Brodogradnja Volume 77 Number 4 (2026) 77411

University of Zagreb Faculty of Mechanical Engineering and Naval Architecture

journal homepage: www.brodogradnja.fsb.hr

Brodogradnja

An International Journal of Naval Architecture and Ocean Engineering for Research and Development

A step-sensitive LightGBM framework with dynamic consistency correction for predicting turbocharger exhaust outlet temperature in marine diesel engines

Yanxian Cai[1], *Jianfan Wang*[1, *], *Zhehui Hou*[2], *Hong Zeng*[2]

[1]*China Classification Society, Beijing 100007, China*
[2]*Marine Engineering College, Dalian Maritime University, Dalian 116026, China*

ARTICLE INFO

Keywords:
Marine diesel engine
Turbocharger exhaust outlet temperature
Bayesian optimization
LightGBM
Dynamic consistency correction
Residual compensation

ABSTRACT

Turbocharger exhaust outlet temperature is a key indicator for assessing thermal load and operational safety in marine diesel engines. Accurate prediction remains challenging under transient operating conditions because the temperature signal is governed by strong thermal inertia, nonlinear coupling, and step-response behavior. To address these challenges, this study develops a data-driven prediction framework based on LightGBM. The framework introduces a step-sensitive weighting strategy to strengthen model learning in transient regions and uses Bayesian optimization for automatic hyperparameter tuning. A dynamic consistency correction module and a residual compensation module are further integrated to improve temporal smoothness and prediction accuracy. The method is validated using real shipboard data collected from the vessel "Xin Hong Zhuan". On the test set, the proposed model achieves a mean absolute error (MAE) of 6.41 °C, a root mean square error (RMSE) of 7.38 °C, and a coefficient of determination (R^2) of 0.990. Compared with the best-performing baseline model, the proposed approach reduces RMSE by approximately 33 %, indicating clear advantages over traditional regression methods and representative deep learning baselines. These results suggest that the proposed framework provides an accurate and robust approach for temperature prediction in complex marine engineering systems.

1. Introduction

With the increasing adoption of data-driven operation and maintenance in intelligent shipping, accurate condition monitoring of marine electromechanical equipment has become an important engineering challenge [1]. Traditional periodic inspection is costly, lacks real-time capability, and is inadequate for systems operating under dynamically varying conditions [2]. As marine diesel engines move toward higher power density, greater efficiency, and lower emissions, the thermal loads imposed on engines and exhaust systems continue to increase [3]. Turbocharger exhaust outlet temperature directly reflects the combustion efficiency of marine diesel engines, the operating state of turbochargers, and the thermal load level of exhaust systems. Therefore, this temperature is critical for engine operation monitoring, performance evaluation, and fault prediction [4, 5].

[*] Corresponding author.
E-mail address: wangjianfan@ccs.org.cn

https://doi.org/10.21278/brod77411
Received 11 February 2026; Received in revised form 19 May 2026; Accepted 09 June 2026
Available online 06 July 2026
ISSN 0007-215X; eISSN 1845-5859

Traditional exhaust temperature prediction methods rely mainly on empirical formulas or simplified mechanistic models with multiple prior assumptions, which limits their applicability under complex real-world operating conditions. Although data-driven machine learning models have shown advantages in nonlinear modeling, most existing studies still formulate exhaust temperature prediction as a static regression task and do not sufficiently account for the intrinsic dynamic evolution of temperature signals.

Zhang et al. [6] proposed a Long Short-Term Memory (LSTM)-based method to improve the prediction accuracy of exhaust gas temperature in marine diesel engines. The LSTM network was constructed as a multi-input and multi-output model and was trained and validated using real ship operating data. The results indicated that the LSTM model is applicable to main-engine exhaust gas temperature prediction and can provide high prediction accuracy. However, thermal inertia and abrupt step changes in temperature signals were not explicitly considered, and the prediction task was still treated as a general time-series problem. This limitation can lead to prediction lag and large peak-fitting deviations under transient operating conditions. Ji et al. [7] developed a hybrid deep learning model for exhaust gas temperature prediction in marine diesel engines by combining a convolutional neural network, bidirectional long short-term memory network, and attention mechanism (CNN-BiLSTM-Attention). Comparative experiments demonstrated improved prediction accuracy, but the complex network structure contains redundant parameters, is prone to overfitting in small-sample shipboard scenarios, and requires substantial computational resources, which may hinder deployment in embedded ship monitoring systems [8]. Sun et al. [9] combined a temporal pattern attention mechanism with BiLSTM and proposed a method for predicting short-term exhaust gas temperature trends. The mean square error, root mean square error, and coefficient of determination of this model were superior to those of the comparison methods, supporting its stability and accuracy. Nevertheless, this method mainly optimized the network structure and did not impose constraints derived from the physical behavior of the exhaust system, which may allow non-physical fluctuations in the predicted sequence. Gao et al. [10] proposed a real-time exhaust temperature prediction model based on Gradient Boosting Decision Tree (GBDT) to address the limited accuracy of conventional machine learning models. However, this work treated prediction as a static regression problem and did not construct temporal features to characterize the dynamic evolution of temperature, limiting adaptability to transient conditions with sudden load changes. In addition, the absence of automatic hyperparameter optimization and error correction mechanisms may reduce model generalization and stability. Liu et al. [11] compared four machine learning algorithms for exhaust gas temperature prediction and reported that gradient boosting decision trees achieved the best predictive performance, random forest required the least calibration effort but yielded the lowest prediction accuracy, support vector machines achieved the smallest prediction error but required the highest computational resources, and artificial neural networks offered good overall applicability but required careful hyperparameter tuning. Building on these findings, the method proposed in this study improves the prediction pipeline from the perspectives of model selection, feature design, training strategy, and error correction. Existing machine learning and deep learning approaches have demonstrated the feasibility of exhaust temperature prediction, but three limitations remain: insufficient modeling of dynamic evolution, including thermal inertia and step-response behavior; dependence on complex model structures with high computational cost [12]; and lack of mechanisms for enforcing physical consistency and correcting systematic errors. These limitations can cause prediction lag, reduced robustness, and restricted applicability in real shipboard environments. To address these issues, this study proposes a unified framework that integrates step-sensitive feature enhancement, dynamic consistency correction, and residual compensation to improve both prediction accuracy and temporal consistency under complex operating conditions.

2. Theory and Methodology

2.1 Algorithm Principles

Linear regression is a supervised learning algorithm for modeling and predicting continuous variables. The method estimates the linear relationship between input features and a target variable and fits the optimal line or hyperplane by minimizing the mean squared error, commonly through least squares estimation [13]. Linear regression is computationally efficient and highly interpretable. However, the method has limited ability to represent nonlinear relationships or complex interaction patterns [14].

Support Vector Machine (SVM) is a supervised learning model used for classification and regression. The central idea is to identify an optimal hyperplane that separates samples while improving generalization by maximizing the classification margin [15]. For linearly inseparable data, SVM maps samples into a high-dimensional feature space through kernel functions, thereby enabling nonlinear pattern representation. SVM has been widely used in pattern recognition tasks, including image recognition and text classification [16].

Random Forest is an ensemble learning model based on decision trees. The method constructs multiple decision trees through bootstrap sampling and aggregates their outputs by voting or averaging to improve generalization and stability [17]. Random Forest is commonly applied to classification, regression, and feature selection tasks [18].

LightGBM is an efficient machine learning framework based on gradient-boosted decision trees. The method is designed to improve training speed and reduce memory consumption, making it suitable for large-scale data scenarios [19]. Owing to its high efficiency and strong nonlinear modeling capability, LightGBM has been widely used in industrial applications such as search systems, recommendation systems, and click-through-rate prediction [20].

Long Short-Term Memory (LSTM) networks are recurrent neural networks designed for sequential data. By introducing input, forget, and output gates together with cell states, LSTM networks can capture long-term dependencies in sequences [21]. This architecture is particularly suitable for scenarios with complex temporal dynamics and long-range contextual dependencies and represents a foundational model for deep sequential learning [22].

The Gated Recurrent Unit (GRU) is a simplified and efficient variant of the LSTM recurrent neural network [23]. By using update and reset gates, GRU can dynamically control information retention and forgetting, alleviate the vanishing-gradient problem, and usually reduce training time and computational cost relative to LSTM. GRU is widely adopted in sequence modeling tasks, including natural language processing, time-series prediction, and speech recognition [24].

2.2 Experimental Design

This study aims to achieve high-precision prediction of turbocharger exhaust outlet temperature in marine diesel engines using multidimensional historical operating parameters. During operation, exhaust outlet temperature is governed not only by current operating conditions but also by the historical state of the system. Under load variations and similar transient events, the signal exhibits evident step responses and thermal inertia. To capture these properties, this study proposes a machine-learning-centered prediction framework [25]. A gradient-boosted tree model is used as the primary predictor, and feature engineering with optimized training strategies is used to extract dynamic information from operating parameters. Dynamic consistency correction and residual compensation modules are then introduced to refine model outputs from a temporal-consistency perspective, thereby improving prediction accuracy and stability [26]. At each time step, the model predicts exhaust outlet temperature from the corresponding input features.

To quantitatively evaluate model performance, this study uses mean absolute error (MAE), root mean square error (RMSE), and the coefficient of determination (R^2) as assessment metrics, which are calculated as follows:

MAE reflects the average absolute magnitude of prediction errors as follows:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|\hat{y}_i - y_i\right|, MAE \in [0, +\infty) \tag{1}$$

The value range is $[0, +\infty)$. MAE = 0 indicates that the predicted and measured samples agree perfectly. A smaller MAE indicates a lower average prediction error and better model performance; a larger MAE indicates greater prediction error and poorer predictive performance.

RMSE measures the square root of the average squared deviation between predicted and true values as follows:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}, RMSE \in [0, +\infty) \tag{2}$$

The value range is $[0, +\infty)$. When the RMSE = 0, predicted and measured samples agree perfectly. A smaller RMSE indicates a smaller prediction error and better model performance; a larger RMSE indicates a larger prediction error and poorer predictive performance.

R^2 measures the proportion of data variance explained by the regression model as follows:

$$R^2 = 1 - \left[\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \overline{y})^2}\right], R^2 \in (-\infty, 1] \tag{3}$$

The coefficient of determination R^2 ranges from -∞ to 1. A value closer to 1 indicates stronger predictive performance, whereas $R^2 = 0$ indicates that the model performs no better than a mean-value predictor. Negative values indicate performance worse than that of the mean predictor [27].

Figure 1 illustrates the overall methodology. The upper part of the figure presents the data processing pipeline, including data cleaning, feature engineering, and feature selection. The middle part shows the model development process, in which candidate models are trained and evaluated. The final model is selected according to performance metrics and Bayesian optimization. The lower part highlights the components specific to this study: time-series feature engineering is used to capture temporal dependencies, and dynamic consistency correction with residual compensation is integrated into the final model to improve prediction accuracy and maintain physically consistent temporal behavior.

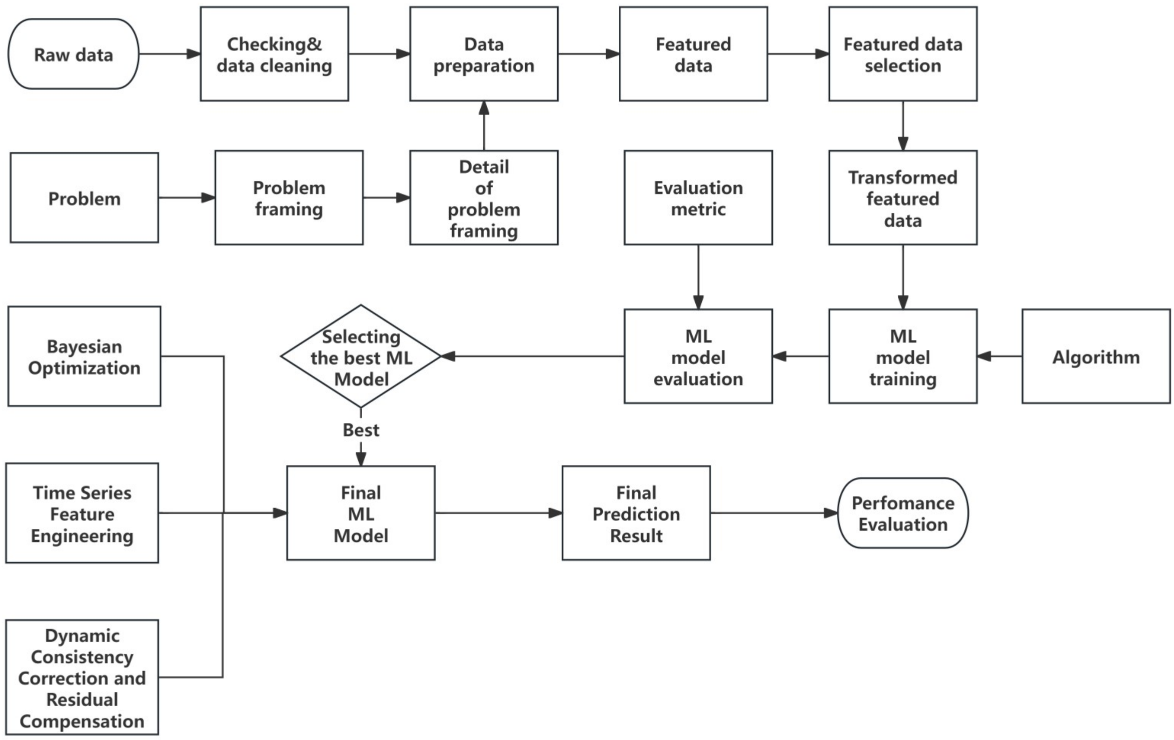

Fig. 1 Overall experimental methodology flowchart

3. Experiments

3.1 Data collection and preprocessing

The “Xinhongzhuan” is a teaching vessel independently developed by Dalian Maritime University. The vessel has an overall length of 69.83 m, a beam of 10.9 m, a fully electric twin-pod propulsion system, a design speed exceeding 18 knots, and an endurance of 2,500 nautical miles. The key engine parameters are listed in Table 1. The experimental data were obtained from the No. 1 diesel engine of the vessel, a Wrtsil 8L20 engine, using real operational data. This engine is equipped with multiple real-time sensors that sample at fixed intervals and transmit data to a local data management platform, forming the dataset used in this study [28]. Turbocharger exhaust gas temperature was measured using a Greystone TE511 high-precision platinum resistance temperature detector (RTD) transmitter. The sensing element is Pt1000 (1000 at 0 °C), with a measurement accuracy of ±0.3 °C, a measurement range of -200 to 850 °C, and a 4-20 mA current signal output. The probe is a 6.35 mm stainless-steel insertion type (TE511B) installed at the exhaust outlet section through a 1/2 NPT interface, with a sampling frequency of once per minute. The overall measurement uncertainty of the system is less than ±0.5 °C, satisfying the requirements for temperature monitoring under transient diesel engine operating conditions. Table 2 provides the specifications of the key sensors deployed in the monitoring system. The raw dataset contains 11,520 samples collected at one-minute intervals and covers complex operating conditions, including shutdown, cold start, and medium-to-high load operation. Thirty-five sensor variables were selected based on expert judgment and the set of monitored variables, including fuel-system parameters, main-engine load, diesel engine speed, scavenging pressure, lubricating oil temperature, and vibration characteristics. The dataset was divided into training and test sets at a chronological ratio of 7:3. The training set was used to fit model parameters, whereas the test set was used for hyperparameter tuning and model selection.

Table 1 Engine parameters

Parameter Name	Unit	Value
Bore	mm	200
Rotational Speed	r/min	1000
Power	kW	1600
Piston stroke	mm	280
Length/Width/Height	mm	3783 × 1713 × 2089
Weight	kg	11000
Ignition sequence	-	1-3-7-4-8-6-2-5

Table 2 Key sensor specifications

Measurement	Sensor	Physical Quantity	Accuracy
Turbocharger outlet temperature	TE517	Exhaust gas temperature	Pt1000 (1000 Ω at 0 °C), ±0.3 °C
Turbocharger inlet temperature	TE511	Exhaust gas temperature	Pt1000 (1000 Ω at 0 °C), ±0.3 °C
Charge air temperature (after cooler)	TE601	Intake air temperature	±0.3 °C
Lubricating oil temperature	TE201	Oil temperature	±0.3 °C
Jacket cooling water temperature	TE401/TE402	Water temperature	±0.3 °C
Boost pressure	PT601	Air pressure	±0.01 bar

The experimental environment was configured as follows:

Hardware configuration

Processor (CPU): Intel Core i7

Graphics card (GPU): NVIDIA GeForce RTX 3070 with 8 GB GDDR6 VRAM, multiple CUDA cores, and support for DirectX 12 Ultimate

Memory (RAM): 16 GB

Storage: 215 GB SSD (Solid State Drive)

Operating system (OS): Windows 11

Software configuration

Operating System: Windows 11

Development environment: PyCharm

Because industrial data are collected in complex operating environments, raw monitoring data inevitably contain missing values, outliers, and inconsistent formats. Standardization before modeling is therefore essential. To ensure stable model training and reliable prediction, this study systematically cleans and preprocesses the raw monitoring data before model development [29].

(1) Data integrity and format processing

First, the integrity of the raw data is verified. The dataset file is decoded using utf-8-sig encoding, and numerical conversion methods are then used to convert variables stored as strings or containing delimiters, such as decimal commas, into numeric variables. Records that cannot be converted are treated as missing values. To avoid excessive loss of valid information, only numeric variables with a missing-value ratio no greater than 50 % are retained for subsequent modeling and analysis.

(2) Chronological verification and sorting

Because turbocharger exhaust temperature has pronounced time-series characteristics, the data are strictly sorted by timestamp to ensure that the sample sequence is consistent with the actual operating process and that the training set precedes the test set in chronological order. During testing, only historical data are used to predict future values to prevent leakage of future information and to preserve the authenticity and engineering relevance of the experimental results [30].

(3) Missing-value handling

For localized data gaps caused by sensor faults or communication interruptions, feature median imputation is applied. This strategy preserves the overall statistical properties of the data without introducing additional distributional bias and is therefore suitable for model training [31].

3.2 Feature Engineering

To reduce the effects of redundant variables on model training stability and generalization performance, this study first uses Spearman rank correlation coefficients to evaluate the relationships between raw monitoring variables and turbocharger exhaust outlet temperature. Spearman coefficients characterize monotonic correlations between variables [32].

$$\rho_S = 1 - \frac{6\sum_{i=1}^{n} d_i^2}{n\left(n^2 - 1\right)} \tag{4}$$

where d_i denotes the rank difference between the i th sample in the ordered pairs of variables, and n represents the sample size.

The absolute value of the Spearman rank correlation coefficient is commonly used to characterize the strength of monotonic relationships between variables. When ρ_S falls within the range 0.00-0.19, the variables exhibit only an extremely weak correlation, and may be regarded as noise variables; When ρ_S falls within the range 0.20-0.39, the variable exhibits a weak correlation with the target, showing some trend but making a limited independent contribution to the prediction; When ρ_S falls within the range 0.40-0.59, the variable demonstrates moderate correlation with the target, exerting a noticeable influence on the target variation; When ρ_S reaches 0.60-0.79, the variable is considered highly correlated with the target and constitutes a key input variable; when ρ_S exceeds 0.80, the relationship between the variable and the target approaches a monotonic function, indicating extremely strong correlation.

Figure 2 presents the Spearman correlation matrix heatmap for the selected monitoring variables and the target variable, namely turbocharger exhaust outlet temperature. Color intensity represents the strength of the monotonic relationship between variables, with warmer colors indicating stronger positive correlations.

Several variables, including turbocharger exhaust inlet temperature, lubricating oil outlet temperature, scavenging pressure, and engine load, show strong correlations with the target variable. In particular, exhaust inlet temperature has an extremely high correlation (= 0.9962), indicating an almost deterministic relationship with turbocharger exhaust outlet temperature.

To prevent potential information leakage and avoid trivial prediction, exhaust inlet temperature is excluded from the input feature set. The remaining variables with moderate-to-high correlations (> 0.3) are retained as core input features for subsequent modeling. This selection strategy enables the model to capture meaningful physical relationships while preserving generalization capability. Although this sensor may be available in practical applications, including it would substantially simplify the prediction task.

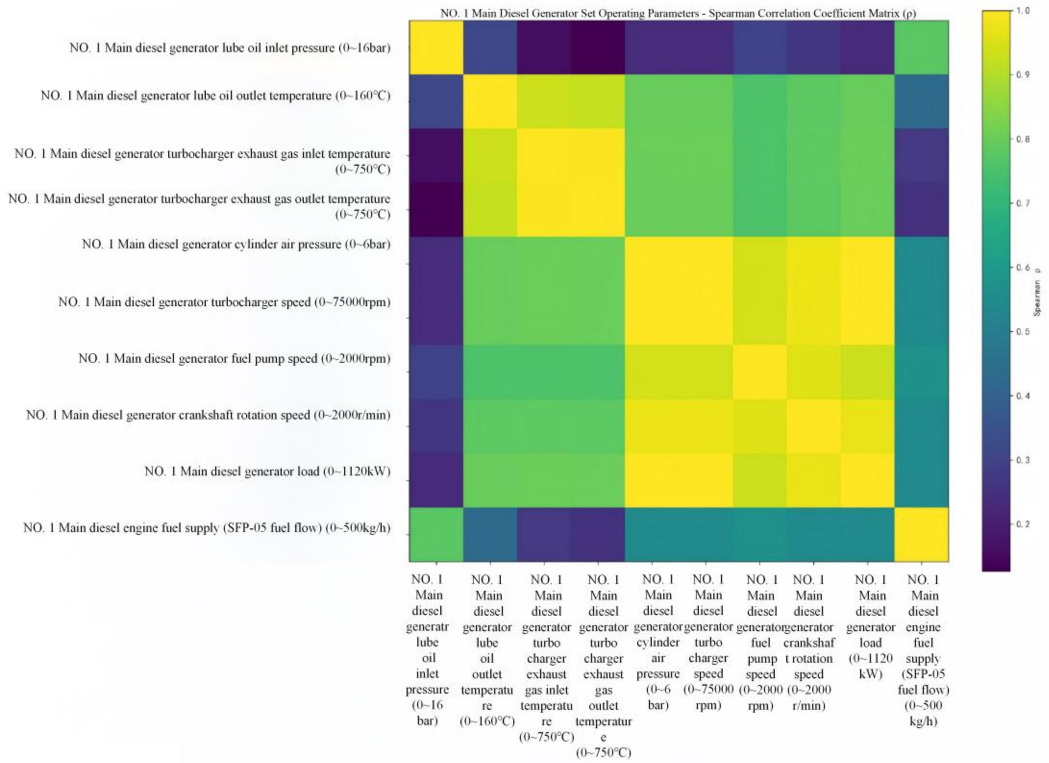

Fig. 2 Feature correlation matrix heatmap

Table 3 Correlation coefficients between monitoring variables and target variables

Serial No.	Variable Name and Unit	Correlation Coefficient
1	Turbocharger exhaust inlet temperature (0–750 °C)	0.9962
2	Lubricating oil outlet temperature (0–160 °C)	0.9249
3	Scavenging pressure (0–6 bar)	0.8012
4	Turbocharger speed (0–75,000 rpm)	0.8008
5	Engine load (−10–120 %)	0.8003
6	Torsional vibration (0–2000 mdeg)	0.7812
7	Diesel engine speed (0–2000 rpm)	0.7544
8	Fuel supply mass flow rate FSF06 (0–500 kg/h)	0.2535
9	Fuel oil return mass flow FSF07 (0–500 kg/h)	0.2479
10	Fuel oil inlet pressure (0–16 bar)	0.1264

During feature selection, variables with absolute Spearman correlation coefficients below 0.3 are excluded to reduce the influence of redundant and noisy inputs on training stability and generalization performance. This threshold is used as a general guideline rather than a strict physical cutoff because variables with low global correlation, such as fuel flow rate, may still contain useful information under transient conditions. In this study, only variables with moderate or high correlations with the prediction target are retained for subsequent modeling, which improves training efficiency and reduces the risk of overfitting.

Variables 2-7 are used as a fixed input feature set throughout the subsequent experiments to ensure reproducibility and fairness in comparative testing. Although turbocharger exhaust inlet temperature shows an extremely high correlation with the target variable, it is excluded to avoid potential information leakage. In practical applications, this sensor may be available, but its inclusion would make the prediction task substantially easier. Therefore, this study evaluates model capability under a more challenging and realistic setting that relies on indirect features. The correlation coefficients between monitoring variables and the target variable are reported in Table 3. Spearman analysis is used only for initial variable screening, and no subsequent adaptive filtering is performed. Further performance improvements are mainly derived from temporal feature expansion of the fixed input variables and optimization of the model training strategy.

3.3 Predictive Model Training Approach

After comparative evaluation of several models, the gradient-boosted tree model LightGBM is selected as the primary predictor for characterizing the nonlinear mapping between input features and turbocharger exhaust outlet temperature. Compared with deep learning models, LightGBM provides stable training, strong generalization on small-to-medium industrial datasets, high computational efficiency, and relatively strong interpretability. Considering the time-series nature of the data, this study uses time-series cross-validation (TimeSeriesSplit) for model training and validation. This strategy preserves the temporal order between training and validation sets and prevents information leakage. TimeSeriesSplit is used during model development for hyperparameter tuning and model selection. Because operating conditions differ substantially across time segments, cross-validation folds may exhibit considerable variability. Therefore, final performance is evaluated on an independent hold-out test set, which more closely reflects real-world prediction scenarios.

3.3.1 Step Window Identification

During actual operation, prediction errors are mainly concentrated in step conditions caused by load variations. To improve model learning for these samples, this study constructs a step-discrimination metric based on the rate of change of the target variable as follows:

$$\Delta y(t) = \left| y(t) - y(t-1) \right| \tag{5}$$

A time point is identified as a potential step-change point as follows:

$$\Delta y(t) > P_{95}(\Delta y)$$

where P95(Δy) denotes the 95th percentile of the absolute first-order difference computed from the training dataset. This percentile is used as a selection criterion to identify significant changes, rather than as a magnitude constraint.

To account for the temporal evolution of transient processes, each detected step point is expanded into a window as follows:

$$W_t = [t-K, t+K]$$

where K denotes the window size. This design is motivated by the physical characteristic that temperature dynamics do not change instantaneously but evolve over a short period because of thermal inertia.

3.3.2 Weighted Loss Function Design

During model training, this study introduces a step-sensitivity mechanism through sample weighting. Samples within the step window are assigned higher weights, guiding the model to focus more on transient operating conditions without altering the model structure or input features.

3.4 Model Optimization

To fully exploit temporal information in monitoring data and improve predictive accuracy under complex operating conditions, this study optimizes the LightGBM model in three dimensions: hyperparameter tuning, temporal feature engineering, and prediction refinement. This design establishes a method for predicting turbocharger exhaust outlet temperature under dynamic operating conditions.

3.4.1 Bayesian hyperparameter optimization

The predictive performance of LightGBM depends strongly on hyperparameter configuration, and inappropriate parameter combinations may cause underfitting or overfitting. To avoid the subjectivity and inefficiency of manual parameter tuning, this study uses a tree-structured Bayesian optimization algorithm, namely the tree-structured Parzen estimator (TPE), to automatically search key model hyperparameters.

During optimization, hyperparameter tuning is formulated as a black-box function optimization problem [33], with the prediction error (RMSE or MAE) under time-series cross-validation serving as the objective function. The TPE method constructs a probabilistic model of the parameter space and adaptively updates the sampling strategy according to previous trial results. This process enables efficient approximation of the optimal solution within a finite number of search iterations.

Compared with grid search and random search, Bayesian optimization improves model generalization while maintaining search efficiency, thereby providing a stable base model for subsequent feature engineering and dynamic correction modules.

3.4.2 Temporal Feature Engineering

Turbocharger exhaust outlet temperature exhibits pronounced thermal inertia and dynamic lag, so current operating parameters alone are insufficient for fully characterizing the system state. To enhance the ability of the model to capture dynamic processes and step responses, this study constructs multiscale temporal derivative features based on the fixed input feature set, as described below.

(1) Lag features

The multiorder lag terms for each fundamental variable $x(t)$ are constructed as follows:

$$x(t-\tau), \tau \in \{1,2,3,5,10,15,30,60\} \tag{6}$$

These features characterize short-term system response and long-term thermal inertia, enabling the model to learn explicitly how historical operating conditions influence current exhaust temperature.

(2) Rolling statistics

The rolling mean and rolling standard deviation are constructed for each fundamental variable as follows:

$$\mu_w(t) = \frac{1}{w}\sum_{i=0}^{w-1} x(t-i) \tag{7}$$

$$\sigma_w(t) = \sqrt{\frac{1}{w}\sum_{i=0}^{w-1} (x(t-i)-\mu_w(t))^2} . \tag{8}$$

where the window length is $w \in \{3,5,10,15\}$. Rolling statistics balance noise and trend information while providing a description of local steady-state strength.

(3) Difference features

$$\Delta x(t) = x(t) - x(t-1) \tag{9}$$

$$\Delta_3 x(t) = x(t) - x(t-3) \tag{10}$$

Difference features emphasize dynamic information at operating-state transitions, enhancing model sensitivity to transient changes and step processes and helping reduce peak prediction errors.

Through temporal feature expansion, the model input includes not only raw operating-state information but also historical dependencies, local statistical characteristics, and rate-of-change information. This expanded representation provides richer features for machine learning models to characterize complex thermal system dynamics. The selection ranges for each feature are listed in Table 4.

Table 4 Feature engineering configuration

Feature Selection	Value Selection	Purpose
Basic Feature Selection	Select variables 2-7	Dimension reduction and stability enhancement
Lagged features	$\tau = \{1,2,3,5,10,15,30,60\}$ min	Characterization of thermal inertia and dynamic lag
Rolling statistics	$w = \{3,5,10,15\}$	Description of local steady-state levels and fluctuations
Difference features	First- and third-order differences	Enhanced sensitivity to step responses

3.4.3 Dynamic Consistency Correction and Residual Compensation

Although the LightGBM base model with Bayesian hyperparameter optimization and temporal feature engineering can capture the overall trend of turbocharger exhaust outlet temperature, local prediction biases may still occur during abrupt operating-condition transitions or load changes. These deviations are usually not random noise; rather, they reflect transient nonstationarity in system dynamic response and local limitations in model expressiveness.

To further improve predictive accuracy and temporal consistency under dynamic conditions, this study introduces a two-stage optimization strategy that combines dynamic consistency correction with residual compensation based on B-LightGBM predictions.

(1) Dynamic consistency correction mechanism

Let $\hat{y}$ denote the predicted temperature at time step t. The first-order difference of the prediction sequence is defined as follows:

$$\Delta\hat{y}_t = \hat{y}_t - \hat{y}_{t-1} \tag{11}$$

To suppress non-physical abrupt changes, a variation constraint is imposed according to the empirical distribution of $|\Delta\hat{y}_t|$:

$$\theta = \alpha \bullet Q_{0.95}(|\Delta\hat{y}|) \tag{12}$$

where $Q_{0.95}(\bullet)$ denotes the 95th percentile, and α is a scaling coefficient that controls the constraint intensity. In this study, α is determined through validation.

The constrained increment is then defined as follows:

$$\Delta\hat{y}_t^* = clip(\Delta\hat{y}_t, -\theta, \theta) \tag{13}$$

Finally, the corrected prediction sequence is reconstructed recursively as follows:

$$\hat{y}_t^{(1)} = \hat{y}_{t-1}^{(1)} + \Delta\hat{y}_t^* \tag{14}$$

The initial condition is $\hat{y}_0^{(1)} = \hat{y}_0$.

This mechanism reduces high-frequency oscillations while preserving the global trend, ensuring that the prediction sequence remains physically consistent over time.

(2) Residual compensation model

To further improve prediction accuracy, especially under transient conditions, a residual compensation model is introduced to capture systematic errors that are not learned by the primary model.

The residual is defined as follows:

$$e_t = y_t - \hat{y}_t^{(1)} \tag{15}$$

where y_t is the ground truth, and $\hat{y}_t^{(1)}$ is the dynamically corrected prediction.

A secondary regression model is trained to approximate the residual as follows:

$$\hat{e}_t = f_{res} = (X_t, \hat{y}_t^{(1)}, \Delta X_t, I_t) \tag{16}$$

where:

X_t denotes the original input features,

ΔX_t represents the differential features,

$I_t \in \{0,1\}$ is the step-window indicator.

A LightGBM model is used as the residual regressor because of its efficiency and robustness.

To enhance learning in transient regions, samples within step windows (I_t = 1) are assigned higher weights during training.

The final prediction is obtained as follows:

$$\hat{y}_t^{final} = \hat{y}_t^{(1)} + \hat{e}_t \tag{17}$$

This two-stage correction framework enables targeted error compensation without introducing excessive model complexity, thereby improving both accuracy and robustness.

Compared with direct stacking of complex models, the residual compensation strategy offers two advantages. First, the strategy limits model complexity and reduces the risk of overfitting. Second, the strategy provides targeted correction of prediction errors from the main model, thereby improving overall prediction accuracy and robustness.

The overall procedure of the proposed method is summarized in Table 5.

Table 5 Proposed prediction framework

Step	Description
Input	Feature sequence X_t, trained base model f_{base}, parameter α.
Output	Final prediction $\hat{y}_t = f_{base}(X_t)$
1	Compute base prediction: $\hat{y}_t = f_{base}(X_t)$
2	Compute first-order difference: $\Delta\hat{y}_t = \hat{y}_t - \hat{y}_{t-1}$
3	Compute threshold: $\theta = \alpha \bullet Q_{0.95}(\lvert\Delta\hat{y}_t\rvert)$
4	Apply constraint: $\Delta\hat{y}_t^* = clip(\Delta\hat{y}_t, -\theta, \theta)$
5	Reconstruct corrected sequence: $\hat{y}_t^{(1)} = \hat{y}_{t-1}^{(1)} + \Delta\hat{y}_t^*$
6	Compute residual: $e_t = y_t - \hat{y}_t^{(1)}$
7	Train residual model: $\hat{y}_t = f_{res} = (X_t, \hat{y}_t^{(1)}, \Delta X_t, I_t)$
8	Final prediction: $\hat{y}_t^{final} = \hat{y}_t^{(1)} + \hat{e}_t$

4. Results and Discussion

4.1 Prediction results of different models

4.1.1 Linear regression model

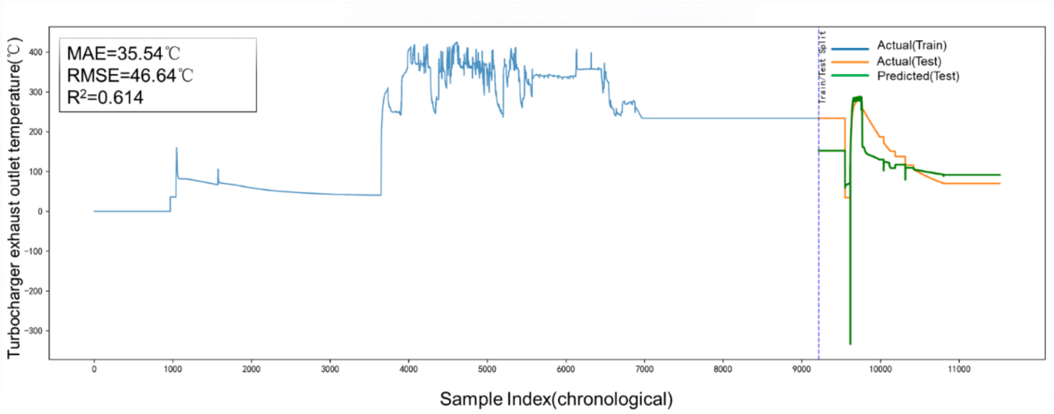

Fig. 3 Prediction results of the linear regression model

The linear regression model is used as the baseline for evaluating prediction performance under purely linear mapping conditions. Figure 3 presents the prediction results of this model. The solid line represents the measured turbocharger exhaust outlet temperature, and the dashed line represents the predicted values. The results indicate that the model can roughly capture the overall trend of exhaust outlet temperature but produces substantially larger errors during transient conditions, such as sudden temperature rises or drops. The predicted curve also shows a clear lag relative to the measured values.

This behavior occurs because linear regression assumes a linear relationship between input variables and the output and therefore has limited ability to capture nonlinear coupling and thermal inertia in diesel engine systems. Under step conditions induced by load variations, the model cannot adequately describe the dynamic evolution of temperature, resulting in high RMSE and MAE values. Consequently, this model serves only as a lower-bound reference for performance comparison.

4.1.2 Support vector machine model

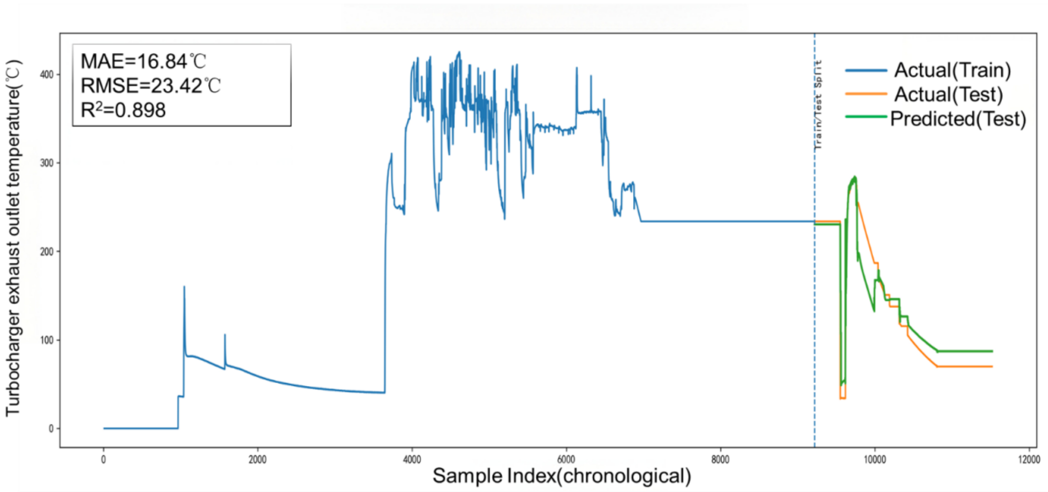

Fig. 4 Prediction results of the support vector machine model

Compared with linear regression, the support vector machine regression model can capture a degree of nonlinear relationship and therefore improves prediction accuracy relative to the baseline. Figure 4 shows the prediction results of the support vector machine model. The experimental results indicate that SVM provides good fitting capability under steady-state operating conditions but still produces noticeable errors during rapid operating-condition transitions. This limitation arises because SVM remains essentially a static mapping model and does not explicitly incorporate time-series structure. Although kernel functions enhance nonlinear representation, SVM cannot directly use historical information to describe thermal inertia and dynamic lag, resulting in insufficient predictive stability under transient operating conditions.

4.1.3 Random forest regression model

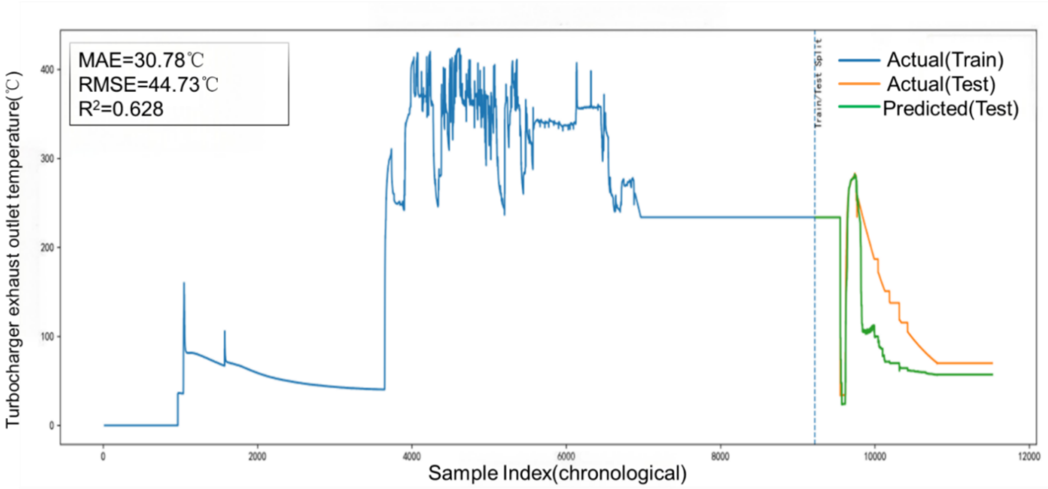

Fig. 5 Prediction results of the random forest regression model

The Random Forest regression model improves nonlinear modeling capability by aggregating multiple decision trees. Figure 5 shows the prediction results of the Random Forest regression model. The experimental results indicate that its predictive performance on this dataset is slightly inferior to that of the support vector machine model.

This limitation can be attributed to the bagging-based ensemble strategy, in which individual trees are constructed independently and do not include an explicit mechanism for sequential error correction. As a result, the model may have limited ability to capture complex dynamic behavior. In addition, Random Forest does not inherently model temporal dependencies. Without explicit time-series features, the model may struggle to represent cumulative and lagged effects in temperature signals.

4.1.4 LightGBM model

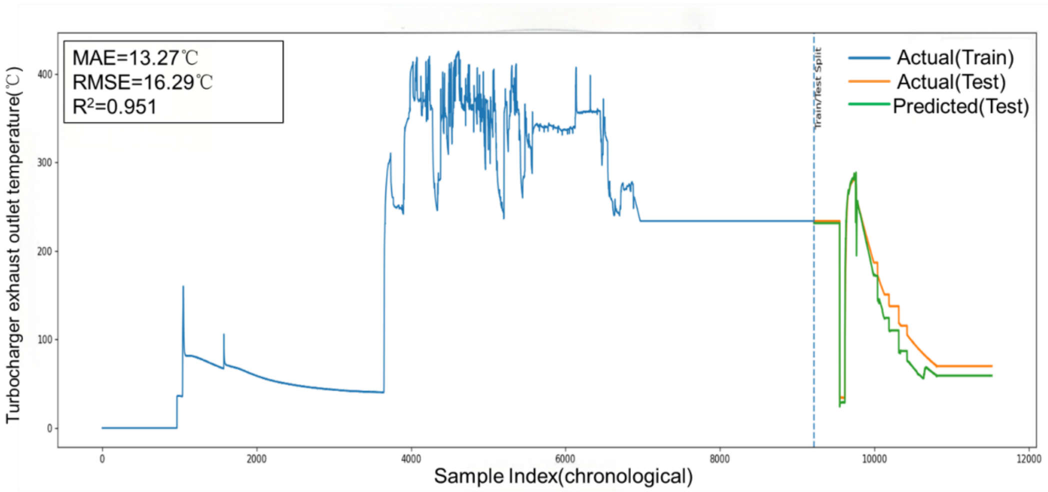

Fig. 6. Prediction results of the LightGBM model

As a gradient-boosted decision tree model, LightGBM iteratively refines prediction errors and therefore has clear advantages in nonlinear modeling. Figure 6 presents the prediction results of the LightGBM model. The experimental results indicate that LightGBM provides stable overall trend fitting and steady-state interval prediction, with RMSE and MAE values substantially lower than those of the linear regression and random forest models.

However, conventional LightGBM still has limitations under transient conditions. During load-surge phases, the prediction curve shows a smoothing effect and cannot fully keep pace with rapid changes in measured temperature. This result indicates that the model structure alone is insufficient for fully capturing the step-response characteristics of the system.

4.1.5 B-LightGBM model

Bayesian optimization for automatic hyperparameter search further improves LightGBM performance. As shown in Figure 7, the B-LightGBM model outperforms the default LightGBM configuration across MAE, RMSE, and R^2, highlighting the importance of optimized parameter combinations for prediction performance.

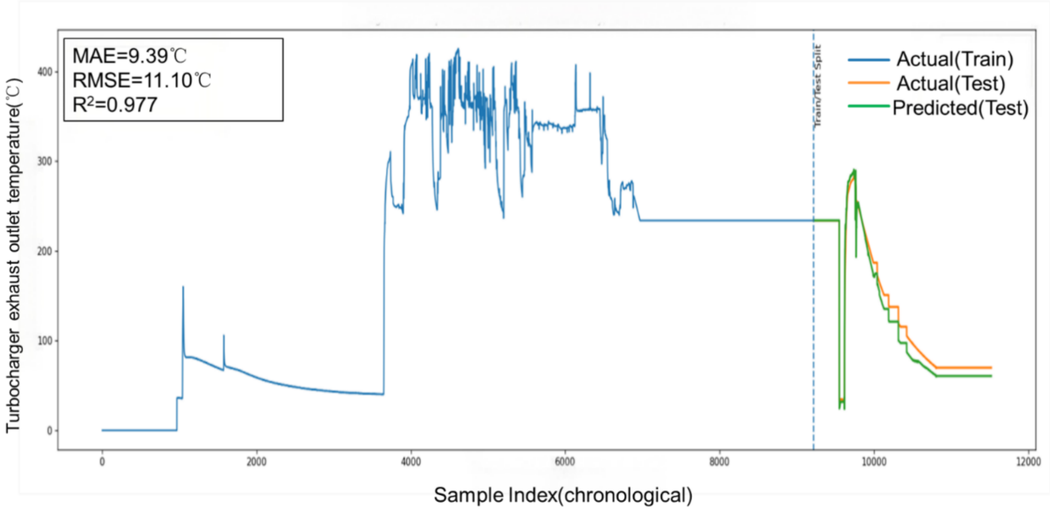

Fig. 7 Prediction results of the B-LightGBM model

Although Bayesian optimization improves overall model accuracy, the improvement is mainly reflected in lower steady-state and overall error levels, with limited reduction of prediction lag under step conditions. This result indicates that hyperparameter optimization alone cannot resolve insufficient dynamic response.

4.1.6 B-SD-LightGBM model

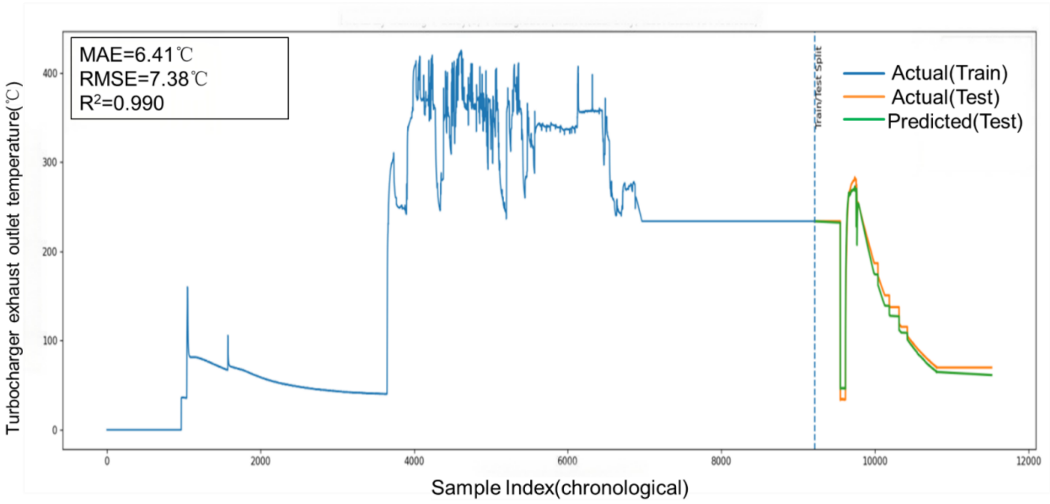

Fig. 8 Prediction results of the B-SD-LightGBM model

Building on B-LightGBM, the B-SD-LightGBM model integrates step-sensitive training, dynamic consistency correction, and residual compensation. Figure 8 presents the prediction results of the proposed B-SD-LightGBM model. The model achieves an MAE of 6.41 °C, an RMSE of 7.38 °C, and an R^2 of 0.990 on the test set, outperforming all comparison models. The prediction curve shows that B-SD-LightGBM accurately fits the steady-state operating range and responds rapidly to temperature step changes induced by load fluctuations, substantially reducing peak errors and lag. These results indicate that step-weighted training strengthens model learning for critical samples, whereas dynamic consistency correction and residual compensation further improve the temporal continuity and physical plausibility of the prediction sequence.

4.1.7 LSTM model

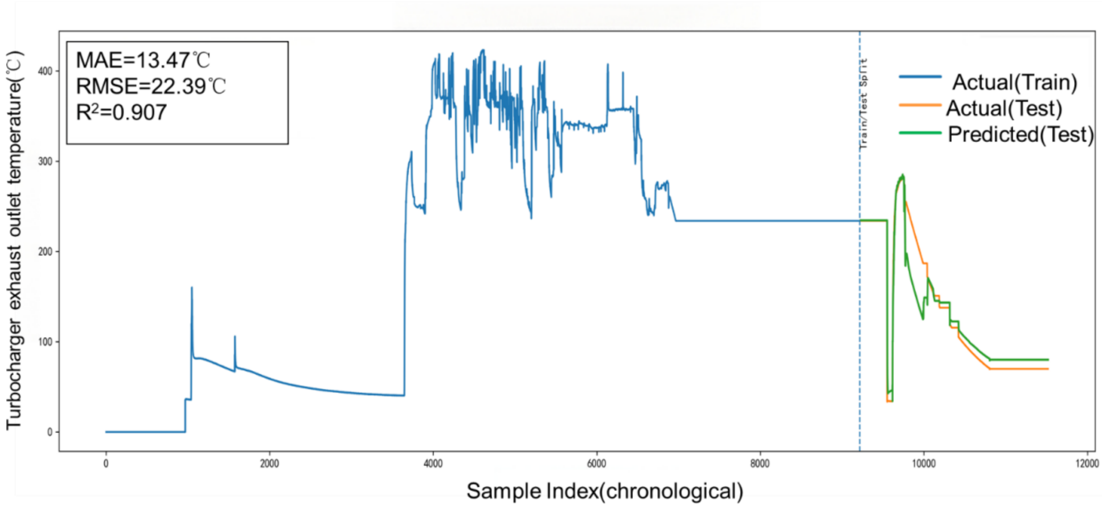

Fig. 9 Prediction results of the LSTM model

The Long Short-Term Memory (LSTM) network can retain long-term dependency information in time series through input, forget, and output gates. This capability makes LSTM suitable for exhaust temperature prediction problems characterized by thermal inertia.

Figure 9 shows the prediction results of the LSTM model. The experimental results indicate that LSTM provides good overall trend fitting and can reflect exhaust outlet temperature variations in medium-to-high load ranges. Compared with traditional static machine learning models, LSTM reduces prediction errors during steady-state operation, indicating an advantage in modeling temporal correlations. However, under step conditions induced by load transients, the LSTM model still exhibits noticeable prediction lag. The prediction curve shows pronounced smoothing during rapid temperature increases or decreases and cannot promptly track abrupt changes in measured values. This behavior is mainly related to the weighted accumulation of historical information through the gating mechanism of LSTM. When sudden changes occur, the model tends to preserve existing state memory, which weakens responsiveness to transient information.

Furthermore, LSTM contains a large number of parameters and is sensitive to training sample size and hyperparameter configuration. These characteristics increase the requirements for computational resources and parameter tuning in engineering applications.

4.1.8 GRU model

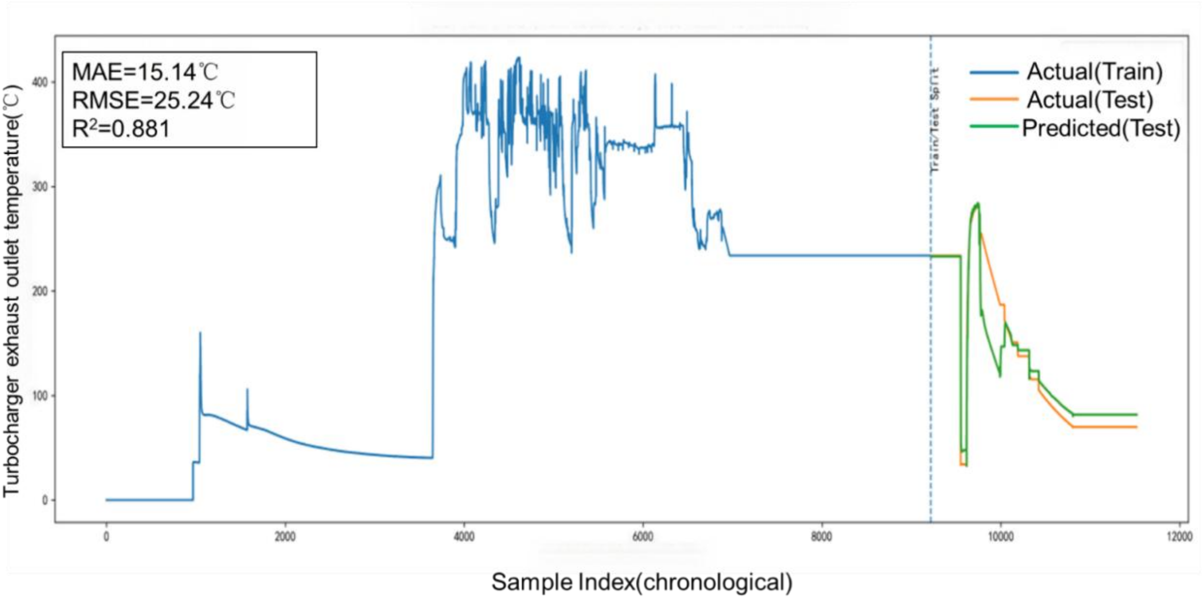

Fig. 10 Prediction results of the GRU model

The Gated Recurrent Unit (GRU) simplifies LSTM by retaining only the update gate and reset gate, which reduces the number of parameters while preserving the ability to model temporal dependencies. Figure 10 presents the prediction results of the GRU model. The experimental results indicate that GRU achieves prediction accuracy comparable to that of LSTM and slightly higher stability under some operating conditions. This result suggests that a more compact GRU structure can reduce overfitting to some extent under limited sample sizes.

However, similar to LSTM, GRU has difficulty accurately capturing rapid temperature changes during step conditions. The predictions show noticeable lag during abrupt transition intervals and local underfitting during sudden temperature drops. Therefore, despite its simpler structure, GRU remains a smoothing-type time-series model with limited sensitivity to sudden operational changes.

4.2 Step condition identification results

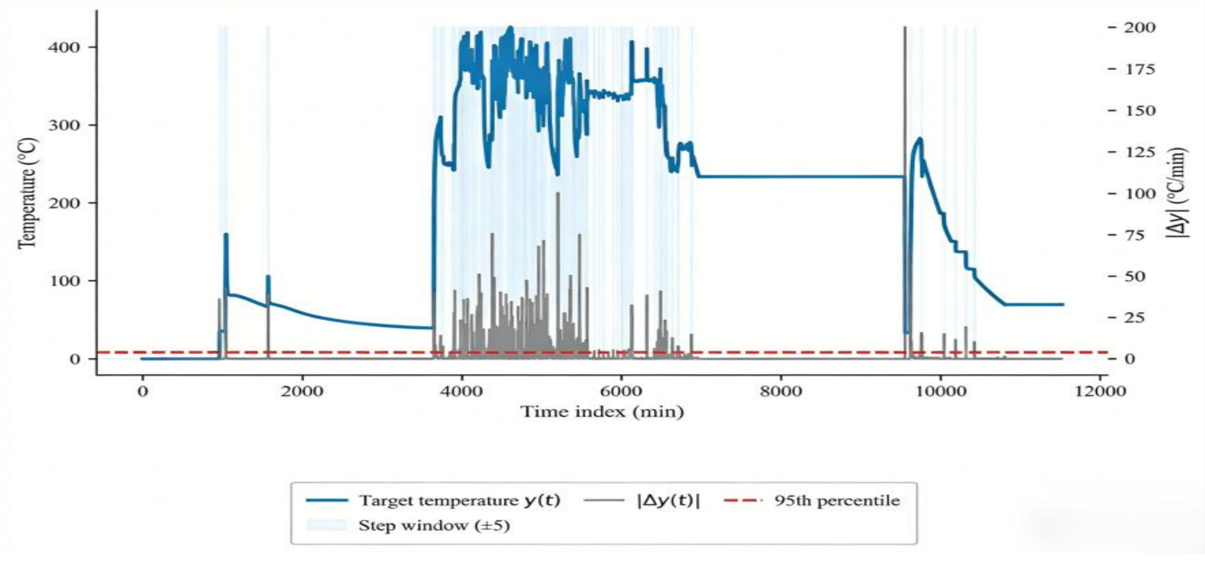

Fig. 11 Step condition prediction results

Figure 11 presents the step condition identification results based on the exhaust outlet temperature variation characteristics. The solid line denotes the original turbocharger exhaust outlet temperature time series y(t), with the right-hand axis representing the absolute value of the first-order temperature difference

$|\triangle y(t)|$. The dashed line indicates the 95th-percentile threshold, whereas the shaded region denotes the identified step window (±5 min). The overall trend reveals distinct multi-condition characteristics in the exhaust outlet temperature throughout the operational cycle. Temperature variations encompass both prolonged steady-state intervals and abrupt transition phases triggered by engine load adjustments or operating mode changes. For example, between 3800 and 6000 min, the engine generally operates under high-load conditions, and the exhaust outlet temperature exhibits frequent fluctuations and localized abrupt changes. At approximately 9800 min, a marked rapid temperature rise followed by a subsequent decay process occurred, reflecting typical load reduction or operating mode switching behavior.

Statistical analysis of $|\triangle y(t)|$ shows that during periods with smooth temperature variation, its values remain at relatively low levels, indicating gradual thermal evolution. These low values are consistent with the expected behavior of thermally inert systems. In contrast, $|\triangle y(t)|$ exhibits pronounced peaks at certain time points, indicating rapid temperature variations. These peaks significantly exceed the 95th-percentile threshold and can therefore be used to identify transient events in the time series. $|\triangle y(t)|$ reflects only the rate of temperature change. The association between these peaks and changes in operating conditions is based on corresponding operational data.

Further examination of the step-window distribution shows that the windows are concentrated mainly during transitional phases of rapid temperature increase or decrease, with almost no step windows occurring within extended steady-state intervals. This distribution indicates that although step conditions account for a small proportion of the time series, they align closely with critical moments of operating-state change. During high-load operation, step windows are densely distributed, reflecting frequent operational adjustments and complex system dynamics. This behavior places higher demands on model responsiveness. Step windows cover not only the instantaneous transition points but also the intervals before and after abrupt changes, which is consistent with the physical characteristic that exhaust temperature cannot stabilize immediately because of thermal inertia. Modeling only a single transition point may overlook dynamic evolution during temperature transitions and may lead to prediction lag or magnitude deviation near the step. Therefore, combining step identification with fixed-width windows provides a more complete characterization of exhaust temperature variation during operating-mode transitions.

To further evaluate the robustness of the proposed method, sensitivity analysis of the percentile threshold and window size K is conducted, as shown in Figure 12. The results indicate that although both parameters influence prediction performance, the model remains relatively stable within a reasonable range. Specifically, increasing the percentile threshold generally increases prediction error because fewer transition regions are detected, whereas larger window sizes help mitigate this effect by incorporating broader temporal context. The configuration using the 95th percentile and K = 5 achieves a favorable balance between sensitivity and stability, supporting the robustness of the selected parameters.

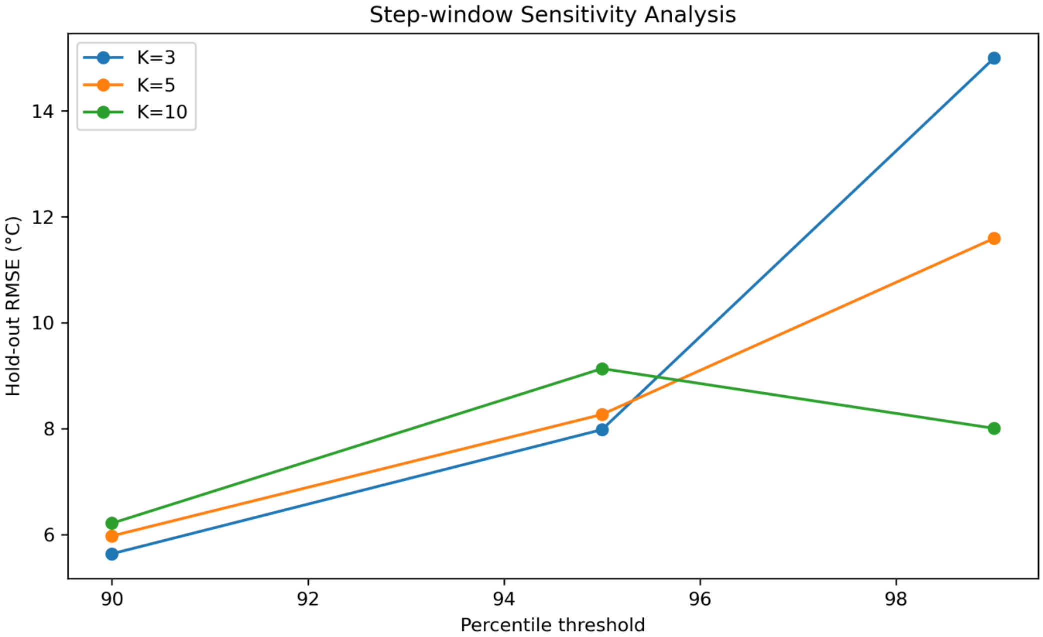

Fig. 12 Sensitivity of prediction performance to percentile threshold and window size K

In summary, the results in Figures 11 and 12 show that the proposed step detection method can identify key transition intervals and distinguish effectively between steady-state and dynamic phases. Together with the sensitivity analysis, these findings support the effectiveness and robustness of the method and provide a foundation for subsequent step-aware weighting and dynamic correction strategies.

Table 6 Comparison of prediction performance across different models

Model	MAE (°C)	RMSE (°C)	R^2
Linear Regression	35.45	45.64	0.614
Support Vector Machine	16.84	23.42	0.898
Random Forest	30.78	44.73	0.628
LightGBM	13.27	16.29	0.951
B-LightGBM	9.39	11.10	0.977
B-SD-LightGBM	6.41	7.38	0.990
LSTM	13.47	22.39	0.907
GRU	15.14	25.24	0.881

As shown in Table 6, predictive performance gradually improves as model complexity and targeted modeling strategies increase. Linear models and traditional machine learning methods struggle to meet forecasting requirements under complex operating conditions. Gradient boosting and Bayesian optimization substantially improve model performance. Further integration of step-sensitive training and dynamic consistency correction improves prediction accuracy under transient conditions while maintaining overall precision.

4.3 Discussion

The experimental results show clear performance differences among the evaluated models in predicting turbocharger exhaust outlet temperature. Traditional methods, including linear regression and support vector machines, have limited ability to capture nonlinear and dynamic behavior in complex thermal systems, resulting in relatively low predictive accuracy.

Tree-based models, particularly LightGBM, show strong capability for modeling nonlinear relationships. Even the non-optimized LightGBM model achieves performance comparable to or slightly better than that of the baseline deep learning models (LSTM and GRU), suggesting that tree-based ensemble methods are effective for this type of data. Bayesian hyperparameter optimization and temporal feature expansion further improve model performance.

The proposed B-SD-LightGBM should be interpreted as a fully optimized prediction framework that includes extensive feature engineering, step-sensitive weighting, and post-processing correction mechanisms. By contrast, the deep learning models are implemented as baseline configurations. Therefore, the comparison reflects the performance of a fully optimized tree-based pipeline relative to baseline deep learning implementations.

From a dynamic perspective, the main performance gains arise from explicit treatment of transient behavior. Multiscale temporal features help the model capture thermal inertia and lag effects, while dynamic consistency correction and residual compensation reduce systematic errors during step-change conditions. These mechanisms improve peak prediction accuracy and temporal continuity.

The model still has limitations under some operating conditions. Prediction errors tend to increase in operating regimes that are underrepresented in the training data, such as extreme load transitions or rare transient events. This result suggests that model performance is influenced by the distribution of training data and highlights the need for more diverse datasets.

From an engineering perspective, the proposed method offers practical advantages. Compared with the baseline deep learning models under the current experimental setup, the framework has lower computational complexity. In addition, its modular structure allows incremental updates and integration into existing monitoring pipelines.

Nevertheless, several limitations remain. The model relies on manually defined parameters, such as step-window size and temporal feature configurations, which may affect robustness under different operating conditions. Furthermore, the study uses data from a single engine and vessel, which limits generalizability to other systems.

5. Conclusions

This study proposes a B-SD-LightGBM framework for predicting turbocharger exhaust outlet temperature under complex and transient operating conditions. The method integrates multiscale temporal feature engineering, step-sensitive weighting, dynamic consistency correction, and residual compensation to improve prediction accuracy and temporal stability.

The results demonstrate that the proposed framework captures the nonlinear and dynamic characteristics of exhaust temperature evolution, particularly under step-change conditions. The observed improvements are not attributable solely to model selection but arise from the integrated design of the prediction pipeline, including feature engineering, optimization strategies, and post-processing mechanisms.

The model achieves strong predictive performance on the test set, with R^2 = 0.990 and RMSE = 7.38 °C. Compared with the best-performing baseline model, the proposed approach reduces RMSE by approximately 33 %.

Several limitations should be acknowledged. First, the model is developed and validated using data from a single engine and vessel, which may limit generalizability. Second, the dataset size is relatively limited. Third, key parameters, including the percentile threshold, step-window size K, and temporal feature spans, are manually defined, which may affect robustness under varying operating conditions. In addition, cross-engine validation has not been conducted, restricting assessment of model transferability.

Future work will address these limitations more directly. Adaptive parameter selection methods will be developed to reduce reliance on manual tuning; cross-engine and multi-vessel datasets will be used to evaluate generalization performance; physics-informed constraints will be incorporated to improve robustness under unseen conditions; and lightweight online updating strategies will be explored to enable real-time deployment and continuous model adaptation.

FUNDING

This research was funded by China Classification Society through the project “Research on Verification Methods for Digital Inspection Systems”.

REFERENCES

[1] Su, S., Miao, Z., Zhao, Y., Song, N., 2024. Digitally twin driven ship cooling pump fault monitoring system and application case. *Brodogradnja*, 75(4), 75403. https://doi.org/10.21278/brod75403

[2] Tan, Y., Tian, H., Jiang, R., Lin, Y., Zhang, J., 2020. A comparative investigation of data-driven approaches based on one-class classifiers for condition monitoring of marine machinery system. *Ocean Engineering*, 201, 107174. https://doi.org/10.1016/j.oceaneng.2020.107174

[3] Inal, O., Charpentier, J., Deniz, C., 2022. Hybrid power and propulsion systems for ships: Current status and future challenges. *Renewable & Sustainable Energy Reviews*, 156, 111965. https://doi.org/10.1016/j.rser.2021.111965

[4] Han, Z., Deng, B., Wu, X., Tian, W., Fang, J., Wu, Y., et al., 2025. Analysis of energy utilization and thermal efficiency in diesel engines under varying EGR rates. *Energy*, 319, 135029. https://doi.org/10.1016/j.energy.2025.135029

[5] Hu, Z., Fan, A., Mao, W., Shu, Y., Wang, Y., Xia, M., et al., 2025. Ship energy consumption prediction: Multi-model fusion methods and multi-dimensional performance evaluation. *Ocean Engineering*, 322, 120538. https://doi.org/10.1016/j.oceaneng.2025.120538

[6] Zhang, Y., Liu, P., He, X., Jiang, Y., 2020. A prediction method for exhaust gas temperature of marine diesel engine based on LSTM. *2020 IEEE 2nd International Conference on Civil Aviation Safety and Information Technology* (ICCASIT). 14-16 October, Weihai, China. https://doi.org/10.1109/ICCASIT50869.2020.9368826

[7] Ji, Z., Gan, H., Liu, B., 2023. A Deep Learning-Based Fault Warning Model for Exhaust Temperature Prediction and Fault Warning of Marine Diesel Engine. *Journal of Marine Science and Engineering*, 11(8),1509 https://doi.org/10.3390/jmse11081509

[8] Dutta, S., Khan, M., Hoque, M., Jin, Y., Alam, F., Trinuruk, P., 2025. Sustainable diesel engine performance enhancement using pyrolytic plastic oil blends: experimental investigation and artificial neural network-based prediction. *Energy*, 333, 137469. https://doi.org/10.1016/j.energy.2025.137469

[9] Sun, J., Zeng, H., Ye, K., 2024. Short-Term Exhaust Gas Temperature Trend Prediction of a Marine Diesel Engine Based on an Improved Slime Mold Algorithm-Optimized Bidirectional Long Short-Term Memory-Temporal Pattern Attention Ensemble Model. *Journal of Marine Science and Engineering*, 12(4), 24. https://doi.org/10.3390/jmse12040541

[10] Gao, C., Sun, S., Xie, L., Huang, D., 2023. Study on Exhaust Gas Temperature Prediction Model Based on GBDT. *35th Chinese Control and Decision Conference* (CCDC), 20-22 May, Yichang, China. https://doi.org/10.1109/ccdc58219.2023.10327412

[11] Liu, J., Huang, Q., Ulishney, C., Dumitrescu, C. E., 2021. Machine learning assisted prediction of exhaust gas temperature of a heavy-duty natural gas spark ignition engine. *Applied Energy*,300, 117413. https://doi.org/10.1016/j.apenergy.2021.117413

[12] Cesur, I., Uysal, F., 2024. Experimental investigation and artificial neural network-based modelling of thermal barrier engine performance and exhaust emissions for methanol-gasoline blends. *Energy*, 291, 130393. https://doi.org/10.1016/j.energy.2024.130393

[13] Bartlett, P., Long, P., Lugosi, G., Tsigler, A., 2020. Benign overfitting in linear regression. *Proceedings of the National Academy of Sciences of the United States of America*, 117(48), 30063-30070. https://doi.org/10.1073/pnas.1907378117

[14] Etemadi, S., Khashei, M., 2021. Etemadi multiple linear regression. *Measurement*, 110080. https://doi.org/10.1016/j.measurement.2021.110080

[15] Vos, K., Peng, Z., Jenkins, C., Shahriar, M., Borghesani, P., Wang, W., 2022. Vibration-based anomaly detection using LSTM/SVM approaches. *Mechanical Systems and Signal Processing*, 169, 108752. https://doi.org/10.1016/j.ymssp.2021.108752

[16] Malashin, I., Tynchenko, V., Gantimurov, A., Nelyub, V., Borodulin, A., 2025. Support Vector Machines in Polymer Science: A Review. *Polymers*, 17(4), 491. https://doi.org/10.3390/polym17040491

[17] Borup, D., Christensen, B., Mühlbach, N., Nielsen, M., 2023. Targeting predictors in random forest regression. *International Journal of Forecasting*, 39(2), 841-868. https://doi.org/10.1016/j.ijforecast.2022.02.010

[18] Sun, Z., Wang, G., Li, P., Wang, H., Zhang, M., Liang, X., 2024. An improved random forest based on the classification accuracy and correlation measurement of decision trees. *Expert Systems with Applications*, 237(B), 121549. https://doi.org/10.1016/j.eswa.2023.121549

[19] Bentéjac, C., Csörgo, A., Martínez-Muñoz, G., 2021. A comparative analysis of gradient boosting algorithms. *Artificial Intelligence Review*, 54(3), 1937-1967. https://doi.org/10.1007/s10462-020-09896-5

[20] Ke, G. L., Meng, Q., Finley, T., Wang, T. F., Chen, W., Ma, W.D., et al., 2017. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. *31st Annual Conference on Neural Information Processing Systems (NIPS)*, 04-09 December, Long Beach, California, USA. https://dl.acm.org/doi/10.5555/3294996.3295074

[21] Gong, J., Ni, P., Fu, Z., Qin, Y., 2026. Study on the prediction performance of ship motion in waves by LSTM under missing data. *Brodogradnja*, 77(3), 77311. https://doi.org/10.21278/brod77311

[22] Cao, Y., Zhang, J., Ma, A., Xu, H., Liu, J., 2026. Marine engine cylinder exhaust temperature prediction based on PSO-optimized CNN-LSTM-attention network. *Brodogradnja*, 77(1), 77101. https://doi.org/10.21278/brod77101

[23] Lawrynczuk, M., Zarzycki, K., 2025. LSTM and GRU type recurrent neural networks in model predictive control: A Review. *Neurocomputing*, 632, 129712. https://doi.org/10.1016/j.neucom.2025.129712

[24] Meng, L., Gan, H., Liu, H., Lu, D., 2025. Deep learning-based research on fault warning for marine dual fuel engines. *Brodogradnja*, 76(3), 76303. https://doi.org/10.21278/brod76303

[25] Thurston, M., Sullivan, M., McConky, S., 2023. Exhaust-gas temperature model and prognostic feature for diesel engines. *Applied Thermal Engineering*, 229, 120578. https://doi.org/10.1016/j.applthermaleng.2023.120578

[26] Sekar, M., 2026. Machine learning based fault detection and classification for predictive maintenance of gas turbine engines: a comprehensive benchmarking analysis on various models. *Aircraft Engineering and Aerospace Technology*, 98(2), 313-324. https://doi.org/10.1108/aeat-04-2025-0160

[27] Hoang, A., Bui, T., Nguyen, X., Bui, V., Nguyen, Q., Truong, T., Chung, N., 2025. Explainable machine learning-based prediction of fuel consumption in ship main engines using operational data. *Brodogradnja*, 76(4), 76405. https://doi.org/10.21278/brod76405

[28] Gupta, P., Rasheed, A., Steen, S., 2022. Ship performance monitoring using machine-learning. *Ocean Engineering*, 254, 111094. https://doi.org/10.1016/j.oceaneng.2022.111094

[29] Karagiannidis, P., Themelis, N., 2021. Data-driven modelling of ship propulsion and the effect of data pre-processing on the prediction of ship fuel consumption and speed loss. *Ocean Engineering*, 222, 108616. https://doi.org/10.1016/j.oceaneng.2021.108616

[30] Liu, W., Cao, Y., 2025. Research on Offshore Vessel Trajectory Prediction Based on PSO-CNN-RGRU-Attention. *Applied Sciences*, 15(7), 19. https://doi.org/10.3390/app15073625

[31] Ma, A., Zhang, J., Shen, H., Cao, Y., Xu, H., Liu, J., 2025. Research on Fault Diagnosis of Marine Diesel Engines Based on CNN-TCN-ATTENTION. *Applied Sciences*, 15(3), 24. https://doi.org/10.3390/app15031651

[32] Selmy, H., Mohamed, H., Medhat, W., 2024. Big data analytics deep learning techniques and applications: A survey. *Information Systems*, 120, 102318. https://doi.org/10.1016/j.is.2023.102318

[33] Bergstra, J., Bengio, Y., 2012. Random Search for Hyper-Parameter Optimization. *Journal of Machine Learning Research*, 13, 281-305. https://dl.acm.org/doi/10.5555/2188385.2188395