Sensors and Materials, Vol. 38, No. 8 (2026) 4697–4718
MYU Tokyo

S & M 4589

Sensor Data Processing for Wind Power Forecasting Based on Bidirectional Long Short-term Memory Network Optimized with *K*-dimensional Tree, Density-based Spatial Clustering, and Robust Iterative Multi-objective Engine

Jing Wang,[1*] HongXin Hu,[2] Yue Chu,[3] Xinyun Xia,[4] Xiongfei Wei,[5]
Yi Ruan,[1] Yuanjie Fang,[1] Hao Cong,[1] and Chuanliang Cheng[1]

[1]School of Electronic Information Engineering, Chaohu University, Hefei 238024, China
[2]School of Electrical Engineering, Anhui Polytechnic University, Wuhu 241000, China
[3]Wuwei Municipal Power Supply Company, State Grid Anhui Electric Power Co., Ltd., Wuhu 241000, China
[4]Anhui Huadian Engineering Consulting and Design Co., Ltd., Hefei 230600, China
[5]Wugou Coal Mine of Anhui Hengyuan Coal and Electricity Co., Ltd., Huaibei 235131, China

(Received May 11, 2026; accepted August 20, 2026)

Keywords: wind power forecasting, DBSCAN, KD-Tree, RIME, BiLSTM

Accurate wind power forecasting is vital for grid stability, but sensor data are frequently compromised by environmental degradation and hardware anomalies. To address this limitation, we developed an intelligent forecasting model that enhances sensor reliability by integrating *K*-dimensional tree-accelerated density-based spatial clustering of applications with noise (KD-DBSCAN) with a bidirectional long short-term memory (BiLSTM) network optimized by a rime optimization algorithm (RIME). KD-DBSCAN reduces spatial search complexity to $O(N\log N)$ and purifies multi-dimensional sensor data. The purified data feed into the RIME-BiLSTM architecture. Tested on two commercial supervisory control and data acquisition (SCADA) datasets, KD-DBSCAN anomaly filtering improved the coefficient of determination (R^2) by up to 48.4%, reducing forecasting errors by 39.6% (inland commercial wind farms) and 32.8% (coastal wind farms). The optimized RIME-BiLSTM showed an R^2 of 0.9876 and reduced the mean absolute error by up to 48.9%. Beyond soft-sensing, isolating physical anomaly signatures, such as sensor drift and signal dropout, provides diagnostic metrics to guide the development of novel physical sensors, such as solid-state ultrasonic wind transducers with anti-icing hydrophobic coatings and self-calibrating microsensor modules. Despite these results, reliance on offline batch processing poses latency constraints for streaming SCADA. Therefore, an online incremental KD-DBSCAN framework must be developed to be paired with adaptive continual learning for real-time edge deployment.

*Corresponding author: e-mail: 052046@chu.edu.cn
https://doi.org/10.18494/SAM6424

ISSN 0914-4935

1. Introduction

Global reserves of renewable energy, including wind, solar, and hydroelectric power, offer a sustainable solution to the depletion of fossil fuels and the increasing global energy demand.(1) Among the sources of renewable energy, wind power generation has been widely adopted owing to its environmental benefits and cost-effectiveness.(2) However, the inherent variability, intermittency, and stochastic nature of wind speed significantly affect the wind power generation process and outcomes.(3) Such challenges hinder accurate wind power forecasting, posing substantial challenges to grid integration, load dispatch, and efficient resource utilization.(4)

To overcome these challenges, robust forecasting frameworks have been developed, among which physical methods are predominantly used.(5) In physical methods, meteorological and topological data are used to model the power conversion process in wind farms.(6) Although these models are advantageous for wind farms where historical records are lacking,(7) they are computationally intensive and require high-quality physical parameters, making them more suitable for medium- to long-term forecasting applications.(8) In physical methods, statistical methods are mainly used to map historical observations and power outputs. Traditional mapping methods include time-series analysis,(9) such as autoregressive integrated moving average(10) and Kalman filtering.(11) However, linear models employed in traditional mapping methods struggle to accurately capture the nonstationary characteristics of wind data.(12)

In modern wind energy conversion systems, the reliability of forecasting models is strictly dictated by the fidelity of the physical sensor layer deployed.(13) Sensors, such as optoelectronic cup anemometers, wind vanes, temperature sensors, and power transducers, are mounted on wind turbine nacelles and meteorological towers to monitor environmental and turbine operational parameters. However, sensors deployed in harsh or remote outdoor environments are vulnerable to severe environmental stressors, including mechanical wear, ice accretion, dirt accumulation, temperature fluctuations, and electromagnetic interference, leading to signal drift, intermittent noise, and anomalous readings.(14) Although robust algorithms are used to filter stochastic noise in soft computing layers, wind power forecasting is not possible without sensor technology.(15) Therefore, analyzing physical sensor data is essential in enhancing model accuracy and providing vital feedback for engineering novel, highly resilient physical sensing devices and materials capable of surviving extreme operational environments.

Such a data-to-decision pipeline requires advanced, data-driven approaches. With the rapid advancement of AI technology, machine learning and deep learning (DL) models have been adopted in forecasting methods beyond conventional statistical techniques. Specifically, shallow architectures with fewer computational layers, such as multilayer perceptrons,(16) support vector machines (SVMs),(17) and extreme learning machines(18) have been widely adopted. However, these models often lack the nonlinear fitting capacity that is required to capture deep features from high-dimensional datasets. In contrast, deep DL architectures, including long short-term memory (LSTM) networks,(19) gated recurrent units,(20) and Transformers(21) exhibit better capabilities than shallow architectures in extracting intricate temporal dependencies. Among the models, bidirectional LSTM (BiLSTM) models have demonstrated their effectiveness in ultrashort-term forecasting by capturing bidirectional temporal dynamics of wind.(22)

Nevertheless, DL models are highly sensitive to data quality, since anomalies in the training dataset significantly degrade the forecasting reliability of models.

Therefore, hybrid forecasting methods are used to enhance accuracy, leveraging the strengths of individual models while mitigating their inherent limitations.(23) In these methods, the whale optimization algorithm (WOA)(24) or the sparrow search algorithm(25) is used to tune model hyperparameters or decompose complex signals and reduce forecasting error. However, in these algorithms, outlier detection remains a problem to solve. Although image-segmentation-based detection methods, such as the quartile methods, are promising to solve the problem,(26) they are also computationally complex for real-time applications.(27)

To improve the precision of hybrid wind power forecasting, we integrate the rime optimization algorithm (RIME) and an enhanced anomaly detection module into a BiLSTM architecture. The RIME algorithm is employed to systematically fine-tune the hyperparameters of the BiLSTM model, ensuring an optimal balance between learning efficiency and predictive generalization. To ensure data quality, the density-based spatial clustering of applications with noise (DBSCAN) algorithm is utilized in conjunction with a *K*-dimensional tree (KD-Tree) structure. This KD-DBSCAN approach significantly enhances the efficiency of neighbor-point queries, and the silhouette score is used to evaluate clustering quality and facilitate the optimal selection of the neighborhood radius, epsilon (Eps), and minimum points (MinPts). By effectively identifying and mitigating the influence of outliers, the framework developed in this study ensures high-fidelity data input, significantly enhancing the overall forecasting accuracy and robustness of the model.

2. Model Construction

The framework developed in this study serves as an intelligent raw sensor data processing unit. The KD-DBSCAN module acts as a front-end signal conditioner, identifying sensor malfunctions or communication errors represented as outliers. Subsequently, the RIME-BiLSTM model functions as a high-level feature extractor, modeling the complex, nonlinear dependencies inherent in multi-sensor wind datasets.

2.1 BiLSTM

LSTM networks are inherently unidirectional, meaning the hidden state at a specific time step is governed solely by preceding historical information.(28) Consequently, they lack the capacity to incorporate future contextual dependencies within a sequence. To address this, a BiLSTM architecture is used to introduce a dual-layer mechanism consisting of forward and backward propagation layers.(21) This enables the network to capture dependencies from past and future states at a specific relative time point, enhancing the complex temporal dynamics of the model. In the architecture of the BiLSTM (Fig. 1), at each time step t, the inputs X_t are simultaneously processed by forward and backward hidden layers. The forward layer calculates the state $\vec{h}_t$ on the basis of the sequence from $t - 1$ to t, whereas the backward layer calculates $\overleftarrow{h}_t$ by processing the sequence in reverse from $t + 1$ to t.

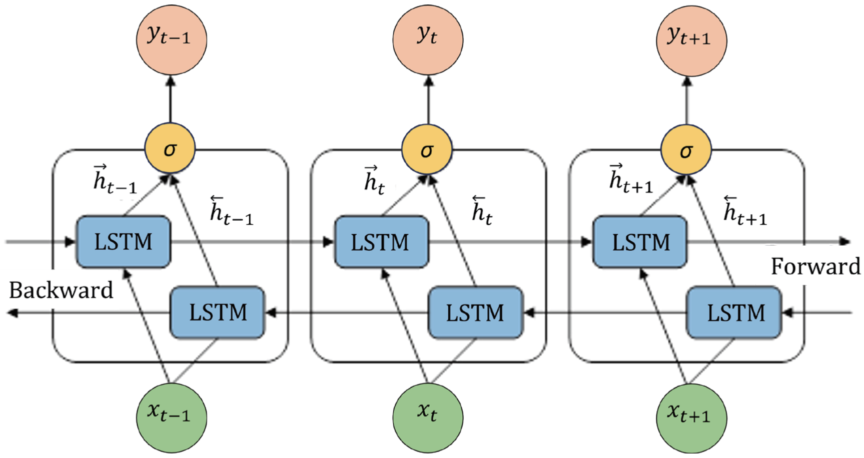


Fig. 1. (Color online) Structure of BiLSTM.

$$
\begin{aligned}
\vec{h}_t &= \sigma\left(W_{x\vec{h}} X_t + W_{\vec{h}\vec{h}} \vec{h}_{t-1} + b_{\vec{h}}\right) \\
\overleftarrow{h}_t &= \sigma\left(W_{x\overleftarrow{h}} X_t + W_{\overleftarrow{h}\overleftarrow{h}} \overleftarrow{h}_{t-1} + b_{\overleftarrow{h}}\right)
\end{aligned}
\tag{1}
$$

Here, X_t is the input vector at time t, $\vec{h}_t$ and $\overleftarrow{h}_t$ are respectively the hidden state vectors for the forward and backward layers, $W_{x\vec{h}}$ and $W_{x\overleftarrow{h}}$ are the weight matrices connecting the input to the hidden layers, $W_{\vec{h}\vec{h}}$ and $W_{\overleftarrow{h}\overleftarrow{h}}$ are the recurrent weight matrices for the hidden-to-hidden transitions, $b_{\vec{h}}$ and $b_{\overleftarrow{h}}$ are the bias vectors for each layer, and σ is the nonlinear activation function (typically *tanh* or rectified linear unit). The final output of the BiLSTM network, y_t, is generated by concatenating or summing the hidden states from both directions, as expressed in Eq. (2).

$$
y_t = g\left(W_{\vec{h}y} h_t + W_{\overleftarrow{h}y} + b_y\right) \tag{2}
$$

Where, $W_{\vec{h}y} h_t$ and $W_{\overleftarrow{h}y}$ represent the weight matrices mapping the bidirectional hidden states to the output layer, b_y is the bias term, and g denotes the output activation function. By integrating information from both temporal directions, BiLSTM provides a comprehensive representation of power waveform fluctuations compared with standard LSTM models.

2.2 RIME

RIME is a physics-inspired metaheuristic optimization algorithm characterized by rapid convergence and high precision of solutions. By adaptively modifying search positions during the iterative process, RIME enhances global exploration capabilities while effectively circumventing local optima. RIME significantly outperforms traditional metaheuristics, such as particle swarm optimization (PSO) and the WOA, across various complex optimization scenarios.[29] The execution of the RIME algorithm is structured into the following four operational phases.

(1) Initialization phase

The RIME algorithm begins by defining the population size and initializing control parameters. A population of RIME crystals (potential solutions) is randomly generated within the predefined search space. The initial fitness value for each particle is then computed on the basis of the objective function.

(2) RIME search phase

In this phase, the physical process of soft condensation is simulated. In the simulation, the position of each RIME particle is updated according to the following mechanism.

$$R_{i,j}^{new} = \begin{cases} R_{best,j} + r_1 \cdot \cos(\theta) \cdot \beta \cdot \left(Ub_j - Lb_j\right), & \text{if } r_2 < E \\ R_{best,j}, & \text{if } r_2 < E \end{cases} \tag{3}$$

Here, $R_{i,j}^{new}$ represents the updated position of the jth dimension of the ith particle, $R_{best,j}$ corresponds to the jth dimension of the optimal crystal in the current population, r_1 and r_2 are random variables uniformly distributed in the interval [0, 1], Ub_j and Lb_j respectively denote the upper and lower bounds of the jth dimension, and E is the attachment factor, which dictates the probability of a position update and is defined as follows.

$$E = \sqrt{\frac{t}{T_{max}}} \tag{4}$$

Here, t denotes the current iteration, and T_{max} is the maximum allowable iteration. The environmental factor β is governed by a step function,

$$\beta = round\left(\frac{T}{T_{max}} \cdot \omega\right) \cdot \frac{1}{\omega}, \tag{5}$$

where the default value for the parameter ω is typically set to 5.

(3) Hard RIME permutation phase

To refine the search and maintain population diversity, a hard condensation simulation is applied using the following permutation equation.

$$R_{i,j}^{new} = R_{best,j} \cdot \left(1 + r_3 \cdot f_{norm,j}\right) \tag{6}$$

Here, r_3 is a random variable in the range [−1, 1], and $f_{norm,j}$ represents the normalized fitness value of the ith crystal. This ensures that particles with better fitness exert a more focused effect on the search trajectory.

(4) Positive greedy selection phase

In the final stage, a greedy selection mechanism is implemented. The fitness value of the updated position $R_{i,j}^{new}$ is compared against the fitness of the previous position. The algorithm retains the updated solution only if it yields superior performance, thereby ensuring monotonic improvement in the population's overall fitness.

2.3 KD-DBSCAN: intelligent sensor data preprocessing

Although the RIME-optimized BiLSTM architecture handles complex wind forecasting, the accuracy of such soft sensor models is largely dependent on the integrity of the input signals. In practical sensor networks, raw data are corrupted by noise or outliers owing to hardware instability or environmental interference. In addition, high-precision wind power forecasting using deep neural networks such as BiLSTM relies on the signal integrity of input data. Sensor data are prone to signal dropouts, calibration drift, and transient noise. Therefore, to construct a lean, high-fidelity dataset, we employed KD-DBSCAN. Although conventional DBSCAN efficiently isolates sparse spatial noise and arbitrary nonlinear cluster boundaries without prior assumptions on cluster numbers, its standard pairwise distance calculation incurs a high time complexity of $O(N^2)$, limiting its scalability on large supervisory control and data acquisition (SCADA) datasets.(13) $O(N^2)$ complexity is a measure of time complexity that describes an algorithm whose performance is directly proportional to the square of the size of the input data. It is commonly referred to as quadratic time.

By integrating KD-Tree spatial indexing into the ε-neighborhood search phase,(14) the spatial search space can be recursively partitioned along orthogonal hyperplanes. We can recursively partition the spatial search space along orthogonal hyperplanes. This reduces the time complexity of range queries from $O(N^2)$ to $O(N \log N)$. $O(N \log N)$ denotes quasilinear time complexity, whose growth rate is faster than that of linear time complexity $O(N)$ but slower than that of quadratic time complexity $O(N^2)$. As the input size N increases, the total number of operations grows proportionally to $N \log N$. In this study, multi-dimensional sensor data, including wind speed, wind direction, ambient temperature, air density, and active power, are mapped into a normalized feature space. Points lying in high-density regions are retained as true operational states, whereas low-density noise points corresponding to sensor faults or curtailments are automatically isolated and filtered before model training. In this study, KD-DBSCAN is used to identify anomalous sensor observations by grouping high-density data points into clusters and isolating low-density noise. The following definitions characterize such density-based data grouping.

(1) Definition 1 (Eps-neighborhood): For a sensor data point M, the Eps-neighborhood denotes the hyperspace within a radius Eps centered at M.

(2) Definition 2 (core point): The point M becomes a core point if its Eps-neighborhood contains at least MinPts observations, representing a stable signal region.

(3) Definition 3 (direct density reachability): A point N becomes directly density-reachable from a core point M if N lies within the Eps radius of M.

(4) Definition 4 (density reachable): Points M and N are density-reachable if there exists a chain of points connecting them through successive direct density-reachability. A point N is density-reachable from a point M if there exists a chain of core points $p_1, p_2, \ldots, p_n$ such that $p_1 = M$, $p_n = N$, and each p_{i+1} is directly density-reachable from p_i. This chain rule characterizes the transitive closure of density reachability, allowing the algorithm to follow a trail of valid sensor data and aggregate clusters of arbitrary geometric shapes.

(5) Definition 5 (density connected): Points M and N are density-connected if there exists an intermediate point P from which both are density-reachable.

Standard DBSCAN requires $O(N^2)$ complexity, which is often prohibitive for high-frequency sensor streams.

To optimize the neighbor search process, we use a KD-Tree, a hierarchical binary search structure that partitions the sensor data space by alternating dimensions. The construction process (Fig. 2) is conducted to organize multi-dimensional features from sensor data into a balanced tree, reducing the search complexity to $O(N\log N)$.

The neighbor-point query is optimized by integrating the KD-Tree into the clustering framework. The process initiates from the root node, comparing the sensor node values with the predefined Eps range to prune irrelevant branches. The Euclidean distance is calculated using Eq. (7) to verify point proximity within the multi-dimensional feature space.

$$d(x,y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \tag{7}$$

Here, $d(x, y)$ is the Euclidean distance between two data points, x and y, in a multi-dimensional feature space. In the developed framework, the Euclidean distance is compared with the Eps

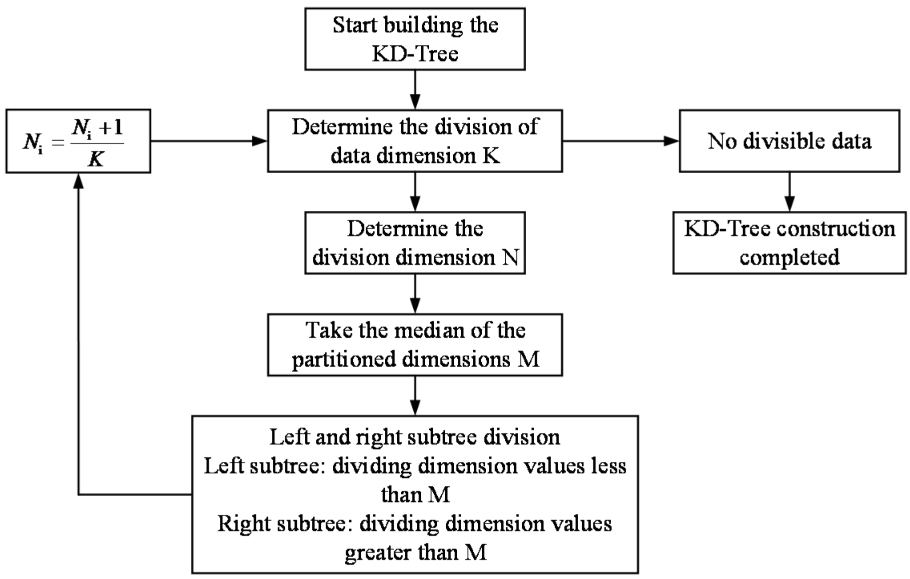


Fig. 2. KD-Tree construction process.

threshold to decide if y is a neighbor of x. x and y are sensor data vectors representing two observations captured by the sensor network. In wind power generation, these vectors contain multiple features (dimensions) such as wind speed, rotor speed in rotations per minute, and temperature at a specific timestamp. n denotes dimensionality, indicating the total number of input features (dimensions) being analyzed. For example, when clustering data based only on wind speed and power output, $n = 2$.

As shown in Fig. 3, the KD-Tree structure guides the DBSCAN algorithm to efficiently traverse only those nodes to satisfy the MinPts density threshold. This optimized search enables real-time anomaly detection in large-scale sensor deployments, ensuring that only representative signal features are passed to the RIME-BiLSTM forecasting model.

To eliminate the subjectivity of manual parameter tuning in varying sensor environments, the silhouette score (S) is applied to Eps and MinPts. The score, a robust metric for evaluating clustering cohesion and separation, is calculated as

$$\begin{cases} S = \dfrac{1}{N}\sum_{i=1}^{N} s(i), \\ s(i) = \dfrac{b-a}{\max(a,b)}, \end{cases} \tag{8}$$

where a denotes the cohesion, which is the average distance between a sensor data point and all other points in the same cluster, and b denotes the separation, which is the minimum average distance between the data point and points in the nearest neighboring cluster. Optimal sensor data clustering can be achieved when the intracluster distance a is minimized, indicating high signal stability, and the intercluster distance b is maximized, ensuring clear separation between valid operational data and stochastic sensor noise. The value of S ranges from −1 to 1. A score near 1 indicates well-separated and compact clusters, whereas a negative score indicates possible cluster misassignment or inappropriate parameter settings. The algorithm to select the optimal epsilon and minPts is created as follows.[30]

2.4 RIME-BiLSTM forecasting model

The developed model integrates anomaly detection with a RIME-optimized BiLSTM network, as shown in Fig. 4. The model was constructed in three phases: data purification, parameter optimization, and predictive modeling.

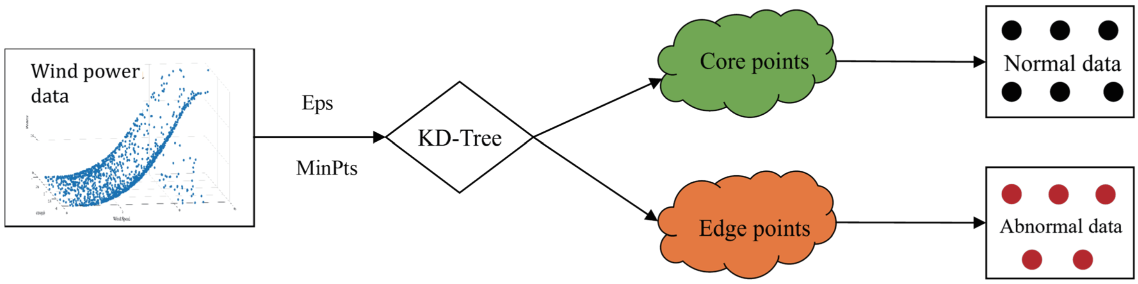


Fig. 3. (Color online) KD-Tree-improved DBSCAN structure.

Algorithm 1
Adaptive parameter selection for KD-DBSCAN.

```
Input: Sensor data X; epsilon range [a, b] with step c; MinPts range [d, e]
Output: Optimal epsilon_best, MinPts_best, and cluster labels L_best
Initialize:
  epsilon_values = {a, a + c, a + 2c, ..., b}
  MinPts_values = {d, d + 1, ..., e}
  S_best = −1
  L_best = ∅
For each epsilon in epsilon_values, do
  For each MinPts in MinPts_values, do
    [L, Noise] ← Perform_KD_DBSCAN(X, epsilon, MinPts)
    If number of clusters in L > 1, then
      S_curr ← Calculate_Silhouette(X_valid, L_valid)
      If S_curr > S_best then
        S_best ← S_curr
        epsilon_best ← epsilon
        MinPts_best ← MinPts
        L_best ← L
      End If
    End If
  End For
End For
Return (epsilon_best, MinPts_best, L_best)
```

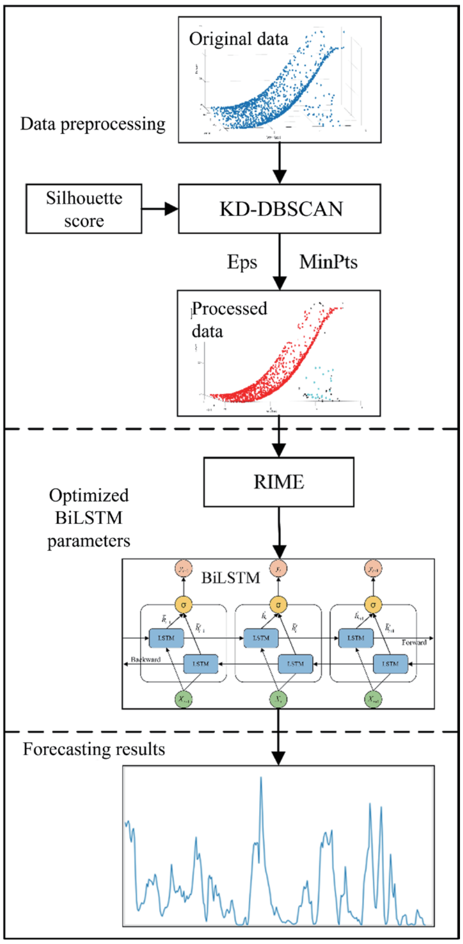


Fig. 4. (Color online) Workflow of wind power forecasting.

- Phase 1: Sensor signal conditioning using KD-DBSCAN
  The process begins with the acquisition of raw wind power sensor data. To handle high-frequency data streams efficiently, a KD-Tree is constructed to partition the feature space. This structure accelerates the neighborhood point queries within the DBSCAN algorithm, significantly reducing the computational overhead required to identify spatial densities. This step ensures that stochastic noise and sensor malfunctions are filtered out in real-time.
- Phase 2: Autonomous validation and feature extraction
  To ensure the robustness of the clustering process, S is calculated to evaluate the quality of the KD-DBSCAN results. By iteratively testing combinations of the neighborhood radius (Eps) and MinPts, the model identifies the optimal parameters that maximize cluster cohesion and separation. The resulting high-fidelity dataset, stripped of outliers, is partitioned into training and testing sets for DL.
- Phase 3: RIME-driven hyperparameter optimization and forecasting
  The purified sensor data are fed into the BiLSTM network. To avoid the limitations of manual trial-and-error, the RIME optimization algorithm searches for the optimal hyperparameters, including the learning rate, L_2 regularization coefficient, and number of hidden neurons. These three hyperparameters are adaptively tuned to minimize the regularized mean squared error (*MSE*) on the validation dataset, thereby maximizing the predictive performance of the BiLSTM model. To prevent overfitting during training, an L_2 regularization penalty (weight decay) is incorporated into the objective loss function. The total loss function to be minimized is defined as

$$L(W)=\frac{1}{N}\sum_{i=1}^{N}(y_i-\hat{y}_i)^2+\lambda\,\|W\|_2^2, \tag{9}$$

  where N is the number of training samples, y_i is the actual wind power output, $\hat{y}_i$ is the predicted output, W represents the trainable weight matrices of the BiLSTM layers, and λ is the L_2 regularization coefficient. λ is used to control the weight decay rate, stabilizing convergence and improving generalization across noisy sensor evaluation sets. In model training, λ was tuned with the RIME algorithm within a logarithmic search range of $[10^{-6}, 10^{-1}]$, and the optimal value selected for the final forecasting model was $\lambda = 1.0 \times 10^{-4}$.(15) The optimized parameters are shown in Table 1.

The initial learning rate is the most critical hyperparameter, controlling the step size at each iteration of the Adam optimizer. It is used to determine how quickly or slowly the model weights are updated in response to the estimated error. If the initial learning rate is too high, the training process might overshoot the global minimum, causing instability. If it is too low, the model might become trapped in local optima or require excessive computational time. RIME searches for an initial learning rate that ensures rapid convergence while maintaining stability. The L_2 regularization coefficient (weight decay) is integrated into the loss function to prevent overfitting, which is common in high-frequency sensor data. The RIME algorithm optimizes the L_2 regularization coefficient to balance the trade-off between fitting the training data and

Table 1
Parameter optimization for BiLSTM model.

Parameter	Search range	Optimal value	Description (location)
L_2 regularization (λ)	$[10^{-6}, 10^{-1}]$	1.0×10^{-4}	Loss function weight penalty [Eq. (9)]
Learning rate	$[10^{-4}, 10^{-2}]$	0.001	Adam optimizer step size
Hidden unit (BiLSTM)	[32, 256]	128	Hidden layer feature dimensionality
Batch size	[16, 128]	64	Training batch size

maintaining a simple model. Then, the model generalizes well to unseen sensor observations from different geographical locations. The number of neurons is used to define the dimensionality of the hidden state in each BiLSTM layer, representing the model's memory capacity. A larger number of neurons allows the model to capture more complex, high-dimensional temporal dependencies within the wind power data. By increasing the number of neurons, the model increases the risk of overfitting and the computational cost. The RIME algorithm identifies the optimal number of neurons to provide sufficient complexity to track nonlinear wind patterns without introducing redundant parameters.

The RIME algorithm also simulates the physical process of crystal condensation to balance global exploration and local exploitation. Once the optimal configuration is identified, the BiLSTM model is trained to capture the bidirectional temporal dependencies of the wind signal, yielding the final high-accuracy forecasting results.

3. Data Collection

3.1 Dataset

For the validation of the developed model, two wind power datasets, including data from diverse geographical regions in China, were acquired to test the model's robustness in different sensor environments. Dataset A was collected from an inland commercial wind project in Henan Province, Central China (May 1–June 30, 2023), a site characterized by complex terrain, strong microclimatic turbulence, and rapid ambient temperature shifts.(31) Dataset B was obtained from a coastal wind project in Shandong Province, Eastern China (April 1–May 31, 2024), featuring coastal plains influenced by seasonal monsoon fronts and high atmospheric salinity and humidity.(32) These datasets were selected to test the KD-DBSCAN RIME-BiLSTM framework under contrasting operational environments with distinct hardware degradation drivers—thermal fluctuations inland versus salt spray and corrosion noise along the coast.(32) Both wind farms operate GW_1.5_70 variable-speed, variable-pitch turbines manufactured by Goldwind Science and Technology Co., Ltd.(33) The GW_1.5_70 employs a doubly fed induction generator with a rated capacity of 1.5 MW.(34) The specifications include a rotor diameter of 70 m (swept area 3,848.5 m^2), hub height of 65–70 m, cut-in/cut-out wind speeds of 3.0 and 25.0 m/s, respectively, and a rated wind speed of 11.5 m/s.(33) To capture full operational dynamics beyond power and wind speed, five multi-dimensional variables were recorded by the SCADA system at 1 Hz sampling and aggregated into hourly records.(35)

- Active generated power (MW): measured using Hall effect transducers at the turbine converter output.
- Wind speed (m/s): measured using nacelle-mounted optoelectronic and ultrasonic anemometers.
- Wind direction (°): measured via potentiometric wind vanes aligned to true north.
- Ambient temperature (°C): measured using platinum resistance temperature detectors.
- Air density (kg/m^3): calculated from synchronized barometric pressure and ambient temperature data.

3.2 Sensor deployment

In the wind farm, sensors were mounted on turbine nacelles and a nearby 70-m meteorological mast (Table 2 and Fig. 5). Wind speed was measured with optoelectronic cup anemometers such as the Thies First Class Advanced (Germany), which use optical encoder disks and silicon photodiodes to detect rotation (a range of 0.5–50 m/s, an accuracy of ±0.2 m/s).(36) Ultrasonic anemometers (the Gill WindMaster Pro, the United Kingdom) were used to measure the transit time of acoustic pulses along orthogonal paths to capture three-dimensional wind velocity.(37) Wind direction was monitored with continuous-rotation potentiometric vanes [NRG 200P, the United States of America (USA)], and the Vaisala WMT700 series (Finland) was used as optical shaft encoders. Both devices have an angular accuracy of ±1.5°.(38) Ambient conditions were monitored using platinum resistance temperature detectors [Omega PR-21 series, platinum resistance thermometer with a nominal electrical resistance of 100 Ω at 0 °C, an international-standard-grade tolerance of ±0.15 °C at 0 °C defined by the International Electrotechnical Commission (IEC) 60751, USA] for air temperature, and barometric pressure was recorded with piezo-resistive MEMS transducers (Bosch BMP388, Germany), which combine capacitive sensing with digital compensation.(39) Electrical output was recorded using Hall-effect current

Table 2
Specifications of sensors deployed in wind farm.

Measurement	Model (manufacturer)	Principle	Range/accuracy	Specification
Wind speed	Thies First Class Advanced (Germany)	Optoelectronic cup with encoder + photodiodes	0.5–50 m/s, ±0.2 m/s	IEC 61400-12-1-compliant
Wind speed (3D)	Gill WindMaster Pro (UK)	Ultrasonic transit time	0–65 m/s, ±1.5% RMS	Measures 3D velocity
Wind direction	NRG 200P (USA)	Potentiometric vane	±1.5°	Continuous rotation
Wind direction (encoder)	Vaisala WMT700 (Finland)	Optical shaft encoder	±1.5°	Ingress protection 66/67 protection
Temperature	Omega PR 21 (USA)	Pt100 RTD	−50 to 250 °C, ±0.15 °C at 0 °C	IEC 60751 Class A
Pressure	Bosch BMP388 (Germany)	Piezo resistive MEMS	300–1250 hPa, ±8 Pa	Digital compensation
Electrical current	LEM LF 205 S (Switzerland)	Hall effect closed loop	Nominal 200 A, ±0.5%	Bandwidth DC–100 kHz
Data acquisition	SCADA RTU with 16 bit ADC	ADC	1 Hz sampling, 60 min aggregation	Time stamped hourly means

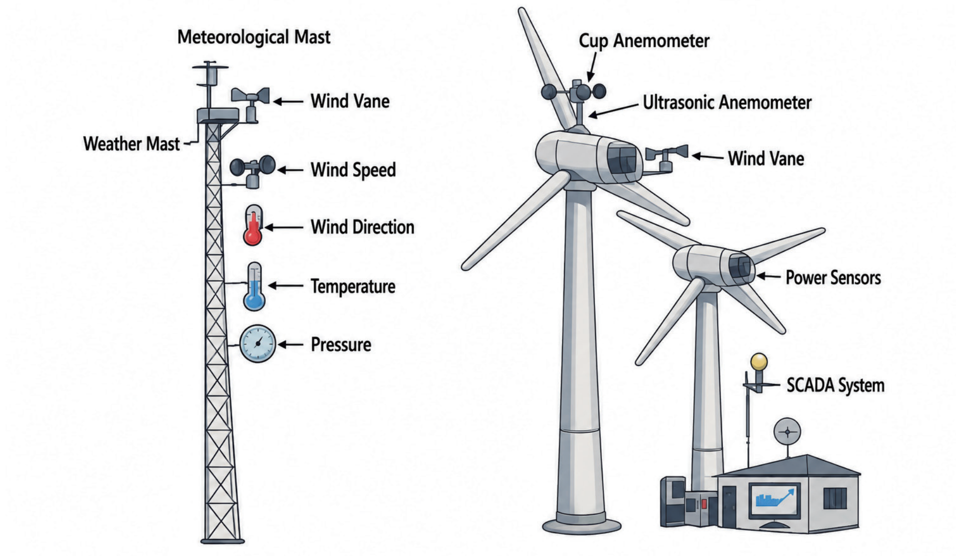


Fig. 5. (Color online) Schematic of wind farm sensor deployment (created using AI).

transducers (LEM LF 205-S, Switzerland) and potential transformers integrated into turbine converters, enabling precise measurements of three-phase voltage and current to compute active power in megawatts.[40] Data were collected using a SCADA system. Analog signals from sensors were digitized by 16-bit analog-to-digital converters (ADCs) in a remote terminal unit (RTU) at 1-Hz sampling. Sensor data collected were filtered, time-stamped, and aggregated into hourly means (60-min resolution).[41]

Since sensors degrade, bearings in cup anemometers wear, and negative speed biases (Type 2 anomalies) occur. Wind vanes ice over, freezing directional output, and data cables suffer transient electromagnetic interference, causing voltage spikes and spatial noise (Type 3 anomalies). KD-DBSCAN is used to cluster multi-dimensional sensor time-series data that include wind speed, direction, temperature, air density, and power, into a K-dimensional space, isolating corrupted signals while preserving dense clusters that represent true turbine operating states.[42]

4. Simulation

The simulations were conducted using the MATLAB 2021b software environment. All computations were carried out on a workstation equipped with an Intel Core i7-7700HQ CPU (2.80 GHz), 8 gigabytes of random access memory, and the Windows 10 Professional operating system. For the training and validation phase, the processed sensor time series were partitioned into a training set (80%) for model calibration and a testing set (20%) to evaluate predictive generalization. To evaluate the performance of the forecasting model, *MAE*, root mean squared error (*RMSE*), and the coefficient of determination (R^2) were used. These metrics were used to evaluate the deviation between the sensor-predicted values and the actual observed values.[43]

In wind energy conversion systems, the acquisition of sensor datasets can be compromised by anomalous data points. These irregularities are attributed to operational factors, including maintenance schedules, turbine curtailment, equipment failures, and sensor degradation. Therefore, robust anomaly detection must be conducted to improve the fidelity of the datasets used for model training. By mapping raw signals into a KD-Tree, the model identifies high-density clusters representing valid turbine states. Data points failing the density-reachability criteria, resulting from sensor drift or mechanical downtime, were removed to ensure that the subsequent BiLSTM model is trained only on high-fidelity, representative signal patterns.

Datasets A and B included multi-dimensional sensor time series, including wind speed, wind direction, and the corresponding power output. To handle the high volume of sensor observations, the data were partitioned using a KD-Tree structure. This allows the DBSCAN algorithm to perform nearest-neighbor queries with significantly reduced computational overhead by categorizing points into core, boundary, and noise candidates. This optimization is effective for large-scale datasets, since it minimizes the search space and accelerates clustering execution time.

S was used to identify the optimal Eps and MinPts. In the optimization process, the ideal parameters for Dataset A were determined to be *Eps* = 0.8, *MinPts* = 9, whereas for Dataset B, *Eps* = 0.7 and *MinPts* = 12. The anomaly detection results are shown in Fig. 6. The results showed that the developed model successfully categorized the raw sensor data into Type 1

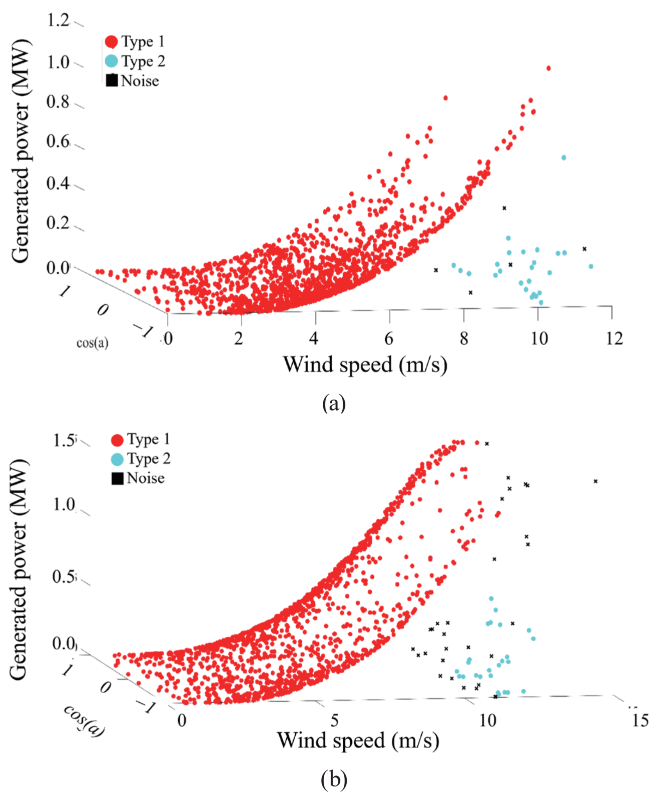


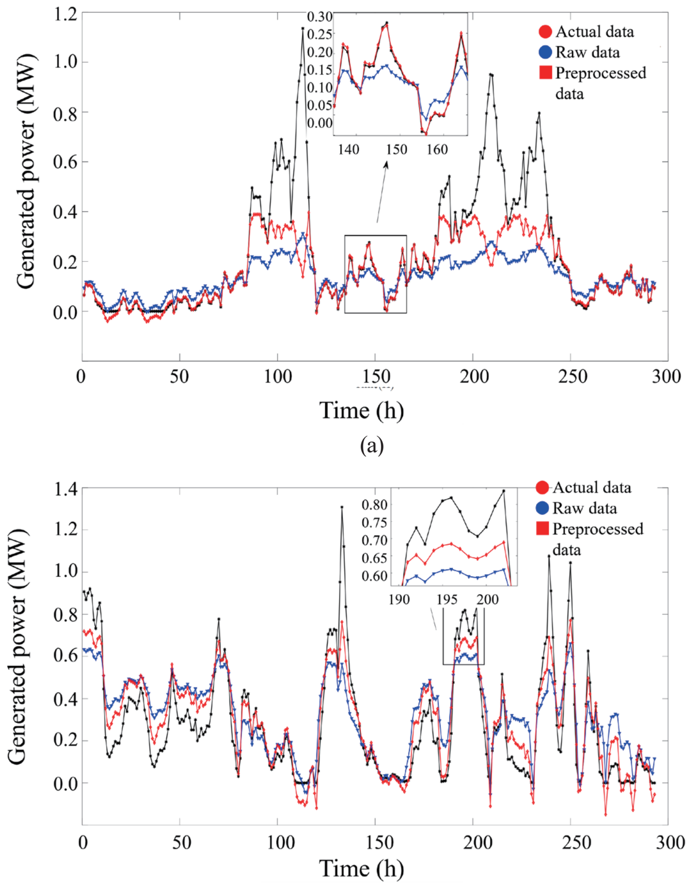
Fig. 6. (Color online) Anomaly detection results from (a) Dataset A and (b) Dataset B [cos(a) in RIME represents the position of the optimization agents for searching the best parameters].

(normal data, representing standard operational states and high-fidelity signal acquisition), Type 2 (abnormal data, observations characterized by sensor malfunctions or non-representative turbine behavior), and Type 3 (spatial noise, outliers that are spatially isolated from the primary data density, resulting from stochastic communication errors or extreme transient interference).

To evaluate its effectiveness, the model was trained and tested using the raw and preprocessed sensor datasets. The data preprocessing led to significant improvements across all evaluation metrics. For Dataset A, R^2 improved by 48.37%, increasing from 0.3568 to 0.5294. Concurrently, *MAE* and *RMSE* were reduced by 39.56% and 14.41%, respectively. For Dataset B, R^2 reached 0.8217, with a 36.42% increase, whereas *MAE* and *RMSE* decreased by 32.81% and 30.02%, respectively (Table 3). Figure 7 illustrates the improvement in forecasting performance before and after KD-DBSCAN preprocessing. In Datasets A [Fig. 7(a)] and B [Fig. 7(b)], the forecasts

Table 3
Comparison of forecasting parameters before and after KD-DBSCAN preprocessing of data.

Dataset	Data	R^2	*MAE*	*RMSE*
A	Raw data	0.3568	0.1127	0.1853
	Preprocessed data	0.5294	6.812×10^{-2}	0.1586
B	Raw data	0.6023	0.1345	0.1689
	Preprocessed data	0.8217	9.037×10^{-2}	0.1182



(b)

Fig. 7. (Color online) Comparison of wind power forecasting results before and after KD-DBSCAN preprocessing for (a) Dataset A and (b) Dataset B.

generated from raw data (blue line) deviate more noticeably from the actual wind power values (black line). After preprocessing, the cleaned forecasts (red line) align much more closely with the actual measurements, particularly in the zoomed-in insets. This visual evidence reinforces the quantitative results in Table 3, confirming that KD-DBSCAN preprocessing substantially reduces noise and enhances the model's ability to capture temporal fluctuations in wind power.

4.1 Forecasting results

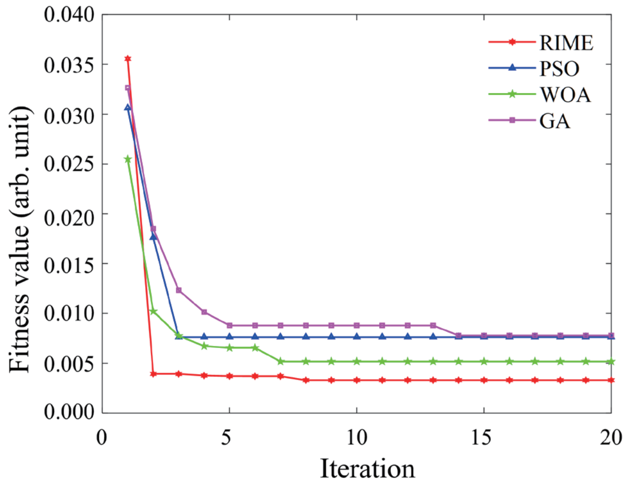
To evaluate the performance of the developed RIME-BiLSTM model, the predictive capability was compared with the standalone BiLSTM and BiLSTM optimized by genetic algorithm (GA), WOA, and PSO. Since BiLSTM accuracy is sensitive to internal hyperparameters, three parameters were selected for optimization: the initial learning rate, L2 regularization, and the number of neurons. Table 4 shows the optimized values obtained by each algorithm, alongside the manually selected baseline values.

The convergence characteristics of the optimization algorithms are shown in Fig. 8. The fitness value represents the error minimized during hyperparameter tuning. For Dataset A, the RIME algorithm demonstrated the fastest convergence, reaching the minimum fitness value of 2.91×10^{-3} as early as the second iteration. In contrast, PSO, WOA, and GA required 4, 7, and 12 iterations, respectively, to achieve similar values and stabilized at higher fitness levels. For Dataset B, RIME again outperformed the other models, achieving the lowest fitness value of 0.103 by the 7th iteration. These results highlight RIME's superior capability in navigating complex search spaces to identify global optima for sensor network parameters.

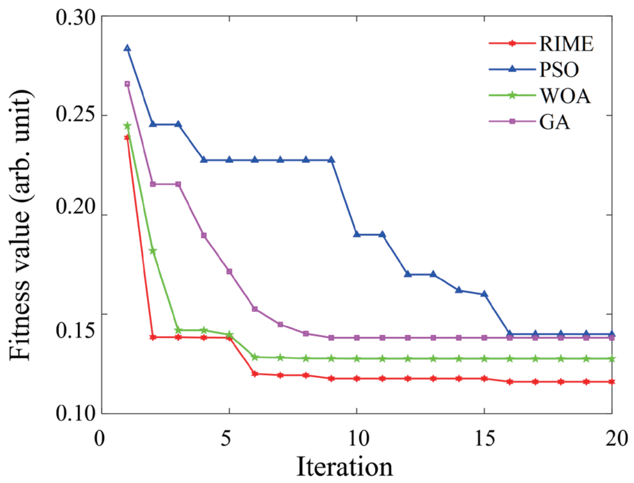
The predictive performance for Datasets A and B is presented in Table 5 and visualized in Figs. 9–10. For Dataset A, the wind power forecasting results of the BiLSTM, GA-BiLSTM, WOA-BiLSTM, PSO-BiLSTM, and RIME-BiLSTM models are shown in Fig. 9. BiLSTM yielded relatively poor accuracy (R^2 = 0.5294, $MAE = 6.812 \times 10^{-2}$, and $RMSE$ = 0.1586). In contrast, GA-BiLSTM, WOA-BiLSTM, and PSO-BiLSTM all achieved substantial improvements, with R^2 exceeding 0.96. Among these, the RIME-BiLSTM model achieved the best performance, with R^2 = 0.9876, $MAE = 8.214 \times 10^{-3}$, and $RMSE = 1.873 \times 10^{-2}$. Compared

Table 4
Hyperparameter optimization results for various algorithms.

Dataset	Model	Initial learning rate	L_2 regularization coefficient	Number of neurons
Data A	BiLSTM	0.012	0.18	12
	GA-BiLSTM	4.6×10^{-3}	0.085	14
	WOA-BiLSTM	1.7×10^{-3}	2.1×10^{-5}	96
	PSO-BiLSTM	0.042	0.014	58
	RIME-BiLSTM	0.0368	8.5×10^{-7}	64
Data B	BiLSTM	0.013	0.21	11
	GA-BiLSTM	0.0092	0.12	17
	WOA-BiLSTM	1.2×10^{-3}	2.8×10^{-4}	23
	PSO-BiLSTM	0.047	0.038	52
	RIME-BiLSTM	0.0286	6.3×10^{-7}	91



(a)

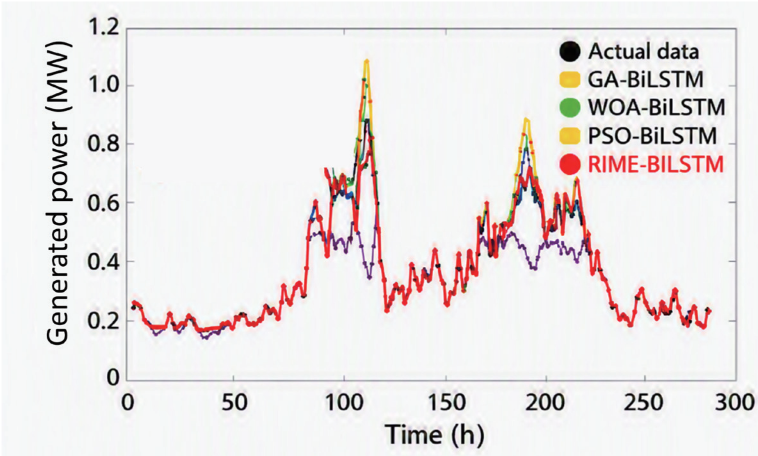


(b)

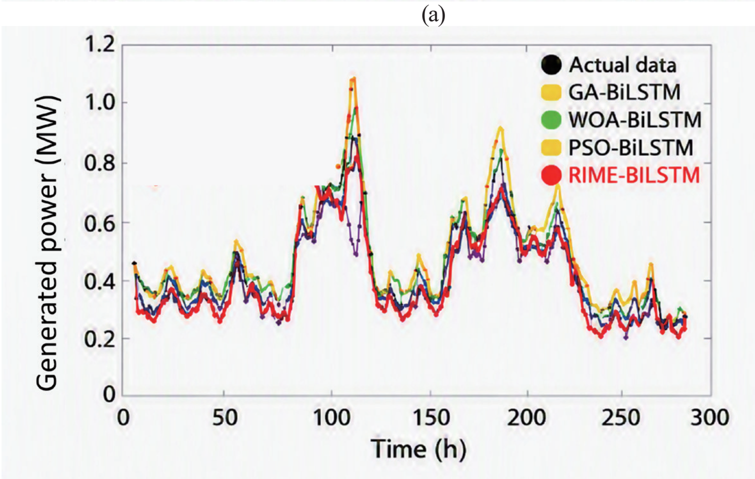
Fig. 8. (Color online) Convergence characteristics (fitness value) of different optimization algorithms for BiLSTM hyperparameter tuning for (a) Dataset A and (b) Dataset B.

Table 5
Performance comparison of models for Datasets A and B.

Dataset	Model	R^2	*MAE*	*RMSE*
Data A	BiLSTM	0.5294	6.812×10^{-2}	0.1747
	GA-BiLSTM	0.9623	1.521×10^{-2}	3.982×10^{-2}
	WOA-BiLSTM	0.9687	1.384×10^{-2}	3.441×10^{-2}
	PSO-BiLSTM	0.9641	1.607×10^{-2}	3.915×10^{-2}
	RIME-BiLSTM	0.9876	8.214×10^{-3}	1.873×10^{-2}
Data B	BiLSTM	0.8217	9.037×10^{-2}	0.1182
	GA-BiLSTM	0.9475	2.438×10^{-2}	6.214×10^{-2}
	WOA-BiLSTM	0.9327	3.265×10^{-2}	7.102×10^{-2}
	PSO-BiLSTM	0.9489	2.781×10^{-2}	6.587×10^{-2}
	RIME-BiLSTM	0.9786	1.936×10^{-2}	4.126×10^{-2}



(a)



(b)

Fig. 9. (Color online) Comparison of wind power forecasting results between RIME-BiLSTM and benchmark models for (a) Dataset A and (b) Dataset B.

with GA-BiLSTM, WOA-BiLSTM, and PSO-BiLSTM, the RIME-BiLSTM model reduced *MAE* by 46.0, 40.6, and 48.9%, respectively, and *RMSE* decreased by 52.9, 45.6, and 52.1%.

5. Discussion

The high R^2 of up to 0.9876 for the results demonstrates that the RIME-BiLSTM model can act as a highly reliable virtual sensor. In industrial settings where physical sensors fail or provide intermittent signals, this model can reconstruct missing data with high fidelity, ensuring continuous system monitoring. Through the integration of the KD-Tree, the $O(N_2)$ complexity of the standard is optimized to $O(N\log N)$. This improvement is vital for hardware implementations where computational resources are limited, enabling real-time anomaly detection and signal conditioning directly at the edge of sensor networks. By utilizing L_2 regularization coefficients and the RIME search strategy, the RIME-BiLSTM model avoids overfitting to stochastic noise. This makes the model well-suited for sensors deployed in harsh environments, such as offshore

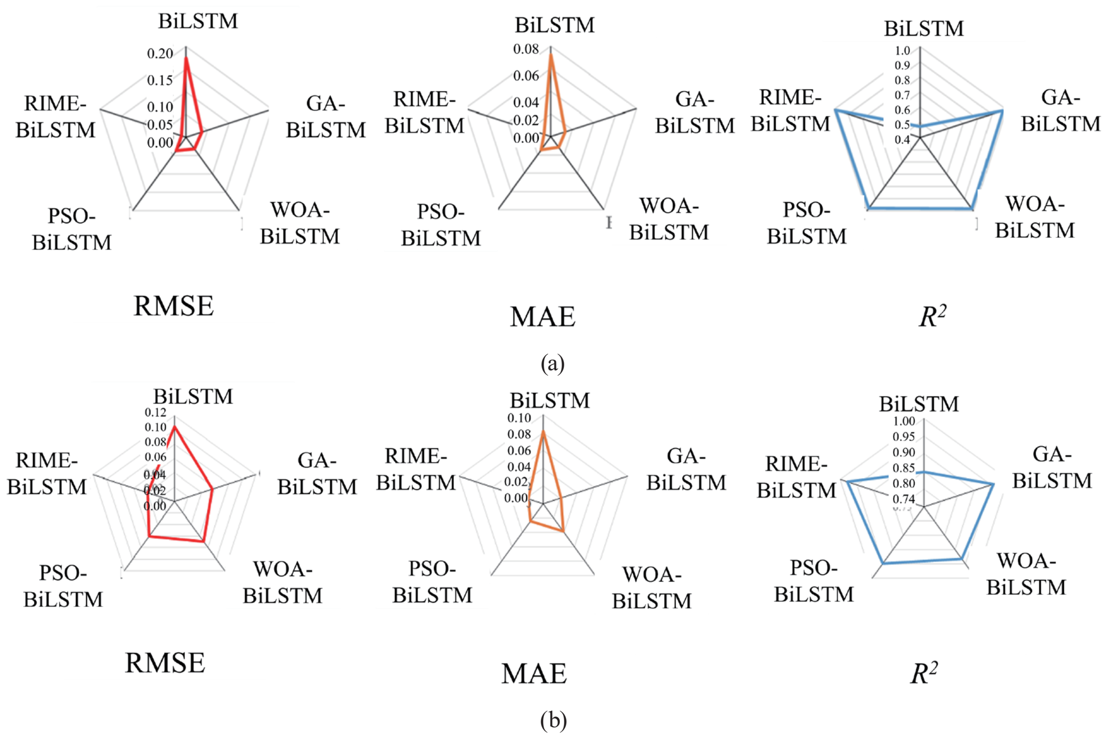


Fig. 10. (Color online) Comparison of multiple forecasting evaluation metrics across different models for (a) Dataset A and (b) Dataset B.

wind farms, where signal drift and degradation are common. In addition, the autonomous selection of clustering and network parameters using S and the RIME algorithm enables self-calibrating sensor systems. This reduces the need for manual intervention and allows the sensor network to adaptively maintain its accuracy as environmental conditions change.

A consideration for deploying soft-computing forecasting frameworks in commercial wind energy systems is offline numerical validation as well as feasibility. Although we evaluated the model computationally in this study, experiments were conducted using high-resolution SCADA from active 1.5 MW commercial wind turbines under operating conditions, rather than synthetic or simulated datasets. To ensure seamless integration into operational wind farm SCADA networks, two requirements must be met.

Standard DBSCAN clustering requires $O(N^2)$ time complexity, making real-time execution difficult on industrial controllers. By accelerating spatial neighborhood queries to $O(N \log N)$ using a KD-Tree, the data cleaning module operates with minimal latency. The KD-DBSCAN-RIME-BiLSTM model developed in this study runs locally on industrial edge gateways or wind turbine RTUs equipped with embedded ARM or x86 processors. As 1-Hz raw physical sensor signals, such as wind speed, direction, temperature, pressure, and power, are sampled by local RTUs, the KD-DBSCAN module filters hardware artifacts locally before streaming clean, 10-min or 1-h aggregated features to the central energy management system for real-time power forecasting and grid dispatch.

6. Conclusion

We developed and validated an intelligent RIME-BiLSTM forecasting model that integrates advanced spatial anomaly detection to address signal corruption and computational latency in physical sensor streams. By accelerating DBSCAN clustering through KD tree partitioning, the framework reduces spatial query complexity to $O(N\log N)$ and autonomously calibrates clustering parameters using S, ensuring consistent anomaly detection without manual intervention. Validation results on two commercial SCADA datasets showed that KD-DBSCAN effectively isolates physical sensor faults such as anemometer drift and signal dropouts, enhancing signal integrity and reducing forecasting errors by 39.6% in Dataset A and 32.8% in Dataset B. The anomaly detection process improved the base BiLSTM R^2 by up to 48.4%. Hyperparameter optimization using RIME outperformed GA, WOA, and PSO benchmarks, achieving $R^2 = 0.9876$ and reducing MAE by up to 48.9%.

The results of this study can be used to advance sensor technology in two directions. First, the developed model functions as a high-fidelity, self-calibrating virtual sensor capable of maintaining reliable output under physical sensor degradation and extreme field conditions. Its $O(N\log N)$ enables deployment on resource-constrained industrial edge controllers and wind farm energy management systems. Second, by categorizing distinct physical failure modes, including thermal drift, bearing friction, and icing, the isolated anomaly signatures provide valuable feedback for next-generation sensor design. These insights support the development of solid-state ultrasonic anemometers with hydrophobic anti-icing nanocomposites and smart self-calibrating transducer circuits.

Although the developed model shows high accuracy and robust anomaly filtering, limitations remain. The reliance on offline batch processing introduces latency when handling ultrahigh-frequency streaming SCADA data, and adaptation to abrupt, nonstationary microclimatic changes continues to pose challenges. Therefore, it is necessary to develop an online, incremental variant of KD-DBSCAN for real-time signal conditioning, integrated with adaptive continual transfer learning architectures. Validated on empirical field data, the computationally efficient model developed offers a practical, field-deployable solution for reliable renewable energy monitoring and grid integration.

Acknowledgments

This research was supported by the National Natural Science Foundation of China (Project No. 62303076), the National Key Research and Development Program of China (Project No. 2024YFE03250300), the Natural Science Research Programme of Colleges and Universities of Anhui Province, China (Project Nos. 2023AH052105 and 2023AH052109), the Anhui Provincial University Outstanding Research and Innovation Team Program (Project No. 2024AH010022), and the Subject Construction Promotion Project of Chaohu University, China (Project Nos. hxkt20230149 and hxkt2607012).

Declarations

The datasets supporting the results are partially available from the corresponding author upon reasonable request. Some data and code cannot be shared owing to confidentiality restrictions and ongoing research.

References

1 G. Mitrentsis and H. Lens: Appl. Energ. **309** (2022) 118473. https://doi.org/10.1016/j.apenergy.2021.118473
2 W. Xu, P. Liu, L. Cheng, Y. Zhou, Q. Xia, Y. Gong, and Y. Liu: Renew. Energy **163** (2021) 772. https://doi.org/10.1016/j.renene.2020.09.032
3 Y. Liu, D. Ćetenović, H. Li, E. Gryazina, and V. Terzija : Int. J. Electr. Power Energy Syst. **136** (2022) 107764. https://doi.org/10.1016/j.ijepes.2021.107764
4 W. Wu, J. Zhu, Y. Chen, T. Luo, P. Shi, and W. Guo : IEEE Trans. Ind. Appl. **59** (2022) 276. https://doi.org/10.1109/TIA.2022.3208866
5 S. Hanifi, X. Liu, Z. Lin, and S. Lotfian: Energies **13** (2020) 3764. https://doi.org/10.3390/en13153764
6 C. Wang, H. Zhang, and P. Ma: Appl. Energ. **259** (2020) 114139. https://doi.org/10.1016/j.apenergy.2019.114139
7 A. Liu, K. Zhang, M. Jia, Z. Liu, Q. Yang, and Y. Liu: Energy **350** (2026) 140691. https://doi.org/10.1016/j.energy.2026.140691
8 L. Li, Y. Li, B. Zhou, Q. Wu, X. Shen, H. Liu, and Z. Gong: Int. J. Electr. Power Energy Syst. **118** (2020) 105814. https://doi.org/10.1016/j.ijepes.2019.105814
9 W. Xiaoyun, J. Yu, Z. Ying, and W. Wang: Proc. 2021 Int. Conf. Power System Technology (POWERCON, 2021) 916. https://doi.org/10.1109/POWERCON53785.2021.9697766
10 H. Liu, C. Chen, X. Lv, X. Wu, and M. Liu: Energy Convers. Manag. **195** (2019) 328. https://doi.org/10.1016/j.enconman.2019.05.020
11 W. Yueheng: Proc. 4th Int. Conf. Energy, Electrical and Power Engineering (CEEPE, 2021) 507. https://doi.org/10.1109/CEEPE51765.2021.9475780
12 P. Du, J. Wang, Z. Guo Z, and W. Yang: Energy Convers. Manag. **150** (2017) 90. https://doi.org/10.1016/j.enconman.2017.07.065
13 M. Ester, H.-P. Kriegel, J. Sander, and X. Xu: Proc. 1996 Int. Conf. Knowledge Discovery and Data Mining (AAAI Press, 1996) 226. https://doi.org/10.5555/3001460.3001507
14 J. L. Bentley: Commun. ACM **18** (1975) 509. https://doi.org/10.1145/361002.361007
15 Z. Wang, Y. Ying, L. Kou, W. Ke, J. Wan, Z. Yu, H. Liu, and F. Zhang: Sensors **24** (2024) 444. https://doi.org/10.3390/s24020444
16 S. Samadianfard, S. Hashemi, K. Kargar, M. Izadyar, A. Mostafaeipour, A. Mosavi, N. Nabipour, and S. Shamshirband: Energy Rep. **6** (2020) 1147. https://doi.org/10.1016/j.egyr.2020.05.001
17 L. Yu, G. Meng, G. Pau, Y. Wu, and Y. Tang: Energies **16** (2023) 2079. https://doi.org/10.3390/en16042079
18 M. Singh and S. Chauhan: Electr. Power Syst. Res. **214** (2023) 108923. https://doi.org/10.1016/j.epsr.2022.108923
19 I. Karijadi, S.-Y. Chou, and A. Dewabharata: Renew. Energy **218** (2023) 119357. https://doi.org/10.1016/j.renene.2023.119357
20 M. Ding, H. Zhou, H. Xie, M. Wu, Y. Nakanishi, and R. Yokoyama: Neurocomputing **365** (2019) 54. https://doi.org/10.1016/j.neucom.2019.07.058
21 L. Liu, X. Wang, X. Dong, K. Chen, Q. Chen, and B. Li: Appl. Energy **374** (2024) 124035. https://doi.org/10.1016/j.apenergy.2024.124035
22 K. Moharm, M. Eltahan, and E. Elsaadany: Proc. 2020 Int. Conf. Smart Grids and Energy Systems (SGES. 2020) 922. https://doi.org/10.1109/SGES51519.2020.00169
23 J. Wang, Y. Qian, L. Zhang, K. Wang, and H. Zhang: Energy Convers. Manag. **299** (2024) 117818. https://doi.org/10.1016/j.enconman.2023.117818
24 B. Gu, H. Hu, J. Zhao, H. Zhang, and X. Liu: Energy Rep. **9** (2023) 807. https://doi.org/10.1016/j.egyr.2022.11.202
25 X. Gao, W. Guo, C. Mei, J. Sha, Y. Guo, and H. Sun: Energy Rep. **9** (2023) 335. https://doi.org/10.1016/j.egyr.2023.05.181
26 X. Shen, X. Fu, and C. Zhou: IEEE Trans. Sustain. Energy **10** (2018) 46. https://doi.org/10.1109/TSTE.2018.2822682

27 G. Liang, Y. Su, F. Chen, H. Long, Z. Song, and Y. Gan: IEEE Trans. Sustain. Energy **12** (2020) 1383. https://doi.org/10.1109/TSTE.2020.3045782
28 S. Yin and H. Liu: Energy **250** (2022) 123857. https://doi.org/10.1016/j.energy.2022.123857
29 Y. Wu, P. Su, T. Wu, J. Hong, and M. Hassan: IEEE Trans. Ind. Appl. **54** (2018) 5609. https://doi.org/10.1109/TIA.2018.2858183
30 M. Ester, H.-P. M., Kriegel, J. Sander, and X. Xu: Proc. 2nd Int. Conf. Knowledge Discovery and Data Mining (KDD, 1996) 226. https://dl.acm.org/doi/10.5555/3001460.3001507
31 J. Chen, J. Pan, C. Zhang, X. Luo, Z. Zhou, and B. Wang: Renew. Energy **111** (2017) 735. https://doi.org/10.1016/j.renene.2016.11.054
32 X. Kong, R. Huang, H. Li, and C. Guedes Soares: Reliab. Eng. Syst. Saf. **273** (2026) 112378. https://doi.org/10.1016/j.ress.2026.112378
33 Y. Xi, X. Wang, L. Cheng, M. Chen, I. Lestas, F. Shi, and K. Huang: Appl. Energy **417** (2026) 128055. https://doi.org/10.1016/j.apenergy.2026.128055
34 V. Bulaevskaya, S. Wharton, A. Clifton, G. Qualley, and W. O. Miller: J. Renew. Sustain. Energy **7** (2015) 013103. https://doi.org/10.1063/1.4904430
35 J.-W. Ding, M.-J. Chuang, J.-S. Tseng, and I.-Y. L. Hsieh: Appl. Energy **375** (2024) 124129. https://doi.org/10.1016/j.apenergy.2024.124129
36 Thies CLIMA: https://www.thiesclima.com/en/first-class--ice-classified/ (accessed August 2026).
37 Gill Instrument: https://gillinstruments.com/compare-3-axis-anemometers/windmaster-3axis/ (accessed August 2026).
38 NRG Systems: https://www.nrgsystems.com/assets/resources/NRG-200P-Wind-Vane-Instructions.pdf (accessed August 2026).
39 Omega Engineering: https://br.omega.com/omegaFiles/temperature/pdf/PR-21.pdf (accessed August 2026).
40 Bosch: https://www.bosch-sensortec.com/media/boschsensortec/downloads/datasheets/bst-bmp388-ds001.pdf (accessed August 2026).
41 LEM: https://www.lem.com/en/product-list/lf-205s (accessed August 2026).
42 Analog Devices: https://www.analog.com/media/en/technical-documentation/data-sheets/ad7606c-16.pdf (accessed August 2026).
43 R. Teixeira, A. Cerveira, E. J. S. Pires, and J. Baptista: Energies **17** (2024) 3480. https://doi.org/10.3390/en17143480