TRANSPORT
Systems and Technologies
https://tstjournal.org.ua/en

VOL. 29
№1
2026
ISSN 2617-9040 e-ISSN 2617-9059

UDC 625.151.2:004.942:004.89

DOI: 10.32703/2617-9040-2026-47-6

Predictive maintenance of railway point machines using digital twin and deep learning

Ihor Katkalo*
Postgraduate Student
National Transport University
01010, 1 M. Omelyanovych-Pavlenko Str., Kyiv, Ukraine
https://orcid.org/0009-0008-2672-1005

Halyna Holub
PhD in Technical Sciences, Associate Professor
National Transport University
01010, 1 M. Omelyanovych-Pavlenko Str., Kyiv, Ukraine
https://orcid.org/0000-0002-4028-1025

Abstract. Point machine drives are critical components of railway signalling infrastructure, with their failures accounting for 15-20% of all signal system malfunctions across European networks, causing significant operational and economic losses. This study aimed to develop a predictive maintenance approach for 1,520 mm-gauge point machine drives using a digital twin, enabling early fault detection from power electrical signatures recorded during each switch cycle. The methodology combined a four-layer digital twin architecture (physical layer, data layer, analytics layer, and decision-making layer) with a comparison of three deep learning architectures: one-dimensional convolutional neural network (1D-CNN), long short-term memory network (LSTM), and a hybrid CNN-LSTM model. These were benchmarked against logistic regression, support vector machines (SVM), and Random Forest on a synthetic, physics-based dataset comprising 5,000 power curves for SP-6M and VSP-220 point machines. Models were evaluated on five-class state classification and 50-cycle degradation forecasting. The hybrid CNN-LSTM achieved the highest classification performance (macro F1 = 0.938), while LSTM yielded the best degradation forecast (R^2 = 0.937); both substantially outperformed the strongest baseline model (Random Forest: F1 = 0.871, R^2 = 0.845). Cross-generator validation and artefact ablation experiments confirmed that deep learning advantages arise from capturing transferable signal shape features rather than memorising simulation artefacts. A recommended deployment configuration includes 1D-CNN at the edge for real-time classification, LSTM at the server level for degradation forecasting, and Random Forest as an intermediate solution for operators with limited labelled data. The practical significance of these results lies in providing evidence-based guidelines for implementing the digital twin analytics layer on 1,520 mm railway networks, supporting a transition from scheduled to condition-based maintenance and reducing unplanned downtime

Keywords: control; diagnostics; power signature analysis; monitoring; architecture; modelling

Introduction

Railway signalling infrastructure depends on point machines to direct rolling stock through track switches, and the operational consequences of their failures are disproportionate to the size of the equipment. Across European networks, signalling-related disruptions account for a substantial fraction of unplanned service interruptions, and within that fraction point machines remain a leading contributor. The cost is measured in delay-minutes, missed connections, freight throughput lost during single-line working, and on the worst occasions in compromised safety margins. Three maintenance regimes dominate the field. Reactive maintenance addresses failures as they occur, accepting the disruption cost in exchange for procedural simplicity. Preventive maintenance services equipment on a fixed schedule, replacing components by calendar time or switching-cycle count regardless of actual condition. Predictive maintenance, by contrast, infers the condition of each device from sensor data and intervenes only when degradation evidence warrants it. The recent systematic review by I. Hector & R. Panjanathan (2024) mapped 124 publications on predictive maintenance under the Industry 4.0 umbrella and showed that, across sectors, the technology has matured from research demonstrations into deployment-ready engineering, with machine learning the most common analytics layer. The digital twin paradigm offers an architectural framework

Received: 25.01.2026. Revised: 18.05.2026. Accepted: 25.06.2026. Published: 04.07.2026.

Suggested Citation: Katkalo, I., & Holub, H. (2026). Predictive maintenance of railway point machines using digital twin and deep learning. *Transport Systems and Technologies*, 29(1), 71-84. doi: 10.32703/2617-9040-2026-47-6.

*Corresponding author (*ihor.katkalo@icloud.com*)

for integrating predictive maintenance with data acquisition, condition monitoring, fault diagnosis and decision support. F. Tao *et al.* (2022) formalised the components of digital twin systems – physical asset, data acquisition, analytics, and decision support – and argued that the analytics layer is where the largest engineering choices accumulate. Operating on this framing, R. van Dinter *et al.* (2022) reviewed 42 papers combining digital twins with predictive maintenance and identified the analytics layer as both the most variable and the most consequential component, since model choice determines whether the digital twin produces actionable insight or merely mirrors the physical world. In the railway sector specifically, E.A. Thompson *et al.* (2025) surveyed 91 papers on digital twin applications and reported predictive maintenance as the dominant use case, with deep learning the most common analytics-layer choice. S. Kaewunruen *et al.* (2023) and J. Sresakoolchai & S. Kaewunruen (2023) demonstrated the approach for railway bridge maintenance under climate change conditions, showing that digital twins can integrate heterogeneous sensor streams and forecast component-level degradation at fleet scale. The Ukrainian context for this work has its own constraints. The 1,520 mm gauge networks operated by Joint Stock Company (JSC) "Ukrzaliznytsia" use electromechanical point machines, predominantly the SP-6M and VG-32 families, with supply standards and operational practices that differ from Western European 1,435 mm networks in voltage configuration, switching cycles, and ambient operating ranges. K.L. Kostenko *et al.* (2023) reviewed the prospects for robotisation of diagnostic procedures in Ukrainian railway automation systems and concluded that the technical foundations for predictive maintenance exist, but field deployments are constrained by data accumulation and the absence of validated reference models tailored to 1520 mm gauge equipment. H. Holub *et al.* (2025) addressed the deployment layer of any digital twin intended for network-scale operation, focusing on the integration of machine-learning workloads with cloud resource optimisation for transport big data.

A clear gap emerges from this literature. Digital twin frameworks and predictive maintenance models are individually mature, and both have been demonstrated on railway components. However, no published work has carried out a systematic, multi-task comparison of deep learning architectures specifically for railway point machines on 1,520 mm gauge infrastructure, embedded within a digital twin architecture, with the experimental controls needed to separate genuine learning from simulation-specific artefacts. Closing this gap is the contribution of the present study. The purpose of this study was to design and assess a digital twin framework for forecasting maintenance needs in railway point machines by evaluating the performance of deep learning models on electrical power signal patterns captured during switching operations. The objectives were: (1) to develop a conceptual digital twin architecture for a railway turnout with an integrated predictive diagnostics module that formalises point machine condition assessment as two complementary tasks: multi-class fault classification and degradation trend prediction; (2) to design, implement, and comparatively evaluate three deep learning architectures (1D-CNN, LSTM, hybrid CNN-LSTM) against three classical baselines across both tasks, using a multi-criteria framework covering accuracy, computational efficiency, and robustness to limited training data; (3) to formulate evidence-based recommendations for digital twin deployment on 1520 mm gauge networks. The scientific contribution of this study lies in a systematic evaluation of three deep learning architectures – a one-dimensional convolutional neural network (1D-CNN), a long short-term memory network (LSTM), and a hybrid convolutional neural network-long short-term memory model (CNN-LSTM) – in comparison with three classical baseline models for multi-class fault classification and degradation trend prediction. The robustness of the comparison was strengthened through cross-generator validation and artefact-ablation experiments. The practical contribution is a set of deployment recommendations calibrated to the operational and data-availability constraints typical of 1,520 mm gauge railway networks.

Materials and Methods

The proposed digital twin is organised as a four-layer structure (Fig. 1). The Physical Layer is the actual turnout installation with its sensors. The primary data source is the point machine power monitoring system, which records electrical current, voltage and computed power consumption during every switching cycle. Supplementary sensors include displacement transducers, accelerometers on the crossing nose and stock rails, and ambient temperature sensors. This configuration is roughly what is deployed on monitored turnouts in modern signalling systems, including those used on Ukrainian railways (Siroklyn & Kladko, 2018). In the present study, all signals analysed in subsequent sections come from a physics-informed simulation of this sensor configuration rather than from a deployed field installation. The Data Layer handles acquisition, transmission, preprocessing and storage. Raw signals are sampled at 1,000 Hz during each switching event, producing 3-7 second records depending on point machine type and switching direction. This layer runs initial quality checks, timestamps each record, and attaches metadata (turnout ID, switching direction, ambient temperature, cumulative switching count since last maintenance) before storing into a time-series database. The Analytics Layer hosts the predictive models that are the focus of this study. Two tasks run here: fault classification (current condition category) and degradation trend prediction (forecast over future switching cycles). A model management component handles periodic retraining and versioning. The output of this layer feeds into a decision-making structure that follows established system models for transport infrastructure management, where multi-level decomposition narrows control objects at each level. The Decision Layer translates the analytical

output into maintenance recommendations: alerts on threshold approach, remaining useful life estimates, and schedules that respect operational constraints like traffic windows and crew availability. The first research task focused on fault classification. Given a discrete time series P = {p_1,...,p_n} representing a switching power curve, assign it to one of K = 5 condition categories defined in Table 1, were chosen to cover the dominant failure modes reported in operational maintenance records for electromechanical point machines, with each class corresponding to a physically distinct mechanism that leaves a characteristic imprint on the power signature.

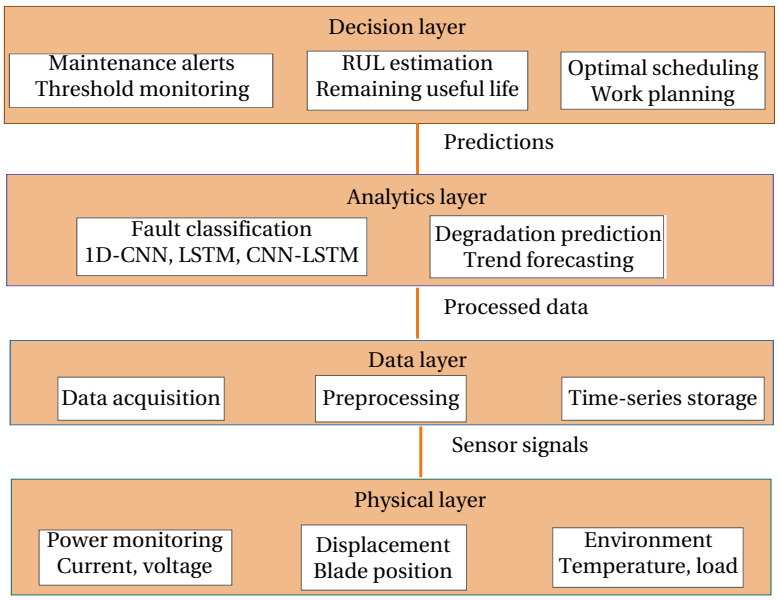

Figure 1. Conceptual architecture of the digital twin for a railway turnout

Source: developed by the authors

Table 1. Point machine condition categories

Class	Label	Physical description
C_1	Normal	Switching characteristics within normal limits
C_2	Increased friction	Elevated resistance from contamination, insufficient lubrication, or wear
C_3	Obstruction	Partial blockage of blade movement by foreign objects or geometry deformation
C_4	Electrical degradation	Brush, commutator or contact deterioration altering current characteristics
C_5	Misadjustment	Incorrect setting of switching stroke, detection rod, or locking mechanism

Source: developed by the authors

C_1 is the fault-free reference; C_2 (mechanical friction) and C_3 (mechanical obstruction) capture degradation in the rod-and-rail interface; C_4 (electrical degradation) covers motor and contactor wear; C_5 (misadjustment) covers geometric drift in the lock-and-detection mechanism. Five is a practical compromise: coarser taxonomies lose the distinctions that drive different maintenance interventions, while finer ones produce classes too rare to learn reliably from the data volumes typical of point machine fleets. The categories were treated as mutually exclusive in the simulation: each cycle was generated under exactly one active failure mode. This is a deliberate simplification of operational reality, where two or more degradation processes can coexist – friction often accompanies misadjustment, and electrical degradation can mask mechanical wear – so the single-label formulation is an approximation appropriate to early-stage diagnosis. Multi-label and hierarchical taxonomies are noted in the Prospects for further research as a natural extension once field data become available. The classification task is the mapping function:

$$f_{class}: P \rightarrow \{C_1, C_2, C_3, C_4, C_5\}. \quad (1)$$

Minimising the cross-entropy loss:

$$L_{CE} = -\left(\frac{1}{M}\right) \sum_{i=1}^{M} \sum_{k=1}^{K} y_{ik} * log(\hat{y}_{ik}), \quad (2)$$

where M is the number of training samples, y_{ik} the binary class indicator, and $\hat{y}_{ik}$ the predicted class probability.

The second task involved degradation trend prediction. For each cycle j, a feature vector x_j is computed from the corresponding power curve. Given a sliding window $X = \{x\_(j - v + 1),...,x\}$, predict the value of a degradation indicator at horizon H:

$$f_{degr}: X = \{x_{j-v+1}, \ldots, x_j\} \rightarrow \hat{y}_{j+h}. \quad (3)$$

The degradation indicator is the normalised peak power during initial blade movement, which is monotonically correlated with mechanical wear. The regression loss is mean squared error:

$$L_{MSE} = (\frac{1}{M}) \sum_{i=1}^{M} (y_i - \hat{y}_i)^2. \quad (4)$$

The dataset has 5,000 power curves (1,000 Hz sampling, 3-7 s duration) covering five condition classes. Class distribution is intentionally imbalanced to reflect operational reality: C_1: 2,000, C_2: 900, C_3: 500, C_4: 800, C_5: 800. Two point machine types are represented (SP-6M and VG-32) and both switching directions. A separate subset of 200 degradation sequences (500-1,000 cycles each) is used for second task, with parameters calibrated to documented by M. Hamadache *et al.* (2019) wear rates of electromechanical point machines. Generation runs in two stages: deterministic baseline curves from an electromechanical model, then stochastic injection of operational artefacts. Electromechanical model. Curves are computed as $P(t) = U(t) * I(t)$ from coupled electrical and mechanical equations. The electrical subsystem is modelled as a DC series-wound motor (the dominant type in SP-6M and VG-32):

$$U(t) = I(t) * (R_a + Rf) + L_a + \frac{di}{dt} + K_e * \omega(t), \quad (5)$$

where R_a = 2.8 Ω at 20°C (temperature coefficient +0.00393/°C, copper resistivity), Rf = 1.4 Ω, L_a = 15 mH, K_e = 0.42 V·s/rad, supply voltage 160 V DC (Siroklyn & Kladko, 2018).

The mechanical subsystem couples motor torque to the switching mechanism through a gear-worm reducer (n = 35, η = 0.45):

$$J * \frac{d\omega}{dt} = K_t * I(t) - T_{load}(t)/n - T_{friction}(\omega), \quad (6)$$

where J = 0.008 kg·m², K_t = 0.42 N·m/A, and Stribeck friction:

$$T_{friction}(\omega) = T_c * sign(\omega) + (T_s + T_c) * \\ * exp(\left|-\frac{\omega}{\omega_s}\right|^2) * sign(\omega) * \beta * \omega, \quad (7)$$

where T_c = 0.15 N·m, T_s = 0.28 N·m, ω_s = 0.5 rad/s, β = 0.005 N·m·s/rad. These were calibrated against the SP-6M technical specifications.

Equations (5)-(7) are integrated with fourth-order Runge-Kutta at 0.1 ms (10 kHz internal rate) and downsampled to 1,000 Hz. Validation against the published reference signatures in M. Hamadache *et al.* (2019) resulted in a Mean Absolute Percentage Error (MAPE) of 3.2% for peak power, 4.1% for steady-state power, and 2.8% for switching duration, confirming the adequacy of the electromechanical calibration. Class-distinguishing load profiles. The load torque $T_{load}(t)$ is what differs between classes (Table 2). Peak power for C_1 falls in 480-560 W, for C_2 in 620-740 W, for C_3 in 750-900 W, consistent with the references.

Table 2. Load profile parameters per class

Class	Modification of nominal model
C_1 Normal	Phase I unlock 18-22 N·m; Phase II peak 25-32 N·m; Phase III steady 12-16 N·m; Phase IV lock 15-20 N·m
C_2 Increased friction	Load profile × U[1.3, 1.6] across all phases; β × [1.5, 2.5]
C_3 Obstruction	Localised torque spike 40-65 N·m, duration 0.1-0.4 s, random position in Phase II/III; stall threshold 70 N·m
C_4 Electrical degradation	R_a + 15-40%; stochastic contact resistance Rc_0 = 0.3-0.8 Ω + $Rc_1 \cdot n(t)$, Rc_1 = 0.1-0.3 Ω, $n(t)$ band-limited 50-200 Hz
C_5 Misadjustment	Stroke ±5-12% from nominal 154 mm; Phase IV torque +30-60%; rising trend in Phase III

Source: developed by the authors based on M. Hamadache *et al.* (2019)

Table 2 specifies how each fault class from Table 1 is reproduced in the physics-informed simulation through targeted modifications of parameters in the electromechanical model (Eqs. 5-7), with the mapping driven by the physical mechanism rather than by arbitrary perturbation. Mechanical friction (C_2) is reproduced by increasing the damping and static-torque terms, raising the steady-state power level and lengthening the switching phase. Obstruction (C_3) is introduced as a transient torque spike at a randomised position within the cycle, mimicking the encounter with a foreign object or a binding point in the slide chairs. Electrical degradation (C_4) modifies the supply-side parameters – the effective armature resistance is increased and the supply voltage reduced – which together depress the peak power and slow the rise time. Misadjustment (C_5) shifts the lock-engagement point in time, producing the asymmetric end-of-stroke signature characteristic of detection-bar drift. Severity for each class was sampled uniformly across a physically plausible range, and independent stochastic noise was added per cycle so that no two curves within a class are identical. The parameter-to-class mapping is one-to-one by construction, but the resulting power signatures overlap in several diagnostic regions, which is precisely why end-to-end learning from the raw curve outperforms hand-crafted feature pipelines. Operational artefacts. Six categories of real-world interference are layered onto the baseline curves so the models train under realistic conditions rather than on idealised signatures (Table 3). Each category can be independently enabled or disabled.

Table 3. Operational artefacts injected into the simulation

Artefact	Model	Coverage
EMI from traction current (catenary 25 kV, 50 Hz)	Sum of sinusoids at 50/100/150 Hz with amplitudes 2-8%	All electrified-line records
Voltage sag during train passage	0.3-1.5 s dip, magnitude 5-12% below nominal, randomised timing	All records, randomised

Table 3. Continued

Artefact	Model	Coverage
Temperature-dependent variation	Temperature-dependent electrical resistance and lubricant viscosity variations; ±3-7% baseline shift for temperatures below –10°C and above +35°C	All records, sampled from temperature distribution
Mechanical vibration from passing trains	Band-limited noise 20-80 Hz, amplitude proportional to train speed/load	~15% of records
Power supply fluctuation	Slow drift ±3-5% over cycle; spikes 2-5 ms up to 15% above nominal	Remote-station records
Sensor noise and quantisation	12-bit ADC quantisation; Gaussian SNR uniform in [35, 45] dB, simulating typical experimental data noise; 0.5% probability of single-sample outlier	All records

Source: developed by the authors based on M. Hamadache *et al.* (2019)

Mitigation of simulation-induced overfitting. A common concern with simulated datasets is that a deep network might learn to invert the generator rather than the diagnostic problem itself. Three measures address this. First, structural perturbation: each record draws independent perturbations on R_a (±12%), η (±8%), T_c and T_s (±15%), and load anchor points (±10%), so no two curves share identical model parameters even within the same class. Second, boundary blurring: about 8% of training records are sampled from overlap zones between adjacent classes (e.g., friction multiplier 1.25 paired with misadjustment 4%), preventing single-parameter threshold shortcuts. Third, randomised phase timing: phase boundaries vary stochastically by ±8-15% per record. The effectiveness of these measures is evaluated experimentally in the following sections. Preprocessing operates on signals carrying the full set of operational artefacts, without explicit removal. This is deliberate: the deep learning models are forced to learn diagnostic features in the presence of realistic interference rather than rely on pre-cleaned signals that would not be available in an operational digital twin deployment anyway. Curves are resampled to N = 4,000 samples (linear interpolation) and normalised to [0, 1] per record:

$$\hat{p}_i = (p_i - p_{min}) / (p_{max} - p_{min}). \quad (8)$$

Phases are segmented automatically by threshold detection and gradient analysis: Phase I (Unlocking), II (Blade movement initiation, peak power), III (Steady-state running), IV (Locking and detection). Figure 2 shows representative curves for each class with phase boundaries.

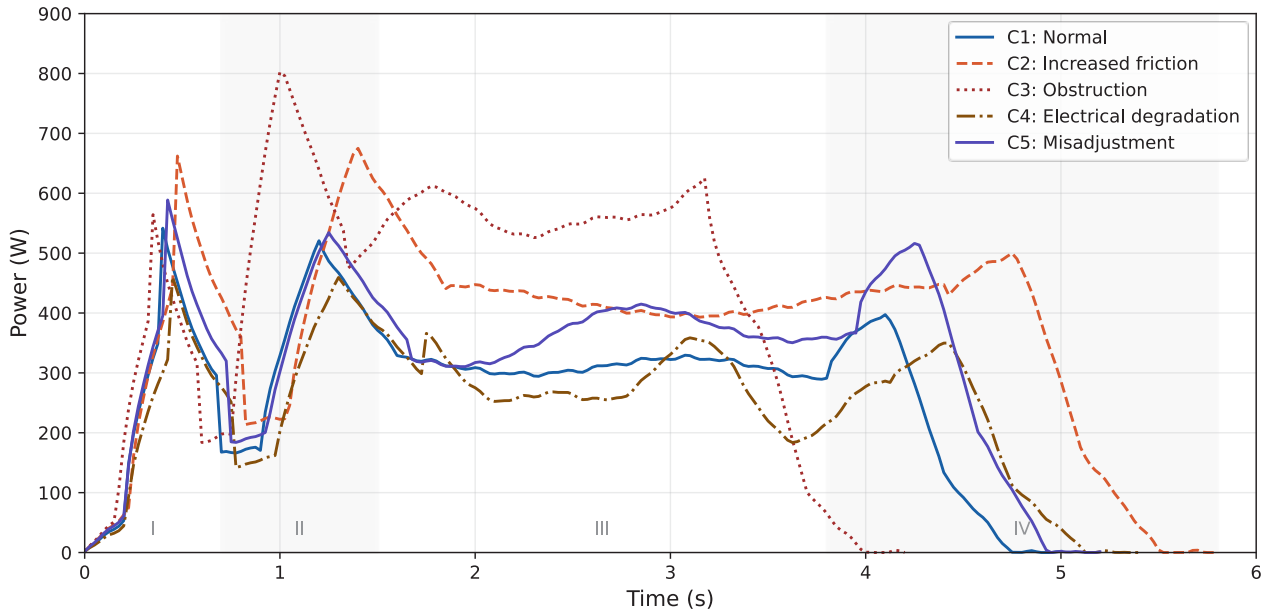

Figure 2. Switching power curves for five-point machine condition categories

Source: developed by the authors

The visible patterns match what the literature suggests. C_1 peaks near 520 W and runs at ~300 W steady-state for 5.0 s. C_2 shows elevated power across all phases (peak ~680 W, steady-state +30-40%, duration 5.8 s). C_3 has a sharp peak of ~820 W in Phase II with high Phase III variability. C_4 has a reduced peak (~460 W) but high-frequency fluctuations throughout. C_5 has a normal-looking peak but a rising Phase III trend and elevated Phase IV power (~520 W). For second task, twelve features per curve are extracted (Table 4). These were chosen to span the patterns visible in Figure 2 P_{peak} separates $C_1/C_2/C_3$ at the amplitude level; P_{std} catches the noisy character of C_4; P_{lock} flags the misadjustment locking signature; phase durations and energies cover the temporal aspects.

Table 4. Features extracted from switching power curves

No.	Feature	Description	Phase
1	P_{peak}	Maximum power	II
2	P_{mean}	Mean power during steady-state running	III
3	P_{std}	Standard deviation of power	I-IV
4	T_{total}	Total switching duration	I-IV
5	T_{unlock}	Unlocking phase duration	I
6	T_{run}	Steady-state phase duration	III
7	E_{total}	Total energy (integral of power)	I-IV
8	E_{phase2}	Energy during blade movement initiation	II
9	Skew	Skewness of power distribution	I-IV
10	Kurt	Kurtosis of power distribution	I-IV
11	$Slope_{rise}$	Gradient at Phase I-II transition	I-II
12	P_{lock}	Power level during locking	IV

Source: developed by the authors

Table 4 lists the twelve hand-crafted features supplied to the three classical baselines, organised along four dimensions: amplitude (P_{peak}, P_{mean}, P_{std}, P_{lock}), duration (T_{total}, T_{unlock}, T_{run}), energy (E_{total}, E_{phase2}), and shape (Skew, Kurt, $Slope_{rise}$), with at least one feature anchored to each phase of the switching cycle. The mapping back to Table 1 follows physically grounded patterns: friction (C_2) raises P_{mean} and T_{run}; obstruction (C_3) shows up in P_{peak} and the shape statistics through the transient torque spike; electrical degradation (C_4) depresses P_{peak} and $Slope_{rise}$ while extending T_{total}; misadjustment (C_5) shifts P_{lock} and the T_{unlock}/T_{total} ratio. Although physically interpretable, these twelve numbers compress a curve of order 10^3 time samples, and the ~7 F1 percentage points between Random Forest and CNN-LSTM measure how much diagnostic information is lost in that compression. The deep learning models for first time receive the full normalised curve as input (end-to-end learning). Split is time-aware to prevent leakage. For first task: 70/15/15 with chronological ordering, so all test records postdate all training records. For second task: split at the sequence level (140/30/30), no overlapping windows. SMOTE is applied only to the training subset for class balancing; validation and test subsets keep the original class distribution. Three classical baselines work on the 12-feature vectors of Table 4; three deep learning models work on the raw curve for classification (and on the sliding-window feature matrix for degradation).

Logistic Regression (multinomial, L2, C = 1.0) and Ridge Regression (cross-validated regularisation) serve as linear lower bounds. SVM/SVR with RBF kernel covers nonlinear kernel methods; hyperparameters were grid-searched over $C \in \{0.1, 1, 10, 100\}$, $\gamma \in \{0.001, 0.01, 0.1, 1\}$. Random Forest (300 trees, grid-searched depth, leaf size, and feature count) is the strong tabular baseline; it has been used in prior railway diagnostic work. All baselines run in scikit-learn 1.3 with feature standardisation (zero mean, unit variance). The deep learning architectures included a 1D-CNN consisting of three convolutional blocks (Conv1D + BatchNorm + ReLU + MaxPool, filters 32 → 64 → 128), global average pooling, and two fully connected layers (128, 64) with dropout 0.3. The output is softmax (5) for classification, linear for regression. The LSTM has two stacked layers (64 and 32 hidden units), dropout 0.3, and two fully connected layers (Sherstinsky, 2020). For classification, the input curve is split into 100 subsequences of 40 samples each; for degradation, the input is a sequence of feature vectors. The hybrid CNN-LSTM applies a shared 1D-CNN feature extractor (two Conv1D blocks, 32 and 64 filters) to non-overlapping segments of the curve, then feeds the resulting per-segment feature sequence into a single LSTM layer (64 units). The motivation for this hybrid is that switching curves carry both local diagnostic patterns (sharp friction spikes within Phase II), which are best captured by convolution, and global sequential structure (the relationship between Phase I and Phase III), which benefits from recurrence (Sherstinsky, 2020).

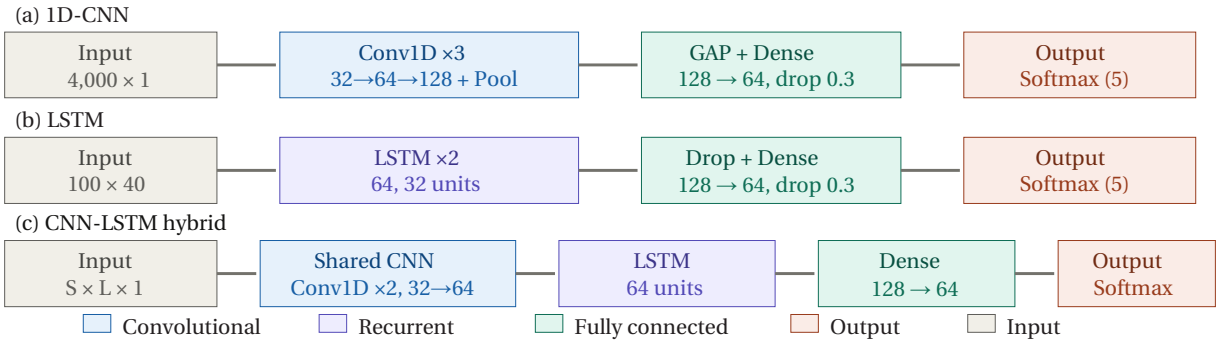

Figure 3. Schematic diagrams of the three deep learning architectures

Source: developed by the authors

All deep models train in PyTorch 2.0 on an NVIDIA RTX 5070 (12 GB), with Adam ($\beta_1 = 0.9$, $\beta_2 = 0.999$), initial learning rate 10^{-3}, batch size 64, max 150 epochs, early stopping (patience 15) on validation loss, He initialisation for convolutional layers and Xavier for dense layers. Hyperparameters are grid-searched per architecture; reported numbers are means over three runs with different random seeds. Classification metrics are accuracy, precision, recall, F1 per class, and macro F1 (the primary metric, given class imbalance): Computational metrics: training time, inference time per record, model size. Practical criteria: interpretability, robustness to reduced data (50% and 25% subsets), real-time suitability (target inference <50 ms).

Results and Discussion

Table 5 presents the test-set classification performance of all six models, covering three classical baselines and three deep learning architectures, with evaluation based on accuracy, macro F1-score and class-wise F1-scores for the five fault classes.

Table 5. Classification performance (test set)

Metric	LogReg	SVM	RF	1D-CNN	LSTM	CNN-LSTM
Accuracy	0.821	0.873	0.889	0.934	0.912	0.951
Macro F1	0.793	0.854	0.871	0.918	0.894	0.938
F1 (C_1)	0.891	0.924	0.932	0.962	0.948	0.971
F1 (C_2)	0.802	0.861	0.882	0.931	0.908	0.947
F1 (C_3)	0.694	0.772	0.801	0.878	0.852	0.912
F1 (C_4)	0.771	0.836	0.858	0.905	0.881	0.924
F1 (C_5)	0.807	0.878	0.880	0.916	0.882	0.935

Source: developed by the authors

Random Forest is the strongest classical baseline at macro F1 = 0.871. Each deep model exceeds it: 1D-CNN by 5.4 percentage points (F1 = 0.918), CNN-LSTM by 7.7 (F1 = 0.938). Logistic Regression at F1 = 0.793 is far behind, since linear decision boundaries simply do not fit this signal-feature relationship. SVM with RBF (F1 = 0.854) sits in between, beating LogReg but not RF. The widest gap is in C_3 (obstruction): 0.801 for RF against 0.912 for CNN-LSTM, a 13.9-point difference. The cause is straightforward. The obstruction-induced power spike has variable timing, irregular shape, and short duration; none of that survives reduction to twelve summary statistics. The deep models look at the raw curve and recover what the features lose. It should be noted that the full operational artefacts (EMI, voltage sags, temperature, vibration, sensor noise) were active during both training and evaluation. The models were not given a sanitised signal; the numbers above hold under realistic interference. The 1D-CNN's competitiveness despite its simpler architecture is partly explained by what convolutional filters do well: they suppress high-frequency components like EMI and vibration while preserving diagnostic content. C_1 is the easiest class for every deep model (largest sample count); C_3 is the hardest, both because the sample size is small and because obstruction signatures are highly variable. The most consistent error mode in the confusion matrices (Fig. 4) is $C_2 \leftrightarrow C_5$ confusion. That makes physical sense: both increased friction and misadjustment elevate mechanical resistance and produce overlapping signatures.

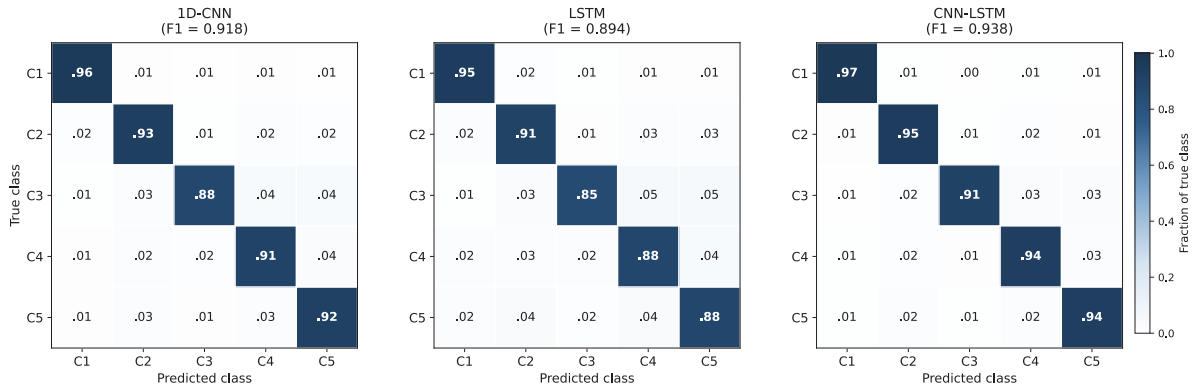

Figure 4. Confusion matrices for the three deep learning architectures

Source: developed by the authors

Figure 4 shows per-class confusion matrices for the three deep learning models. Two patterns are common to all of them: C_1 (Normal) is the easiest class (0.95-0.97 on the diagonal), C_3 (Obstruction) the hardest (0.85-0.91), with the dominant misclassification directing C_3 toward C_4 (electrical degradation) at 0.03-0.05. This is physically interpretable, since both perturb power in the same phases and only the sharpness of the obstruction spike

distinguishes them. The differences between architectures concentrate on these harder classes. CNN-LSTM lifts diagonal accuracy on C_3 from 0.85 (LSTM) to 0.91 and on C_4 from 0.88 to 0.93, while 1D-CNN sits between the two and is two to three points above LSTM on C_2, C_4 and C_5. The pattern matches the underlying inductive biases: convolutional layers localise the short-duration spikes and slope changes that pure recurrence misses, and the hybrid gains by combining local detection with sequence-level context, which is why its advantage concentrates exactly on the failure modes that matter most for predictive maintenance.

Table 6. Degradation prediction performance (test set, H = 50 cycles)

Metric	Ridge	SVR	RF	1D-CNN	LSTM	CNN-LSTM
MAE	0.071	0.058	0.052	0.042	0.031	0.033
RMSE	0.094	0.078	0.069	0.058	0.044	0.046
R^2	0.713	0.802	0.845	0.891	0.937	0.931

Source: developed by the authors

The architecture ranking inverts. LSTM, which placed second on classification, gives the best degradation prediction at $R^2 = 0.937$. CNN-LSTM follows closely ($R^2 = 0.931$); 1D-CNN sits a step below at 0.891. Random Forest reaches 0.845, which is 4.6 R^2 points behind 1D-CNN. Ridge Regression collapses to 0.713. The wear trajectory is too nonlinear for straight-line extrapolation. The narrower gap between baselines and deep models on this task is informative. The twelve features in Table 4 were chosen specifically as wear indicators, guided by multiphase curve analysis principles standard in the field, so Random Forest can work with input that already encodes most of what matters. Multi-class fault classification, on the other hand, asks for distinctions that summary statistics do not capture, and the deep models recover that information from the raw curve. Why does LSTM beat CNN-LSTM here while losing to it on classification? Classification is a per-curve decision: spatial patterns within a single signal carry the answer, and convolutional filters are the natural tool. Degradation is a per-sequence decision: features evolve slowly across hundreds of cycles, and the recurrent cell state is what tracks that evolution (Sherstinsky, 2020). CNN-LSTM has the recurrent component, but the convolutional preprocessing it applies to each segment adds capacity that helps less here, and probably hurts slightly, because it gives the model more parameters to fit on the same data.

Figure 5 shows a typical LSTM prediction trajectory on a held-out 800-cycle sequence. The model tracks both the slow degradation regime in the first ~470 cycles and the transition to accelerated wear that follows, with the 95% confidence band remaining narrow throughout; visible widening occurs at the regime transition rather than at long prediction horizons, indicating that residual uncertainty is dominated by the change-of-dynamics moment rather than by error accumulation. The pointwise error at the early-life sample shown in the inset is 0.012 in absolute terms (cycle 4: actual 0.328, predicted 0.340), or 3.7% relative; the trajectory tracking remains comparably tight throughout both regimes, with no systematic bias visible at the transition or near the maintenance threshold. The prediction horizon for this trajectory is H = 50 cycles. The actual degradation indicator crosses the maintenance threshold (0.74) at cycle ~750, while the model's H = 50 prediction reaches that same threshold at cycle ~700, giving a 50-cycle lead time to schedule intervention before the device crosses the action boundary. At a typical mainline turnout throughput of 10-30 switching operations per day, this corresponds to roughly two to five days of warning, which is the practically useful margin of the predictive maintenance approach.

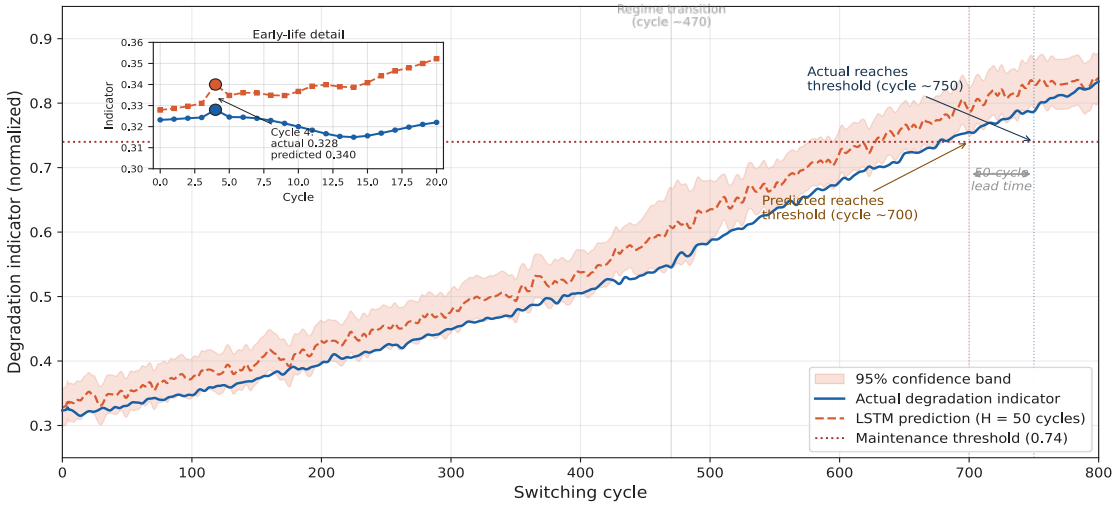

Figure 5. Predicted vs actual degradation trajectory, LSTM model

Source: developed by the authors

Table 7. Computational efficiency

Parameter	LogReg/Ridge	SVM	RF	1D-CNN	LSTM	CNN-LSTM
Training time	0.3 s	12 s	8 s	~18 min	~42 min	~55 min
Inference (ms/rec)	0.02	0.8	0.4	1.8	5.4	6.1
Model size (MB)	<0.01	0.15	38	0.72	0.48	0.84
GPU required	No	No	No	Recommended	Recommended	Recommended

Source: developed by the authors

Baselines are orders of magnitude faster in both training and inference. All six models, however, finish inference well below the 50 ms real-time threshold, so deployment inside the digital twin pipeline is feasible for any of them. The 1D-CNN at 1.8 ms is roughly 4× slower than RF but still leaves substantial margin for edge deployment on modest hardware. Baselines lose 3.7-4.7% under data reduction; deep models lose 6.1-11.4%. This is not unexpected. Tree ensembles and SVMs working on 12 features need fewer samples to fit a decent decision boundary than a deep network learning representations from 4,000 raw samples. LSTM is hit hardest at -11.4%, consistent with the well-known fact that recurrent networks are data-hungry. 1D-CNN is the most stable deep model at -6.1%, which can be attributed to the convolutional inductive bias (locality, translation invariance) acting as an implicit regulariser. A practical question follows: at what training-set size does Random Forest become competitive with the best deep model? At 25% training data, 1D-CNN reaches F1 = 0.862, and Random Forest at full data reaches 0.871. They sit within 0.009 of each other, RF marginally ahead. An operator with about a fifth of the data used here would do roughly as well with Random Forest on hand-crafted features as with a deep model. Below that threshold, RF is the safer bet; above it, deep learning pulls ahead. This has a clean deployment implication. Operators accumulating their first labelled dataset should not skip directly to deep learning. Random Forest delivers comparable accuracy faster and at lower compute cost. Migration to 1D-CNN or CNN-LSTM becomes worthwhile once the dataset reaches roughly five times the crossover threshold (full scale here).

Table 8. Macro F1-score under reduced training data

Training data	LogReg	SVM	RF	1D-CNN	LSTM	CNN-LSTM
100%	0.793	0.854	0.871	0.918	0.894	0.938
50%	0.781	0.838	0.855	0.896	0.851	0.908
25%	0.764	0.814	0.836	0.862	0.792	0.867
Drop 100 → 25%	-3.7%	-4.7%	-4.0%	-6.1%	-11.4%	-/–7.6%

Source: developed by the authors

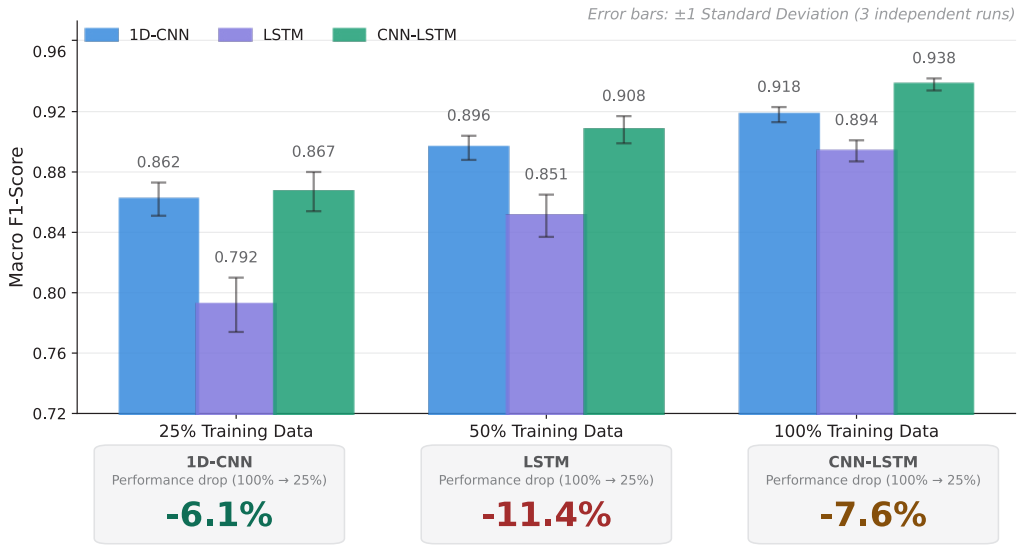

Figure 6. Classification F1 vs training data volume

Source: developed by the authors

Table 9. Multi-criteria comparison

Criterion	LogReg	SVM	RF	1D-CNN	LSTM	CNN-LSTM
Classification F1	-	-	-	++	+	+++
Degradation R^2	- -	-	-	+	+++	++

Table 9. Continued

Criterion	LogReg	SVM	RF	1D-CNN	LSTM	CNN-LSTM
Training speed	+++	+++	+++	++	+	+
Inference speed	+++	+++	+++	++	++	++
Robustness (limited data)	++	++	++	++	–	+
Feature engineering	Yes	Yes	Yes	No	No	No
Interpretability	+++	+	++	+	-	-

Note: legend: +++ excellent; ++ good; + acceptable; - weak; - - poor
Source: developed by the authors

Classical baselines train in seconds, infer in fractions of a millisecond, run on a CPU, and are easy to interpret. They also lose accuracy on both tasks, and they require manual feature engineering. That last point matters: feature engineering assumes domain expertise, may not generalise to other point machine types, and discards most of what the raw curve contains. Among the deep models, no single architecture wins everywhere. CNN-LSTM is the right pick when classification accuracy is the priority and resources allow. LSTM is preferred for degradation prediction. 1D-CNN is the practical choice for edge deployment and data-scarce scenarios; at 25% training data it still reaches F1 = 0.862, which is close to what RF achieves on full data (0.871). The numbers in Tables 5 and 6 are what justify deep learning here. The weakest deep model (LSTM, F1 = 0.894) beats the strongest baseline (RF, F1 = 0.871) on classification by 2.6 points, and the gap reaches 7.7 for CNN-LSTM. On degradation prediction, LSTM beats RF by 10.9 R^2 points. Whether those margins justify the extra training time depends on operational priorities, which connects to the cross-generator validation – the question of whether the gap is real or simply a simulation artefact. The recommended deployment uses two deep models. A 1D-CNN at the edge handles real-time classification on each switching event; an LSTM at the server level handles periodic degradation forecasting on aggregated features. Operators without the data or compute for this configuration should start with Random Forest. At full-data F1 = 0.871 it serves as a competent bootstrap, and migration to deep learning becomes worthwhile at around five times the crossover threshold. At the network level, where dozens or hundreds of turnouts feed the analytics layer simultaneously, distributed training and cost-aware resource scheduling become the binding constraints for cloud-deployed transport analytics at scale.

A reasonable concern about simulation-based studies is whether the models actually learned the diagnostic problem or just memorised the generating function. Two experiments test this directly. Cross-generator validation was performed using a second parameterisation of the electromechanical model (Generator B) was created by altering motor constants ($R_a = 3.2\ \Omega$, $K_e = 0.38$ V·s/rad), gearbox parameters ($n = 40$, $\eta = 0.42$), and load anchors (shifted ±15% from Generator A). Generator B produces curves with visibly different absolute amplitudes, durations, and phase proportions, but the five fault classes remain physically the same. All six models, trained only on Generator A, were evaluated on 750 records from Generator B without retraining or fine-tuning.

Table 10. Cross-generator validation, trained on A, tested on B

Metric	LogReg	SVM	RF	1D-CNN	LSTM	CNN-LSTM
F1 (same-gen)	0.793	0.854	0.871	0.918	0.894	0.938
F1 (cross-gen)	0.602	0.681	0.871	0.831	0.798	0.862
Drop	-24.1%	-20.3%	-18.0%	-9.5%	-10.7%	-8.1%

Source: developed by the authors

Every model loses something on the cross-generator data, which means the simulation parameters do influence what was learned; the models have not converged on a perfectly general diagnostic rule. The size of the loss separates the two groups sharply, though. Baselines lose 18-24%. Deep models lose 8-11%. CNN-LSTM keeps 91.9% of its accuracy across the shift; Random Forest keeps only 82.0%. The likely cause lies in what each kind of model is sensitive to. Random Forest, SVM, and LogReg work on absolute feature values: peak power in watts, duration in seconds. When Generator B shifts those numbers by 10-15%, the decision boundaries learned on A no longer hit them in the right places. The deep models work on the raw curve and end up learning relative shape information: the proportion of Phase II peak to Phase III steady state, the slope of power rise, the symmetry of locking. Those proportions are largely preserved across the parameter shift, which is why the deep models survive it. One comparison is particularly telling: CNN-LSTM's cross-generator F1 of 0.862 is essentially the same as Random Forest's same-generator F1 of 0.871. Under a domain shift Random Forest itself cannot survive without an 18% drop, CNN-LSTM almost matches Random Forest on home turf. This alone is arguably the strongest single piece of evidence that the deep model gain is not a simulation artefact. Artefact ablation served as a reverse test: if the deep models had memorised generator-specific artefacts, then removing the noise that obscured those artefacts should have improved their performance. The underlying pattern would be easier to read. If instead the models learned features that hold up under noise, performance should stay roughly where it was when artefacts are removed.

Table 11. Artefact ablation, classification with artefacts removed at test time

Artefact removed	1D-CNN ΔF1	LSTM ΔF1	CNN-LSTM ΔF1
None (baseline)	0.918	0.894	0.938
EMI	+0.003	+0.005	+0.002
Voltage sags	+0.008	+0.011	+0.006
Vibration	+0.002	+0.004	+0.001
Temperature	-0.004	-0.002	-0.003
All artefacts	+0.012	+0.019	+0.008
F1 without artefacts	0.930	0.913	0.946

Source: developed by the authors

Removing all artefacts at once moves F1 by only +0.008 to +0.019, small enough to confirm that none of the deep models depends on the noise structure for classification. Voltage sags account for most of the visible improvement (+0.6 to +1.1 percentage points), which fits the physics; sags directly modulate amplitude and shrink the gap between classes whose signatures already differ mainly in amplitude. The pattern across models is also instructive. LSTM, the most sequence-dependent architecture, shows the largest aggregate gain when artefacts are removed (ΔF1 = +0.019); 1D-CNN and CNN-LSTM, with their convolutional feature extractors, gain only +0.012 and +0.008 respectively. This is consistent with convolution acting as an implicit denoising operation. The filters suppress high-frequency artefacts during the forward pass, so removing those artefacts at test time produces a smaller change than for a purely sequential model. None of the three deep models gains more than two percentage points from a fully clean test set, and the individual artefact effects (≤1.1 percentage points) fall within the magnitude range typical of per-seed F1 fluctuations in deep-learning training at this data scale. The conclusion is qualitative but robust: none of the models relies on any specific artefact class as a discriminative cue – the precise property artefact ablation is designed to detect. Temperature artefacts behave the opposite way. Removing them costs the models 0.2-0.4 percentage points. A plausible explanation is that the models learned to use winding-resistance variation as a supplementary cue. Cold conditions raise resistance and also increase mechanical friction, so the two effects co-vary in the training data. Remove the temperature signal and a weak but real cue is gone. A cleaner deep-vs-classical comparison emerges in this artefact-free regime. CNN-LSTM on clean signals reaches F1 = 0.946. Random Forest, evaluated separately on clean feature vectors, reaches 0.904. The 4.2-point gap that survives in the absence of any noise rules out the explanation that the deep models' advantage came from exploiting noise patterns. What is left is the difference in input itself: twelve summary statistics on one side, four thousand raw samples on the other.

The numerical results in Tables 5-11 invite comparison with the broader literature on point machine fault diagnosis. Seven works covering different methodological strands frame this comparison. D. Ou *et al.* (2019) used hand-crafted features from switching force curves with classical classifiers and reported accuracies above 90% on multi-class problems. The Random Forest baseline reported here at F1 = 0.871 sits close to that range – the gap maps to the difference between accuracy and macro F1 on imbalanced data, and the additional 7.7 percentage points achieved by CNN-LSTM quantify how much diagnostic information the raw curve carries beyond hand-crafted features. Z. Shi *et al.* (2023) applied an improved deep CNN combined with Support Vector Data Description (SVDD) classification to ZDJ7 point machine current curves and reported 96.59% accuracy with robust performance across varying signal-to-noise ratios; this places between the same-generator CNN-LSTM result (93.8%) and the cleaner artefact-free regime (94.6%), consistent with their explicit focus on noise immunity through adaptive batch normalisation. C. Zhang *et al.* (2019) and S. Givnan *et al.* (2022) applied a locally connected autoencoder to point machine current curves, anomaly-flagging via reconstruction error. Their architecture is unsupervised in spirit, while the present approach is supervised end-to-end, complementary rather than competing: autoencoders work where labelled fault examples are scarce, supervised models pull ahead once labels accumulate. M. Sysyn *et al.* (2019) worked on instrumented 1,520 mm gauge turnouts using vehicle-based inertial sensors with classical ML, reporting accuracies in the 0.85-0.90 range, close to the Random Forest baseline reported here. The cross-modality coincidence suggests that the classical-features ceiling for this infrastructure sits independently around F1 = 0.87, a ceiling the deep models of the present study exceed by 5-8 points.

Z. Hu *et al.* (2022) surveyed data-driven fault diagnosis for point machines and observed that hybrid CNN+RNN architectures outperform single-paradigm models at the cost of higher data requirements. The reduced-data experiment gives that observation a measurable form: CNN-LSTM at 25% training data drops to F1 = 0.867, statistically indistinguishable from Random Forest at full data. Z. Hu's *et al.* (2022) crossover threshold thus has a quantifiable location for this application. Y. Sun *et al.* (2020) used SVM on acoustic signals and reported accuracy in a range similar to electrical-signature classical methods. The convergence across orthogonal sensor modalities indicates that the ~0.87-0.89 ceiling reflects the information content of any single channel rather than the feature extractor – a direct argument for multi-modal fusion as a future direction. S. Reetz *et al.* (2024) developed a Bayesian network combining expert

knowledge with limited data. Their motivation overlaps with the present study, but the methodological direction inverts: knowledge-based versus data-driven. In low-data regimes the Bayesian route may close the gap exposed by the 25%-data experiment, and a hybrid deep-learning + Bayesian architecture is a natural target for integrating symptom classification with root-cause inference within the digital twin. Beyond the point-machine-specific literature, T. Zonta *et al.* (2020) surveyed predictive maintenance implementations across Industry 4.0 sectors and reported that deep learning consistently outperforms classical methods at the cost of higher training data and computational requirements, a cross-industry finding the reduced-data experiment (Table 8) and computational efficiency analysis (Table 7) confirm specifically for railway point machine diagnosis. The results indicate that advanced data-driven models can consistently outperform classical approaches, revealing additional diagnostic information and highlighting the value of combining methodologies for improved fault detection.

Conclusions

A four-layer digital twin architecture (Physical, Data, Analytics, Decision) for predictive maintenance of railway point machines on 1,520 mm gauge networks was developed and validated through systematic comparison of deep learning and classical machine learning architectures on a physics-informed simulated dataset of 5,000 switching power curves. The hybrid CNN-LSTM achieved the highest fault classification accuracy (macro F1 = 0.938), exceeding the strongest classical baseline (Random Forest, F1 = 0.871) by 7.7 percentage points. For degradation trend prediction over a 50-cycle horizon, LSTM reached $R^2 = 0.937$ against 0.845 for Random Forest. Cross-generator validation showed deep models losing 8-11% accuracy under parameter shift versus 18-24% for classical baselines, and artefact ablation moved F1 by only 0.8-1.9 percentage points – confirming that the deep learning advantage stems from learning relative shape information rather than memorising simulation artefacts or noise patterns. The recommended deployment combines 1D-CNN inference at the edge (1.8 ms per switching event) for real-time fault classification with LSTM-based degradation forecasting at the server level. Random Forest serves as an interim solution for operators with limited labelled data, becoming competitive with deep models at approximately 25% of the training data volume used here. The most consequential next step is validation on field operational data from Ukrainian railways in collaboration with JSC "Ukrzaliznytsia". Transfer learning would help adapt models across point machine types; online learning would enable continuous adaptation as new switching data accumulates; lightweight architectures may close the data-efficiency gap with classical baselines. The digital twin scope can extend to crossing-nose wear, rail fastenings, and ballast condition, transforming the framework into full turnout health management. Integration of deep reinforcement learning for track geometry maintenance, would shift the framework from passive diagnosis to closed-loop maintenance policy optimisation.

Acknowledgements

None.

Funding

None.

Conflict of Interest

None.

References

[1] Givnan, S., Chalmers, C., Fergus, P., Ortega-Martorell, S., Whitaker, R., & Al-Jumeily, D. (2022). Anomaly detection using autoencoder reconstruction upon industrial motors. *Sensors*, 22(9), article number 3166. doi: 10.3390/s22093166.

[2] Hamadache, M., Dutta, S., Olaby, O., Kestelyn, X., & Raison, B. (2019). On the fault detection and diagnosis of railway switch and crossing systems: An overview. *Applied Sciences*, 9(23), article number 5129. doi: 10.3390/app9235129.

[3] Hector, I., & Panjanathan, R. (2024). Predictive maintenance in Industry 4.0: A survey of planning models and machine learning techniques. *PeerJ Computer Science*, 10, article number e2016. doi: 10.7717/peerj-cs.2016.

[4] Holub, H., Voronko, I., Bachynskyi, V., & Korchovyi, V. (2025). Integrated optimization of machine learning algorithms and cloud resources for big data processing in transportation systems. *Science and Technology Today*, 11(52), 1990-2000. doi: 10.52058/2786-6025-2025-11(52)-1990-2000.

[5] Hu, Z., Zhang, Y., Lv, R., Yan, J., & Wang, M. (2022). Data-driven technology of fault diagnosis in railway point machines: review and challenges. *Transportation Safety and Environment*, 4(4), article number tdac036. doi: 10.1093/tse/tdac036.

[6] Kaewunruen, S., Sresakoolchai, J., & Lin, Y.-H. (2023). Digital twins for managing railway bridge maintenance, resilience, and climate change adaptation. *Sensors*, 23(1), article number 252. doi: 10.3390/s23010252.

[7] Kostenko, K.L., Serdiuk, T.M., & Skalko, V.V. (2023). Robotization of diagnostics of railway automation and communication systems. *Science and Transport Progress*, 3(103), 5-12. doi: 10.15802/stp2023/292720.

[8] Ou, D., Xue, R., & Cui, K. (2019). A data-driven fault diagnosis method for railway turnouts. *Transportation Research Record*, 2673(4), 448-457. doi: 10.1177/0361198119837222.

[9] Reetz, S., Neumann, T., Schrijver, G., van den Berg, A., & Buursma, D. (2024). Expert system based fault diagnosis for railway point machines. *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit*, 238(3), 322-334. doi: 10.1177/09544097231195656.

[10] Sherstinsky, A. (2020). Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network. *Physica D: Nonlinear Phenomena*, 404, article number 132306. doi: 10.1016/j.physd.2019.132306.
[11] Shi, Z., Du, Y., & Yao, X. (2023). Fault diagnosis of ZDJ7 railway point machine based on improved DCNN and SVDD classification. *IET Intelligent Transport Systems*, 17(8), 1649-1674. doi: 10.1049/itr2.12357.
[12] Siroklyn, I.M., & Kladko, A.S. (2018). Modern approaches to building intelligent systems for monitoring the technical condition of rolling stock. *Information and Control Systems on Railway Transport*, 4, 49-49.
[13] Sresakoolchai, J., & Kaewunruen, S. (2023). Railway infrastructure maintenance efficiency improvement using deep reinforcement learning integrated with digital twin based on track geometry and component defects. *Scientific Reports*, 13(1), article number 2439. doi: 10.1038/s41598-023-29526-8.
[14] Sun, Y., Cao, Y., Xie, G., & Wen, T. (2020). Condition monitoring for railway point machines based on sound analysis and support vector machine. *Chinese Journal of Electronics*, 29(4), 786-792. doi: 10.1049/cje.2020.06.007.
[15] Sysyn, M., Gruen, D., Gerber, U., Nabochenko, O., & Kovalchuk, V. (2019). Turnout monitoring with vehicle based inertial measurements of operational trains: A machine learning approach. *Communications – Scientific Letters of the University of Zilina*, 21(1), 42-48. doi: 10.26552/com.C.2019.1.42-48.
[16] Tao, F., Xiao, B., Qi, Q., Cheng, J., & Ji, P. (2022). Digital twin modeling. *Journal of Manufacturing Systems*, 64, 372-389. doi: 10.1016/j.jmsy.2022.06.015.
[17] Thompson, E.A., Lu, P., Alimo, P.K., Atuobi, H.B., Akoto, E.T., & Abbew, C.K. (2025). Revolutionizing railway systems: A systematic review of digital twin technologies. *High-speed Railway*, 3(3), 238-250. doi: 10.1016/j.hspr.2025.05.005.
[18] van Dinter, R., Tekinerdogan, B., & Catal, C. (2022). Predictive maintenance using digital twins: A systematic literature review. *Information and Software Technology*, 151, article number 107008. doi: 10.1016/j.infsof.2022.107008.
[19] Zhang, C., Zhang, X., Peng, X., & Li, Y. (2019). Fault diagnosis of railway point machines using the locally connected autoencoder. *Applied Sciences*, 9(23), article number 5139. doi: 10.3390/app9235139.
[20] Zonta, T., da Costa, C.A., da Rosa Righi, R., de Lima, M.J., da Trindade, E.S., & Li, G.P. (2020). Predictive maintenance in the Industry 4.0: A systematic literature review. *Computers & Industrial Engineering*, 150, article number 106889. doi: 10.1016/j.cie.2020.106889.

Прогнозне технічне обслуговування залізничних стрілочних приводів із використанням цифрового двійника та глибокого навчання

Ігор Каткало
Аспірант
Національний транспортний університет
01010, вул. М. Омеляновича-Павленка, 1, м. Київ, Україна
https://orcid.org/0009-0008-2672-1005

Галина Голуб
Кандидат технічних наук, доцент
Національний транспортний університет
01010, вул. М. Омеляновича-Павленка, 1, м. Київ, Україна
https://orcid.org/0000-0002-4028-1025

Анотація. Стрілочні електроприводи є критично важливими компонентами інфраструктури залізничної сигналізації, а їх відмови становлять 15-20 % усіх збоїв у роботі сигнальних систем європейських мереж, спричиняючи значні експлуатаційні та економічні втрати. Метою дослідження була розробка підходу до предиктивного обслуговування стрілочних електроприводів мереж колії 1520 мм на основі цифрового двійника, що забезпечує раннє виявлення несправностей за електричними сигнатурами потужності, зареєстрованими під час кожного циклу переведення. Методологія дослідження поєднала чотиришарову архітектуру цифрового двійника (фізичний шар, шар даних, аналітичний шар і шар прийняття рішень) із порівнянням трьох архітектур глибокого навчання: одновимірної згорткової нейронної мережі (1D-CNN), мережі довгої короткочасної пам'яті (LSTM) та гібридної CNN-LSTM-моделі. Їх зіставляли з логістичною регресією, методом опорних векторів (SVM) і випадковим лісом на синтетичному фізично обґрунтованому наборі з 5000 кривих потужності для стрілочних електроприводів СП-6М та ВСП-220. Моделі оцінювали за багатокласовою класифікацією п'яти категорій стану та прогнозуванням деградації на горизонті 50 циклів. Гібридна CNN-LSTM досягла найвищої точності класифікації (macro F1 = 0,938), а LSTM – найкращого результату для прогнозу деградації (R^2 = 0,937); обидві істотно перевершили найкращу baseline-модель (Random Forest: F1 = 0,871, R^2 = 0,845). Експерименти cross-generator validation та artefact ablation підтвердили, що перевага глибокого навчання походить з виявлення переносних ознак форми сигналу, а не з запам'ятовування симуляції. Рекомендовано конфігурацію розгортання: 1D-CNN на edge для класифікації у реальному часі, LSTM на серверному рівні для прогнозу деградації, Random Forest як проміжне рішення для операторів з обмеженими розміченими даними. Практична цінність отриманих результатів полягає у формуванні науково обґрунтованих рекомендацій з розгортання аналітичного шару цифрового двійника на мережах колії 1520 мм, що сприяє переходу від планового до обслуговування за фактичним станом та зменшенню непланових простоїв

Ключові слова: керування; діагностика; аналіз енергетичних сигнатур; моніторинг; архітектура; моделювання