Key Technologies for Digital Transformation and Performance Optimization of Manufacturing Enterprises Based on Deep Learning

Wei Qiu[1], Ying Zhao[1], Mengxia Ren[1], Donghe Liang[2]
[1]Shandong Xiehe University, Jinan 250109, Shandong, China
[2]Xi'an FanYi University, Xi'an 710105, Shaanxi, China
Corresponding author: Wei Qiu (e-mail: 13793175324@163.com)

ABSTRACT With the rapid development of intelligent sensing, industrial wireless communication, and Electromagnetic Waves, Antennas and Propagation technologies, the efficient integration of heterogeneous manufacturing data has become essential for digital transformation and operational optimization. This study proposes a deep learning-driven technology framework for performance optimization in manufacturing enterprises by integrating multimodal data fusion, attention-enhanced long short-term memory (LSTM) networks, deep reinforcement learning, and edge deployment. Multi-source equipment signals, including vibration, current, and temperature data, are aligned and fused to construct representative health features for remaining useful life prediction and fault diagnosis. The predicted health states are further utilized to dynamically optimize maintenance scheduling through reinforcement learning, forming a closed-loop architecture from state perception to intelligent decision-making. Experimental validation demonstrates that the proposed framework achieves 96.8% fault prediction accuracy, improves Overall Equipment Effectiveness by 11.8%, and significantly reduces unplanned downtime while enabling low-latency edge inference for real-time industrial applications. The proposed methodology provides an effective solution for data-driven digital transformation and offers valuable engineering references for distributed sensing, industrial wireless monitoring, and intelligent communication systems associated with Electromagnetic Waves, Antennas and Propagation.

INDEX TERMS deep learning, industrial intelligent sensing, wireless monitoring; digital transformation, predictive maintenance

I. INTRODUCTION

THE core goal of digital transformation for manufacturing enterprises suggests that converting industrial data assets from processes into quantifiable operational performance remains the key objective. This integration ensures that distribution data are effectively leveraged to enhance the precision and efficiency of complex production systems. However, the significant evidence could indicate that no effective conversion path currently exists between massive multi-source heterogeneous data and performance indicators such as equipment comprehensive efficiency and unplanned downtime. Moreover, the findings may suggest that this article explores how to build a technical closed loop from state perception to decision execution, achieving accurate perception of manufacturing system operating status. Furthermore, the important results might indicate that real-time prediction of remaining life and adaptive scheduling of maintenance tasks could demonstrate critical value in this context. Study shows urgency exists solving this problem. Given that the evidence demonstrates that traditional manufacturing models appear unable to cope with equipment load fluctuations and uncertainty caused by customized production [1], the significant findings could suggest that these limitations may indicate substantial operational risk. Additionally, the results might indicate that the full release of data value could directly determine the effectiveness of digital transformation in manufacturing enterprises. In light of these key findings, the evidence may suggest that reducing operation and maintenance costs and improving equipment utilization could demonstrate the core path for improving operational performance. Research shows data-driven decision-making deepens industrial application [2].

The transformation of data into performance may suggest that three core challenges could fundamentally shape the field. Moreover, the significant findings indicate that manufacturing data exhibits multi-source heterogeneity characteristics, with signals such as vibration, current, and temperature having different sampling frequencies and severe noise interference. However, the key evidence may suggest that the primary challenge is how to extract effective features that characterize equipment degradation patterns from mixed data [3,4]. Given the high demand for decision reliability in industrial scenarios, research findings may indicate that despite the powerful pattern recognition capabilities of deep learning models, there are still critical limitations. Interpretability limits engineers' confidence [5,6]. Furthermore, the significant evidence may suggest that prediction and decision-making processes are disconnected from each other, and the existing systems could indicate a tendency to stop at fault warning, failing to dynamically transform prediction information into maintenance task scheduling strategies [7,8]. Therefore, the key findings might indicate that this disconnection results in technical investment that cannot ultimately be reflected in actual performance improvements such as reduced downtime. In light of these results, the important evidence may suggest that relying solely on optimization of a single technical link appears to demonstrate insufficient capacity to achieve a complete transformation of data into performance. Notwithstanding these observations, the significant findings could indicate that there is an urgent need for a comprehensive technical solution that might demonstrate clear guidance by operational performance. Additionally, the key results may suggest that this solution could integrate data fusion, interpretable prediction, and adaptive decision-making into a unified framework. Both academia and industry may suggest that a wealth of methodological achievements has emerged in the field of device health management and intelligent maintenance. In terms of fault feature extraction, methods based on convolutional neural networks can automatically learn deep features from vibration signals, but such research mostly focuses on the classification and recognition of fault patterns, without establishing a correlation between feature evolution trends and equipment remaining life [9,10]. In terms of modeling degradation trends, long short-term memory networks are widely used for time-series data prediction, which can effectively fit the trajectory of device performance degradation. However, the predicted probabilities output by the model lack interpretability support, making it difficult to gain the trust of on-site engineers [11,12]. Moreover, the significant findings could indicate that deep reinforcement learning demonstrates the ability to generate dynamic strategies and adaptively schedule maintenance tasks based on environmental conditions. However, the evidence may suggest that existing decision models appear to assume that state information is completely observable. In light of these results, the data could indicate that the input is mostly idealized simulation data, which appears disconnected from the uncertain temporal characteristics of the actual prediction output [13,14]. Research shows deployment schemes lack interpretability. Furthermore, the significant findings may suggest that model compression technology demonstrates the ability to enable deep learning algorithms to run in real-time on industrial sites. Nevertheless, the key results could indicate that deployment schemes often appear to focus only on improving inference speed. Given that the evidence demonstrates that interpretability requirements remain unaddressed, the findings might suggest that availability of decision outputs appears absent from the overall design [15]. Although the above methods have made progress in their respective directions, a complete technical solution that can integrate multisource data fusion, interpretable state prediction, and adaptive maintenance decision-making has not yet been formed.

To bridge the gap between prediction and decision-making in existing research, this paper proposes a technical architecture that integrates morphology state prediction with maintenance scheduling. This architecture takes multi-source temporal data as input, extracts device degradation features through attention enhanced temporal networks, and generates interpretable remaining life predictions; The predicted results then drive deep reinforcement learning models to establish dynamic mapping relationships between device health status and maintenance tasks, achieving online optimization of maintenance strategies; The whole model is deployed at the edge computing node after lightweight processing, forming a complete loop from data acquisition, state analysis, strategy generation to task execution. In the validation of a precision machining workshop in a certain automotive parts manufacturing enterprise, the architecture achieved accurate fault prediction while successfully converting the predicted output into executable maintenance scheduling plans, achieving significant improvements in key operational performance such as equipment comprehensive efficiency and unplanned downtime, and verifying the feasibility of the data to performance conversion path.

II. DEEP LEARNING TECHNOLOGY SOLUTIONS FOR PERFORMANCE OPTIMIZATION

OVERALL TECHNICAL FRAMEWORK

To systematically solve the problem of transforming multi-source heterogeneous data into operational performance, this chapter designs a deep learning technology solution consisting of four core modules: data fusion and feature construction, state prediction, maintenance decision-making, and edge deployment. Each module is sequentially connected through standardized data interfaces to form a complete processing chain from raw sensor data to final maintenance instructions. The overall architecture is shown in Figure 1.

The architecture in Figure 1 focuses on the flow of data and divides the data processing process into four logical stages. The data fusion and feature construction module is located at the starting point of the architecture, responsible for aligning and converting the original time-series data of vibration, current, and temperature into a sample matrix with clear labels. Its design is based on the fact that the sampling frequency of the original data is different and contains noise, which must be uniformly preprocessed before it can be used

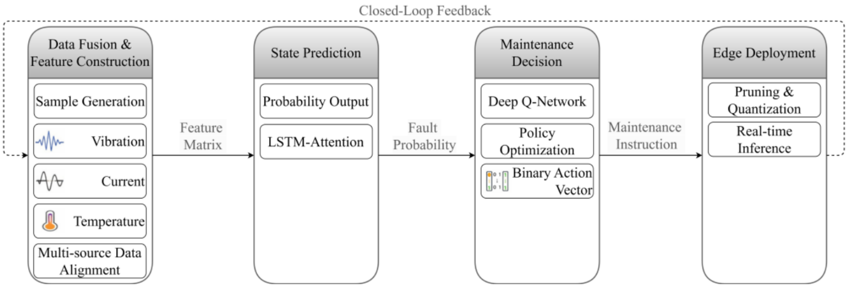

FIGURE 1. Overall architecture of deep learning technology solution for performance optimization

as a reliable input for subsequent models [16]. The state prediction module receives a feature matrix and uses an attention enhanced long short-term memory network to extract device degradation features and output the probability of failure within the next 24 hours. The introduction of attention mechanism enables the model to locate key degradation periods and provide interpretable basis for decision-making [17]. The maintenance decision-making module takes the failure probability of each device and the interval time from the last maintenance as state inputs, and uses deep reinforcement learning to solve the optimal maintenance strategy. The predicted information is converted into specific maintenance instructions, which solves the problem of the disconnection between prediction and decision-making. The edge deployment module compresses the trained decision model and deploys it to the workshop edge computing node to achieve real-time reasoning and send instructions to the manufacturing execution system. Its necessity lies in the strict requirements of the industrial site on response delay [18]. The four modules are connected in a series fashion with a solid line indicating the direction of travel. There is a dashed line connecting the edge deployment module to the data fusion module that represents the closed-loop optimization process of adding continuously retrieved data into the overall system during actual running conditions. The entire architecture connects the entire process from data perception to decision execution, providing a logical framework for the specific implementation of subsequent chapters.

MULTI-SOURCE DATA FUSION AND FEATURE CONSTRUCTION

30 CNC machine tools in the precision machining workshop are equipped with vibration, current, and temperature sensors. Among them, the vibration sensor collects signals in the X, Y, and Z directions with a sampling frequency of 10 kHz; The current sensor monitors the load changes of the spindle motor, with a sampling frequency of 1 Hz; the temperature sensor measures the temperature rise of the spindle bearing, with a sampling frequency of 0.1 Hz. The raw data is stored in comma separated value format in the data acquisition and monitoring control system [19]. Due to network fluctuations or sensor failures during data collection, missing values and abnormal spikes are inevitably present in the raw data [20]. For data segments with continuous missing duration less than 1 second, linear interpolation is used for filling. For outliers with continuous missing duration exceeding 1 second or values exceeding three times the range, they are directly removed and marked as empty. The cleaned data is stored in a time-series database with device ID and timestamp as the primary keys, laying the foundation for subsequent multi-source data fusion [21]. Due to significant differences in sampling frequency among the three types of sensors, joint analysis cannot be directly performed and must be aligned uniformly in the time dimension [22]. To preserve transient impulses and high-frequency signatures indicative of early mechanical wear, this article uses a dual-stream feature extraction approach. While the baseline target frequency for multi-source alignment is 1 Hz, the vibration signals are first processed at the original 10 kHz sampling rate to extract high-frequency statistical features (such as Kurtosis and Crest Factor) and spectral energy in specific bands. These high-dimensional features are then integrated into the 1 Hz synchronized feature matrix, ensuring that critical diagnostic information is not discarded during the downsampling process. The effective value calculation formula is:

$$V_{\mathrm{Ras}} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} v_i^2} \tag{1}$$

In the formula, v_i represents the i-th vibration sampling point per second, and N represents the total number of samples taken within that second. The original sampling frequency of the temperature signal is 0.1 Hz, which is lower than the target frequency. The cubic spline interpolation method is used to upsample to 1 Hz [23]. On the interval $[t_j, t_{j+1}]$, the interpolation function $S(t)$ is defined as a cubic polynomial:

$$S(t) = a_j + b_j(t - t_j) + c_j(t - t_j)^2 + d_j(t - t_j)^3 \tag{2}$$

At the original sampling time of t_j in the equation, the coefficients a_j, b_j, c_j, d_j are uniquely determined by the continuity of the function values, the continuity of the first derivative, the continuity of the second derivative, and the natural boundary conditions $S''(t_0) = 0$ and $S''(t_n) = 0$. After alignment, each record contains six feature dimensions: timestamp, root mean square (Rms) of three-axis vibration, spindle current value, and spindle temperature value.

Although the data has been aligned in the time dimension, there is still environmental noise interference in the original signal, which may mask the true trend of device degradation. Therefore, noise reduction processing is needed for key features. To eliminate environmental noise interference, wavelet thresholding is used to denoise the vibration effective value sequence. Daubechies 4 (db4) wavelet basis is selected for three-layer decomposition, and the soft threshold function is used to reconstruct the signal [24]. The threshold λ is determined by the following equation:

$$\lambda = \sigma\sqrt{2\log_e M} \tag{3}$$

In the formula, σ is the estimated noise standard deviation value, and M is the signal length. Through noise reduction processing, environmental interference can be effectively suppressed, highlighting the characteristic components related to equipment degradation. To verify the ability of feature construction process to characterize early degradation behavior of equipment, representative time series samples were selected from actual operating data after completing time alignment and wavelet denoising of multi-source time-series data. The evolution characteristics of vibration Rms in different operating stages were compared and analyzed, providing intuitive basis for subsequent sliding window sampling and label construction. On this basis, draw the curve of the effective value of vibration as a function of time, as shown in Figure 2.

From the typical sample shown in Figure 2, it can be observed that the vibration Rms in the early part of the time series fluctuates slightly around the stable baseline, indicating that the equipment is in a structurally complete and uniformly stressed operating state. At this time, the signal changes mainly come from measurement noise and small operating condition disturbances; After entering the subsequent stage, a transient impact with a significant increase in Rms gradually appears in the signal, accompanied by exponential decay characteristics. This phenomenon reflects the periodic impact load caused by local damage to mechanical components, and the impact interval shows a certain regularity, which is related to the natural motion cycle of the rotating components. At the same time, the rise in background noise level indicates an increase in the overall vibration energy of the system, reflecting the energy accumulation effect during the degradation process. The above characteristics indicate that the vibration signal already contains identifiable structural changes before the fault occurs, providing effective discrimination basis for fault prediction based on time series models.

After completing data alignment and denoising, continuous time-series data needs to be constructed as the sample sequence required for supervised learning. To balance historical context with sensitivity to recent environmental noise, the sliding window length is dynamically adjusted, with a primary focus on the most recent 24-hour high-resolution sequence within a broader 7-day trend analysis. Furthermore, instead of a simple binary output, the model provides a multi-level health index (0 to 1). This continuous risk score allows the subsequent reinforcement learning module to perform fine-grained maintenance scheduling, moving beyond 'all-or-nothing' decisions to support more nuanced, real-time maintenance prioritizing. The sliding step size between adjacent windows is 1 hour to ensure overlap between samples and expand the data volume [25]. The multivariate time-series data within each window form a feature matrix $X \in R^{T\times F}$ for a sample, where $T = 6048$ is the time step and $F = 6$ is the feature dimension. The sample label is generated based on maintenance records, and the types of faults recorded in the repair work order mainly include spindle bearing wear and tool fracture. Due to the fact that the prediction target of this article is whether the device experiences a malfunction that causes shutdown within the next 24 hours, rather than distinguishing specific failure modes, the two types of failures are uniformly labeled as fault samples, and the remaining windows are labeled as healthy samples. After aligning, denoising, and sliding window sampling, the raw vibration, current, and temperature data are converted into a sample set with clear labels for subsequent prediction model training.

TIME SERIES PREDICTION MODEL BASED ON ATTENTION MECHANISM

The sample matrix X consists of 6048 time steps of vibration, current and temperature characteristics. Each time step represents one second state of device operation. In order to model the degradation characteristics of devices using long-term data and further predict the health status of devices in the next 24 hours, this section constructs a long short-term memory network with attention mechanism. This network first utilizes the gate units of LSTM to extract temporal dependent features, and then locates key degradation periods through attention mechanisms, thereby improving the interpretability of the model while modeling long-range dependencies [26,27].

The LSTM network effectively alleviates the gradient vanishing problem of traditional recurrent neural networks when dealing with long-term dependencies by controlling the information transmission path through forget gates, input gates, and output gates [28]. At time step t, the input feature vector $x_t \in R^F$ interacts with the hidden state h_{t-1} and cell state c_{t-1} from the previous time step, and the update process is controlled by the following equation system:

$$\begin{cases} f_t = \sigma(W_f[h_{t-1}, x_t] + b_f), \\ i_t = \sigma(W_i[h_{t-1}, x_t] + b_i), \\ \tilde{c}_t = \tanh(W_c[h_{t-1}, x_t] + b_c), \\ c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t, \\ o_t = \sigma(W_o[h_{t-1}, x_t] + b_o), \\ h_t = o_t \odot \tanh(c_t) \end{cases} \tag{4}$$

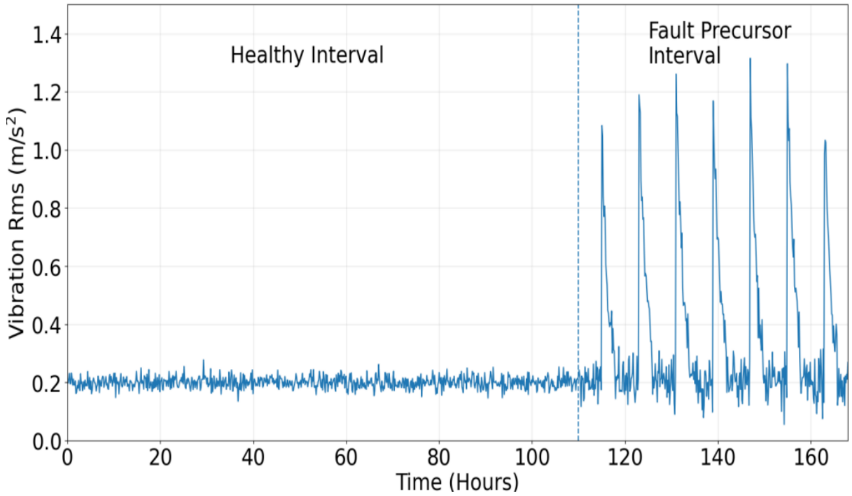

FIGURE 2. Time sequence evolution characteristics of equipment vibration Rms in healthy state and fault precursor stage

In the formula, f_t, i_t and o_t are the activation values of the forget gate, input gate, and output gate, respectively. $\tilde{c}_t$ is the candidate cell state, c_t is the updated cell state, and h_t is the hidden state output of the current time step. σ is the Sigmoid function, tanh is the hyperbolic tangent function, $\odot$ represents element-wise multiplication, and W and b are trainable parameters. The LSTM layer sets 128 hidden units, processes 6048 steps along the time dimension, and outputs the hidden state sequence $H = [h_1, h_2, \ldots, h_T]$ for all time steps.

In the process of equipment degradation, specific periods before the occurrence of faults often contain key symptoms, while the information redundancy in other periods is relatively high [29]. If the hidden state

sequence output by LSTM is directly aggregated with equal weights, it weakens the influence of critical periods, making it difficult for the model to accurately capture degradation inflection points. To this end, an attention mechanism is introduced after the LSTM layer to weight and aggregate the hidden state sequence, enabling the model to automatically focus on time periods strongly related to faults. The attention mechanism learns the importance weights of each time step through trainable parameters, and the weight calculation method is as follows:

$$\alpha_t = \frac{\exp\left(v^\top \tanh(W_{\text{att}} h_t + b_{\text{att}})\right)}{\sum_{j=1}^{T} \exp\left(v^\top \tanh(W_{\text{att}} h_j + b_{\text{att}})\right)} \tag{5}$$

In the formula, W_{att} and b_{att} are the attention parameter matrix and bias, v is the trainable weight vector, and α_t is the normalized attention weight at the t-th time step. This weight distribution directly reflects the

importance of each time step to the final prediction. After obtaining the weights, the hidden states of each time step are weighted and summed to obtain the context vector:

$$c_{\text{att}} = \sum_{t=1}^{T} \alpha_t h_t \tag{6}$$

This vector aggregates the temporal information of the entire time period, and due to the introduction of attention weights, the model can clearly identify which time periods contribute the most to fault prediction. To enhance industrial interpretability, these high-weight segments are cross-referenced with physical degradation signals, such as periodic transient impacts in vibration data, which directly correlate with specific physical failure modes like spindle bearing wear or tool fracture. To visually demonstrate the localization effect of attention mechanism on key time periods, taking typical spindle bearing wear samples as an example, the distribution of normalized attention weights between the time dimension and multi-source sensor features is visualized, as shown in Figure 3.

From Figure 3, it can be observed that in the time interval far from the occurrence of the fault, the attention weights of various sensors are generally maintained at a low level, reflecting that when the equipment is in a stable operation stage, the contribution of temporal features to the prediction results is relatively uniform and the information density is limited; As time gradually approaches the fault moment, the weight of the vibration signal in three directions shows a continuous upward trend. This change is due to the accumulation of vibration energy caused by the degradation of the mechanical structure, which makes the relevant features dominant in the model discrimination process. At the same time, the current signal shows concentrated enhancement within a specific time window, indicating a coupling relationship between load fluctuations and local damage, and is recognized by the model as a key discrimination basis. The gradient of temperature heature is relatively flat, which means the sensitivity of short-term fault prediction is relatively low compared with vibration current feature. The trend of weight spread from dispersion to concentration, which means that model gradually concentrates on few periods with high information and some key sensor channels in the later stage of degradation and then gets fault precursors effectively.

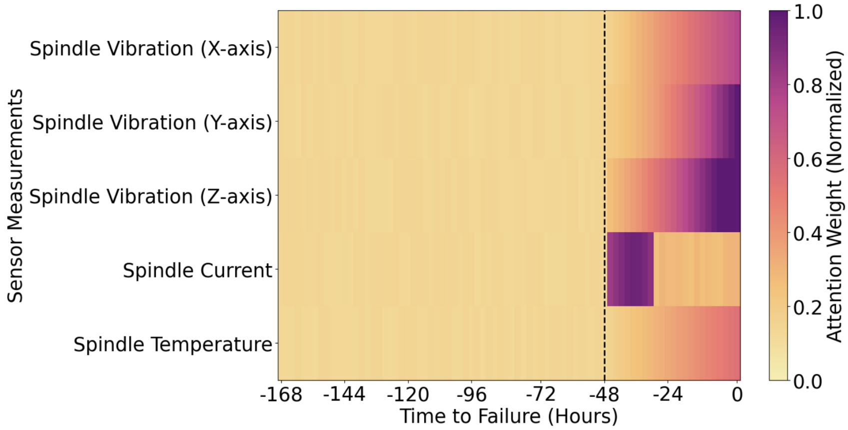

FIGURE 3. Time series distribution of attention weights for multi-source sensors on the main axis

After completing the modeling of long temporal features and attention weighted aggregation, the model inputs the context vector c_{att} into the fully connected layer, and outputs the probability of fault occurrence through Sigmoid activation [30]:

$$\hat{y} = \sigma(W_p c_{\text{att}} + b_p) \tag{7}$$

In the formula, W_p is the output layer weight matrix, b_p is the bias, and $\hat{y} \in [0, 1]$ represents the predicted

probability of faults occurring within the next 24 hours. The model training adopts a binary cross entropy loss function, and the optimizer selects Adaptive Moment Estimation (Adam) with an initial learning rate of 0.001. During the training process, the validation set loss is used as an indicator for early stopping to prevent overfitting. After training convergence, the model can receive real-time collected multi-source time-series data, output probability estimates of device health status, and provide input for subsequent maintenance decision-making modules.

MAINTENANCE DECISION-MAKING AND LIGHTWEIGHT DEPLOYMENT BASED ON DEEP REINFORCEMENT LEARNING

The failure probability $\hat{y}_k$ output by the prediction model for each device is a static risk indicator, but a single probability value cannot directly drive specific maintenance actions. To transform predictive information into executable maintenance instructions, this paper constructs the multi-device maintenance scheduling problem as a sequential decision problem and uses deep reinforcement learning to solve the optimal strategy [31,32]. Specifically, the problem can be modeled as a Markov Decision Process (MDP), with a scheduling environment consisting of K CNC machine tools. The state at time t is characterized by the predicted probabilities $\hat{y}_{t,k}$ of each device and the interval time $d_{t,k}$ from the last maintenance. The two are concatenated to form a state vector $s_t = [\hat{y}_{t,1}, d_{t,1}, \ldots, \hat{y}_{t,K}, d_{t,K}]$, with a dimension of $2K$. The precision machining workshop involved in this study has a total of $K = 30$ CNC machine tools, so the dimension of the state vector is 60.

The action space is defined as a binary vector, where $a_t \in \{0, 1\}^K$ represents the specific action performed at time t, and the component $a_{t,k} = 1$ represents the maintenance performed on the k-th device. To avoid resource conflicts caused by simultaneous maintenance of multiple devices, the constraint $\sum_{k=1}^{K} a_{t,k} \leq 1$ must be met. For the convenience of subsequent expression, this article uses the symbol a to refer to any action in the action space. After executing the action, the environment transitions to the next state s_{t+1} based on the actual operating status of the device, and provides an immediate reward r_t as feedback. The design of this reward function aims to balance maintenance costs and fault losses:

$$r_t = -\sum_{k=1}^{K} (c_m a_{t,k} + c_f I_{\text{fail},t,k} + c_e E_{t,k} + c_y(1 - Y_{t,k})) \tag{8}$$

In the formula, c_m represents the cost of a single maintenance, c_f represents the downtime loss caused by equipment failure, $I_{\text{fail},t,k}$ is indicator function determined from historical maintenance work order data, $c_e E_{t,k}$ accounts for energy consumption fluctuations and $c_y(1 - Y_{t,k})$ represents yield losses related to production rhythm. This ensures the agent considers operational constraints like production rhythm and energy efficiency beyond simple binary maintenance decisions. When the kth equipment actually fails between time t and $t+1$, take 1; otherwise, take 0. Although the fault probability in the state vector is output by the prediction model, the reward signal is constructed based on real fault records to ensure the alignment of the reinforcement learning process with the actual operating results. In addition, reinforcement learning has a certain robustness to prediction errors through long-term cumulative reward optimization mechanisms, enabling decision-making strategies to gradually approach the optimal in dynamic environments. Under this reward structure, fault losses have a higher weight relative to maintenance costs, making delayed maintenance inferior to early intervention in expected returns, thereby guiding the agent to adopt preventive maintenance strategies during the risk accumulation stage to achieve optimal long-term cumulative returns [33]. Through the above modeling, the maintenance scheduling problem is transformed into a solvable reinforcement learning problem. In terms of model solving, this article uses a deep Q-network to solve the MDP process mentioned above. The network structure is a three-layer fully connected neural network, with the input layer receiving the state vector s_t and the output layer corresponding to the Q values of all optional actions. The training process is based on experience replay and target network mechanism, randomly sampling quadruple (s_t, a_t, r_t, s_{t+1}) from experience pool D, and updating network parameters by minimizing Bellman error:

$$L = \mathbb{E}_{(s_t,a_t,r_t,s_{t+1})\sim D}\Big[\big(r_t + \gamma \max_{a'} Q_{\text{target}}(s_{t+1}, a') - Q(s_t, a_t)\big)^2\Big] \tag{9}$$

In the formula, γ is the discount factor, a' represents the candidate action under state s_{t+1} at the next moment, Q_{target} is the target network parameter, which is periodically replicated from the main network for stable training. During the training process, the intelligent agent continuously interacts with the environment to generate trajectories, gradually converging to an approximate optimal strategy. In actual decision-making, for the current state s_t, iterate through all possible actions a, select the action with the highest Q value as $a_t^* = \arg\max_a Q(s_t, a)$, and issue the maintenance instruction corresponding to that action to the manufacturing execution system. To characterize the dynamic evolution characteristics of the policy performance of reinforcement learning models during the training process, and to analyze the changing trends of policy return levels at different stages from a statistical distribution perspective, the training process was divided into multiple typical stages and visualized. The results are shown in Figure 4.

Figure 4 shows the cumulative return distribution changes of the reinforcement learning model at different training stages. In the initial stage, the distribution is concentrated in a lower range with small fluctuations, reflecting that the strategy has not yet formed effective decision-making ability and mainly relies on random exploration, resulting in an overall low return level; As the training progresses, the distribution gradually moves towards the high return range and the degree of dispersion increases significantly. This change is due to the agent's continuous correction of value estimates in environmental interactions, allowing some strategy paths to begin to obtain better returns, while strategy instability still exists; After entering the later stage, the distribution center further shifts upward and the shape tends to converge, indicating that high-value strategies gradually dominate and the impact of randomness significantly weakens; In the stable stage, the distribution range significantly shrinks and the mean remains at a high level, indicating that the strategy update has reached equilibrium. The model forms a stable decision-making mode under the goal of maximizing returns. Therefore, this result indicates that the constructed reinforcement learning method can achieve an effective convergence process from random exploration to stable optimization. After obtaining the deep Q network after training, it is necessary to deploy it to the workshop edge computing node to support real-time operation [34]. Considering that edge devices use NVIDIA Jetson Nano, which has limited computing resources, it is necessary to compress and optimize the model to meet the requirements of low latency and low power consumption in industrial sites. To this end, an amplitude based pruning method is first adopted to remove weights with absolute values less than the threshold θ in the fully connected layer. After pruning, the network is fine tuned to restore accuracy. Then, weight quantization techniques are applied to further reduce model storage and computational overhead. The compressed model can run stably on Jetson Nano, receiving the updated state s_t of the prediction module every 10 seconds, outputting maintenance decisions, and writing them into the manufacturing execution system through the Open Platform Communications Unified Architecture (OPC UA) protocol, triggering the generation of maintenance work orders, thus achieving a complete closed-loop from state perception, prediction analysis to decision execution [35,36]. Experimental results show that after pruning and quantization, the inference latency was reduced from 85ms to 12ms per sample, and the memory footprint decreased by 74% (from 120MB to 31MB), validating the necessity of these lightweight techniques for real-time edge execution.

MECHANISM ANALYSIS OF TECHNICAL SOLUTIONS SUPPORTING DIGITAL TRANSFORMATION

The technical solution proposed in this article forms a complete closed loop from data fusion, state prediction, maintenance decision-making to edge deployment, and its mechanism supporting the digital transformation of manufacturing enterprises is reflected in the following four levels.

Firstly, this article constructs a mechanism for data assetization and feature engineering, which opens up a channel for the transformation of multi-source heterogeneous data into structured information. Manufacturing enterprises generally face the dilemma of complete data collection but insufficient value mining, which is rooted in the failure to form a unified feature expression for heterogeneous data such as vibration, current, and temperature. The previously introduced time alignment, denoising and sliding window sampling methods convert raw time-series data into sample matrix with semantic labels, i.e., the data from "non" to "learnable". This conversion process is actually the process of converting industrial data assets into structured information assets that can be used by deep learning models, which paves the way for subsequent intelligent analysis. This mechanism solves the primary challenge of extracting effective features from multi-source heterogeneous data. Secondly, this article establishes an interpretable state prediction mechanism to bridge the cognitive gap between model output and engineer

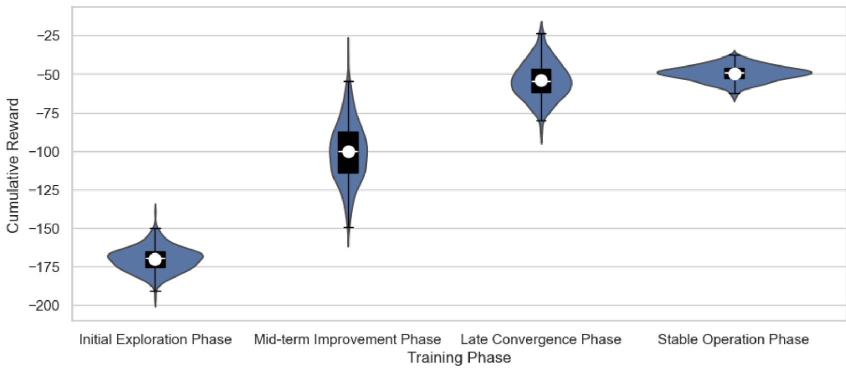

FIGURE 4. Evolution of cumulative reward distribution during reinforcement learning training phase

trust. The "black box" nature of deep learning models is the main obstacle to their implementation in industrial scenarios. In the state prediction stage, this article introduces an attention mechanism to allocate weights to key degradation periods, so that the model not only outputs the probability of failure, but also outputs the contribution of each time step to the prediction results. As shown by the high consistency between attention weights and the evolution trend of vibration signals, on-site engineers can intuitively understand the model judgment basis through a visual interface, thereby enhancing their willingness to adopt the system output. This mechanism directly addresses the key challenge of insufficient interpretability of the model.

Thirdly, a predictive driven closed-loop decision-making mechanism can be formed to achieve dynamic coupling between state perception and maintenance execution. Traditional predictive maintenance systems are limited to fault warning, and there is a gap between predictive information and maintenance actions. This article introduces deep reinforcement learning into maintenance scheduling, encoding the fault probability output by the prediction model and the equipment maintenance history into a state space, and guiding the agent to seek the optimal balance between maintenance costs and fault losses through a reward function. The non-linear shrinkage characteristics of decision boundaries with increasing fault probability indicate that the system can adaptively adjust maintenance thresholds based on real-time risk levels, allowing predicted information to directly drive the issuance of specific maintenance instructions. This mechanism solves the core problem of the disconnection between prediction and decision-making processes.

Fourthly, a lightweight deployment mechanism can be established at the edge to ensure real-time response and stable system operation. Industrial sites have strict requirements for decision latency, and cloud based inference models are difficult to meet real-time requirements. The decision model is compressed to the edge computing node through pruning and quantification technology, so that the complete decision closed-loop can meet the real-time response requirements of the industrial site, and ensure the stability of the system under the condition of long-term continuous operation. The effectiveness of this mechanism can be verified in subsequent experiments, ensuring the engineering feasibility of the technical solution in actual production environments and enabling the continuous conversion of digital transformation technology investment into operational performance. This technical solution achieves information assetization through a data fusion module, improves interpretability through an attention prediction module, bridges the gap between prediction and execution through a reinforcement learning decision module, and ensures the feasibility of project implementation through a lightweight deployment module. The four modules are sequentially connected to form a complete value chain that transforms data assets into operational performance, connecting the entire process of data collection, status analysis, strategy generation, and task execution.

III. EXPERIMENTAL SETUP AND DATA PREPARATION

To verify the feasibility and effectiveness of the technical solution proposed in this article, this chapter constructs an experimental environment based on actual production data from a precision machining workshop of a certain automotive parts manufacturing enterprise, and forms a standardized experimental dataset according to the aforementioned data fusion and feature construction methods. The experimental configuration and parameter settings follow the principles of model design, providing a reliable experimental basis for subsequent result analysis.

DATASET DESCRIPTION

The data for this study was collected from the precision machining workshop of a certain automotive parts manufacturing enterprise. This workshop contains a total of 30 CNC machine tools, all of which are mainstream models that have been put into use in recent years, and have a complete data acquisition and monitoring control system. The data collection period is 9 consecutive months, with the benchmark period data from October to December 2023 before system deployment, and the data from January to June 2024 used to build the model dataset, which is further divided into training set, validation set, and testing set. This workshop is in a typical stage of digital transformation in manufacturing enterprises. Before deploying this technology solution, real-time collection and storage of equipment status data have been achieved. However, there are still significant shortcomings in data value mining: predictive maintenance has not yet been carried out, equipment maintenance is still mainly based on post maintenance and regular maintenance, and unplanned downtime events have caused frequent impacts on production plans. The pain points of workshop management are mainly reflected in three aspects: unpredictable equipment health status, lack of basis for maintenance resource allocation, and delayed abnormal response to production needs. These pain points are the common bottlenecks for manufacturing enterprises in the process of digital transformation from data collection to intelligent decision-making. The application of this technical solution in this scenario aims to transform existing data assets into executable maintenance decisions. The implementation process involves data integration with the existing manufacturing execution system in the workshop, adaptation and adjustment of maintenance work order processes, and other aspects. The above characteristics of application scenarios provide a real and representative industrial environment for the verification of technical solutions, and also provide clear benchmark references for subsequent performance comparisons.

Each machine tool is equipped with three types of sensors: vibration, current, and temperature. Among them, the vibration sensor collects acceleration signals in the X, Y, and Z directions of the main shaft, with a sampling frequency of 10 kHz; The current sensor monitors the load current of the spindle motor, with a sampling frequency of 1 Hz; the temperature sensor measures the temperature of the spindle bearing, with a sampling frequency of 0.1 Hz. The raw data is in comma separated value format and amounts in total to around 4.2 TB. In the time period covered, there were a total of 146 equipment failure shutdown events in the workshop, and the maintenance work order described when and where (equipment number) and what fault mode occurred. This article aims to explore whether any faults that could cause downtime will occur in the equipment within the next 24 hours. However, this article does not concern itself with specific fault modes, so all types of faults are uniformly labeled as fault samples.

After preprocessing such as cleaning, time alignment, and noise reduction, the raw data is used to construct a sample set using the sliding window method. The window length is set to 7 days (corresponding to 6048 time steps), and the sliding step size is 1 hour. The multivariate time-series data within each window forms a sample feature matrix, which includes six feature dimensions: timestamp, effective values of three-axis vibration, spindle current value, and spindle temperature value. The sample label is generated based on the maintenance work order record: if a fault occurs within 24 hours after the end of the window that causes a shutdown, the window can be marked as a faulty sample; otherwise, it can be marked as a healthy sample. The model training and validation only use data from January to June 2024, and data from October to December 2023 as the benchmark period for performance comparison, without participating in model training. To ensure long-term robustness and generalizability across full equipment degradation cycles, a rolling update mechanism is implemented where the model is periodically re-trained with new operational data every quarter, effectively extending the historical coverage and capturing long-term wear patterns that exceed a six-month window. After the above processing, a total of 17280 valid samples were generated, including 14976 healthy samples and 2304 faulty samples. The dataset is divided into training set, validation set, and testing set in chronological order, with a ratio of 7:1:2. That is, the training set contains 12096 samples, the validation set contains 1728 samples, and the testing set contains 3456 samples. The test set is entirely taken from the last month of the dataset (June 2024) to evaluate the model's predictive ability for future unknown data.

EXPERIMENTAL ENVIRONMENT AND PARAMETER SETTINGS

In order to make the experiments reproducible and validate the possibility of running the model in an industrial edge environment, this paper constructed a unified hardware/software experimental environment. The experiments of model training and comparison were conducted on the server, and the experiments of model deployment and real-time performance testing were conducted on the edge computing nodes. The server is equipped with Intel Xeon Gold 6248R Central Processing Unit (CPU) (2.7 GHz, 24 cores), NVIDIA Tesla V100 GPU (32 GB VRAM), and 128 GB DDR4 memory; while the edge nodes are equipped with NVIDIA Jetson Nano, which has 128-core Maxwell Graphics Processing Unit (GPU) (4 GB shared VRAM, Quad-core ARM Cortex-A57 CPU, and 4 GB memory). In terms of the software environment, the server uses Ubuntu 20.04 LTS operating system, and the edge nodes use Ubuntu 18.04 LTS. The deep learning framework uses PyTorch 1.12.0 (edge nodes use aarch64 version), the programming language uses Python 3.8, data processing and analysis depend on NumPy 1.21.0, Pandas 1.3.0 and Scikit-learn 1.0.0, and visualization uses Matplotlib 3.4.0 and Seaborn 0.11.0.

Regarding model training hyperparameter settings, the fault prediction model uses the Adam optimizer, with an initial learning rate of 0.001, a batch size of 64, 100 iterations, and an early stopping mechanism (training stops if the validation set loss does not decrease within 20 iterations). The loss function is the binary crossentropy loss function. The Long Short-Term Memory (LSTM) network has 128 hidden units, and the attention mechanism parameter dimension is set to 64. The maintenance decision model also uses the Adam optimizer, with a learning rate of 0.0005, a batch size of 32, an experience replay pool size of 50,000, a target network update frequency of once every 200 steps, and a discount factor of 0.95. Its network structure is a three-layer fully connected neural network, with the input layer dimension corresponding to the state vector, the hidden layer units being 128 and 64 respectively, and the output layer corresponding to the number of selectable actions. In the reward function, the cost of a single maintenance operation is set to 1, and the equipment failure loss is set to 5.

To evaluate the performance of the fault prediction model proposed in this article, four baseline models were selected for comparative experiments, including a standard long short-term memory network (2 layers, 128 hidden units per layer), a convolutional long short-term memory network (2 convolutional layers followed by 2 LSTM layers), a Transformer (4 encoder layers, 8 multi-head attention heads, 256 feedforward network dimensions), and a temporal convolutional network (8 dilated convolutions with dilation factors of 1, 2, 4, 8, 16, 32, 64, 128). All baseline models use the same dataset partitioning and training strategy to ensure fairness in comparison results.

In response to the deployment requirements at the industrial edge, lightweight compression was applied to the trained deep Q network. Firstly, amplitude based weight pruning is adopted, with a pruning threshold set at 0.02, which removes weights with absolute values less than 0.02. After pruning, 5 rounds of fine-tuning are performed to restore model accuracy; Subsequently, 8-bit weight quantization was performed on the pruned model to further reduce its size. The compressed model is deployed on the edge computing node of the Jetson Nano. The decision reasoning cycle is set to once every 10 seconds, consistent with the update frequency of the prediction module. The decision results are written into the manufacturing execution system through the OPC UA protocol, triggering the generation of a repair order.

PERFORMANCE INDICATOR COLLECTION AND STATISTICAL CALIBER

To quantify the performance optimization effect before and after system deployment, this study synchronously collected four key operational indicators: OEE, unplanned downtime, monthly average maintenance cost, and monthly average failure loss. The comprehensive efficiency of the equipment is calculated according to international standards, which is obtained by multiplying the time utilization rate, performance utilization rate, and qualified product rate. The data is sourced from the production records and quality inspection system of the workshop manufacturing execution system. The statistics of unplanned downtime caused by equipment failures are sourced from the equipment status monitoring system and maintenance work order system. The monthly average maintenance cost includes direct expenses such as spare parts replacement, manual repair, and maintenance consumables. The monthly average failure loss is calculated as the production capacity loss caused by equipment failure resulting in production stagnation. The above cost and loss data are sourced from the actual settlement records of the enterprise's financial system and production management system. All performance indicators are calculated on a monthly basis. The benchmark period before system deployment is based on historical data from October to December 2023 for three consecutive months. The evaluation period after system deployment is based on actual data from April to June 2024 for three consecutive months after stable system operation. The two sets of data are consistent in statistical caliber to ensure the effectiveness of performance comparison.

IV. RESULT ANALYSIS

FAULT PREDICTION PERFORMANCE EVALUATION

To further analyze the effectiveness of the proposed deep learning technology in the fault prediction task, this study chose standard long short-term memory networks, convolutional long short-term memory networks, Transformers, and time-series convolutional networks as the baseline model for comparison experiments on the same data set. The four indicators of accuracy, precision, recall, and F1 score were used to quantitatively evaluate the performance of each model. The experimental results are shown in Figure 5.

From the comparison results presented in Figure 5, it can be seen that each model shows a higher level of accuracy than the other three indicators, reflecting the overall distribution characteristics of healthy samples dominating the dataset. The model in this article achieved the best values in all four indicators, with an accuracy of 96.8%, which is 2.3 percentage points higher than the suboptimal Transformer model. This difference is mainly attributed to the effective focus of the attention mechanism on the critical period of equipment degradation, which enables the model to capture fault precursor features even in noisy environments. In terms of recall rate, proposed model achieved 92.0%, slightly higher than the Transformer model with the highest recall rate in the baseline model (91.8%) by 0.2 percentage points, indicating that the recall performance was not sacrificed for the pursuit of accuracy in fault sample recognition ability. The error bar data shows that the standard deviation of proposed model is between 0.5% and 0.7%, significantly smaller than other baseline models, indicating that the introduction of attention mechanism enhances the robustness of the model to different random initialization and data sampling, thereby maintaining more stable predictive performance in multiple repeated experiments.

EFFECTIVENESS OF ATTENTION MECHANISMS

After obtaining the model prediction results, in order to further characterize the evolution characteristics of attention distribution in the time dimension and verify its correlation with the equipment degradation process, a visual analysis was conducted on the continuous time interval of typical fault samples before the fault occurred. By aligning the statistical results of attention weights with the normalized effective values of vibration signals, the trend of the model's attention intensity over time and its focusing behavior in key time intervals are reflected, providing a basis for subsequent

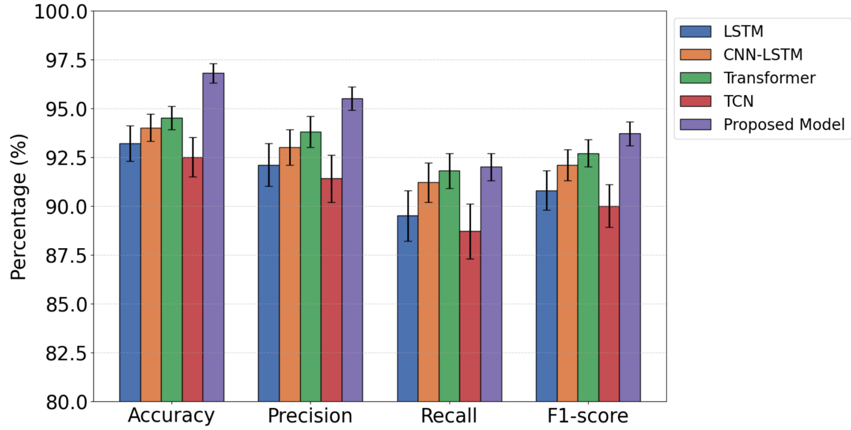

FIGURE 5. Comparison of fault prediction performance among different models

analysis of the effectiveness of attention mechanisms, as shown in Figure 6.

As shown in Figure 6 (1), the attention weight fluctuates little in the same time interval far away from the fault moment, and the curve changes relatively smoothly. This is because the operation state of the equipment is stable, and there is little discrimination for the temporal features, so the model maintains an approximately equally strong attention intensity on every time step. With the time gradually approaching to the fault point, the weight curve gradually increases and breaks 0.50 around -18h, and then enters into rapid growth stage; When approaching to 0h before the fault, the weight can reach about 0.75; This phenomenon shows that the model's sensitivity to the accumulation of poor features is gradually enhanced. And the standard deviation interval range expands from about $\pm$ 0.05 around the initial stage to about $\pm$ 0.15 around the later stage. At this time, the degradation path of different samples show certain differences, so the dispersion of attention distribution can increase. As shown in Figure 6 (2), the vibration Rms increases slowly from a certain level close to 0 on the same time interval, and then rapidly increases to its maximum when approaching to the fault. The vibration Rms and the change of the attention weight have the same trend of change with time. This phenomenon comes from the accumulation of vibration energy caused by the damage of the mechanical structure, which gradually increases the density of discriminative information contained in the signal, and then promotes the model to aggregate more weights on the critical interval around the fault. Within the critical window of -18h to 0h indicated by the gray area, the vibration Rms rapidly increased from about 0.4 to its peak, indicating that the model has effectively located key information and strengthened its contribution to the prediction results at this stage. This indicates that the constructed attention mechanism can accurately capture the key evolutionary stages of device degradation in the time dimension. The above analysis indicates that the model can effectively focus attention on the key degradation interval 18 hours before the fault within the 24-hour prediction window, indicating its early warning capability, and the attention mechanism is not dispersed to redundant periods due to window length.

REINFORCEMENT LEARNING DECISION STRATEGIES

To characterize the distribution characteristics of maintenance decisions in continuous operation and analyze the coupling relationship between equipment risk level and maintenance interval, the probability of equipment failure and the distance from the last maintenance time are constructed as a two-dimensional state space. The maintenance response behavior under different operating states is presented through the visualization of sample point classification results and decision boundaries, as shown in Figure 7.

The horizontal axis in Figure 7 reflects the change in fault probability from 0 to 1, and the vertical axis describes the distribution of maintenance intervals within the range of 0 to 60. The sample as a whole shows a band like distribution along the upper left to lower right direction, indicating that as the fault probability increases from about 0.2 to 0.8, the maintenance interval gradually shrinks from about 50 days to nearly 0 days. This phenomenon also represents the compression effect of the increase of risk level on time to maintenance. It can be seen that when $x \approx 0.3$, the corresponding critical maintenance interval for the decision boundary x=0.2 is approximately 33 days; when $x \approx 0.6$, the corresponding critical maintenance interval is approximately 15 days; and when $x \to 0.8$, the corresponding critical maintenance interval tends to 0 days. The nonlinear concave shape of the boundary curve shows that when the risk continues to increase, the conditions for maintenance do not change linearly, but have an accelerating tightening characteristic. Most of the red samples are above the boundary. In the $x \in [0.4,0.7]$ interval and $y \in [20,40]$ range, there is a relatively concentrated area, indicating that the system can make more maintenance decisions when the risk is medium to high and the maintenance interval is longer. Most of the blue samples are below the boundary and form a relatively concentrated area in the $x \in [0.3,0.7]$ range and $y<20$, indicating that when there is a certain degree of risk,

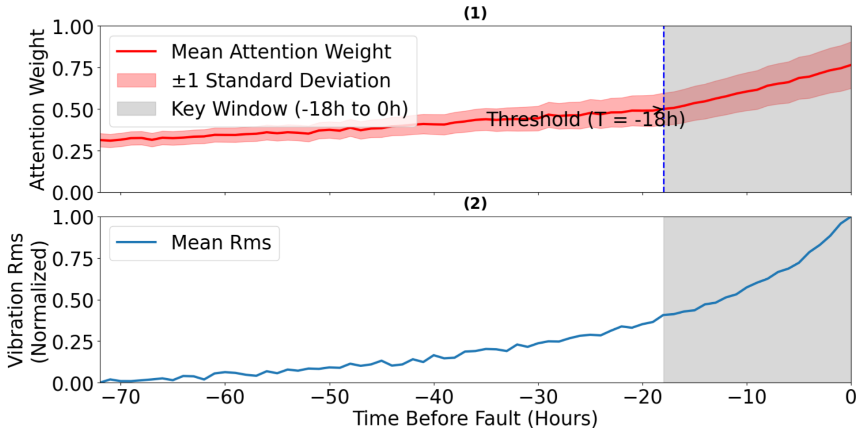

FIGURE 6. Attention weight and vibration response time evolution before fault: (1) Changes in attention weight trends; (2) Trend change of vibration response

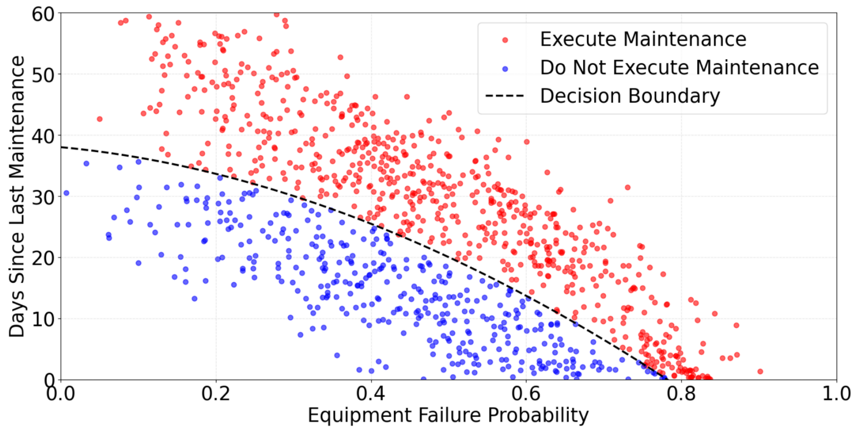

FIGURE 7. Distribution of maintenance decision results driven by equipment failure probability and maintenance interval

in order to maintain operation with a shorter maintenance interval, the system still generates maintenance decisions.

REAL-TIME VERIFICATION OF MODEL DEPLOYMENT

To evaluate the reliability and real-time performance of the lightweight deployment solution in an industrial edge environment, the following experiment was conducted. The entire loop of the deployment from status acquisition to maintenance command output was completed every 10 seconds for 120 hours. In Table 1, the real-time verification results of the edge deployment are shown based on the measured data at the time of 1-hour operation, 24 hours, 48 hours, 72 hours, and 120 hours.

TABLE 1. Real-time performance verification results for edge deployment

Runtime (h)	Average closed-loop delay (s)	Maximum closed-loop delay (s)	Delayed standard deviation (s)	Model memory usage (MB)	Peak CPU utilization (%)
1	1.58	1.72	0.09	312	44
24	1.61	1.79	0.11	312	46
48	1.67	1.88	0.15	314	51
72	1.63	1.82	0.12	312	48
120	1.69	1.93	0.16	315	52
Mean ± Standard Deviation	1.64 ± 0.04	1.83 ± 0.08	0.13 ± 0.03	313 ± 1.4	48.2 ± 3.3

In Table 1, the average closed-loop delay fluctuates within the range of 1.58s to 1.69s and remains below 1.7s overall, indicating that the compressed decision model can meet the basic requirements of real-time response in industrial sites on edge devices. The average latency increased to 1.67s and the standard deviation expanded to 0.15s in the 48th hour, which corresponds to the peak CPU usage rate of 51% during the same period. This reflects a brief performance fluctuation of the system after two consecutive days of operation due to cache accumulation and competition for background resources; The average delay dropped to 1.63s by the 72nd hour, indicating that the resource management mechanism of edge devices can effectively release accumulated load and restore stable operation of the system. The memory usage of the model remained between 312MB and 315MB during the testing period, and did not show a continuous growth trend, proving that there is no memory leakage problem in the deployment plan. According to the statistic information of all testing cycles, the average closed-loop delay is 1.64s and the standard deviation is only 0.04s. The maximum average closed-loop delay is 1.83s and the maximum delay of single value in each testing node is 1.93s. Under the condition of long-term continuous operation, the complete decision closed-loop delay can be controlled within 2s with the lightweight deployment scheme, and the result meets the requirement of real-time response of the application in the industrial site.

QUANTIFICATION OF PERFORMANCE OPTIMIZATION EFFECTS

To quantify the actual impact of the proposed technical solution on the operational performance of manufacturing enterprises, this study selected four core indicators: OEE, unplanned downtime, monthly average maintenance cost, and monthly average failure loss. The measured data for three consecutive months before and after system deployment were compared with the historical benchmark period statistical values. The relevant statistical results are summarized in Table 2.

TABLE 2. Comparison of key operational performance indicators before and after system deployment

Index	Before deployment (baseline period)	Post-deployment (evaluation period)	Change
OEE(%)	71.2	83.0	+11.8
Unplanned downtime (hours/month)	112	44	-68
Average monthly maintenance cost (ten thousand yuan)	18.6	12.3	-6.3
Average monthly failure loss (ten thousand yuan)	24.5	9.8	-14.7

Table 2 shows that after system deployment, the comprehensive efficiency of equipment increased from 71.2% to 83.0%, and the unplanned downtime decreased from 112 hours per month to 44 hours, a decrease of 60.7%. The monthly average maintenance cost and failure loss decreased by 63000 yuan and 147000 yuan, respectively, with a decrease of 33.9% and 60.0%. The improvement of overall equipment efficiency comes from the effective positioning of degradation periods by the prediction module, which makes maintenance timing more accurate and avoids the dual losses of excessive maintenance and sudden failures; The significant reduction in downtime is directly attributed to the reinforcement learning decision module transforming predictive information into preventive actions, changing the long-term downtime state in passive response mode; The dual decrease in maintenance costs and fault losses reflects the balance mechanism between maintenance costs and fault losses in the reward function, which guides the agent to choose the economically optimal maintenance strategy in actual operation. The synchronous optimization of the above indicators in the three dimensions of efficiency, time, and cost verifies that the technical solution proposed in this article has connected the complete conversion path from data assets to operational performance.

COMPREHENSIVE ANALYSIS OF THE EFFECTIVENESS OF DIGITAL TRANSFORMATION

This article analyzes the comprehensive effectiveness of deep learning technology solutions in supporting the digital transformation of manufacturing enterprises from three dimensions: digital capability building, digital operation transformation, and digital engineering implementation.

At the level of digital capability construction, this article adopts multi-source data fusion and feature construction technology to aggregate three kinds of heterogeneous signals of vibration, current and temperature into a sample matrix with clear semantics, completing the assetization conversion of industrial data from “storable” to “learnable”. The attention-enhanced temporal prediction model achieves better performance than the baseline models of LSTM, Convolutional Neural Network (CNN), Transformer and Temporal Convolutional Network (TCN) in the evaluation metrics of accuracy, precision, recall and F1-score, respectively. In addition, the evolution trend of attention weight is basically consistent with the change of vibrational Rms, which focuses on the degradation period around the occurrence of failure in the 18th hour, indicating that the output of the model is not only high-precision failure probability (96.8% accuracy), but also has more interpretable decision-making support, that is, the digital system is not only a “black box output”, but also an “understandable and trustworthy” intelligent decision-making tool. This overcomes the cognitive difference of technology and business in digital transformation.

In the dimension of digital operations, the deep reinforcement learning decision-making based on predictive information directly outputs maintenance scheduling strategies, and the decision boundary shrinks accordingly with the equipment risk level, thereby realizing the balance of maintenance cost and failure loss. After field deployment, OEE increased by 11.8 percentage points, and unplanned downtime dropped to 44 hours per month, down 60.7%; the average monthly maintenance cost and failure loss dropped by 33.9% and 60.0%, respectively, which shows a good effect of balancing improvement in efficiency, cost and stability. The results validate the significant improvement of operation performance brought by digital technology, and the achievement of digital transformation goal of “transferring O&M model from “experience-based” to “databased” and from “passively taking measures when trouble occurs” to “proactively taking measures when signs are detected”, i.e. using digital investment to bring about quantifiable business benefits.

In terms of digital engineering implementation, the lightweight deployment solution can upload the decision-making model and run it on NVIDIA Jetson Nano edge nodes. This paper tested it for 120 hours, and the average latency of the entire loop of the decision-making process was still 1.64 seconds, and the maximum latency did not exceed 1.93 seconds. The model memory is not constantly increasing, and the average CPU utilization is only 48.2%. This result demonstrates that the digital technology solution can meet real-time response requirements in resource-constrained industrial edge environments and possesses the reliability for long-term continuous operation. It solves the engineering bottleneck of "advanced technology but difficult implementation" in digital transformation, providing a practical foundation for the large-scale promotion of digital technologies in manufacturing scenarios.

The deep learning technology system guided by performance optimization constructed in this article has connected the complete path of digital transformation from data assetization, decision intelligence to operational performance. This paper has achieved deep mining of data value at the technical level, structural improvement of performance indicators at the operational level, and practical verification of technical solutions at the engineering level, providing a key technological paradigm for digital transformation of manufacturing enterprises.

V. CONCLUSION

This article constructs a performance-guided deep learning system that integrates multi-source data, attention-enhanced temporal prediction, and deep reinforcement learning into a unified process. This technical architecture effectively solves the core challenge of converting complex fiber morphology and yarn spatial distribution data assets into quantifiable operational performance for manufacturing enterprises. This system establishes a multimodal feature alignment mechanism in the data fusion layer, introduces attention to locate degraded key periods in the prediction layer to improve interpretability, and dynamically maps fault probabilities to maintenance scheduling strategies in the decision layer, forming a closed loop from state perception to task execution, shifting the operation and maintenance mode from passive response to active intervention. The practical application results show that this technical solution has connected the conversion path between equipment health status and operational performance indicators, and achieved consistent improvements in improving equipment comprehensive utilization efficiency, reducing unplanned downtime, and optimizing maintenance cost structure. Subsequent research can further explore the generalization mechanism of reward functions across spinning and weaving scenarios and adaptive deployment methods for heterogeneous industrial hardware. This exploration aims to enhance the transferability of technical solutions across diverse fiber manipulation environments and continuously empower the digital transformation of manufacturing enterprises through optimized yarn spatial distribution and fiber morphology data.

AUTHOR CONTRIBUTIONS

Conceptualization – Wei Qiu, Ying Zhao, Mengxia Ren and Donghe Liang; methodology – Wei Qiu, Mengxia Ren and Donghe Liang; investigation – Wei Qiu, Ying Zhao, Mengxia Ren and Donghe Liang; writing-original draft preparation – Wei Qiu, Ying Zhao, Mengxia Ren and Donghe Liang. All authors have read and agreed to the published version of the manuscript.

CONFLICTS OF INTEREST

The authors declare no conflict of interest.

FUNDING

This research received no external funding.

ACKNOWLEDGEMENTS

Not applicable.

REFERENCES

[1] J. Matschewsky, L. Kambanou M, and T. Sakao, "From mass customization to circular customization-measuring and managing variety in product service systems," Resources, Conservation and Recycling, vol. 219, no. 1, pp. 108261-108268, 2025, doi: 10.1016/j.resconrec.2025.108261.

[2] R. Lorenz, M. Kraus, H. Wolf, S. Feuerriegel, and H. Netland T, "Selecting advanced analytics in manufacturing: a decision support model," Production Planning & Control, vol. 35, no. 7, pp. 711-724, 2024, doi: 10.1080/09537287.2022.2126951.

[3] A. Shokrani, H. Dogan, D. Burian, D. Nwabueze T, P. Kolar, Z. Liao, A. Sadek, et al., "Sensors for in-process and on-machine monitoring of machining operations," CIRP Journal of Manufacturing Science and Technology, vol. 51, no. 1, pp. 263-292, 2024, doi: 10.1016/j.cirpj.2024.05.001.

[4] U. Hassan I, K. Panduru, and J. Walsh, "Review of data processing methods used in predictive maintenance for next generation heavy machinery," Data, vol. 9, no. 5, pp. 69-77, 2024, doi: 10.3390/data9050069.

[5] B. Yu, Z. Yang, Y. Yu, and G. Xiang, "GMVE: Graph-Mamba Variational Encoder for Interpretable Remaining Useful Life Prediction with Uncertainty Quantification," Knowledge-Based Systems, vol. 328, no. 25, pp. 114217-114226, 2025, doi: 10.1016/j.knosys.2025.114217.

[6] A. Andres, A. Martinez-Seras, I. Laña, and D. Ser J, "On the black-box explainability of object detection models for safe and trustworthy industrial applications," Results in Engineering, vol. 24, no. 1, pp. 103498-103505, 2024, doi: 10.1016/j.rineng.2024.103498.

[7] Y. Wei, Y. Wang, S. Lu, X. Li, and H. Su, "Adaptive integration of closed-loop manufacturing scheduling and predictive maintenance based on RUL prediction by CNN-SVR model," Expert Systems with Applications, vol. 311, no. 15, pp. 131269-131276, 2026, doi: 10.1016/j.eswa.2026.131269.

[8] F. Wei, X. Ma, Q. Qiu, Y. Ma, J. Wang, and L. Yang, "Adaptive mission risk control under incomplete health information and resource limitation: A constrained multi-state predictive maintenance model," Reliability Engineering & System Safety, vol. 266, no. 1, pp. 111697-111704, 2025, doi: 10.1016/j.ress.2025.111697.

[9] Q. Zhao, X. Zhang, J. Xie, et al., "KFC-Former: An interpretable kernel function convolutional-former with phase space reconstruction for remaining useful life prediction of mechanical equipment," Expert Systems with Applications, vol. 302, no. 15, pp. 130472-130481, 2025, doi: 10.1016/j.eswa.2025.130472.

[10] P. Huang, Y. Wang, Y. Gu, and G. Qiu, "A bearing RUL prediction approach of vibration fault signal denoise modeling with Gate-CNN and Conv-transformer encoder," Measurement Science and Technology, vol. 35, no. 6, pp. 066104-066112, 2024, doi: 10.1088/1361-6501/ad2cd9.

[11] Y. Li, N. Li, J. Ren, and W. Shen, "An interpretable light attention-convolution-gate recurrent unit architecture for the highly accurate modeling of actual chemical dynamic processes," Engineering, vol. 39, no. 1, pp. 104-116, 2024, doi: 10.1016/j.eng.2024.07.009.

[12] B. Zhao, W. Zhang, Y. Zhang, C. Zhang, C. Zhang, and J. Zhang, "Lithium-ion battery remaining useful life prediction based on interpretable deep learning and network parameter optimization," Applied Energy, vol. 379, no. 1, pp. 124713-124720, 2025, doi: 10.1016/j.apenergy.2024.124713.

[13] J. Zeng and Z. Liang, "Predictive group maintenance using probabilistic prognostics and deep reinforcement learning," Computers & Industrial Engineering, vol. 212, no. 1, pp. 111738-111745, 2025, doi: 10.1016/j.cie.2025.111738.

[14] Y. Liu, J. Gao, T. Jiang, and Z. Zeng, "Selective maintenance and inspection optimization for partially observable systems: An interactively sequential decision framework," IISE Transactions, vol. 55, no. 5, pp. 463-479, 2023, doi: 10.1080/24725854.2022.2062627.

[15] T. Wang, J. Guo, B. Zhang, G. Yang, and D. Li, "Deploying AI on edge: Advancement and challenges in edge intelligence," Mathematics, vol. 13, no. 11, pp. 1878-1886, 2025, doi: 10.3390/math13111878.

[16] M. Wang, H. Liu, H. Tang, et al., "IFNet: Data-driven multisensor estimate fusion with unknown correlation in sensor measurement noises," Information Fusion, vol. 115, no. 1, pp. 102750-102759, 2025, doi: 10.1016/j.inffus.2024.102750.

[17] X. Yan, W. Liang, and S. Sun, "An improved method for predicting the remaining useful life using a spatial-temporal feature extraction network with attention mechanism," IEEE Access, vol. 12, no. 1, pp. 66587-66604, 2024, doi: 10.1109/ACCESS.2024.3396156.

[18] Z. Zhou, D. Yu, M. Chen, Y. Qiao, Y. Hu, and W. He, "An edge intelligence-based model deployment method for CNC systems," Journal of Manufacturing Systems, vol. 74, no. 1, pp. 716-751, 2024, doi: 10.1016/j.jmsy.2024.04.029.

[19] A. Sama, S. Warnars H L H, H. Prabowo, Meyliana, and N. Hidayanto A, "Acquiring Automation and Control Data in The Manufacturing Industry: A Systematic Review," Procedia Computer Science, vol. 227, no. 1, pp. 214-222, 2023, doi: 10.1016/j.procs.2023.10.519.

[20] J. Ni, S. Yang, and Y. Liu, "Data Cleaning Model of Mine Wind Speed Sensor Based on LOF-GMM and SGAIN," Applied Sciences, vol. 15, no. 4, pp. 1801-1809, 2025, doi: 10.3390/app15041801.

[21] A. Ghosh, "Advanced time-series data management for continuous process optimization in manufacturing execution system," International Research Journal of Modernization in Engineering Technology and Science, vol. 4, no. 1, pp. 2681-2688, 2022, doi: 10.56726/IRJMETS29797.

[22] M. Barros, P. Fontenele L, S. Silva M, T. Rissi, E. Cabrera, A. Tannuri E, S. Gomi E, et al., "Embracing data irregularities in multivariate time series," Applied Soft Computing, vol. 186, no. 1, pp. 114039-114045, 2025, doi: 10.1016/j.asoc.2025.114039.

[23] C. Kihal M, H. Moulai, H. Azizi, C. Smartt, and P. Thomas D W, "Sequential adaptive sampling based on cubic spline interpolation to accelerate the near field scanning," Journal of Electromagnetic Waves and Applications, vol. 36, no. 16, pp. 2237-2259, 2022, doi: 10.1080/09205071.2022.2071767.

[24] Y. Lv, J. Wang, C. Zhang, and J. Ding, "Composite fault feature extraction for gears based on MCKD-EWT adaptive wavelet threshold noise reduction," Measurement and Control, vol. 58, no. 2, pp. 185-195, 2025, doi: 10.1177/00202940241253173.

[25] S. Guha and A. Datta, "GDRNet: A Deep Learning Framework for Dynamic Sliding Window Determination and Improved Prediction Accuracy in Online IIoT Systems," IEEE Access, vol. 13, no. 1, pp. 200382-200393, 2025, doi: 10.1109/ACCESS.2025.3634671.

[26] H. Bao, L. Song, Z. Zhang, B. Han, J. Wang, J. Ma, and X. Jiang, "Prediction of the remaining useful life of rolling bearings by LSTM based on multidomain characteristics and a dual-attention mechanism," Journal of Mechanical Science and Technology, vol. 37, no. 9, pp. 4583-4596, 2023, doi: 10.1007/s12206-023-0814-x.

[27] S. Rathore M and P. Harsha S, "Prognostics analysis of rolling bearing based on bi-directional LSTM and attention mechanism," Journal of Failure Analysis and Prevention, vol. 22, no. 2, pp. 704-723, 2022, doi: 10.1007/s11668-022-01357-1.

[28] C. Coelho, P. Costa M F, and L. Ferrás L, "Enhancing continuous time series modelling with a latent ODE-LSTM approach," Applied Mathematics and Computation, vol. 475, no. 1, pp. 128727-128735, 2024, doi: 10.1016/j.amc.2024.128727.

[29] C. He, L. Chen, N. Sun, P. Chen, X. Xu, and S. Lu, "Redundancy modification and potential feature reactivation network for predicting the remaining useful life of machines," IEEE Sensors Journal, vol. 25, no. 3, pp. 5060-5072, 2024, doi: 10.1109/JSEN.2024.3514168.

[30] R. Dutta Baruah and M. Munoz-Organero, "Recurrent neural network with contextual attention for prognostics," Evolving Systems, vol. 16, no. 4, pp. 117-125, 2025, doi: 10.1007/s12530-025-09744-3.

[31] H. Zhang, X. Wei, Z. Liu, Y. Ding, and Q. Guan, "Condition-based maintenance for multi-state systems with prognostic and deep reinforcement learning," Reliability Engineering & System Safety, vol. 255, no. 1, pp. 110659-110667, 2025, doi: 10.1016/j.ress.2024.110659.

[32] S. Eidi, F. Haghighi, A. Safari, and E. Zio, "Optimal dynamic state-dependent maintenance policy by deep reinforcement learning," Quality and Reliability Engineering International, vol. 41, no. 6, pp. 2715-2728, 2025, doi: 10.1002/qre.3806.

[33] Y. Zhang, B. Cai, C. Gao, Y. Zhao, X. Shao, and C. Yang, "A system-centred predictive maintenance re-optimization method based on multi-agent deep reinforcement learning," Expert Systems with Applications, vol. 274, no. 1, pp. 127034-127041, 2025, doi: 10.1016/j.eswa.2025.127034.

[34] X. Ji, F. Gong, N. Wang, J. Xu, and X. Yan, "Cloud-edge collaborative service architecture with large-tiny models based on deep reinforcement learning," IEEE Transactions on Cloud Computing, vol. 13, no. 1, pp. 288-302, 2025, doi: 10.1109/TCC.2024.3525076.

[35] M. Ladegourdie and J. Kua, "Performance analysis of opc ua for industrial interoperability towards industry 4.0," IoT, vol. 3, no. 4, pp. 507-525, 2022, doi: 10.3390/iot3040027.

[36] I. Liu H, M. Galindo, H. Xie, K. Wong L, H. Shuai H, H. Li Y, and H. Cheng W, "Lightweight deep learning for resource-constrained environments: A survey," ACM Computing Surveys, vol. 56, no. 10, pp. 1-42, 2024, doi: 10.1145/3657282.