

A Semi-supervised Industrial Time Series Anomaly Detection Method Based on Deep Reinforcement Learning

Liming Du,^{1,2} Guolin Wang,^{2,3} and Fengying Wang^{1*}

¹School of Information Engineering, Suqian University,
No. 399, South Huanghe Road, Suqian City, Jiangsu Province 223800, P.R.China

²School of Computer Science and Engineering, Shenyang Jianzhu University,
No. 25, Hunnan Middle Road, Hunnan District, Shenyang, Liaoning 110168, P.R. China

³School of Electronics and Information Engineering, Huaibei Institute of Technology,
No. 8, Qingnian Road, Gao'yu Subdistrict, Duji District, Huaibei, Anhui 235000, P.R.China

(Received February 9, 2026; accepted September 17, 2026)

Keywords: sensor signal processing, industrial time series data, deep reinforcement learning, anomaly detection, semi-supervised learning

Industrial manufacturing systems increasingly rely on large-scale sensor networks to monitor equipment states and production processes. For example, in continuous casting, casting-speed signals are continuously acquired by encoders and transmitted to a programmable logic controller (PLC) or a distributed control system (DCS) through industrial sensing links. These sensor-generated time-series signals provide essential information for intelligent perception and fault diagnosis, yet often lack high-quality annotations. To address this issue, in this paper, we proposed the Industrial Deep Time Series Detection-Q-Network (IDTS-DQN), a semi-supervised deep reinforcement learning algorithm, for anomaly detection in industrial sensing systems. The method integrates deep learning and reinforcement learning through a dual reward mechanism and an anomaly-based experience replay strategy, enabling the agent to learn temporal patterns with limited labeled samples and dynamically improve detection performance. Extensive experiments on real-world industrial datasets demonstrate that IDTS-DQN outperforms baseline methods in recall, precision, and F1-score, providing an adaptive sensing-data analysis approach for industrial monitoring and demonstrating the potential of deep reinforcement learning in sensor-based anomaly perception. However, the framework has been evaluated only offline on historical sensing data, and its applicability to broader signal types and fault modes remains to be examined. Future work will focus on multimodal sensor fusion, edge-based online deployment, and domain-knowledge integration for industrial sensor-signal analysis.

1. Introduction

The evolution of industrial production—from the mechanization of Industry 1.0, through the electrification and automation of Industry 2.0 and Industry 3.0, to the intelligent manufacturing

*Corresponding author: e-mail: wangfengying@squ.edu.cn
<https://doi.org/10.18494/SAM6284>

era of Industry 4.0—has fundamentally transformed modern production systems. A key feature of Industry 4.0 is the integration of cyber-physical systems, in which various sensors and monitoring devices are deployed throughout industrial processes to continuously acquire information about equipment operation and production conditions. These sensor-generated signals provide the foundation for intelligent perception, process monitoring, and fault diagnosis in modern manufacturing systems.⁽¹⁾ With the rapid growth of sensor deployment, industrial systems continuously generate massive amounts of time-series data, and the efficient analysis and utilization of these signals has become a critical challenge for both academic research and industrial applications.⁽²⁾ Specifically, these signals are continuously acquired by industrial sensing chains embedded along the production line, such as encoder-based measurement systems interfaced with a programmable logic controller (PLC) or a distributed control system (DCS). In these systems, physical quantities are converted into digital time-series data, which are then aggregated via edge gateways and transmitted to process-monitoring platforms.

Among the various types of industrial sensing data, time-series signals are the most fundamental and widely collected, because they directly reflect the temporal evolution of equipment states and production processes. Unlike general time-series data, industrial time-series data are usually generated by distributed sensors embedded in manufacturing systems and are closely coupled with physical operating conditions. These signals often exhibit complex characteristics, including high dimensionality, noise interference, incomplete observations, and strong temporal dependencies, which are further compounded by extremely large data volumes, heterogeneous sources and formats, and high generation velocity.⁽³⁾ Consequently, accurately identifying abnormal states from massive sensor-generated time-series data remains a critical problem in intelligent industrial perception.⁽⁴⁾

In manufacturing systems, distributed sensor networks continuously acquire physical quantities along the production line that reflect equipment operating states, forming multi-source heterogeneous sensor data streams. Affected by field conditions, these sensor signals typically exhibit complex characteristics—including high dimensionality, strong noise, incomplete observations, and pronounced temporal dependencies—that challenge conventional fixed-threshold and rule-based monitoring. With the advancement of intelligent manufacturing, the volume and velocity of sensor-generated data continue to grow, creating a strong demand for methods that can adaptively extract state information from sensor data and identify abnormal patterns. This need constitutes the core motivation of the present study.

To address the above need, a wide range of anomaly detection methods have been proposed, which can be roughly grouped into three categories: traditional statistical and distance-based methods, classical machine-learning methods, and deep learning-based methods.⁽⁵⁾ In the first category, traditional statistical methods identify anomalies by modeling data distributions and temporal trends; for instance, Falcão *et al.* detected faults in industrial equipment through time-series stationarity analysis.⁽⁶⁾ Distance-based methods determine anomalies according to the distances between data points: the k-nearest neighbors (KNN) algorithm defines the anomaly degree of a point as its distance to its KNN,^(7,8) whereas the local outlier factor algorithm introduces the concept of local density and identifies anomalies by comparing the density of a point with that of its neighbors.⁽⁹⁾ Although these traditional methods perform well in specific

scenarios, statistical approaches rely on strong assumptions about the underlying data distribution and are sensitive to parameter choices, whereas distance-based methods suffer from high computational cost and the curse of dimensionality on high-dimensional time series.

In the second category, classical machine-learning techniques have been widely adopted for anomaly detection. Support vector machines identify an optimal separating hyperplane to discriminate between normal and anomalous samples,⁽¹⁰⁾ and one-class support vector data description methods construct a hypersphere enclosing normal samples to discriminate anomalies.⁽¹¹⁾ Other methods, such as multivariate regression, linear discriminant analysis, decision trees, and Gaussian mixture models (GMMs), have also been employed to address different aspects of the anomaly detection problem.⁽¹²⁾ However, these supervised methods require large quantities of labeled anomaly data during training, which are frequently difficult to obtain in real industrial environments, particularly for novel or previously unseen fault modes. Unsupervised methods instead learn the distributional characteristics of normal behavior from large volumes of unlabeled data and have therefore been widely adopted for anomaly detection in industrial processes,^(13,14) but their accuracy typically lags behind that of supervised methods, and their performance can be affected by implicit assumptions about data distributions and anomaly types.

In the third category, deep neural networks have been extensively applied to anomaly detection in industrial time-series data. Reconstruction-based methods,^(15,16) particularly those utilizing autoencoders (AEs),^(17,18) learn representations of time-series data and determine anomalies according to the reconstruction error. For instance, Qian *et al.* proposed RGAAnomaly, a data reconstruction-based generative adversarial network for multivariate time-series anomaly detection in the IoT.⁽¹⁹⁾ It extracts and fuses temporal and metric features through a Transformer with cross-attention, constructs a joint generator composed of an AE and a variational AE, and finally uses a combination of reconstruction loss and discrimination loss as the anomaly score. Sellam *et al.* further proposed the Mamba Adaptive Anomaly Transformer (MAAT), an enhanced anomaly Transformer that combines sparse attention with a Mamba state-space model gated by attention, achieving a balance between reconstruction fidelity and contextual enhancement on noisy, nonstationary data.⁽²⁰⁾ However, existing deep learning models still have limitations: they mainly rely on reconstruction error and may overlook substantial prior knowledge; moreover, industrial data are generated at high speed and high complexity and exhibit pronounced drift owing to changes in production plans, which makes it urgent to develop models capable of continuous training and updating.

On the basis of the above analysis, existing approaches still exhibit four fundamental limitations when confronting the complexities of real-world industrial scenarios.

- (1) Inadequate generalization capability, as traditional algorithms are typically effective only for specific data distributions or anomaly patterns.
- (2) Limited sensitivity to nonperiodic patterns, because most algorithms focus on univariate time series with distinct periodicity, and their sensitivity deteriorates for abrupt, nonperiodic anomalies.
- (3) Excessive reliance on annotated data, since mainstream deep learning methods depend heavily on large volumes of high-quality labeled data, which are prohibitively expensive to obtain in complex industrial systems.

- (4) The absence of model evolvability and adaptability, as contemporary models typically freeze their parameters and decision logic after training and lack mechanisms to continuously improve using newly generated data and detection feedback.

To address the above urgent intelligent-perception needs of industrial sensor-based systems, in this work, we systematically integrated deep reinforcement learning (DRL) as a supporting technique into the anomaly analysis of industrial sensor data, and proposed a semi-supervised algorithm, namely, Industrial Deep Time Series Detection-Q-Network (IDTS-DQN). By synergizing the representational power of deep learning with the exploratory dynamics of reinforcement learning (RL), IDTS-DQN employs dual reward functions (intrinsic and extrinsic) together with an experience replay strategy, enabling the agent to receive reward signals and train effectively with minimal labeled data. The system autonomously learns to discriminate anomalies from normal patterns while dynamically refining its detection capability through accumulated experience, which provides the industrial sensor-based system with environmental exploration ability and enhances its adaptability across diverse anomaly contexts.

The principal contributions of this study are summarized as follows.

- (1) By addressing the anomaly-perception needs of industrial sensor-based systems, we develop a DRL framework for the analysis of sensor-generated time-series signals, in which anomaly perception is formulated as a sequential decision-making process, and we construct a Deep Q-Network (DQN) framework that inherently supports evolutionary perception.
- (2) For sensor-signal analysis under scarce annotations, an innovative dual-reward strategy is proposed, which combines limited anomaly labels with the intrinsic characteristics of sensor signals, enabling effective model training under insufficient annotation conditions.
- (3) To meet the identification needs of rare abnormal patterns in industrial sensor-based systems, an anomaly-aware experience replay strategy is designed to improve the recognition capability of sensor-signal analysis for rare abnormal patterns.

The remainder of this paper is organized as follows. In Sect. 2, we present the overall framework design of IDTS-DQN; in Sect. 3, we provide details of the algorithm design; in Sect. 4, we report the experimental results and analysis; in Sect. 5, we discuss the results and their implications for intelligent sensor-based perception in industrial systems; and in Sect. 6, we conclude the paper.

2. Framework Design of IDTS-DQN

To address the challenges described above, we develop a DRL algorithm tailored to sensor-generated industrial time-series anomaly detection, which we call IDTS-DQN. The principal innovation lies in the targeted design of three coupled modules: the interactive environment, the hybrid reward function, and the experience replay mechanism, to accommodate the distinctive statistical and operational characteristics of industrial sensor signals.

By integrating deep learning with RL and adopting a sliding-window scheme to process sequential sensor measurements, we simulate an interactive reinforcement-learning environment that yields meaningful feedback to the agent. The core strategy is a hybrid intrinsic–extrinsic

reward function that supplies the agent with informative learning signals even when labeled data are extremely scarce. This reward fuses intrinsic anomaly indicators derived from the data's internal properties with extrinsic supervisory signals provided by a small labeled subset, guiding the agent to recognize known anomalies while exploring previously unseen patterns—thereby mitigating the label-scarcity problem in industrial sensing applications.

Moreover, to address the ubiquitous class-imbalance issue in industrial datasets, we propose a proportional sampling strategy. Samples are dynamically assigned to two separate experience replay pools according to their reward values or anomaly scores. During training, mini-batches are sampled from the two pools in proportions that reflect the prior anomaly-to-normal ratio in the original dataset, ensuring stable convergence and sufficient exposure of the agent to rare anomalous examples.

Ultimately, the proposed agent leverages limited prior knowledge of known anomalies while acquiring the capacity to explore and learn unknown anomaly types, enabling continuous model evolution. The IDTS-DQN framework is composed of three core modules—the interactive environment, the hybrid reward function, and the experience replay mechanism—as illustrated in Fig. 1.

2.1 Interaction environment

In DRL, an agent learns a policy through interaction with an environment. Since time-series anomaly detection does not inherently provide a ready-made RL environment, we construct a sequential decision-making environment from the raw sensor time-series signals using a sliding-window procedure. Both the current state and its successor state are fixed-length subsequences extracted from T .

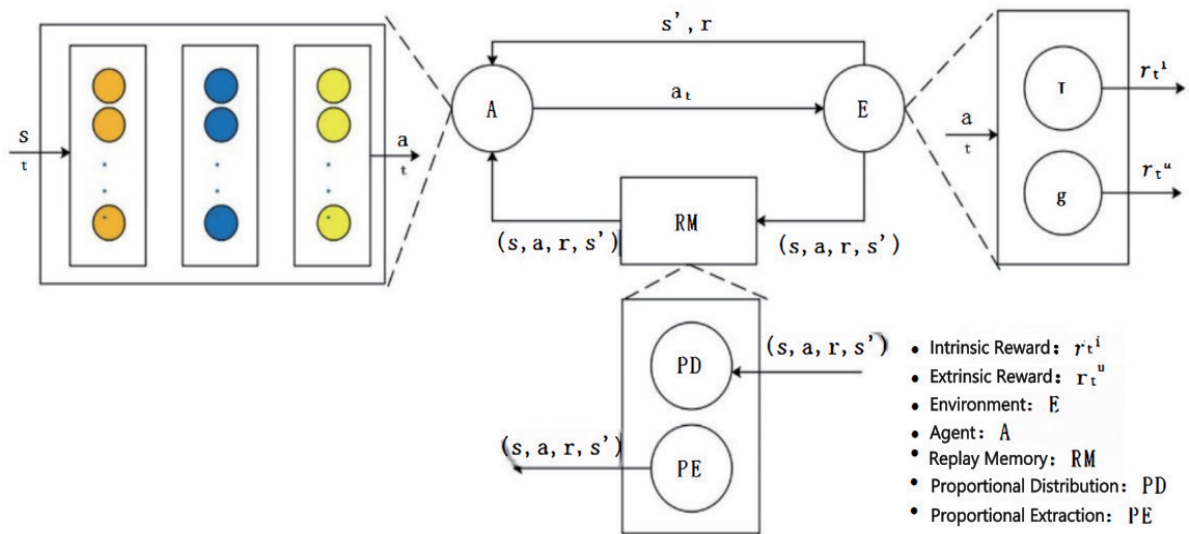


Fig. 1. (Color online) The IDTS-DQN framework for industrial time-series anomaly detection.

Let the observed series be $T = \{(t_1, x_1), (t_2, x_2), \dots, (t_n, x_n)\}$. We define the length- m subsequence ending at (t_i, x_i) as $S_{i,m} = \{(t_{i-m+1}, x_{i-m+1}), (t_{i-m+2}, x_{i-m+2}), \dots, (t_{i-1}, x_{i-1}), (t_i, x_i)\}$, where $m \leq i \leq n$. The successor state corresponding to $S_{i,m}$ is simply $S_{i+1,m}$. This sliding-window construction produces a temporally ordered sequence of decision states that is suitable for training an RL agent to detect anomalies in industrial time-series data.

2.2 Reward function

We have designed a novel reward function for sparse labeled data, which consists of two components: the external reward based on the label of the data, and the internal reward based on the data itself.

The external reward is derived from the action $a = \{0,1\}$, which is taken according to the label of the data and the reward obtained from the data's own label.

The external reward is defined as follows:

$$r_t^u = \begin{cases} 0.9, & \text{if } a_t = 1 \text{ and } lab = 1, \\ -0.1, & \text{if } a_t = 0 \text{ and } lab = 1, \\ -0.1, & \text{if } a_t = 1 \text{ and } lab = 0, \\ 0.1, & \text{if } a_t = 0 \text{ and } lab = 0. \end{cases} \quad (1)$$

Under this scheme, actions coinciding with the anomalous label yield a positive external reward, whereas contradictory actions incur a negative one. This label-driven reinforcement encourages the agent to fully exploit the sparsely labeled data and to learn discriminative features of anomalous events, while explicitly penalizing false positives and false negatives so as to suppress those error types.

The internal reward is derived exclusively from intrinsic properties of the data. Concretely, for every time index t , we compute the anomaly degree of the fixed-length subsequence that terminates at t by comparing it to a reference set of normal subsequences. The scoring mechanism is designed to amplify the disparity between anomalous and normal behavior so that truly anomalous subsequences attain markedly higher anomaly degrees than normal ones. The resulting amplified anomaly degree is used directly as the internal reward.

Algorithm 1 shows a sliding-window procedure for computing the anomaly degree at each time index of a time series. The corresponding pseudocode is given below.

Algorithm 1

Anomaly Calculation.

Input: Time Series (T), Subsequence length m , transition t ;

Output: Anomaly(X_t);

```

1  Anomaly( $X_t$ ) = 0;
2  For i=m to n
3      nearest_neighbor_dist=infinity;
4      For j=i-m+1 to i-1
5          nearest_neighbor_dist = EuclideanDist( $S_{i,m}, S_{j,m}$ );
6      End for
7      Anomaly ← nearest_neighbor_dist;
8  end for

```

Normal subsequences appear frequently in industrial data and are not fixed. We use mean and fitting methods to extract normal subsequences from time series when the industrial data are in a stable state with stable or fluctuating amplitudes. The size of the values of normal subsequences is replaced by the mean. The recursive calculation method for the mean is used to estimate the mean distribution, and the calculation equation is

$$\mu_n = \frac{1}{n}x_n + \frac{n-1}{n}\mu_{n-1}. \quad (2)$$

Euclidean distance: Given two subsequences $S_{i,m}$ and $S_{j,m}$ extracted from the time series T , the Euclidean distance between them is defined as

$$EuclideanDist(S_{i,m}, S_{j,m}) = \sqrt{\sum_{q=0}^{m-1} (x_{i-q} - x_{j-q})^2}, \quad 1 \leq i, j \leq n \quad (3)$$

Anomaly degree: In a time series T , the anomaly degree of the point x_i is defined as the Euclidean distance between the length- l subsequence $S_{i,m}$ that extends forward from X_i and a normal subsequence $S_{j,m}$, i.e.,

$$Anomaly(X_i) = EuclideanDist(S_{i,m}, S_{j,m}). \quad (4)$$

Z-score normalization is applied to $Anomaly(x_i)$ as follows:

$$Anomaly(X_i)_z = \frac{Anomaly(X_i) - \mu}{\sigma}, \quad (5)$$

where μ and σ denote the mean and standard deviation of the anomaly scores, respectively. $Anomaly(X_i)_z$ represents the standardized anomaly score. The internal reward is $r_t^i = Anomaly(X_i)_z$.

Therefore, the reward function we design for industrial time-series anomaly detection is composed of the external reward r_t^i and the internal reward r_t^u and is computed according to the following rule:

$$r_t = r_t^u + r_t^i = \begin{cases} 0.9, & \text{if } a_t = 1 \text{ and } lab = 1, \\ -0.1, & \text{if } a_t = 0 \text{ and } lab = 1, \\ Anomaly(t), & \text{if } a_t = 1 \text{ and } lab = 0, \\ -Anomaly(t), & \text{if } a_t = 0 \text{ and } lab = 0. \end{cases} \quad (6)$$

Under this reward scheme, we adopt a granular allocation that reflects the asymmetric tolerance between error types in operational environments. Correctly identified true positives receive a scalar reward of +0.9, reinforcing the agent's sensitivity to genuine anomalies. False positives are penalized by a deliberately mild -0.1 , because completely eliminating false alarms in real-world deployments is impractical and over-penalization would suppress sensitivity to true events. This asymmetric design lets the agent learn from sparse labels while adaptively balancing sensitivity and specificity at deployment time, and ultimately enhances robust anomaly-detection performance under practical label-constrained scenarios.

2.3 Experience replay

This component comprises two processes—storage and retrieval—designed to manage large-scale and highly heterogeneous industrial sensor streams. Such data are massive and often exhibit prolonged periods in either stable or anomalous states, so an undersized replay buffer risks overfitting to one mode, whereas an oversized one inflates training cost and sampling bias. We address this by introducing two complementary strategies: a reward-driven storage policy that partitions experiences into anomaly and normal pools on the basis of reward magnitude, and an anomaly-aware storage policy that partitions them on the basis of anomaly-score severity. Their joint operation balances training efficiency with data diversity.

The reward-driven strategy is built upon the hybrid reward in Sect. 2.2. It defines a storage policy that assigns each newly collected experience to an anomaly or normal pool according to the reward produced after the agent's interaction, ensuring that the anomaly pool retains sufficient rare-anomaly experiences while the remaining samples populate the normal pool. During sampling, a proportional policy draws mini-batches from the two pools at a ratio aligned with the natural anomaly-to-normal ratio in the original sensor data; this ratio can be flexibly adjusted to specific application scenarios.

On the other hand, the anomaly-based storage strategy adopts a similar principle by partitioning data into anomaly and normal subsets according to the severity of $Anomaly(t)$. Its selection mechanism operates in a manner complementary to the reward-driven strategy. The introduction of this strategy further enhances the robustness of the data-processing framework, ensuring that the agent not only maintains efficient learning but also obtains sufficient data diversity to comprehensively understand and model the characteristics of industrial time-series data. The design and implementation of these strategies collectively aim to optimize the agent's training pipeline, improving both the efficiency and accuracy of handling complex industrial time-series patterns. Ultimately, they provide a solid foundation for the broader deployment of intelligent systems in industrial applications.

We outline the workflow of the proposed experience replay strategy in Fig. 2.

3. Design of the IDTS-DQN Algorithm

Building on the framework described in Sect. 2, in this section, we formalize the IDTS-DQN algorithm and its training pipeline. The complete data flow—from the initial sensor window

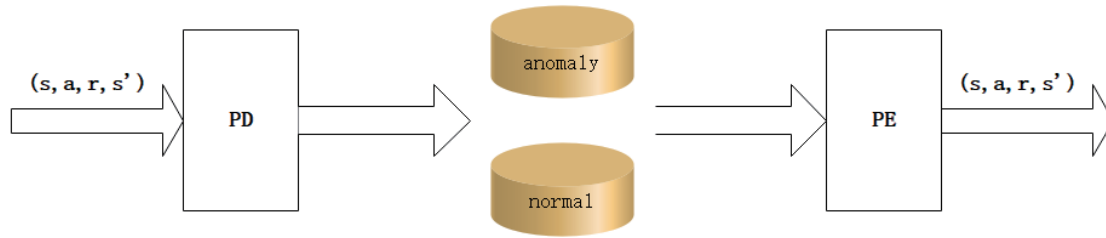


Fig. 2. (Color online) Experience replay strategy. PD denotes proportional distribution, which distributes experiences into anomaly and normal groups; PE denotes proportional extraction, which samples experiences proportionally from the two groups.

through reward computation, experience storage, sampling, and network update—is illustrated in Fig. 3, with the corresponding pseudocode given in Algorithm 2.

The proportional storage and sampling strategies introduced in Sect. 2.3 are instantiated here as concrete procedures: experiences are placed into their designated replay buffers under the reward/anomaly storage policy, and retrieved during training according to the proportional sampling policy, providing stable learning under the natural class imbalance of industrial sensor data.

First, the initial data are processed using the *Anomaly* operation to obtain the anomaly scores of the test set, followed by a normalization step to construct the internal reward r_t^i . In parallel, an external reward function r_t^u is designed, which is then combined with the internal reward to form the overall reward r_t . The model acquires the current state s_t from the environment S , and the agent A selects an action a_t according to its policy. The environment S subsequently returns a corresponding reward on the basis of the action a_t taken by the agent.

Under the proportional distribution (PD) storage policy, each experience tuple (s, a, r, s') is placed into its designated replay buffer. During training, the proportional extraction (PE) sampling policy is adopted to retrieve experiences from the corresponding replay buffers. The agent network is then updated by stochastic gradient descent according to the sampled minibatches and the network parameters.

The industrial time-series anomaly detection algorithm based on the improved DQN is referred to as IDTS-DQN, and its pseudocode is shown in Algorithm 2.

4. Experimental Results and Analysis

4.1 Dataset description

In this study, we utilize real-world CastSpeed time-series data collected from the continuous casting process in steel-making as the research object, focusing on analyzing the temporal variation patterns of the casting speed. During normal production, the casting speed remains relatively stable and exhibits distinct steady states under standard operating conditions. However, when mechanical faults or process anomalies occur, the casting speed fluctuates sharply, demonstrating pronounced instability.

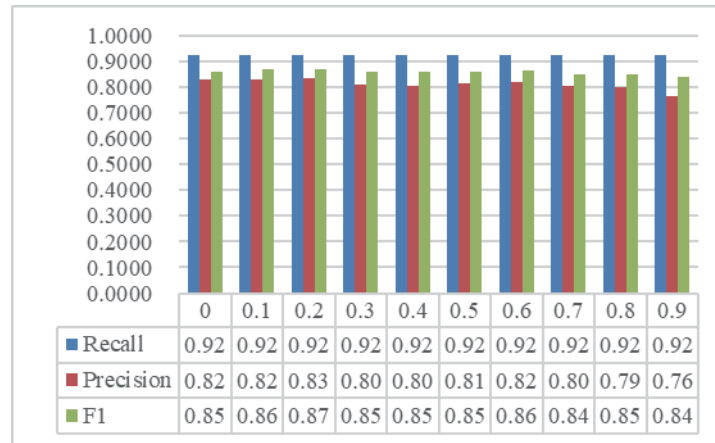


Fig. 3. (Color online) Results of the experience replay strategy on steeldata_1. The horizontal axis denotes the sampling ratio (ranging from 0.1 to 0.9), and the vertical axis denotes the recall, precision, and F1-score of the compared models.

Algorithm 2

IDTS-DQN.

Input: environment;

set of action a ;

set of states S ;

train_interval t ;

replay memory E & E' of DQN;

structured neural network A ;

parameter update ratio r ;

greedy factor ε ;

discount factor γ ;

learning rate α ;

Output: Action a ;

```

1  for e in episodes do
2    Anomaly;
3    normalization  $\rightarrow [-1, 1]$ ;
4    for  $i$  in  $T$  do
5      take a state  $S$  from  $T$ ;
6      Compute  $q_{value} = Q(s, a)$ ;
7      compute  $Prob_{action}$  based on  $q_{value}$  and  $\varepsilon$ ;
8      action  $a = \text{random choice with } Prob_{action}$ ;
9      observe reward  $s$ , next state  $s'$ ;
10     store transition  $(s, a, r, s')$  in  $E$  & store transition  $(s, a, r, s')$  in  $E'$  based on  $PD$ ;
11     if  $E > \text{mini-batch}$  and  $\text{step} \% t == 0$ 
12       randomly sample a  $\text{mini-batch}$   $(s, a, r, s')$  from  $E$  &  $E'$  based on  $PE$ ;
13        $\text{target} = r_j + \gamma \max_a \hat{Q}(s_j, a_j)$ ;
14       perform gradient descent:  $(q_{value} - \text{target})^2$ ;
15     end if
16   end for
17 end for
```

Sensing configuration and data acquisition. The monitored equipment in this study is a continuous casting machine used in the steel-making process; the signal analyzed is a single time-series variable—the casting speed, i.e., the linear speed at which the cast strand is

withdrawn from the mold, measured in meters per minute (m/min). The casting-speed signal is acquired through an industrial sensing chain rather than by direct manual measurement. Specifically, an incremental pulse encoder is mounted on the withdrawal roller of the continuous casting machine and outputs a train of electrical pulses whose frequency is proportional to the rotational speed of the roller. These pulses are fed into a high-speed counter module of a PLC, which performs the counting and timing operations; the pulse count is then converted into a casting speed value. The computed speed value is recorded and stored by the DCS/PLC and is finally sampled at a fixed interval to form the time-series data used in this study. In this sensing chain, the incremental pulse encoder acts as the primary sensing element, whereas the PLC/DCS modules perform signal conditioning, speed conversion, and recording. It should be emphasized that only this single sensor signal—the casting speed—is used in this study; mechanical faults and other anomalies are not measured by dedicated sensors but are reflected in, and inferred from, the temporal patterns of casting-speed signals, which is precisely the scenario that the proposed anomaly-detection method is designed to handle.

Anomaly types. In the collected casting-speed signals, anomalies appear as abrupt, large-amplitude disturbances superimposed on the normal steady state. According to their temporal forms, the anomalies considered in this study can be categorized into three types: (i) sudden drops of the casting speed to approximately zero, which typically indicate severe interruptions of the strand withdrawal process, such as mechanical jamming of the withdrawal roller, a drive-motor fault, an emergency stop triggered by a mold-level incident (e.g., a sticker breakout precursor), or loss of the encoder/measurement signal; (ii) sudden rises of the casting speed, which usually correspond to an abnormal speed command from the control system or a sudden load change caused by strand breakage or slippage; and (iii) large-amplitude fluctuations around the steady state, which generally reflect slippage between the withdrawal roller and the strand, casting-speed oscillations induced by mold-level fluctuations, mechanical wear, unstable tension, or electromagnetic interference in the sensing chain. These temporal forms correspond to different mechanical and process faults of the continuous casting machine and appear as distinctive patterns in the sensor-derived casting-speed signal, providing the fault categories that the proposed algorithm is expected to detect.

This dataset consists of over 70000 industrial time-series samples gathered across distinct time periods and is collectively named *Steeldata*. To facilitate in-depth analysis, we partition it into three subsets: *Steeldata_1*, *Steeldata_2*, and *Steeldata_3*. *Steeldata_1* and *Steeldata_2* feature multiple anomaly types under diverse operational regimes, while *Steeldata_3* only contains one type of anomaly. This data preparation and experimental design enable a comprehensive evaluation of the proposed algorithm, particularly its capability to handle industrial time-series data exhibiting diverse anomaly patterns as well as data characterized by a single abnormality. Note that the datasets used in this study are not the authors' own; they are real-world industrial time-series data obtained from the historical monitoring system of a continuous casting production line at a steel plant (i.e., the casting-speed sensor records at an actual production site). The data were automatically recorded by the plant's DCS/PLC system at a fixed sampling interval, and the anomaly labels were annotated by on-site process engineers on the basis of production logs and fault records.

4.2 Evaluation metrics

In this study, recall, precision, and F1-score are employed as the evaluation metrics for the proposed model, which aligns with the characteristics and requirements of industrial data.

Precision is defined as the proportion of samples predicted as True by the model that are actually True, also referred to as the positive predictive value. It reflects the accuracy of the model's positive predictions. The formulation is given by

$$Precision = \frac{TP}{TP + FP} . \quad (7)$$

Recall represents the proportion of samples that are truly labeled as True and are correctly identified by the model, and is also referred to as the true positive rate. It reflects the model's ability to correctly recognize positive instances. The metric is defined as

$$Recall = \frac{TP}{TP + FN} . \quad (8)$$

The F1-score represents the harmonic mean of precision and recall, providing a balanced assessment of both metrics. It is particularly suitable for scenarios in which there exists a trade-off between precision and recall. Since precision and recall are two complementary evaluation criteria that are often difficult to improve simultaneously, the F1-score combines them through a harmonic average. Consequently, the F1-score is primarily governed by the smaller of the two metrics: only when both precision and recall are relatively high will the F1-score reach a high value. The metric is defined as

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} . \quad (9)$$

In industrial applications, the number of anomalous instances is typically far smaller than that of normal samples, resulting in a pronounced class imbalance. If accuracy is used as the sole evaluation metric, the model may become biased toward predicting the majority (normal) class, thereby failing to adequately capture its ability to identify anomalous patterns. Consequently, employing recall, precision, and the F1-score as comprehensive evaluation metrics provides a more reliable and informative assessment and is better aligned with the characteristics and practical requirements of industrial data.

4.3 Ablation experiments

A series of experiments were conducted on the three datasets to compare the proposed method with IDTS-DQN (with the original experience replay), IDTS-DQN (with the original reward function), and the standard DQN. Specifically, IDTS-DQN (original experience replay)

restores the experience replay mechanism of the standard DQN while keeping all other components of IDTS-DQN unchanged, and IDTS-DQN (original reward) replaces the proposed reward function with the original one. The standard DQN follows the conventional DQN framework without modification.

The experimental results are summarized in Table 1. We conducted experiments under eight configurations (four methods on each of the two datasets), and the results demonstrate that the proposed DRL algorithm, IDTS-DQN, designed for industrial time-series anomaly detection, performs remarkably well. Across the two primary datasets, all evaluation metrics achieve high levels of performance: the Recall consistently exceeds 90%, and the F1-score surpasses 86%. Among the models compared, IDTS-DQN (original experience replay) yields the second-best overall performance, with its recall, precision, and F1-score higher than those of the remaining two baselines. This can be attributed to the industrial-data-oriented reward function it retains, which enables the agent to receive valid rewards even with limited labeled samples, thereby facilitating effective learning. These findings clearly verify that the proposed reward function is both effective and efficient.

In contrast, the standard DQN exhibits relatively poor performance across both datasets. When replaced with our redesigned reward function and storage strategy, however, the model achieves significantly higher accuracy. Moreover, each of the proposed components—the reward function and the sampling strategy—demonstrates competitive performance even when used independently. Collectively, these results confirm the practical effectiveness of the proposed algorithm: IDTS-DQN can accurately identify known anomalies and learn to detect previously unseen anomalies under the constraint of limited industrial labels. Furthermore, it effectively alleviates the inherent data imbalance issue in industrial time-series data, enabling the anomaly detector to exhibit both strategic decision-making capability and continual learning behavior.

In addition, we conducted experiments on the experience replay strategy to examine whether varying the proportions of samples with different anomaly types and quantities would influence the final performance. Specifically, we adopted nine replay-ratio settings (ranging from 0.1 to 0.9) and performed experiments on both the Steeldata_1 and Steeldata_2 datasets, resulting in a total of eighteen experimental configurations. For comparison, we also evaluated IDTS-DQN (original experience replay), in which the original replay mechanism—corresponding to a proportion of 0—was adopted. The results are presented in Figs. 3 and 4.

Table 1
Recall, precision, and F1-score of each algorithm on the two datasets.

Datasets	Method	Recall	Precision	F1-score
Steeldata_1	IDTS-DQN (ours)	0.9151	0.8334	0.8723
	IDTS-DQN-Original Experience Replay	0.9170	0.8063	0.8581
	IDTS-DQN -Original Reward	0.5431	0.9236	0.6840
	DQN	0.5524	1.0000	0.7117
Steeldata_2	IDTS-DQN	0.9176	0.8175	0.8647
	IDTS-DQN-Original Experience Replay	0.9126	0.6460	0.7565
	IDTS-DQN -Original Reward	0.0493	0.4202	0.0882
	DQN	0.0461	0.3978	0.0827

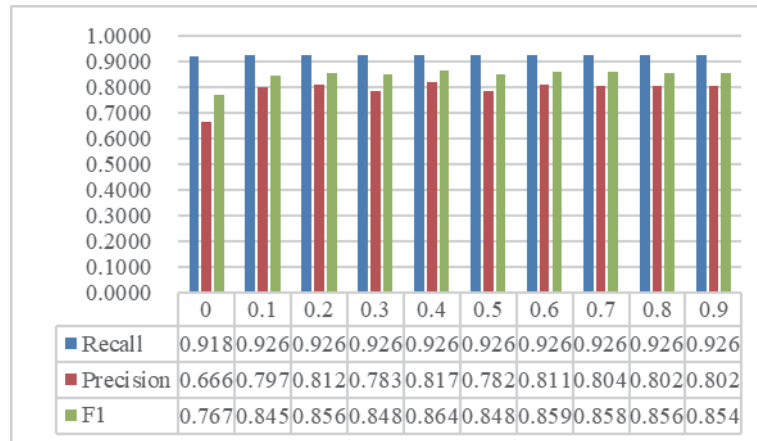


Fig. 4. (Color online) Results of the experience replay strategy on steeldata_2. The horizontal axis denotes the sampling ratio (ranging from 0.1 to 0.9), and the vertical axis denotes the recall, precision, and F1-score of the compared models.

When conducting experiments on the Steeldata_1 and Steeldata_2 datasets, we observed that different experience replay ratios exert a significant impact on model performance. This finding indicates that the ratio configuration within the experience replay strategy is a crucial factor affecting the final outcome of the model. Specifically, on the Steeldata_1 dataset, a ratio of 0.2 achieves the best performance, whereas on the Steeldata_2 dataset, a ratio of 0.4 yields superior results. This difference primarily arises from the varying anomaly types and quantities within the two datasets. Since Steeldata_1 contains more anomalies than Steeldata_2, employing a smaller ratio for datasets with higher anomaly counts and a larger ratio for those with fewer anomalies proves to be more appropriate.

However, the experimental results also demonstrate that simply increasing the ratio does not consistently lead to better performance. As illustrated in the figures, a higher ratio does not necessarily correspond to improved results, implying that selecting an appropriate experience replay ratio must be tailored to the specific characteristics of each dataset. The key lies in identifying a balance point—one that leverages the replay strategy to enhance model performance without causing degradation due to an excessively large ratio.

In summary, regardless of the specific ratio strategy employed, all configurations outperform the traditional experience storage method. This finding underscores the importance and effectiveness of the proposed experience replay ratio strategy in enhancing model performance. By analyzing and comparing reward variations under different ratio settings, we gain deeper insight into how these configurations influence the model's learning process and performance optimization. This ratio-optimization strategy not only improves the model's efficiency and accuracy but also strengthens its adaptability and flexibility across various industrial time-series anomaly detection tasks.

Figure 5 shows reward variations across different ratio strategies in our experiments, revealing the specific influence of each ratio configuration on the model's learning performance.

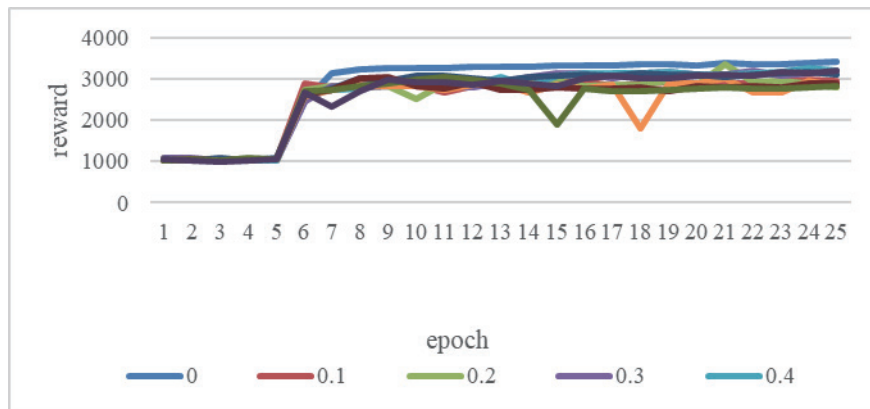


Fig. 5. (Color online) Changes in rewards for each proportion. The horizontal axis denotes the training epoch, and the vertical axis denotes the reward value obtained by the agent under different sampling ratios.

This visual comparison not only facilitates the identification of ratio settings beneficial for performance gains, but also provides valuable insights to guide future model design and optimization in related fields.

4.4 Comparative experiments

We selected a total of six algorithms, including semi-supervised and unsupervised approaches.

DevNet: Deviation Networks (DevNet) is a deep semi-supervised method that leverages a small set of labeled anomalies together with a Gaussian prior over anomaly scores to perform end-to-end anomaly detection.

IForest: Isolation Forest (IForest) is a widely used unsupervised method that detects anomalies on the basis of their relative rarity. It isolates instances through random partitioning in isolation trees, allowing anomalous samples to be identified in fewer steps.

AutoEncoder: This method employs a reconstruction neural network with three hidden layers to estimate the outlierness of data records by measuring reconstruction errors.

DAGMM: Deep Autoencoding Gaussian Mixture Model (DAGMM) utilizes a deep AE to generate low-dimensional representations and reconstruction errors for each input data point, which are subsequently fed into a GMM. The parameters of both the AE and the GMM are jointly optimized.

LSTM-ED: The LSTM encoder–decoder model is trained solely on normal time-series data to reconstruct the input sequences. During testing, anomalous sequences typically produce higher reconstruction errors, indicating that the corresponding time-series data are abnormal.

REBM: This method integrates Energy-Based Models (EBMs) with different types of dataset, including static, sequential, and spatial data. The training algorithm relies on score matching, combining EBMs with a regularized AE to eliminate the need for complex sampling procedures.

By analyzing Steeldata_1 and Steeldata_2, which contain different quantities of anomalies, we demonstrate the effectiveness of our proposed algorithm. Subsequently, we conduct comparative experiments on the Steeldata_1 and Steeldata_3 datasets against six baseline algorithms. In addition, we evaluate performance using standard metrics, including Recall, Precision, and F1-score. Results of the experiments are presented in Table 2.

As presented in Table 2, we evaluated the performance of seven distinct algorithms (six baselines plus IDTS-DQN) via fourteen experimental trials across two datasets, with each algorithm tested twenty times; the optimal results from these runs are reported herein. Notably, our proposed method (IDTS-DQN) exhibits exceptional performance on the Steeldata_3 dataset, achieving the highest scores across all evaluation metrics—with a comprehensive F1-score of 92.36%. Although its performance on the Steeldata_1 dataset shows a moderately lower F1-score of 0.8723, IDTS-DQN still outperforms the other six comparative algorithms by a significant margin.

The DevNet algorithm ranks second to IDTS-DQN on the Steeldata_3 dataset and also performs competitively on Steeldata_1, closely following our method. In contrast, IForest exhibits inferior performance on Steeldata_1 and Steeldata_3 datasets, with F1-scores of 0.0672 on Steeldata_1 and 0.0501 on Steeldata_3. This underperformance may stem from the large scale and high complexity of industrial data, which hinder the algorithm's ability to effectively isolate anomalies, thereby highlighting some limitations of traditional machine learning approaches in industrial anomaly detection.

Although the other algorithms achieve reasonably good results, their performance still falls short when compared with IDTS-DQN, especially in handling industrial data. It is also noteworthy that all seven algorithms perform significantly better on the Steeldata_3 dataset than on Steeldata_1. This discrepancy may be attributed to the fact that Steeldata_1 contains multiple types of anomaly, whereas Steeldata_3 involves only a single anomaly type, making it easier for the algorithms to achieve higher accuracy. Nevertheless, this contrast further underscores the

Table 2
Comparative experimental results of IDTS-DQN and the six algorithms.

Datasets	Method	Recall	Precision	F1-score
Steeldata_3	DevNet	0.8572	1.0	0.9231
	IForest	0.0672	0.0672	0.0672
	AutoEncoder	0.7956	1.0000	0.8862
	DAGMM	0.5956	1.0000	0.7465
	LSTMED	0.6604	1.0000	0.7955
	REBM	0.0025	0.0034	0.0028
	IDTS-DQN (ours)	0.9900	0.8654	0.9236
Steeldata_1	DevNet	0.9645	0.7187	0.8236
	IForest	0.0501	0.0501	0.0501
	AutoEncoder	0.6985	0.9671	0.8111
	DAGMM	0.4001	1.0000	0.5715
	LSTMED	0.7078	0.9561	0.8134
	REBM	0.6398	0.7295	0.6817
	IDTS-DQN (ours)	0.9151	0.8334	0.8723

robustness and superior performance of IDTS-DQN in both single-anomaly and multi-anomaly scenarios, demonstrating its effectiveness and applicability in industrial anomaly detection tasks.

5. Discussion

The comparative results in Table 2 show that IDTS-DQN consistently outperforms six baseline methods on the Steeldata_1 and Steeldata_3 datasets. On Steeldata_1, IDTS-DQN achieves an F1-score of 0.8723, improving by 4.87 percentage points over the strongest baseline, DevNet (0.8236). On Steeldata_3, IDTS-DQN achieves the highest F1-score of 0.9236 among all compared methods. The ablation results in Table 1 further confirm that both the dual-reward mechanism and the anomaly-aware experience replay contribute independently; on Steeldata_1, removing the dual reward decreases the F1-score from 0.8723 to 0.6840, whereas removing the experience replay decreases it to 0.8581, indicating that the reward function plays the dominant role and the replay mechanism provides an additional incremental gain. These observations are consistent with the broader finding that semi-supervised methods can extract substantial diagnostic values from even limited label information when combined with appropriate inductive biases.

From the perspective of sensor technology and sensing applications, the proposed framework addresses a practical need in industrial monitoring: extracting actionable anomaly information from continuous sensor streams with minimal human intervention. In the continuous casting process studied here, a single casting-speed sensor signal—acquired through the encoder–PLC/DCS sensing chain—serves as the sole input. The framework transforms this raw sensor signal into anomaly decisions without requiring additional sensor modalities or manual feature engineering. Such capability aligns with the development of intelligent sensing systems, where the goal is to maximize the diagnostic value extracted from existing sensor infrastructure rather than to deploy additional hardware. Looking forward, the framework can be deployed as a software layer on top of the existing PLC/DCS infrastructure, analyzing sensor signals in real time and alerting operators to developing faults before they escalate into production failures.

Regarding the practicality of the proposed framework, we note that the current evaluation is based on offline analysis of historical sensor data. Nevertheless, several design choices support future online deployment. First, the sliding-window interaction environment processes data sequentially, matching the streaming nature of sensor signals. Second, the anomaly-degree computation relies only on the current window and a reference set of normal subsequences, requiring no access to future data. Third, the model update via experience replay can be performed incrementally as new sensor data arrive, enabling continuous adaptation. The main obstacle to real-time deployment is therefore computational latency. We will characterize inference latency and validate the closed-loop online behavior in a pilot deployment at a continuous casting workshop as part of our future work.

This study falls within the scope of machine learning techniques for sensor data analysis. Specifically, it applies DRL—a paradigm in which an agent learns decision policies through trial-and-error interaction with an environment—to the problem of anomaly detection in

industrial sensor signals. The novelty relative to prior work lies in three aspects: (i) the formulation of time-series anomaly detection as a sequential decision-making problem, which enables the detector to accumulate and leverage detection experience over time; (ii) the dual-reward mechanism that fuses sparse label feedback with intrinsic anomaly measures, addressing the label-scarcity bottleneck that limits supervised and semi-supervised methods; and (iii) the anomaly-aware experience replay strategy, which mitigates the class-imbalance problem inherent in industrial sensor data. In this setting, in the present study, we explore the application of DRL to anomaly detection in continuous-casting casting-speed signals acquired through the encoder–PLC/DCS sensing chain, a scenario that has received limited attention in the literature.

6. Conclusion

In this study, we proposed an innovative anomaly detection framework tailored for industrial time-series data, termed IDTS-DQN. The framework formalizes anomaly perception as a sequential decision-making problem over sliding-window sensor measurements and integrates the strengths of deep learning and DRL. Its design comprises four core components: a sliding-window simulation environment that converts continuous sensor signals into RL states; a dual-reward mechanism that combines sparse label feedback with intrinsic anomaly measures to address the label-scarcity problem; an action space tailored to binary anomaly decisions; and a proportional experience replay strategy that mitigates the inherent class imbalance in industrial sensor data. Extensive ablation studies and comparative experiments on three real-world industrial time-series datasets validate the effectiveness of each component and the overall superiority of the framework.

By coupling DRL with sensor-derived industrial time-series data, the proposed framework provides a practical sensing-data analysis tool for industrial monitoring and fault diagnosis. We therefore position this work as a step toward making anomaly perception a first-class capability of industrial sensor systems, alongside the more traditional measurement and transmission tasks.

Despite the promising results, three limitations should be acknowledged. First, although the proposed framework was validated on real-world industrial sensing data, the experiments were conducted offline on historical datasets rather than in real time at an actual production site; the online practicality of the framework under strict real-time constraints has therefore not been verified, and this is the most pressing issue to be addressed in our future research. Second, the anomaly types considered here correspond to the abrupt disturbances (sudden drops to approximately zero, sudden rises, and large-amplitude fluctuations) observed in the casting-speed signal; extending the framework to a broader range of sensing signals, fault modes, and multi-sensor fusion scenarios requires further investigation. Third, the external reward relies on a small labeled subset, and the sensitivity of the method to the labeling ratio and to the replay-ratio hyperparameter warrants systematic study. In future work, we plan to deploy and validate the framework online in an actual industrial monitoring system, extend it to multiple sensor signals, and develop self-adaptive reward and replay mechanisms to further improve its practicality for intelligent sensing and fault diagnosis.

Acknowledgments

This work was supported by Jiangsu Province Industry University-Research Cooperation Projects (BY20251032 and BY20231236), Suqian Science and Technology Plan Guiding Projects (Z2025040 and Z2023138), and Suqian University Talent Introduction Research Start-up Fund Project (University 2022XRC091).

References

- 1 M. Wadinger and M. Kvasnica: Expert Syst. Appl. **246** (2024) 123200. <https://doi.org/10.1016/j.eswa.2024.123200>
- 2 Z. Zhao, Z. Xiao, and J. Tao: Sensors **24** (2024) 7218. <https://doi.org/10.3390/s24227218>
- 3 Q. Liu, P. Boniol, T. Palpanas, and J. Paparrizos: Proc. VLDB Endow. **17** (2024) 4229. <https://doi.org/10.14778/3685800.3685842>
- 4 F. Wang, Y. Jiang, R. Zhang, A. Wei, J. Xie, and X. Pang: Sensors **25** (2025) 190. <https://doi.org/10.3390/s25010190>
- 5 S. Oswal, S. Shinde, and M. Vijayalakshmi: Commun. Comput. Inf. Sci. **1782** (2023) 221. https://doi.org/10.1007/978-3-031-35644-5_17
- 6 D. Falcão, F. Reis, J. Farinha, N. Lavado, and M. Mendes: Algorithms **17** (2024) 455. <https://doi.org/10.3390/al7100455>
- 7 J. P. S. da Silva and A. L. Maitelli: Sensors **25** (2025) 3532. <https://doi.org/10.3390/s25113532>
- 8 M. Ali, L. T. Jung, A.-H. Abdel-Aty, M. Y. Abubakar, M. Elhoseny, and I. Ali: Expert Syst. Appl. **151** (2020) 113374. <https://doi.org/10.1016/j.eswa.2020.113374>
- 9 Y. Yang, L. Chen, and C. Fan: Soft Comput. **25** (2021) 4283. <https://doi.org/10.1007/s00500-020-05442-1>
- 10 M. Hosseinzadeh, A. M. Rahmani, B. Vo, M. Bidaki, M. Masdari, and M. Zangakani: Soft Comput. **25** (2021) 3195. <https://doi.org/10.1007/s00500-020-05373-x>
- 11 T. J. Alhindi: Mathematics **13** (2025) 3987. <https://doi.org/10.3390/math13243987>
- 12 J. Qu, Q. Du, Y. Li, L. Tian, and H. Xia: IEEE Trans. Geosci. Remote Sens. **59** (2021) 9504. <https://doi.org/10.1109/TGRS.2020.3038722>
- 13 C. Aldrich and X. Liu: Minerals **14** (2024) 76. <https://doi.org/10.3390/min14010076>
- 14 F. Jin, H. Wu, Y. Liu, J. Zhao, and W. Wang: Inf. Sci. **647** (2023) 119479. <https://doi.org/10.1016/j.ins.2023.119479>
- 15 Z. Cheng, S. Wang, P. Zhang, S. Wang, X. Liu, and E. Zhu: Int. J. Intell. Syst. **36** (2021) 7103. <https://doi.org/10.1002/int.22582>
- 16 M. Thill, W. Konen, H. Wang, and T. Bäck: Appl. Soft Comput. **112** (2021) 107751. <https://doi.org/10.1016/j.asoc.2021.107751>
- 17 S. Yan, H. Shao, Y. Xiao, B. Liu, and J. Wan: Robot. Comput.-Integr. Manuf. **79** (2023) 102441. <https://doi.org/10.1016/j.rcim.2022.102441>
- 18 V. T. Le and Y. G. Kim: Appl. Intell. **53** (2023) 3240. <https://doi.org/10.1007/s10489-022-03613-1>
- 19 C. Qian, W. Tang, and Y. Wang: Future Gener. Comput. Syst. **167** (2025) 107751. <https://doi.org/10.1016/j.future.2025.107751>
- 20 A. Z. Sellam, I. Benaissa, A. Taleb-Ahmed, L. Patrono, and C. Distant: Eng. Appl. Artif. Intell. **160** (2025) 111685. <https://doi.org/10.1016/j.engappai.2025.111685>

About the Authors



Liming Du received his B.S. and M.S. degrees from Shenyang Jianzhu University in 2000 and 2003, respectively, and his Ph.D. degree from Northeastern University, China, in 2015. He worked at Shenyang Jianzhu University from 2003 to 2022 and has since been an associate professor at Suqian University and a part-time master's supervisor at Shenyang Jianzhu University. His research interests include machine learning and intelligent software testing. (duliming@squ.edu.cn)



Guolin Wang received his B.S. degree from Hefei University, China, in 2020 and his M.S. degree from Shenyang Jianzhu University, China, in 2024. Since 2024, he has been a Teaching Assistant specializing in Artificial Intelligence at the School of Electronics and Information Engineering, Huaibei Institute of Technology, China. His research interests include fault diagnosis, anomaly detection, time series analysis, and large-model applications. (1337827548@qq.com)



Fengying Wang received her B.S. degree from the University of Science and Technology Liaoning in 2001 and her M.S. degree from Shenyang Jianzhu University in 2004. She worked at Shenyang Jianzhu University from 2004 to 2022 and has since been an associate professor at Suqian University and a part-time master's supervisor at Shenyang Jianzhu University. Her research interests include machine learning, intelligent software engineering, and visual inspection. (wangfengying@squ.edu.cn)