resamples signals, and stores domain, representations with labels.

by band-averaging the magnitude over uniformly spaced frequency intervals. This produces a compact spectral feature vector with consistent dimensionality across all scenarios, while preserving the T_w-limited frequency resolution.

Each sample is annotated with its corresponding fault type, spindle speed, and excitation amplitude. Figure 9 outlines the complete acquisition and processing chain from cloud communication and real-time simulation to steady-state extraction, windowing, and fixed-length encoding.

To illustrate the resulting data representation, Fig. 10 presents examples of resampled active current and power signals for both minimum and maximum spindle speeds under different conditions and severities. A more detailed example of a single acquisition is shown in Fig. 11. The results show the complete active power and stator current waveforms for a representative case, together with the selected steady state window, corresponding to the resampled signals, and frequency-domain representation of the current.

The $N = 60$ encoding is compact, yet the examples in Fig. 10 and Fig. 11 show that class-dependent waveform and energy-pattern differences remain observable after preprocessing. To support the selection of (T_w, N) beyond qualitative inspection, we also report a sensitivity analysis over the representative values of T_w and N, demonstrating that the adopted setting provides a favorable tradeoff for online deployment.

Although HyDT generates fault scenarios with high physical fidelity, its outputs are limited to modeled conditions and do not fully capture the stochastic variability observed in real machines. To address this limitation, two complementary data preparation strategies were employed: data augmentation to increase sample diversity and feature enrichment to enhance signal representation.

Data augmentation involves applying controlled transformations to each resampled segment in the training set only. These transformations emulate realistic acquisition and operating perturbations (sensor noise and minor speed ripple), increasing the sample diversity while preserving the label-informative fault harmonics and modulation patterns. The validation and test subsets were kept clean for an unbiased evaluation.

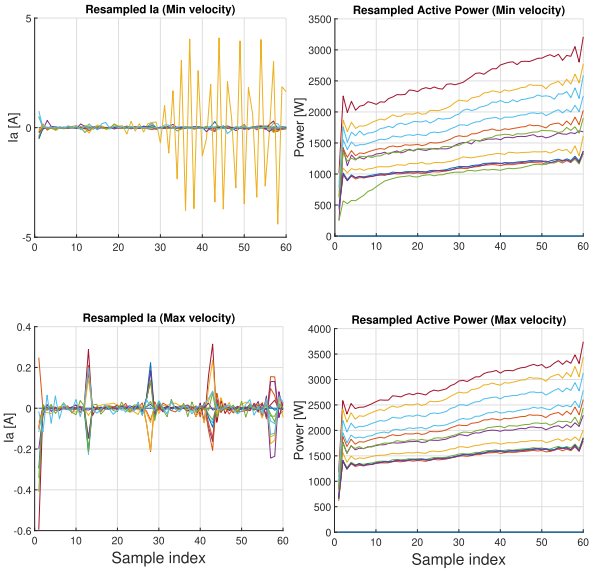

FIGURE 10. Examples of 60-point resampled inputs for I_a and P at minimum and maximum spindle speeds.

TABLE 2. Augmentation parameters applied to resampled segments.

Transform	Value
Gaussian noise	$\sigma = 0.02$ (normalized/resampled domain)
Time warping	warp amplitude 0.04, $f_w \in [0.5, 1.5]$ cycles/window
Envelope jitter	$\epsilon \sim \mathcal{N}(0, 0.06^2)$

For each resampled segment $\mathbf{x}$, we generated an additional augmented sample by injecting small perturbations with fixed magnitudes chosen to remain in a "low-distortion" regime: additive Gaussian noise (5), mild time warping to emulate small within-run speed fluctuations (6), and envelope jitter to account for minor variability in the demodulated amplitude (7).

$$\mathbf{x}' = \mathbf{x} + \eta, \qquad \eta \sim \mathcal{N}(0, \sigma^2). \tag{5}$$

$$t_1 = t_0\,[1 + 0.04\sin(2\pi f_w t_0 + \psi)]\,. \tag{6}$$

$$\mathrm{Env}' = \mathrm{Env} \cdot (1 + \epsilon), \qquad \epsilon \sim \mathcal{N}(0, 0.06^2). \tag{7}$$

These values were selected to be sufficiently small to avoid creating an artificial fault signature while improving the invariance to realistic measurement noise and minor operating variability.

Feature enrichment employs a multi-domain representation approach in which each 4-second window is decomposed into complementary signal representations: the original time-domain signal, its temporal derivatives (deltas) capturing the rate of change, and frequency-domain features revealing spectral characteristics. These representations were concatenated into comprehensive feature vectors to provide a model with complementary perspectives on the signal behavior without increasing the number of physical samples.

It is important to mention that to prevent data leakage, as mentioned in Table 3, all datasets were split at the raw run (file) level before windowing. Sliding windows

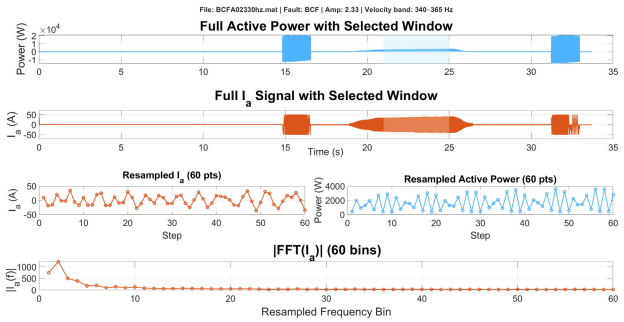

FIGURE 11. Illustrative preprocessing for one experiment: selected steady-state window, corresponding resampled waveforms, and FFT magnitude of I_a.

are generated only after the file-level split, so that the windows derived from a given raw run cannot appear in multiple subsets. Data augmentation was applied only to the training subset to improve the robustness. The validation and test subsets remained clean (non-augmented) to ensure that the reported metrics reflected strict generalization rather than robustness to perturbations introduced during data augmentation. This protocol evaluates the generalization of unseen fault severities under the measured operating speeds. Specifically, the test subset includes the conditions (α_2 and α_5 captured in $f_{r,5}$ and $f_{r,3}$, respectively) that were withheld from training and validation, while the corresponding speed/severity levels remained represented in training via other severities/speeds (Table 3). Testing on entirely unseen speeds is not emphasized here because spindle speed is a directly measured operating condition in practical monitoring systems and strongly reshapes the stator-current signatures. Accordingly, we treated speed as an observed variable and target robustness across unseen fault severities within the validated DT operating envelope.

The resulting dataset achieved a balanced representation of all four fault categories at different operating speeds. Representative examples of parts of the augmented current signals are shown in Fig. 12, where the key waveform characteristics remained intact despite the introduced perturbations.

B. DATASET STRUCTURE

After windowing and augmentation, the synthetic dataset consists of hundreds of labeled signal samples covering all relevant bearing fault conditions.

The windowing and augmentation scheme produced samples from the original DT runs, thereby providing a statistically rich foundation for training the ML architectures. Table 3 illustrates the balanced representation achieved across fault categories after augmentation and windowing, summarizing the dataset distribution for the online multi-class OS-ELM classifier and the CNN verification stage, respectively. To prevent information leakage, the dataset was split at the file level prior to any windowing or augmentation. Each raw run contributes one steady-state record of duration $T_{\text{rec}} = 4$ s. The sliding windows of length T_w and stride T_s

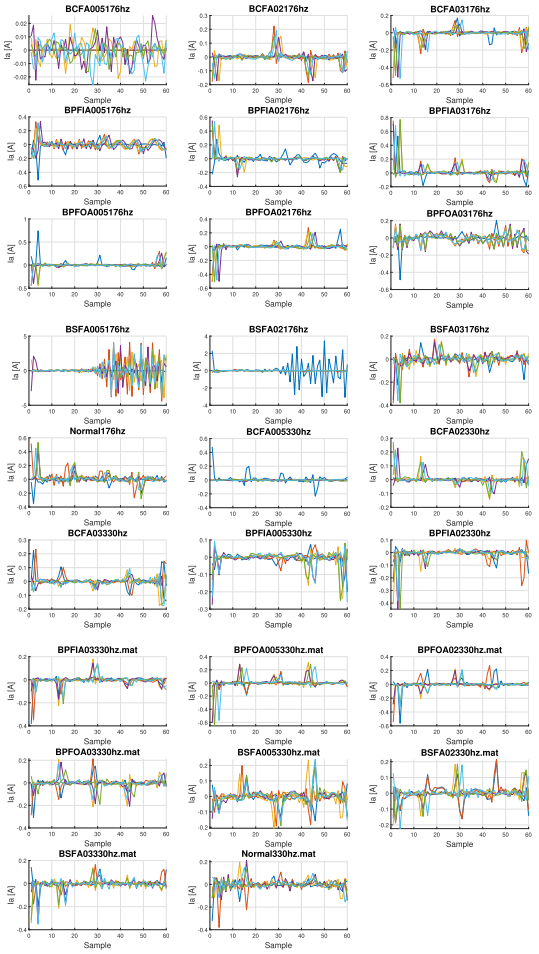

FIGURE 12. Representative augmented I_a segments across fault classes and two speed levels; file labels follow *[Fault][Severity][Speed]* (e.g., *BCFA005176hz* denotes the *BCF* class with severity index A005, i.e., $\alpha = 0.05\,T_{\text{nom}}$, at $f_{r,1} = 176$ Hz).

are extracted only *after* the split, yielding

$$N_{\text{win}} = \left\lfloor \frac{T_{\text{rec}} - T_w}{T_s} \right\rfloor + 1. \tag{8}$$

With $T_w = 0.6$ s and $T_s = 0.1$ s, this results in $N_{\text{win}} \approx 34$ windows per raw run (owing to discrete-time indexing after steady-state cropping and sample rounding). Augmentation is applied only to the training windows, and the validation and test sets remain clean. It is necessary to note that to replicate a more realistic scenario-level sense, we performed an additional holdout evaluation at a known operating speed ($f_r = 300$ Hz) using two severity levels $\alpha = \{0.25, 0.31\}T_{\text{nom}}$ that were not used during training. This would bring us two more unique records and 68 windowed data points per fault.

- **Time-domain matrix:** 2×60 array of I_a and P time-series data
- **Multi-domain feature set:** Comprehensive representation including temporal derivatives and frequency-domain magnitude spectra for both I_a and P

TABLE 3. Dataset distribution used for the proposed fault diagnosis models: numbers of unique records, windowed samples, and augmented samples per subset.

Dataset distribution		Speed level	Severity factor	Fault type				Normal	Total
				BPFO	BPFI	BCF	BSF		
Unique records	Train	$f_{r,1}, f_{r,4}, f_{r,6}$	$\alpha_1, \alpha_3, \alpha_4, \alpha_6, \alpha_7$	8	8	8	8	3	35
		$f_{r,2}$	α_7	1	1	1	1	1	5
		$f_{r,3}$	α_2	1	1	1	1	1	5
		$f_{r,5}$	α_5	1	1	1	1	1	5
	Val	$f_{r,2}$	α_4	1	1	1	1	–	4
	Test	$f_{r,3}$	α_5 (unseen)	1	1	1	1	–	4
		$f_{r,5}$	α_2 (unseen)	1	1	1	1	–	4
Windowed	Train	$f_{r,1}, f_{r,4}, f_{r,6}$	$\alpha_1, \alpha_3, \alpha_4, \alpha_6, \alpha_7$	272	272	272	272	102	1190
		$f_{r,2}$	α_7	34	34	34	34	34	170
		$f_{r,3}$	α_2	34	34	34	34	34	170
		$f_{r,5}$	α_5	34	34	34	34	34	170
	Val	$f_{r,2}$	α_4	34	34	34	34	–	136
	Test	$f_{r,3}$	α_5 (unseen)	34	34	34	34	–	136
		$f_{r,5}$	α_2 (unseen)	34	34	34	34	–	136
Augmented (added)	Train	$f_{r,1}, f_{r,4}, f_{r,6}$	$\alpha_1, \alpha_3, \alpha_4, \alpha_6, \alpha_7$	272	272	272	272	102	1190
		$f_{r,2}$	α_7	34	34	34	34	34	170
		$f_{r,3}$	α_2	34	34	34	34	34	170
		$f_{r,5}$	α_5	34	34	34	34	34	170
	Val	–	–	–	–	–	–	–	–
	Test	–	–	–	—	–	–	–	–

- **Metadata:** Labels for fault type, spindle speed, excitation amplitude, and window index to support model training and evaluation

Figure 13 shows the averaged resampled waveforms for each fault class, confirming that despite augmentation, the distinctive shapes and relative energy profiles of the faults were preserved. The processed and labeled data, stored in the dataset file, form the standardized input for the classification models described in Section V.

V. DT–DRIVEN MACHINE LEARNING FRAMEWORK FOR FAULT DIAGNOSIS

Following HyDT validation and systematic enrichment of synthetic datasets through windowing and resampling, this section presents a two-level ML framework for bearing fault diagnosis. As shown in Fig. 14, the framework first detects abnormal operating conditions, and then precisely identifies specific fault types using electrical measurements of the phase current (I_a) and active power (P).

Within this architecture, HyDT functions as a high-fidelity generator of labeled signal data, whereas the learning algorithms constitute an analytical layer that translates the signal dynamics of diagnostic intelligence. The integrated workflow combines physics-based simulations, feature-based representations, and data-driven inferences into a continuous cycle of detection, classification, and adaptation. Each windowed segment of the current and power signals was transformed into comprehensive time- and frequency-domain features that characterized the energy distribution, waveform morphology, and transient patterns, attributes directly correlated with mechanical health and defect severity.

All signals were z-score normalized (parameters fit on the training subset only) and split into train/val/test using stratified sampling.

The diagnostic framework was organized into three sequential stages, as shown in Fig. 14.

1) **Stage-1, fault detection:** The binary OS-ELM model first discriminates between normal and faulty operating conditions, and acts as the primary diagnostic filter.
2) **Stage-2, fault classification:** Once a fault is detected, the corresponding signal window is processed using an OS-ELM classifier operating at the edge to perform a real-time multi-class analysis.
3) **Stage-3, offline verification:** An offline deep CNN–BiLSTM model was used for performance benchmarking and continuous monitoring of the real-time classifier.

This dual-classifier approach leverages complementary data preparation strategies tailored to the architectural strengths of each model. The CNN processes raw time-domain signals to automatically learn hierarchical features, whereas the OS-ELM utilizes enriched multi-domain representations for rapid sequential learning.

Window Length and Resampling Rationale: To obtain fixed-size inputs while preserving diagnostically relevant signal structures, the current and power signals were divided into windows of duration T_w, and each window was resampled to N points before classification. In a discrete implementation, this corresponds to an effective resampling rate, $f_{s,\text{eff}} = N/T_w$.

The finite window length fundamentally limits the achievable frequency resolution, that is $\Delta f \approx 1/T_w$; longer

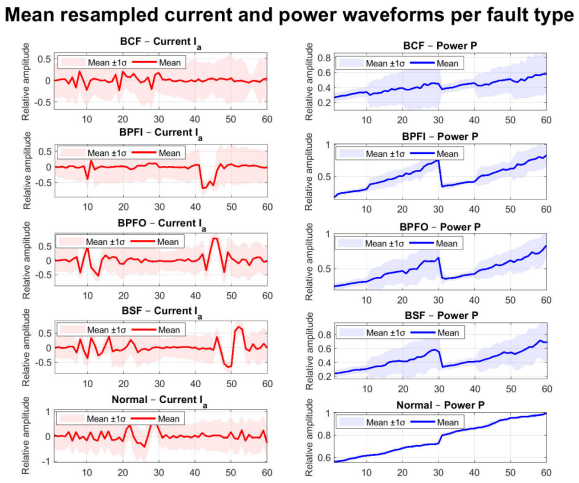

FIGURE 13. Representative check of augmentation fidelity after preprocessing for I_a and P, indicating increased variability without distorting the underlying pattern.

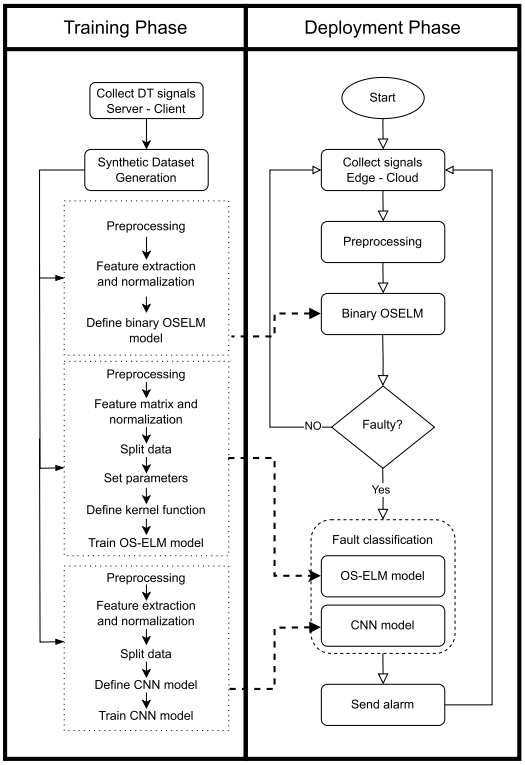

FIGURE 14. Fault detection and classification pipeline.

windows improve the frequency resolution at the cost of temporal localization [51], [52].

In this study, the shaft rotational frequency spans approximately 176.6–353.3 Hz across the validated operating points ($f_r \in [f_{r,1}, f_{r,\mathrm{ref}}]$), therefore a T_w window contains approximately $T_w f_r$ shaft rotational frequency, supporting stability of the modulation pattern while maintaining a low detection latency. With $T_w = 0.6$ s, the corresponding resolution is $\Delta f \approx 1.67$ Hz, which is sufficient to separate the characteristic bearing fault frequencies from their modulation sidebands around the shaft frequency [47], [48].

In practice, the choice of the window length T_w and fixed-length encoding size N is constrained by both the HyDT update cycle and the need for a compact, unified input interface shared by all diagnostic stages. Here, OPC-UA exchanges scenario updates at 1 s granularity, which imposes the practical upper bound $T_w \leq 1$ s. Very short windows, on the other hand, make demodulation and envelope statistics more sensitive to transient fluctuations, edge effects, and noise; therefore, we restrict the sweep to $T_w \geq 0.2$ s. Within this feasible range, we performed a systematic sensitivity sweep over $T_w \in [0.2, 1.0]$ s and $N \in [20, 120]$ to assess how the compact representation preserves fault-informative modulation and envelope structure.

To decouple preprocessing selection from downstream model training, we evaluate representation stability using two criteria: (i) the relative deviation of a fixed-dimensional descriptor aligned with OS-ELM, denoted $\epsilon_\phi^{\mathrm{bin}}$, computed with respect to high-resolution reference encoding, and (ii) envelope-shape deviation after per-window normalization, $\epsilon_\phi^{\mathrm{mc}}$, consistent with fault type discrimination. Both measures are computed with respect to a high-resolution reference encoding, so that lower values indicate higher descriptor preservation. As shown in Fig. 15 (a), at $T_w = 0.6$ s, $\epsilon_\phi^{\mathrm{bin}}$ remains low for approximately $N \approx 50$–70, while larger N increases the input dimensionality (and thus computation burden) without commensurate improvement in descriptor deviation across stages.

Similarly, Fig. 15 (b) shows that for a fixed compact encoding ($N = 60$), both $\epsilon_\phi^{\mathrm{bin}}$ and $\epsilon_\phi^{\mathrm{mc}}$ remain near their low-error regime over a broad region around $T_w \approx 0.55$–0.65 s, indicating stable descriptor preservation while avoiding unnecessary latency increase. Based on these trends and the requirement to enforce a single preprocessing configuration across all stages, we adopt $(T_w, N) = (0.6\ \mathrm{s}, 60)$ as a deployable compromise within the empirically stable region. The downstream validation results reported in the following further support that this common configuration yields a reliable performance across the binary OS-ELM detector, multi-class OS-ELM classifier, and CNN-BiLSTM verifier.

Significantly, resampling does not change the physical spectral resolution dictated by T_w; it normalizes only the input size of the learning models [51], [52]. When down-sampling or interpolation is needed, a light low-pass filter is applied to suppress aliasing, preserving the statistical and envelope characteristics that carry most of the diagnostic information.

This representation strikes a balance between fidelity and compactness, exhibiting good class separability in exploratory analysis. Figure 16 shows the correlation and separability of the extracted features, confirming that the adopted representation effectively distinguishes between

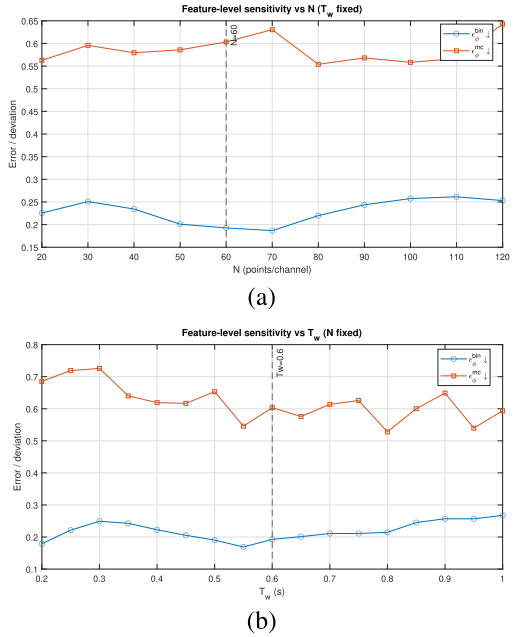

FIGURE 15. Sensitivity of descriptor-level stability to (a) encoding size N with fixed $T_W = 0.6$ s and (b) window length T_W with fixed $N = 60$.

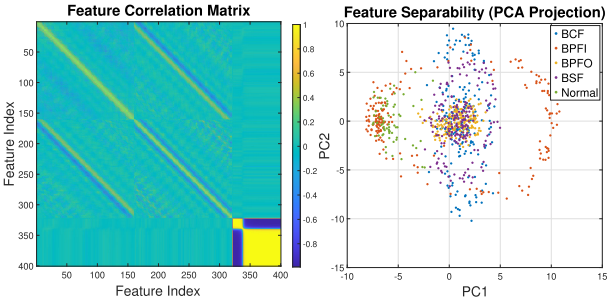

FIGURE 16. Feature correlation matrix and PCA visualization of the enriched feature space used by OS–ELM, highlighting redundancy structure among features and qualitative separability of classes after augmentation.

normal and faulty conditions, and further discriminates among the four fault types.

Furthermore, Fig. 17 shows the projection of the learned features used in the fault classification pipeline onto the first two principal components obtained through PCA. These components accounted for 86% of the total variance ($PC1 = 47\%$, $PC2 = 39\%$), indicating that the $2D$ projection preserved most of the informative structure. The four bearing conditions (*BCF, BPFI, BPFO,* and *BSF*) appear as clearly separated and compact clusters, demonstrating that the feature extractor produces discriminative, well-organized embedding. The evident separability in the PCA space confirms the robustness and suitability of the learned representation for reliable multi-class fault classification.

A. MODEL TRAINING AND COMPARATIVE EVALUATION

The ML stage followed a structured comparative design to evaluate the performance and generalization of each classifier under consistent data conditions.

1) STAGE-1: BINARY OS-ELM CLASSIFIER (NORMAL VS. FAULT)

To ensure prompt detection of abnormal behavior during spindle operation, the first diagnostic gate was a binary OS-ELM. In real time, each incoming window is labeled as either a *Normal* or *Fault*, independent of the fault type.

The hidden layer employs L radial basis functions, defined in (9),

$$\phi_j(\mathbf{x}) = \exp\left(-\gamma \left\|\mathbf{x} - \mathbf{c}_j\right\|^2\right), \tag{9}$$

where $\gamma > 0$ is the hidden layer radial basis function (RBF) width parameter, and the centers $\mathbf{c}_j$ are obtained via k-means.

The OS-ELM sequentially updates the output weights using the recursive least-squares rule in (10) and (11).

$$\boldsymbol{\beta}_k = \boldsymbol{\beta}_{k-1} + \mathbf{K}_k\left(\mathbf{t}_k - \boldsymbol{\Phi}_k\boldsymbol{\beta}_{k-1}\right), \tag{10}$$

$$\mathbf{K}_k = \mathbf{P}_{k-1}\boldsymbol{\Phi}_k^\top\left(\mathbf{I} + \boldsymbol{\Phi}_k\mathbf{P}_{k-1}\boldsymbol{\Phi}_k^\top\right)^{-1}, \tag{11}$$

where $\boldsymbol{\Phi}_k$ is the hidden layer activation matrix for the incoming mini batch.

The binary OS-ELM model was intentionally compact, providing very low latency. Its role is solely to determine whether the machine is operating abnormally; any window flagged as "*Fault*" is immediately forwarded to stage-2.

2) STAGE-2: ONLINE FAULT-TYPE CLASSIFICATION (MULTI-CLASS OS-ELM)

Upon detection of a fault, the same window is analyzed by a *multi-class OS-ELM* configured to discriminate among *BPFI*, *BPFO*, *BSF*, and *BCF*. This classifier utilizes a more comprehensive feature set and a greater number of hidden nodes, thereby enhancing its capacity to discern subtle spectral patterns associated with each defect mechanism. The model shares the same RBF activation, but operates with a higher capacity than the binary detector. Because it is invoked only after a fault is detected, the system maintains high efficiency during normal operation.

This stage provides real-time identification of the fault type and delivers early actionable insights into the DT or maintenance system.

3) STAGE-3: AN OFFLINE CNN-BILSTM FOR VERIFICATION

After each machine runs, the complete signal is analyzed by a 1-D deep *CNN-BiLSTM* that serves as a high-fidelity reference model and operates on a fixed-length multi-channel window. This architecture integrates convolutional layers for local pattern extraction, bidirectional LSTMs for temporal context integration, and a self-attention mechanism for adaptive feature weighting. Each window is represented as a sequence with C channels and N resampled points.

The convolutional operation is defined in (12), the BiLSTM output is formed, as shown in (13), and the attention

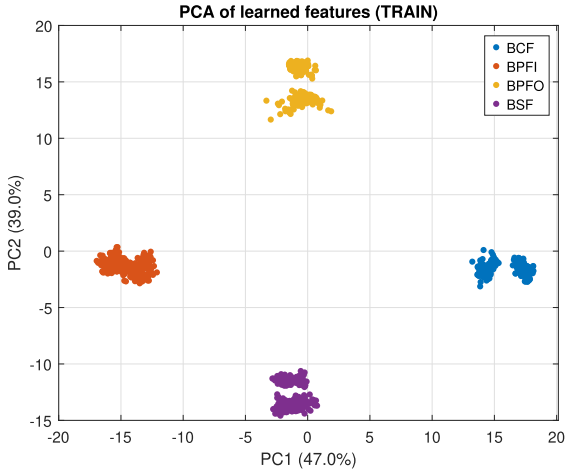

FIGURE 17. PCA projection of learned features (training subset) showing cluster separability among bearing fault classes (*BCF, BPFI, BPFO, BSF*); axis labels indicate explained variance of PC1 and PC2.

weights follow (14).

$$\mathbf{h}_l^{(t)} = \sigma\left(\mathbf{W}_l * \mathbf{h}_{l-1}^{(t)} + \mathbf{b}_l\right). \tag{12}$$

$$\mathbf{h}_t = \left[\overrightarrow{\mathbf{h}}_t; \overleftarrow{\mathbf{h}}_t\right]. \tag{13}$$

$$\alpha_{ij} = \frac{\exp\left(\mathbf{q}_i^\top \mathbf{k}_j/\sqrt{d_k}\right)}{\sum_j \exp\left(\mathbf{q}_i^\top \mathbf{k}_j/\sqrt{d_k}\right)}. \tag{14}$$

Following the dataset construction step, we enriched the input by adding two envelope-derived channels (temporal envelope derivative and normalized envelope), yielding C=5 channels. Each channel was standardized per window via z-score normalization along the temporal dimension. The network architecture and training hyperparameters are presented in Tables 4 and 5, respectively. The validation was performed on a held-out file-level split (no overlap in FileId between splits). To avoid optimistic bias, data augmentation was applied only to the training subset, whereas the validation and test subsets remained clean.

Owing to its computational cost, the CNN-BiLSTM operates offline. Its purpose is to validate and refine real-time OS-ELM classification, ensuring high diagnostic reliability for reporting and archival purposes.

This pipeline achieved immediate fault detection, accurate fault-type classification, and robust offline validation, making it suitable for real-time edge deployment within an industrial DT framework.

B. RESULTS AND DISCUSSION

This section evaluates the performance of the proposed diagnostic framework consisting of an online binary OS-ELM detector, an online multi-class OS-ELM classifier, and an offline CNN-BiLSTM verification network.

All models were trained exclusively on the HyDT dataset generated from a real-time HIL environment, whereas testing was performed on unseen raw DT signals corresponding to the operating conditions and fault severities that were not used during training. Therefore, the test subset performance

TABLE 4. CNN verifier architecture.

Block	Configuration
Input	Sequence input, C=5 channels, length N
Conv-1	1-D conv, kernel 3, filters 64, padding same
	BatchNorm + ReLU
	MaxPool, pool/stride 2
Conv-2	1-D conv, kernel 3, filters 128, padding same
	BatchNorm + ReLU
	MaxPool, pool/stride 2
Regularization	Dropout p=0.35
Temporal model	BiLSTM, 128 hidden units, output mode: last
Regularization	Dropout p=0.45
Classifier head	FC 64 + ReLU + Dropout p=0.35
Output	FC $\lvert\mathcal{F}\rvert$ + Softmax
Loss	Focal cross-entropy, γ=1.5

TABLE 5. CNN training hyperparameters and evaluation protocol.

Item	Value
Optimizer	Adam
Max epochs	150
mini batch size	32
Initial learning rate	3×10^{-4}
LR schedule	Piecewise; drop factor 0.7 every 25 epochs
Gradient clipping	Threshold 1
L_2 regularization	10^{-4}
Shuffle	Every epoch
Validation frequency	Every 40 iterations
Model selection	Final epoch

quantifies the generalization across unseen DT scenarios (speed/severity cells) within the validated twin envelope and is not transferred to real machine faulty measurements.

1) BINARY DETECTION

For the binary OS-ELM, RBF nodes were initialized via k-means clustering, and the output weights were updated sequentially using a recursive least-squares (RLS) rule. This design supports low-latency, window-level inferences that are suitable for online execution.

The binary model was trained on 2108 sliding windows (normal and fault), validated on 150 windows, and tested on an unseen subset of 300 windows to ensure that the test subset did not overlap with the training data. Table 6 reports the standard window-level metrics, showing a favorable tradeoff between sensitivity and precision.

In the evaluation protocol, decisions are not made directly based on a single window threshold. Instead, the detector applies a record-level rule that aggregates consecutive window outputs within each T_{rec} record. Using this aggregation, the record-level performance reached 100% for both validation and test subsets (Table 7). This indicates that the extracted features are sufficiently discriminative and that temporal aggregation suppresses isolated window fluctuations while retaining sensitivity to persistent fault signatures.

The results indicate that the online binary OS-ELM maintains high sensitivity under previously unseen operating conditions, while preserving excellent precision. This behavior is essential for the first stage module, whose primary role

TABLE 6. Window-level binary OS–ELM performance computed per sliding window ($T_W = 0.6$ s, stride $T_S = 0.1$ s).

Metric	Validation	Test
Accuracy [%]	92.16	90.69
Precision (Fault) [%]	93.3	100.0
Recall (Fault) [%]	97.6	90.7
F1-score (Fault) [%]	95.4	95.1

TABLE 7. Record-level performance of the binary OS-ELM fault detector obtained by aggregating window-level outputs within each T_{rec} record. A record is labeled Fault if at least K_{win} windows satisfy $P(\text{Fault}) \geq \tau$ (here $K_{\text{win}} = 4$, $\tau = 0.65$).

Metric	Validation	Test
Precision (Fault) [%]	100.00	100.00
Recall (Fault) [%]	100.00	100.00
F1-score (Fault) [%]	100.00	100.00

is to flag faulted segments reliably for subsequent fault-type classification.

Figure 18 illustrates the detector output for two representative T_{rec} records: (a) faulty and (b) healthy. In each case, the upper plot shows the measured phase current I_a over the diagnostic interval, while the lower plot reports the window-level posterior probability $P(\text{Fault})$ computed using sliding windows of T_w with a stride of T_s.

As previously discussed, to reduce the sensitivity to isolated high-probability windows, we applied the decision rule at the segment level rather than using a single per-window threshold. Specifically, a segment is declared faulty if at least K_{win} windows exceed a probability threshold τ, where (K_{win}, τ). The pair (K_{win}, τ) was selected for the validation set by maximizing the segment-level F1-score. In this study, we used $K_{win} = 4$ and a probability threshold of $\tau = 0.65$, tuned on the validation data across the six tested spindle speeds. If the operating range is extended beyond these speeds, the thresholds may need to be recalibrated (or speed-conditioned tuned). This rule promotes temporal persistence, which is consistent with the physical expectation that bearing-related faults produce sustained signatures across consecutive windows, rather than sporadic spikes.

2) ONLINE MULTI-CLASS CLASSIFICATION (OS-ELM)

Following fault detection, the flagged windows are forwarded to a higher-capacity multi-class OS-ELM for real-time classification of bearing fault types (*BPFI, BPFO, BSF,* and *BCF*). The OS-ELM model offers a lightweight architecture with extremely low computational overhead, making it well-suited for real-time deployment on embedded hardware. Its performance on the dedicated test subset showed an overall accuracy of **93.01%**, with a particularly strong recall for the *BCF, BSF,* and *BPFO* fault types, with an accuracy above **95%**. However, the recall for the *BPFI* faults decreased to **76.5%** recall.

Although all four defect classes were injected using distinct characteristic frequencies, some regimes yielded

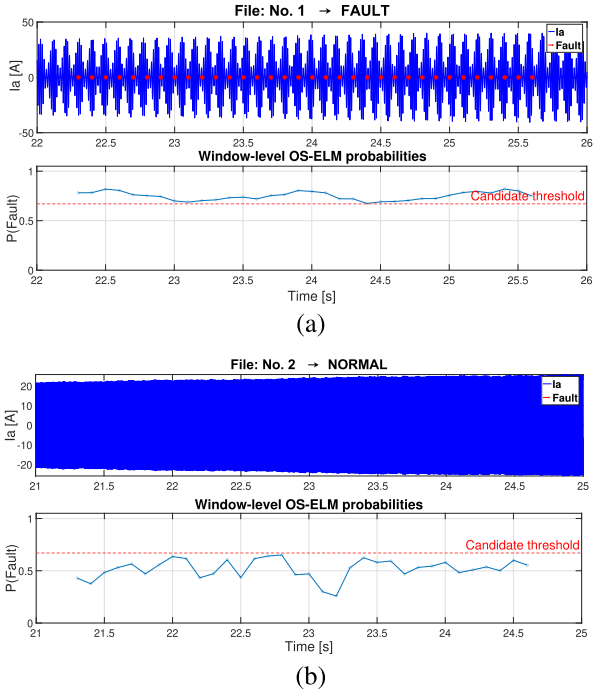

FIGURE 18. Binary OS–ELM detector output on representative T_{rec} test records: (a) faulty and (b) healthy cases; upper plots show $I_a(t)$ and lower plots show window-level posterior $P(\text{Fault})$ over sliding T_W windows with the probability threshold τ used in the record-level aggregation rule.

increased spectral similarity in the current and power signatures. In particular, under a higher severity, the disturbance torque produces stronger harmonics and richer modulation sidebands, which can increase the overlap between the spectral neighborhoods exploited by compact feature sets (e.g., shared modulation around f_r and harmonics in the envelope spectrum). This overlap primarily affects the inner/outer race family (*BPFI*/*BPFO*), and the dominant confusion may shift with operating point and severity distribution. In our primary test split, this is reflected by the reduced recall for *BPFI* (Fig. 19 (a)), whereas in the additional unseen-severity holdout, *BPFO* exhibits a larger performance drop while *BPFI* remains well separated (Table 8). Because the OS-ELM operates on compact multi-domain descriptors with limited capacity, it is more sensitive to such regime-dependent overlap, whereas the CNN-BiLSTM verifier can exploit finer spatiotemporal structures to recover separability at higher computational costs.

In Fig. 19(a), the multi-class OS-ELM confusion matrix demonstrates stable generalization across held-out severity scenarios within the validated operating envelope and highlights the clear separability of the four bearing fault signatures.

3) OFFLINE HIGH-FIDELITY VERIFICATION (CNN-BiLSTM)

Finally, the entire run was analyzed offline using the CNN-BiLSTM network. This deep model incorporates convolutional layers, bidirectional LSTMs, and a temporal self-attention mechanism to capture fine spatiotemporal

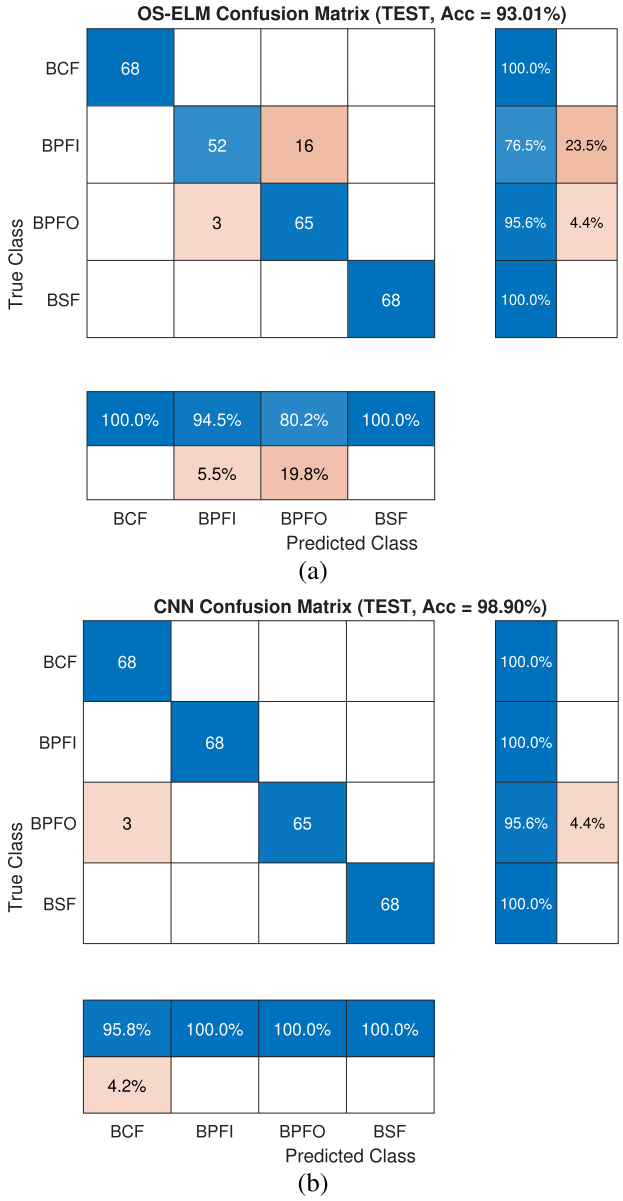

FIGURE 19. Confusion matrices on unseen test subset for (a) online multi-class OS-ELM and (b) offline CNN classifier.

structures that cannot be exploited by online models. This served as a high-fidelity reference that validated the OS-ELM outputs and provided the final fault label for record-keeping and maintenance recommendations.

The CNN–BiLSTM model achieved a much richer representation of the temporal–frequency characteristics of the current and power signals. The confusion matrix (Fig. 19 (b)) shows that the network effectively captures the complex envelope and harmonic patterns associated with most fault conditions, reaching **95.6%** for *BPFO* and **100%** for all remaining fault types (*BCF, BPFI,* and *BSF*).

Moreover, despite its strong representation capacity, CNN–BiLSTM has a considerably higher computational burden and longer inference time and memory footprint, making it unsuitable for on-machine real-time fault monitoring. These limitations are particularly relevant when the goal is online decision support in machining operations, where latency and hardware constraints are strict design requirements.

TABLE 8. Overall accuracy of multi-class classifiers on train/val/test subsets and the additional unseen-severity holdout set.

Model	Train (%)	Val (%)	Test (%)	unseen-severity holdout (%)
CNN	99.5	99.2	98.9	99.63
OS–ELM	96.93	98.75	93.01	97.43

This complementary arrangement preserves the interpretability and low computational cost of OS-ELM for online monitoring while leveraging the higher-capacity CNN-BiLSTM verifier for offline confirmation. Table 8 summarizes the comparative results. In addition to the train/val/test subsets, we report an external unseen-severity holdout check at a known operating speed ($f_r = 300$ Hz). This holdout contains severity levels $\alpha \in \{0.25, 0.31\}T_{\text{nom}}$ that are not used during the OS-ELM and CNN training or validation. The resulting performance (Table 8) provides additional evidence that the reported generalization is not solely attributable to the deterministic simulation grid or overlap-induced sample correlation but reflects robustness to severity levels that were not seen during model development. It is important to note that holdout was applied for evaluation, and no retraining was performed using this new test subset.

By integrating lightweight online learning with high-fidelity offline verification, HyDT becomes an adaptive, real-time diagnostic architecture intended for edge deployment, the performance of which in this study is evaluated within the validated DT operating envelope.

This integration strengthens HyDT from a passive to an active system by enabling online diagnostic inference and continuous data-driven analyses. By coupling physics-based insights with real-time intelligence, the framework achieves both interpretability and adaptability, representing a tangible step toward next-generation PdM systems for smart manufacturing.

VI. RESTRICTIONS AND FUTURE TRENDS

The current HyDT framework is specifically validated for electrospindle bearing faults, which limits its immediate generalizability to other critical failure modes, such as rotor imbalances, misalignment, cooling failure, motor overload, and ball screw wear.

Moreover, the proposed fault injection is implemented as a speed-dependent periodic disturbance torque at the characteristic defect frequency, which captures the dominant harmonic and modulation cues exploited by the current- and power-based diagnosis. This abstraction does not explicitly model impulsive contact forces, intermittent impacts, slips, or the structural transfer path that yields resonance amplification and broadband energy spread in real spindles. Consequently, the DT to real transferability may degrade for fault modes whose discriminative cues are transient rather than harmonic.

Furthermore, the present formulation may represent transient impulsiveness and high-frequency resonance phenomena, which are often prominent in vibrational measurements. Therefore, the present study focuses on the discriminative harmonic and modulation signatures produced by (4) in the electrical channels.

Future work will prioritize validation using real faulty measurements of the target spindle to quantify the DT-to-real domain gap and calibrate the decision thresholds accordingly. Transfer learning and domain adaptation can then be investigated to improve cross-condition and cross-machine generalization, extending the current fault classification pipeline toward prognostics and health management, including remaining useful life (RUL) estimation. Further work will also study deployment on industrial hardware and enrich the twin with multi-physics effects (e.g., thermal and structural dynamics) to increase fidelity and broaden the range of detectable fault modes.

VII. CONCLUSION

This study introduced a comprehensive HyDT framework to address the persistent challenge of data scarcity in fault diagnosis for industrial spindle systems, thereby supporting PdM applications. By integrating a real-time, physics-based DT implemented on an OPAL-RT platform with synthetic dataset generation, multi-domain feature enrichment, and a data-driven approach using supervised machine learning, this study shows that high-fidelity artificial data can effectively support the development and evaluation of diagnostic models when real fault measurements are limited, while preserving physical realism and diagnostic relevance.

A dual classifier architecture was proposed to enable practical deployment. A lightweight OS-ELM model performs real-time condition monitoring at the edge, whereas an offline CNN-BiLSTM model provides more detailed verification to support maintenance analysis and record-keeping. Despite being trained exclusively on DT-generated data, both models demonstrated strong diagnostic performance under previously unseen fault-severity conditions, underscoring the effectiveness of the proposed data generation strategy within the validated operating envelope.

By combining real-time modeling, automated fault scenario simulation, structured data acquisition, and supervised learning, the proposed framework advances DT beyond passive simulation tools, positioning it as an active enabler of a learning-driven diagnostic workflow. This integrated approach offers a transparent and physically consistent pathway for developing and pre-validating an AI-assisted fault diagnosis in machine tools, particularly when acquiring real fault data is costly, risky, or impractical.

This study provides a solid foundation for the next generation of DT-assisted AI systems for industrial diagnostics. By demonstrating that synthetic datasets can support both online and offline fault diagnoses within a well-defined DT scope, this study outlines a realistic pathway for integrating data-driven intelligence into industrial monitoring workflows.

REFERENCES

[1] D. Neupane, M. R. Bouadjenek, R. Dazeley, and S. Aryal, "Data-driven machinery fault diagnosis: A comprehensive review," *Neurocomputing*, vol. 627, Apr. 2025, Art. no. 129588.

[2] P. Zhang, R. Chen, L. Yang, Y. Zou, and L. Gao, "Recent progress in digital twin-driven fault diagnosis of rotating machinery: A comprehensive review," *Neurocomputing*, vol. 634, Jun. 2025, Art. no. 129914.

[3] M. Zeynivand and G. Gruosso, "Data-driven monitoring and benchmarking of a permanent magnet synchronous motor using digital twins," in *Proc. IECON - 50th Annu. Conf. IEEE Ind. Electron. Soc.*, Nov. 2024, pp. 1–6.

[4] L. Boca de Giuli, A. La Bella, E. Masero, R. Scattolini, P. Esmaili, L. Cristaldi, L. Tanca, G. Gruosso, L. Martiri, M. Zeynivand, G. La Vigna, and G. Minei, "The MICS project approach for fault detection in CNC machine bearings," in *Proc. IEEE Int. Conf. Metrology EXtended Reality, Artif. Intell. Neural Eng. (MetroXRAINE)*, Oct. 2024, pp. 674–679.

[5] J. Li, W. Luo, M. Bai, and M. Song, "Fault diagnosis of high-speed rolling bearing in the whole life cycle based on improved grey wolf optimizer-least squares support vector machines," *Digit. Signal Process.*, vol. 145, Feb. 2024, Art. no. 104345.

[6] S. Li, Q. Jiang, Y. Xu, K. Feng, Y. Wang, B. Sun, X. Yan, X. Sheng, K. Zhang, and Q. Ni, "Digital twin-driven focal modulation-based convolutional network for intelligent fault diagnosis," *Rel. Eng. Syst. Saf.*, vol. 240, Dec. 2023, Art. no. 109590.

[7] K. C. Chan, M. Rabaev, and H. Pratama, "Generation of synthetic manufacturing datasets for machine learning using discrete-event simulation," *Prod. Manuf. Res.*, vol. 10, no. 1, pp. 337–353, Dec. 2022.

[8] V. Kannan, T. Zhang, and H. Li, "A review of the intelligent condition monitoring of rolling element bearings," *Machines*, vol. 12, no. 7, p. 484, Jul. 2024.

[9] D. Sahu, R. K. Dewangan, and S. P. S. Matharu, "Long short-term memory based fault diagnosis of rolling element bearings using vibration signals," *J. Vib. Control*, pp. 1–12, Mar. 2025.

[10] B. Luo, H. Wang, H. Liu, B. Li, and F. Peng, "Early fault detection of machine tools based on deep learning and dynamic identification," *IEEE Trans. Ind. Electron.*, vol. 66, no. 1, pp. 509–518, Jan. 2019.

[11] M. Bokhtiar Al Zami, S. Shaon, V. Khanh Quy, and D. C. Nguyen, "Digital twin in industries: A comprehensive survey," *IEEE Access*, vol. 13, pp. 47291–47336, 2025.

[12] W. Yu, G. Liu, L. Zhu, and W. Yu, "Digital twin: A literature review of concepts, technologies, and applications," *IEEE Access*, vol. 13, pp. 183075–183095, 2025.

[13] Y. Lu, C. Liu, K. I.-K. Wang, H. Huang, and X. Xu, "Digital twin-driven smart manufacturing: Connotation, reference model, applications and research issues," *Robot. Comput.-Integr. Manuf.*, vol. 61, Feb. 2020, Art. no. 101837.

[14] M. Zeynivand, P. Esmaili, L. Cristaldi, and G. Gruosso, "Modern digital twin for validation and generation of datasets for machine tool spindle modeling," in *Proc. Int. Symp. Power Electron., Electr. Drives, Autom. Motion (SPEEDAM)*, Jun. 2024, pp. 430–435.

[15] S. Aheleroff, X. Xu, R. Y. Zhong, and Y. Lu, "Digital twin as a service (DTaaS) in Industry 4.0: An architecture reference model," *Adv. Eng. Informat.*, vol. 47, Jan. 2021, Art. no. 101225.

[16] E. Ferko, A. Bucaioni, and M. Behnam, "Architecting digital twins," *IEEE Access*, vol. 10, pp. 50335–50350, 2022.

[17] C. Chen, H. Fu, Y. Zheng, F. Tao, and Y. Liu, "The advance of digital twin for predictive maintenance: The role and function of machine learning," *J. Manuf. Syst.*, vol. 71, pp. 581–594, Dec. 2023.

[18] H. Liu, M. Xia, D. Williams, J. Sun, and H. Yan, "Digital twin-driven machine condition monitoring: A literature review," *J. Sensors*, vol. 2022, pp. 1–13, Jul. 2022.

[19] M. S. Azari, L. Ricci, S. Santini, and F. Flammini, "An intelligent diagnostic framework based on digital twins and partial transfer learning: Methodology and industrial application," *IEEE Trans. Ind. Cyber-Phys. Syst.*, vol. 3, pp. 1–13, 2025.

[20] B. Sicard, Q. Butler, Y. Wu, S. M. Abdolahi, Y. Ziada, and S. A. Gadsden, "Generating synthetic data for data-driven solutions via a digital twin for condition monitoring in machine tools," *Proc. SPIE*, vol. 13035, pp. 9–22, Apr. 2024.

[21] Y. Xu, Y. Sun, X. Liu, and Y. Zheng, "A digital-twin-assisted fault diagnosis using deep transfer learning," *IEEE Access*, vol. 7, pp. 19990–19999, 2019.

[22] Y. Zhang, J. C. Ji, Z. Ren, Q. Ni, F. Gu, K. Feng, K. Yu, J. Ge, Z. Lei, and Z. Liu, "Digital twin-driven partial domain adaptation network for intelligent fault diagnosis of rolling bearing," *Rel. Eng. Syst. Saf.*, vol. 234, Jun. 2023, Art. no. 109186.

[23] K. Feng, Y. Xu, Y. Wang, S. Li, Q. Jiang, B. Sun, J. Zheng, and Q. Ni, "Digital twin enabled domain adversarial graph networks for bearing fault diagnosis," *IEEE Trans. Ind. Cyber-Phys. Syst.*, vol. 1, pp. 113–122, 2023.

[24] X. Yu, Y. Yang, M. Du, Q. He, and Z. Peng, "Dynamic model-embedded intelligent machine fault diagnosis without fault data," *IEEE Trans. Ind. Informat.*, vol. 19, no. 12, pp. 11466–11476, Dec. 2023.

[25] Z. Guo, J. Li, T. Wang, J. Xie, J. Yang, and B. Niu, "Dynamically constrained digital twin-based mechanical diagnosis framework under undetermined states without fault data," *IEEE Trans. Instrum. Meas.*, vol. 74, pp. 1–15, 2025.

[26] G. Pritschow and S. Röck, "'Hardware in the loop' simulation of machine tools," *CIRP Ann.*, vol. 53, no. 1, pp. 295–298, 2004.

[27] Z. Ju, Y. Chen, Y. Qiang, X. Chen, C. Ju, and J. Yang, "A systematic review of data augmentation methods for intelligent fault diagnosis of rotating machinery under limited data conditions," *Meas. Sci. Technol.*, vol. 35, no. 12, Dec. 2024, Art. no. 122004.

[28] X. Hu, J. Li, Y. Huang, X. Zhang, H. Wang, H. Wang, and Y. He, "PCASTNet: A physics-constrained adaptive style transfer network for sample generation in cross-machine small-sample fault diagnosis," *IEEE Trans. Instrum. Meas.*, vol. 74, pp. 1–17, 2025.

[29] Y. Huang, X. Hu, H. Wang, Y. He, and J. Cao, "OAIFAN: A noise-robust discriminative feature unification framework for cross-speed fault transfer diagnosis," *IEEE Trans. Instrum. Meas.*, vol. 74, pp. 1–18, 2025.

[30] M. Zeynivand, P. Esmaili, L. Cristaldi, and G. Gruosso, "A novel approach to digital twin-based energy efficiency monitoring and failure analysis in industrial applications," *J. Manuf. Syst.*, vol. 83, pp. 612–625, Dec. 2025.

[31] N. Berti et al., "Sustainability and resilience in the MICS SPOKE8 project: The role of the digital twin," in *Proc. IEEE Int. Conf. Metrology EXtended Reality, Artif. Intell. Neural Eng. (MetroXRAINE)*, Oct. 2024, pp. 669–673.

[32] B. Du, W. Huang, T. Li, R. Hu, Y. Cheng, and S. Cui, "A digital twin system for two-stage PMSM rotor and bearing faults identification based on deep learning and improved-RGB acoustic image," *IEEE Trans. Power Electron.*, vol. 40, no. 1, pp. 2184–2195, Jan. 2025.

[33] H.-X. Hu, Y. Feng, Q. Hu, and Y. Zhang, "A masked one-dimensional convolutional autoencoder for bearing fault diagnosis based on digital twin enabled industrial Internet of Things," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 10, pp. 3242–3253, Oct. 2023.

[34] R. Huang, J. Li, Y. Liao, J. Chen, Z. Wang, and W. Li, "Deep adversarial capsule network for compound fault diagnosis of machinery toward multidomain generalization task," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–11, 2021.

[35] W. Li, H. Lan, J. Chen, K. Feng, and R. Huang, "WavCapsNet: An interpretable intelligent compound fault diagnosis method by backward tracking," *IEEE Trans. Instrum. Meas.*, vol. 72, pp. 1–11, 2023.

[36] C. Wang, Z. Fu, Z. Zhang, W. Wang, H. Chen, and D. Xu, "Fault diagnosis of power transformer in one-key sequential control system of intelligent substation based on a transformer neural network model," *Processes*, vol. 12, no. 4, p. 824, Apr. 2024.

[37] R. Huang, Y. Chen, and C. Liu, "Revealing state-feature dependencies in dynamic degradation: An exofeature-aware transformer for battery state-of-health prediction," *IEEE Trans. Consum. Electron.*, early access, Nov. 12, 2025, doi: 10.1109/TCE.2025.3631954.

[38] (2025). *Opc Foundation Official Website*. Accessed: Sep. 12, 2025. [Online]. Available: https://opcfoundation.org/

[39] H. Hao, Y. Zhao, Y. Chen, Y. Zhang, and D. Wang, "Incremental single-class fault detection and diagnosis method for rolling bearings based on OS-ELM," *Electronics*, vol. 12, no. 19, p. 4099, Sep. 2023.

[40] R. Khanam, M. Hussain, R. Hill, and P. Allen, "A comprehensive review of convolutional neural networks for defect detection in industrial applications," *IEEE Access*, vol. 12, pp. 94250–94295, 2024.

[41] B. Sicard, Q. Butler, Y. Ziada, and S. A. Gadsden, "Experimental setups for linear feed drive predictive maintenance: A review," in *Proc. IEEE Int. Conf. Prognostics Health Manage. (ICPHM)*, Jun. 2023, pp. 357–367.

[42] P. Chandrasekar and K. Srinivasan, "Inferential based measurement of backlash in servo system," *Mater. Today, Proc.*, vol. 46, pp. 9766–9770, Jan. 2021.

[43] J.-L. Chang, J.-A. Chao, Y.-C. Huang, and J. S. Chen, "Prognostic experiment for ball screw preload loss of machine tool through the Hilbert–Huang transform and multiscale entropy method," in *Proc. IEEE Int. Conf. Inf. Autom.*, Jun. 2010, pp. 376–380.

[44] K. Pichler, J. Klinglmayr, and M. Pichler-Scheder, "Detecting wear in a ball screw using a data-driven approach," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, Oct. 2018, pp. 3123–3128.

[45] L. Zhou, F. Li, Y. Wang, L. Wang, and G. Wang, "A new empirical standby power and auxiliary power model of CNC machine tools," *Int. J. Adv. Manuf. Technol.*, vol. 120, nos. 5–6, pp. 3995–4010, May 2022.

[46] Y. Wu, B. Sicard, and S. A. Gadsden, "A review of physics-informed machine learning methods with applications to condition monitoring and anomaly detection," 2024, *arXiv:2401.11860*.

[47] W. Jung, S.-H. Kim, S.-H. Yun, J. Bae, and Y.-H. Park, "Vibration, acoustic, temperature, and motor current dataset of rotating machine under varying operating conditions for fault diagnosis," *Data Brief*, vol. 48, Jun. 2023, Art. no. 109049.

[48] F. Cong, J. Chen, G. Dong, and M. Pecht, "Vibration model of rolling element bearings in a rotor-bearing system for fault diagnosis," *J. Sound Vib.*, vol. 332, no. 8, pp. 2081–2097, Apr. 2013.

[49] S. Salnikov, E. Solodkiy, and D. Vishnyakov, "Simulation of three-phase induction motor different bearing faults in MATLAB simulink environment," *J. Phys., Conf. Ser.*, vol. 1886, no. 1, Apr. 2021, Art. no. 012009.

[50] H. I. Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P.-A. Müller, "Deep learning for time series classification: A review," *Data Mining Knowl. Discovery*, vol. 33, no. 4, pp. 917–963, 2019.

[51] L. Cohen, *Time-Frequency Analysis*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1995.

[52] S. Sharma, *Digital Signal Processing*. New Delhi, India: Khanna Publishing House, 2025.

MOHSEN ZEYNIVAND (Member, IEEE) was born in Ilam, Iran, in 1993. He is currently pursuing the Ph.D. degree with the Department of Energy, Politecnico di Milano. He is a Ph.D. Researcher with Simlab40. His research explores the integration of machine learning and digital twins to enhance energy efficiency and predictive maintenance in industrial applications, particularly machine tools. At Simlab40, he develops high-fidelity, real-time digital twins using Typhoon HIL and OPAL-RT. He leverages the synthetic datasets from these digital twin models to train machine learning algorithms for predicting power consumption and anticipating system failures. Additionally, he builds communication bridges using protocols, such as OPC-UA to connect digital models with real-world machinery. His peer-reviewed publications demonstrate a commitment to advancing Industry 4.0 goals through sustainable innovation and improved system reliability.

AZADEH KERMANSARAVI (Senior Member, IEEE) received the Ph.D. degree in software engineering from the Department of Computer and Software Engineering, Polytechnique Montréal, Canada, in 2019. From 2019 to 2024, she was with Industry as a Data Analyst and an AI Specialist, contributing to advanced analytics and AI-driven solutions across the energy, CO_2-emissions, and manufacturing sectors. In 2024, she joined Delft University of Technology as a Postdoctoral Fellow and currently a Lecturer with The Hague University of Applied Sciences. Her research interests include AI optimization, reinforcement-learning-based control, and the integration of machine learning techniques into power electronics and power system applications. She is an Active Member of the IEEE Industrial Electronics Society's Students and Young Professionals Committee.

HANI VAHEDI (Senior Member, IEEE) received the Ph.D. (Hons.) degree from École de Technologie Supérieure (ÉTS), Université du Québec, Montreal, in 2016. He has authored/coauthored over 100 papers in IEEE conferences and journals, a Springer Nature book, and a book chapter in Elsevier. He has been an active member of the IEEE Industrial Electronics Society (IES) since 2012, serving on the local organizing committee for IECON12 in Montreal, acting as Students and Young Professionals Chair for numerous IES-sponsored conferences from 2016 to 2024, and serving as Technical Program Chair (GreenTech Cluster) for IECON25 in Madrid, as well as Special Sessions Chair for IECON26 in Qatar. Currently, he is the coordinator of the IES Chapters & Joint Chapters Program. His research interests include multilevel power converter topologies, modulation and control, and applications in renewable energy conversion, EV charging, green hydrogen production, and fuel-cell systems. He was awarded ÉTS's Best Ph.D. Thesis Award (2016-2017). He is an Associate Editor for the IEEE Transactions on Industrial Electronics, the IEEE Open Journal of the IES, and the IEEE Open Journal of Power Electronics. He is the inventor of the 5-level Packed U-Cell (PUC5) converter, holds several U.S. and international patents, and transferred this technology to industry, assisting in the development of a bidirectional EV charger based on his invention. After seven years in the industry, he joined Delft University of Technology as a faculty member to advance electrification for the clean energy transition. Since January 2026, he has been with AASU, Kuwait, and holds a guest Professor position at TU Delft.

GIAMBATTISTA GRUOSSO (Senior Member, IEEE) was born in 1973. He received the B.S. and M.S. degrees in electrical engineering and the Ph.D. degree in electrical engineering from Politecnico di Torino, Turin, Italy, in 1999 and 2003, respectively. From 2002 to 2011, he was an Assistant Professor with the Department of Electronics and Informatics, Politecnico di Milano, Milan, Italy, where he has been an Associate Professor, since 2011. He is the author of more than 200 papers in journals and conferences on these topics. His research interests include electrical engineering, electronic engineering, industrial engineering, electric vehicle transportation electrification, electrical power systems optimization, simulation of electrical systems, digital twins for smart mobility, factory and city, and how they can be obtained from data.