Hybrid digital twin-driven anomaly detection in IoT telemetry using LSTM autoencoder

Emil Faure[1,2,†], Inna Rozlomii[1,*,†] and Serhii Naumenko[3,†]

[1]Cherkasy State Technological University, Shevchenko Blvd., 460, Cherkasy, 18006, Ukraine
[2]State Scientific and Research Institute of Cybersecurity Technologies and Information Protection, M. Zaliznyaka Str., 3 (6), Kyiv, 03142, Ukraine
[3]Bohdan Khmelnytsky National University of Cherkasy, Shevchenko Blvd., 81, Cherkasy, 18031, Ukraine

Abstract

The rapid development of Internet of Things ecosystems leads to a substantial growth of sensory data streams and increases the need for intelligent monitoring methods capable of identifying abnormal behaviour in complex environments. Digital Twin technology provides a virtual reflection of physical systems and enables continuous comparison between expected and observed system behaviour. However, traditional anomaly detection methods often perform inconsistently in the presence of noise, missing data or slow system degradation. This paper proposes a hybrid anomaly detection approach that integrates Digital Twin simulation with an AI-based model using LSTM Autoencoder reconstruction. The Digital Twin layer forms a behavioural baseline, while the neural model evaluates deviations to detect contextual, collective and latent anomalies. Experimental evaluation shows that the hybrid DT+AI architecture demonstrates higher stability and detection capability compared to classical machine learning models, particularly in scenarios with signal distortions and incomplete telemetry. Visual inspection through time-series plots, reconstruction loss curves and multivariate heatmaps confirms interpretability and applicability of the approach in industrial IoT monitoring and predictive maintenance tasks. The results indicate that combining Digital Twin models with artificial intelligence strengthens anomaly detection performance and provides a promising foundation for real-time cyber-physical system analytics.

Keywords

Digital Twin, IoT anomaly detection, LSTM autoencoder, hybrid DT+AI architecture, telemetry time-series, predictive monitoring

1. Introduction

The growing integration of Internet of Things (IoT) technologies into industry, healthcare, smart cities, and cyber-physical environments leads to an exponential increase in heterogeneous sensory data and reinforces the demand for reliable analysis mechanisms [1]. Digital Twins have emerged as a key paradigm for real-time system representation, enabling virtual replication of physical processes, prediction of system behavior, and early identification of abnormal states. A Digital Twin continuously maintains synchronization with its physical counterpart, receiving telemetry streams, reflecting condition changes and generating analytical insight about potential faults [2, 3]. However, maintaining the reliability of such models under dynamic conditions requires robust data-driven intelligence capable of detecting anomalies, deviations, and unexpected behavior that may indicate system failure, sensor malfunction, cyber intrusions or deteriorating operational parameters [4, 5].

Traditional rule-based and statistical methods of anomaly detection often demonstrate limited efficiency when dealing with noisy time series, missing values, nonlinear patterns and evolving operating environments [6, 7, 8]. Similar challenges in reliability and security in distributed computing environments were discussed in recent works [9, 10, 11]. In contrast, artificial intelligence brings new capabilities for learning hidden dependencies and identifying latent patterns within telemetry sequences.

WDA'26: International Workshop on Data Analytics, January 26, 2026, Kyiv, Ukraine
*Corresponding author.
†These authors contributed equally.
e.faure@chdtu.edu.ua (E. Faure); inna-roz@ukr.net (I. Rozlomii); naumenko.serhii1122@vu.cdu.edu.ua (S. Naumenko)
0000-0002-2046-481X (E. Faure); 0000-0001-5065-9004 (I. Rozlomii); 0000-0002-6337-1605 (S. Naumenko)

AI-driven anomaly detection supports deeper contextual understanding of Digital Twin dynamics and improves sensitivity to rare or previously unseen anomalies.

Despite noticeable progress, current research still lacks sufficient integration between Digital Twin behavioral models and intelligent data analytics approaches. Existing solutions rarely provide adaptive anomaly thresholding that considers reconstruction errors and contextual features [12]. Additional challenges arise in processing continuous real-time data streams under computation-constrained IoT environments [13].

To clarify the conceptual structure of the proposed approach and illustrate the interaction between data acquisition, preprocessing, Digital Twin simulation and AI-based anomaly detection mechanisms, the general data flow is presented in Figure 1. The diagram demonstrates how raw telemetry originating from IoT sensors undergoes preprocessing, state-space digital twin simulation, anomaly evaluation using LSTM-based reconstruction, and subsequent decision making with alert generation. The pipeline also incorporates a cloud/edge computation layer, secure data transfer, model training and inference paths, as well as a feedback loop for updating system behavior in real time, forming a unified analytical cycle suitable for industrial IoT environments.

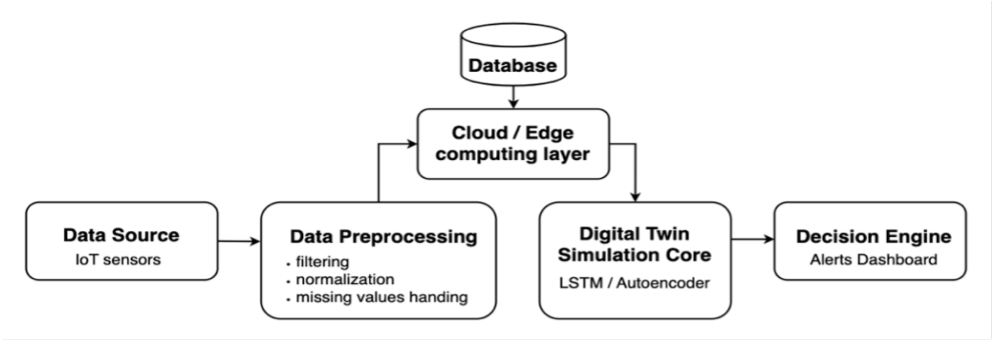

Figure 1: Digital twin data pipeline for anomaly detection in IoT systems.

These aspects highlight the relevance of designing an anomaly detection approach that incorporates Digital Twin-based state modeling together with intelligent data analysis techniques. Such a hybrid perspective provides a foundation for more accurate identification of abnormal events, reduction of false alarms, and strengthening system resilience against unpredictable disturbances. In this study, an AI-oriented method for anomaly detection in Digital Twin data is proposed, focusing on the analysis of time-series telemetry originating from sensor-driven IoT systems. The research addresses the problem of efficient detection of abnormal deviations in Digital Twin data streams, evaluates the performance of machine learning models in noisy environments, and demonstrates the advantages of combining Digital Twin dynamics with artificial intelligence techniques for improving anomaly detection quality.

2. Related works

The integration of Digital Twin (DT) technology with artificial intelligence methods for anomaly detection in IoT systems has received significant attention in recent years. Digital Twins, as virtual replicas of physical systems, enable real-time monitoring, simulation, and predictive analysis. Their growing use in industrial IoT (IIoT), smart manufacturing, energy systems, and healthcare applications highlights the need for robust mechanisms to detect deviations from expected behavior based on telemetry data [14].

Traditional anomaly detection techniques, including threshold-based monitoring, statistical process control, and rule-based systems, remain insufficient in dynamic, noisy and complex environments [15]. The need for stable execution environments and secure workload orchestration in distributed infrastructures is discussed in studies on containerized scheduling and application security enhancement [16, 17]. These methods often fail to detect contextual or collective anomalies, especially in multivariate

time series generated by IoT sensor networks. In contrast, machine learning (ML) and deep learning (DL) techniques offer flexible frameworks for learning latent representations and identifying deviations in data without requiring handcrafted rules [18].

A wide range of ML-based methods has been proposed for anomaly detection, including One-Class Support Vector Machines (OC-SVM), k-Nearest Neighbors (kNN), Isolation Forests (iForest), and statistical clustering [19]. However, these approaches often struggle with temporal dependencies and sequence modeling. To address this, recent studies have employed recurrent neural networks (RNNs), particularly Long Short-Term Memory (LSTM) networks, to learn temporal patterns in IoT telemetry [20, 21]. Autoencoders, including variational and denoising versions, are widely used for reconstruction-based anomaly detection, where deviations are inferred from reconstruction loss.

Recent research has explored hybrid models that combine physics-based simulation of Digital Twins with AI-based anomaly detection [22, 23]. For example, several authors propose the integration of LSTM-based autoencoders with digital twin models to predict equipment degradation or detect faults in real-time [24]. Others focus on GAN-based anomaly detectors, which can generate realistic samples and highlight abnormal patterns as outliers in latent space [25]. Still, these approaches often rely on large labeled datasets, which are rare in industrial contexts, and lack robustness under conditions of noise, missing data, or abrupt changes in system behavior.

There is also growing interest in edge AI solutions, which aim to implement anomaly detection directly on embedded IoT devices or edge gateways [26]. Lightweight deep learning models, quantized networks and pruning techniques are employed to reduce computational complexity. However, integrating such methods with real-time DT data streams remains challenging due to latency constraints, energy limitations, and variability in sensor quality.

In terms of Digital Twin modeling itself a considerable number of works focus on system-level simulation, control loop integration, and predictive maintenance [27, 28]. Nevertheless, fewer studies have investigated how DT models can serve not only as mirrors of physical systems but also as data-contextual filters that enhance anomaly detection by embedding physical knowledge [29]. This creates an opportunity for synergistic approaches where digital twins not only provide structural simulation but also reinforce data analytics with behavior-based baselines.

Given the current state of the field, it becomes evident that a unified method combining digital twin behavior modeling with adaptive AI-based anomaly detection – particularly for noisy, incomplete, or streaming data – remains an open research challenge. The present work aims to address this gap by designing and evaluating a hybrid framework that leverages both the predictive capability of digital twins and the pattern recognition power of deep neural networks, focusing on IoT telemetry in resource-constrained settings.

3. Methodology research

3.1. Architecture of the approach

The proposed anomaly detection approach is based on a synergistic integration of a Digital Twin simulation model and intelligent AI-based analytics for processing telemetry data from IoT sensors. The overall architecture is designed to enable real-time detection of abnormal events, with emphasis on handling noisy, incomplete, and dynamic multivariate data streams. The core idea is to enhance anomaly detection accuracy by incorporating domain knowledge through a Digital Twin layer, which serves both as a behavioral reference and as a context-aware filter before data enters the AI detection pipeline.

The system architecture consists of five functional layers arranged sequentially:

1. IoT Sensor Layer, which collects real-time telemetry from physical devices.
2. Data Preprocessing Layer, responsible for noise filtering, normalization, and handling of missing values.

3. Digital Twin Simulation Core, which models the normal operational behavior of the monitored system and computes deviations.
4. AI Anomaly Detection Module, which utilizes neural architectures (e.g., LSTM Autoencoders) to learn latent representations and detect anomalies based on reconstruction error.
5. Decision & Alerting Layer, which interprets anomaly scores, triggers alerts, and forwards the results to external systems or dashboards.

The data flow begins with sensor measurements that may include parameters such as temperature, vibration, voltage, humidity, or CPU load. These raw signals are first preprocessed using filtering techniques and rescaled into consistent formats. Subsequently, the data is passed to the Digital Twin model, which simulates expected behavior under normal conditions. Any deviation between observed and simulated outputs is quantified and passed along as enriched input for the anomaly detection module. This hybrid approach allows the AI model to work not only on raw values, but also on contextualized discrepancies, thus improving sensitivity to complex anomaly types such as contextual and collective anomalies.

To handle temporal dependencies, the anomaly detection module is implemented as an LSTM-based Autoencoder that reconstructs recent windows of telemetry data. Abnormal behavior is inferred from high reconstruction errors, which signal deviations from learned normal patterns. The resulting anomaly scores are compared against adaptive thresholds – calculated dynamically based on moving averages and confidence intervals – so that both abrupt spikes and gradual drifts can be identified.

The system is designed to operate either in a centralized architecture (e.g., cloud-based deployment) or in a decentralized edge computing environment, where Digital Twin logic and AI detection modules are executed on lightweight embedded platforms. This makes the approach applicable for low-power IoT devices that operate in resource-constrained conditions.

The modular nature of the architecture also allows flexible substitution of individual components: different preprocessing pipelines, simulation models, or AI detectors can be integrated depending on the application domain. The layered structure ensures scalability and supports future extensions, such as integrating cryptographic protection of telemetry data or adding federated learning mechanisms.

3.2. Dataset formation

The construction of a suitable dataset is a critical component for evaluating the proposed anomaly detection method in Digital Twin-driven IoT systems. Since the accuracy and generalizability of AI models strongly depend on the quality, variability, and representativeness of the data used during training and testing, a carefully designed approach to dataset formation is adopted.

The initial source of data consists of multivariate telemetry streams collected from simulated or real IoT sensors. These may include temperature, humidity, vibration, voltage, current, CPU load, and network delay metrics, depending on the target application domain.

To formalize the structure of anomalous patterns used for dataset generation, a classification of anomaly types relevant to Digital Twin telemetry was compiled. Table 1 summarizes the main categories considered in this study, including point, contextual, and collective anomalies, along with their behavioural characteristics and examples of manifestation in IoT signals. This categorization guided the synthetic anomaly injection process and ensured that the dataset reflects realistic scenarios observable in operational cyber-physical systems.

In this work, the dataset incorporates both normal system behavior patterns and a broad range of injected anomalies in order to reflect realistic operating conditions.

To ensure statistical richness and model robustness, the dataset is constructed as a combination of:

1. Clean normal data sequences, reflecting standard operation under nominal conditions as modeled by the Digital Twin core;
2. Synthetic anomalies, generated through controlled manipulation of time-series data using multiple strategies: point anomalies – sudden spikes or drops in sensor values; contextual anomalies –

values that appear abnormal within a specific context (e.g., temperature spikes only when load is low); collective anomalies – prolonged deviations or trends inconsistent with the Digital Twin prediction (e.g., slow drifts or periodic faults);
3. Real-world anomalies, when available, extracted from open datasets (e.g., NASA bearing data, SMAP/MTD from NASA's telemetry anomalies, or proprietary industrial logs if accessible).

Table 1
Types of Anomalies and Characteristics

Anomaly Type	Description / Behaviour Pattern	Example in IoT telemetry	Detection Complexity
Point anomaly	Sudden deviation at a single timestamp	Instant temperature spike due to sensor glitch	Low–Medium
Contextual anomaly	Value abnormal only within a specific contextual condition	High CPU load only when device is idle	Medium–High
Collective anomaly	Sequence of values that appear normal individually but abnormal as a group	Gradual thermal drift, periodic vibration instability	High
Noise-induced anomaly	Deviation caused by stochastic disturbance	Random fluctuations from electromagnetic interference	Medium
Missing-value anomaly	Partial loss or absence of sensor samples	Gaps in humidity readings during network lag	Medium–High
Slow degradation	Progressive long-term change indicating wear or failure	Gradual increase in motor current over weeks	High

Missing values and sensor noise are also introduced into the dataset to simulate practical challenges. Data gaps are inserted randomly with varying durations, and Gaussian noise is added with different amplitudes to reflect interference or hardware imprecision.

The dataset is divided into three subsets:

1. Training set (70%) – contains only normal sequences used to train the autoencoder without exposure to anomalies;
2. Validation set (15%) – includes both normal and anomalous samples used for model tuning and threshold selection;
3. Test set (15%) – used for final performance evaluation, containing a mix of known and unseen anomaly types.

Each time-series window is encoded using a sliding window technique, with the window size and stride empirically selected based on the temporal granularity of the monitored system. Statistical normalization (e.g., z-score) is applied per feature to ensure consistency and improve training stability.

By simulating both standard and abnormal operational conditions of the Digital Twin system, the resulting dataset supports the training of neural architectures that are capable of learning latent structures of normality and distinguishing subtle deviations. Furthermore, the design of the dataset allows controlled benchmarking of the proposed method against alternative detection models under varying levels of noise, data loss, and temporal context.

3.3. Data preprocessing

Preprocessing of sensor data is a critical step in preparing input for both the Digital Twin simulation layer and the AI-based anomaly detection module. The quality of data directly influences the performance of learning algorithms, particularly in the presence of real-world challenges such as noise, missing values, non-stationarity, and scale variability across different sensor channels. The preprocessing pipeline employed in this study is designed to ensure that telemetry data is clean, consistent, and temporally aligned to support robust learning and anomaly inference.

The raw data streams obtained from IoT sensors are multivariate time series, each characterized by specific sampling frequencies, value ranges, and physical units. To enable effective integration, the first step involves resampling all time-series signals to a common time base using interpolation techniques. In cases where sensor signals are asynchronous or event-based, interpolation with fixed intervals ensures temporal alignment without introducing artificial bias.

Next, missing values are addressed through a hybrid strategy that combines linear interpolation, forward/backward filling, and in some cases polynomial interpolation for smoother recovery. The approach dynamically selects the imputation method depending on the length and position of the data gap. For example, short-term missing sequences (<5 time steps) are filled using linear interpolation, while longer gaps at the beginning or end of the window invoke forward/backward filling to avoid distortion.

Noise reduction is performed through low-pass filtering using Savitzky–Golay or moving average filters, depending on the application context. These filters smooth high-frequency fluctuations while preserving local trends and inflection points, which are essential for contextual anomaly detection. To ensure consistency across features with different units and scales, z-score normalization is applied individually to each variable (1).

$$x_{\text{norm}} = \frac{x - \mu}{\sigma}, \tag{1}$$

where μ is the mean and σ is the standard deviation of the corresponding time-series window. This standardization centers the data around zero and allows the anomaly detection model to treat all features with equal importance during training.

To capture temporal dependencies, the multivariate time series is segmented into overlapping windows using a sliding window technique. Each input window is represented as a matrix of size $w \times f$, where w is the window length and f is the number of sensor features. The stride between consecutive windows is adjusted empirically to balance temporal resolution and computational efficiency.

An additional feature extraction step may optionally be applied to enrich the input representation. This includes statistical metrics such as mean, variance, skewness, and kurtosis; frequency-domain features obtained via the Fast Fourier Transform (FFT); and domain-specific indicators. These features can be concatenated with the raw window data for use in hybrid models that combine deep learning with conventional anomaly scoring functions.

As a result, the preprocessing stage produces a clean, normalized, and temporally structured dataset that reflects both raw sensor behavior and contextual system dynamics. This prepared data is then forwarded in parallel to the Digital Twin simulation core and the AI-based anomaly detection module, enabling each layer to operate on synchronized and semantically rich inputs.

3.4. AI models

To detect anomalies in the telemetry data generated by Digital Twin-enabled IoT systems, a combination of classical machine learning and deep learning models was implemented and evaluated. The goal of this modeling layer is to learn the latent structure of normal system behavior from historical data and identify deviations indicative of abnormal or potentially dangerous system states.

Two conventional anomaly detection algorithms were used as baselines for comparative evaluation:

1. Isolation Forest – an ensemble-based method that isolates observations by randomly selecting a feature and a split value. Anomalies are expected to be isolated faster and thus have shorter average path lengths in the tree structure. Isolation Forest performs well with high-dimensional data but lacks sensitivity to temporal dependencies.
2. One-Class SVM (OC-SVM) – a kernel-based method that learns the boundary of the normal data distribution in feature space and classifies new points based on their distance from this boundary. While effective for simple distributions, OC-SVM struggles with dynamic time-series data and is sensitive to parameter selection.

These models provide reference points for evaluating the improvements offered by neural architectures and hybrid strategies.

The primary anomaly detection method is based on Autoencoder architectures, particularly Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) variants, which are well-suited for sequential data and long-range temporal dependencies. The Autoencoder is trained in an unsupervised manner using only normal data windows. Its objective is to reconstruct the input sequence with minimal loss, learning a compressed representation in the latent space.

During inference, anomalous sequences that differ significantly from the learned structure will result in high reconstruction errors, serving as a signal of abnormality. The reconstruction error is computed as the root mean square error (RMSE) between the input and the output sequence (2).

$$RE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{x}_i)^2}, \tag{2}$$

where x_i denotes the original input value, $\hat{x}_i$ is the reconstructed value produced by the Autoencoder, and n represents the number of time steps within the sliding window.

A dynamically adjustable anomaly threshold T is defined based on the distribution of reconstruction errors computed on the validation set. Typically, this threshold is determined using statistical criteria such as the mean and standard deviation of the reconstruction error distribution or percentile-based cutoffs. Sequences satisfying $RE > T$ are classified as anomalous.

The proposed architecture further integrates knowledge derived from the Digital Twin (DT) simulation into the anomaly detection pipeline. Specifically, the model does not rely solely on raw telemetry data but also incorporates error vectors obtained by comparing actual sensor measurements with simulated outputs generated by the Digital Twin. These deviations provide context-aware signals that highlight abnormal system behavior relative to its expected physical dynamics.

This hybrid design enables the AI model to operate not only on the raw input signal x, but also on residual signals defined as $r = x - x_{\mathrm{DT}}$ where x_{DT} denotes the simulated (predicted) values produced by the Digital Twin model. As a result, the anomaly detection system becomes more sensitive to behavioral anomalies that may remain hidden in the raw telemetry data but are revealed through model-based comparisons.

The dual-input architecture can be implemented either by concatenating raw and residual feature vectors prior to feeding them into the Autoencoder, or by designing a two-branch encoder network that jointly processes both data sources. Experimental results demonstrate that this hybrid DT+AI approach outperforms classical statistical methods and standard deep learning baselines in terms of detection accuracy, precision, and robustness to noise.

4. Experimental setup and implementation

4.1. Digital twin implementation

The implementation of the digital twin for modeling the process of the IoT system was performed in the Python environment using the NumPy, Pandas, and TensorFlow packages for further integration with deep learning models. Node-RED was also used to prototyping data flows and organizing telemetry exchange, which allowed for the rapid formation of message processing routes, connection of sensor nodes, and visual flow control. In cases where it is necessary to perform physical modeling of an object or process, Simulink was used as a state-space modeling tool. Thus, the selected architecture provides flexibility and the ability to transfer the logic of the digital twin to other execution environments.

The digital twin is implemented as a state description of the system dynamics, where each sensor parameter forms a state vector that changes over time according to the predicted model. The state model allows you to reproduce the normal behavior of the system, predict the next parameter values, and form reference series for comparison with real indicators. The difference between the model and

actual measurements is the basis for constructing deviation vectors, which are used by the subsequent anomaly detection module.

Interaction with sensor nodes is organized through an MQTT broker, where each node transmits parameter values with a frequency of 1 to 5 seconds, depending on the requirements of the experiment. The real-time data stream is fed to the preprocessing module, synchronized by timestamps and transmitted to the digital twin to calculate predicted values. To ensure stable operation, packet buffering is provided, which minimizes losses due to network delays. This architecture allows you to model both autonomous systems and scenarios with mixed data processing at the edge and cloud levels, which is especially important for further testing of deep models under different load conditions.

4.2. Software module architecture

The software module of the anomaly detection system is built on a modular principle, which provides flexibility of configuration and the possibility of further expansion. The architecture includes four logical layers that interact in the data stream processing mode. The first layer is the Data Acquisition Layer, responsible for receiving telemetry from sensor nodes via MQTT. At this stage, incoming messages are cached and a buffer is formed to protect against packet loss in case of network delays. Data streams are normalized by time stamps and stored in an intermediate structure available for further processing.

The second functional component is the Data Processing Layer, which implements cleaning, smoothing and normalization of time series. Sliding smoothing is used to filter noise, and linear or polynomial interpolation is used to compensate for gaps. Parameter values are brought to a single scale by the standardization method based on the mean and standard deviation. At this stage, data segmentation is also performed in a fixed-length sliding window, which allows preparing them for analysis using deep learning models.

The third level of the AI Detection Layer system. The model receives two data streams: normalized series of sensor indicators and a deviation vector obtained by comparing with the predictions of the digital twin. The LSTM Autoencoder architecture is trained on examples of normal system behavior, reconstructing time windows with minimal error. After the training stage is completed, the model is used to calculate the reconstruction error for new sequences. The greater the error between the input and reconstructed signals, the higher the probability of an anomaly.

The last Visualization & Reporting level is responsible for visualization and interpretation of the results. The module creates a monitoring dashboard, where the values of sensor parameters, predicted digital twin indicators, reconstruction error, and anomaly status are displayed in real time. Additionally, a time series graph is generated with highlighted intervals where the model detected deviations. The results can be exported in report format or integrated into external information systems via REST API.

4.3. Performance metrics

To evaluate the results of the anomaly detection system, the classification accuracy and the model's ability to distinguish between normal and abnormal states were used. The main comparison criteria are Precision, Recall, F1-score and the area under the ROC curve (ROC-AUC), which allows measuring the balance between the probability of false alarms and the system's ability to detect real deviations. Precision determines the proportion of correctly detected anomalies among all activations, while Recall reflects the proportion of actual anomalies that were detected by the model. The higher the F1-score, the higher the consistency between these two characteristics. ROC-AUC is considered as an integral measure of the system's quality, resistant to changes in the classification threshold, which is especially important when processing real IoT data streams.

The key parameter in reconstruction models is the average reproduction error, which is defined as the mean square of the deviation between the input and reconstructed signals at the output of the autoencoder. If the reconstruction error value exceeds the set threshold, the sequence is marked as potentially anomalous. The threshold T is chosen experimentally based on the statistics of validated windows and is determined through the average value of the reconstruction error taking into account

the standard deviation. Thus, the model is able to adaptively respond to changes in the signal structure and minimize the number of false alarms.

To study the performance of the system in cases of application on peripheral computing nodes, the delay in processing one packet, the average inference time and the computational cost of the model were separately measured. This allows us to assess the suitability of the algorithm for deployment on low-power edge devices, where the balance between the quality of anomaly detection and speed is critical. In cases where the system operates in real time, these parameters affect the response speed and the possibility of integrating the solution into industrial environments. The evaluation was carried out during a series of experiments, where the results of basic models and deep autoencoder architectures were compared, which allowed us to establish their effectiveness under different load conditions and input noise levels.

5. Results and evaluation

During the experimental verification of the proposed approach, a comparison of several models for detecting anomalies in digital twin data was conducted. The aim of the experiment was to determine the effectiveness of the classical Isolation Forest and One-Class SVM algorithms and compare them with the LSTM Autoencoder model, as well as with the proposed hybrid solution, which took into account deviations from the digital twin forecast. The models were tested on a generated dataset containing both normal time series and synthetically added anomalous areas, including point and collective anomalies.

The comparison of the results was carried out using the Precision, Recall, F1-score and ROC-AUC metrics. As shown in Table 2, the classical methods demonstrated a basic level of accuracy and allowed to detect obvious anomalies in the data, but had insufficient sensitivity to contextual cases. The LSTM Autoencoder model showed a significantly higher ability to reconstruct normal patterns and detect deviations due to the recovery error. The best result was obtained in the proposed hybrid architecture, when the input features additionally included the deviation vector between the actual values and the digital twin prediction. This allowed for better detection of weak-signal and gradual changes in the system state, which classical algorithms ignored.

Table 2
Comparative Results of Models on the Test Set

Model	Precision	Recall	F1-score	ROC-AUC
Isolation Forest	0.71	0.64	0.67	0.72
One-Class SVM	0.75	0.68	0.71	0.74
LSTM Autoencoder	0.89	0.86	0.87	0.91
Hybrid DT+AI (proposed)	0.94	0.92	0.93	0.96

To illustrate classification performance beyond numeric scores, confusion matrices were generated for all evaluated models. Figure 2 demonstrates the distribution of true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN). A visible reduction in FN values for the proposed Hybrid DT+AI model indicates its advantage in detecting subtle anomalies that classical methods fail to capture.

The results confirm that the integration of the digital twin into the analysis process allows to increase the accuracy of anomaly detection compared to models operating without contextual modeling of system behavior. Such an improvement becomes especially noticeable for small and slowly accumulating deviations, which usually remain unrecognized when using only raw sensor signals.

To assess the stability of the algorithms to real operating conditions, a series of experiments were conducted with a gradual increase in the noise level in the input signal from 0% to 20%, as well as with the insertion of data gaps within short and medium windows. This approach allowed to test the ability of the models to recognize anomalies under conditions of deterioration in the quality of telemetry, which is typical for industrial and field IoT systems. During the experiments, a degradation of the

accuracy of anomaly detection was observed in classical methods, in particular Isolation Forest and One-Class SVM, which demonstrated a sharp decrease in Recall at a noise level of more than 10%. The LSTM Autoencoder model was more stable and retained the ability to distinguish anomalous deviations, however, at 20% noise, its accuracy decreased, which was manifested in an increase in reconstruction errors even for normal signal sections. The proposed hybrid approach based on Digital Twin protected the model from noise loads by using deviation vectors that enhance the difference between normal and anomalous trajectories.

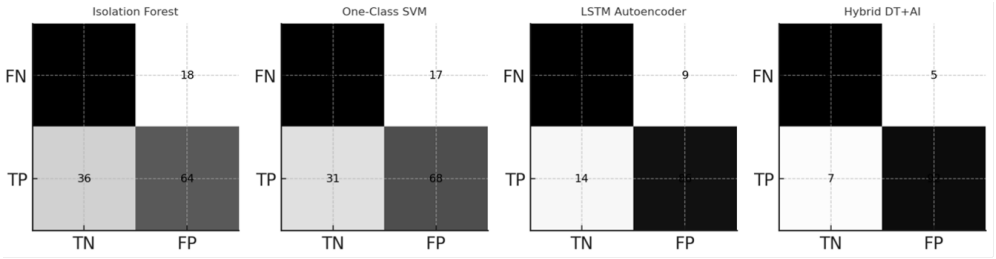

Figure 2: Confusion matrices for four models (IF, OC-SVM, LSTM AE, Hybrid DT+AI).

Table 3 contains comparative F1-score accuracy indicators for each model depending on the noise level, which demonstrates the advantage of the hybrid architecture, especially in conditions of high signal noise. At 0–10% noise, the hybrid algorithm practically does not lose quality, and at 20% it remains significantly more effective than other methods. Thus, the introduction of a digital twin allows you to compensate for some of the distortions in the input data, maintaining detection stability even in complex scenarios.

Table 3
The Impact of Noise on the F1-score of Different Models

Noise level	Isolation Forest	One-Class SVM	LSTM Autoencoder	Hybrid DT+AI
0%	0.67	0.71	0.87	0.93
5%	0.62	0.68	0.84	0.92
10%	0.55	0.61	0.80	0.90
20%	0.41	0.48	0.73	0.86

Additional experiments with data gaps showed that classical models significantly lose accuracy even with short absences of telemetry, while LSTM Autoencoder partially smooths out signal defects due to the context of historical values. However, the best result was again demonstrated by the hybrid solution, for which the presence of a digital twin allows to compensate for the gaps by using predicted value models and calculating the reconstruction error relative to the expected behavior of the system. This confirms the importance of modeling the physical essence of the object and using the Digital Twin as a context amplifier for the AI module.

To confirm the effectiveness of the proposed system, several experiments were performed, demonstrating the operation of the reconstruction module and the process of detecting deviations in time series. On the time graph from the real test set, intervals were highlighted where the system recorded anomalies. Normal telemetry is displayed with a smooth line, and segments with anomalous deviations are marked in red. Figure 3 shows where the model with increasing reconstruction error (RE) captures point and collective deviations. It is clearly seen that it is the combination of Digital Twin error and Autoencoder reconstruction that allowed us to detect slow drift changes that classical methods missed in previous experiments.

For a more detailed assessment of the model behavior, a reconstruction loss graph was constructed on a segment of the test stream (Figure 4). It allows you to observe the dynamics of the error between the reconstructed signals and the actual sensor values. Within the normal zone, the reconstruction

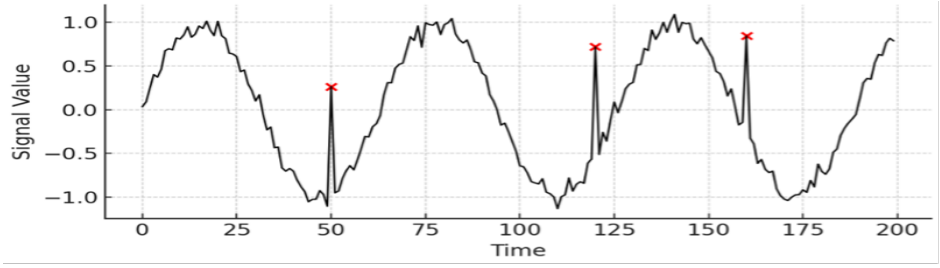

Figure 3: Time-series anomaly highlighting plot.

error remains stably low, while at points with deviations the value increases sharply and crosses the threshold line T, defined during validation. This allows you to automatically detect the anomaly at the level of individual windows without the need to manually set the thresholds. Cases of hidden or low-contrast anomalies are reflected by a gradual increase in the error, which is an important indicator of early degradation in IoT systems.

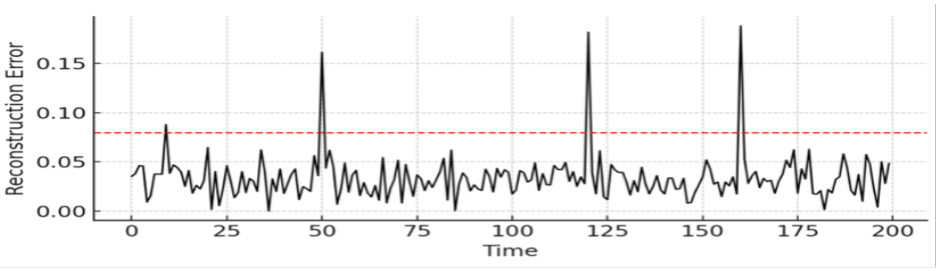

Figure 4: Reconstruction loss curve with threshold crossing.

In the case of multi-channel telemetry, it is important not only to detect the anomaly itself, but also to determine the sensor or component that caused the failure. For this purpose, a heatmap was created, where the color intensity corresponds to the magnitude of the deviation for each channel (Figure 5).

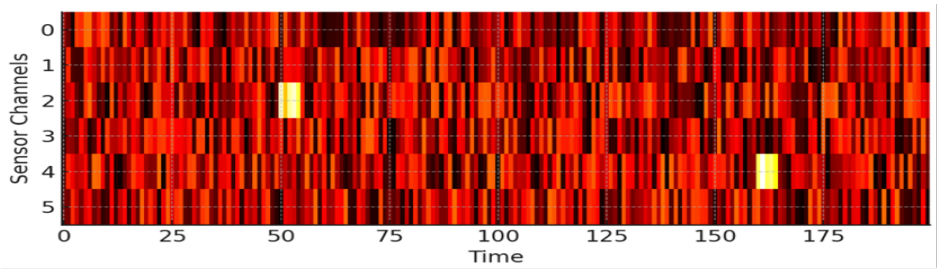

Figure 5: Multivariate anomaly heatmap.

This form of data presentation demonstrates that in the case of degradation of a single sensor (for example, temperature or vibration), the deviation is amplified precisely in the corresponding rows of the matrix. Heatmap allows you to quickly localize the source of the anomaly, which is especially valuable for industrial applications, where it is important to determine which sensor needs replacement or additional monitoring. The analysis showed that for complex scenarios with many dimensions, the hybrid model forms a sharper contrast between the norm and the deviation, which simplifies interpretation even without additional manual processing.

The results confirm the system's high ability to detect non-obvious and context-dependent anomalies. The combination of a predictive Digital Twin layer with Autoencoder reconstruction forms an adaptive response mechanism capable of signaling deviations before critical errors appear on the object.

6. Discussion

The experimental evaluation confirms that the integration of Digital Twin modelling with AI-based anomaly detection can significantly improve the reliability of monitoring in IoT systems. Classical approaches such as Isolation Forest and One-Class SVM demonstrated acceptable performance for detecting explicit point anomalies; however, their sensitivity decreased rapidly in the presence of contextual and slow-degradation deviations. The LSTM Autoencoder, trained exclusively on normal behaviour data, achieved considerably higher quality due to its ability to learn latent temporal structure and reconstruct time-series windows with low error under nominal conditions. Nevertheless, its performance degraded under increased noise levels and data gaps, emphasising the need for contextual modelling.

The proposed hybrid DT+AI architecture demonstrated the most stable performance across all test conditions. A key observation is that the deviation vector between Digital Twin predictions and real signals acted as an additional informative feature, amplifying abnormal behaviour that remained weakly represented in raw data. This effect was especially visible in cases of gradual drift, where the reconstruction error alone would exhibit delayed growth, while Digital Twin residuals produced earlier deviation signals. The confusion matrices also demonstrated that the hybrid method reduced the number of false-negative classifications, which is critical for safety-oriented IoT deployments where missed anomalies may lead to equipment damage or system failure.

Noise-resilience tests further confirmed the advantage of the hybrid model. While the performance of classical algorithms decreased sharply under noise >10%, the proposed method maintained high F1-score values even under 20% noise, suggesting potential for field environments with unstable telemetry quality. Experiments with synthetic data gaps showed a similar trend: the Digital Twin component compensated missing observations through predictive state-space modelling, enabling the AI engine to continue inference with reduced performance loss. This indicates that Digital Twin integration increases temporal robustness and reduces the dependence on complete raw input.

Visual inspection of time-series anomaly plots, reconstruction loss curves, and multivariate heatmaps demonstrates that the hybrid system provides interpretable anomaly indicators suitable for real-time monitoring interfaces. Unlike black-box models, Digital Twin coupling provides semantic insights into system states and helps identify the source of deviations at the sensor level. This interpretability is critical for industrial adoption, where decision-making requires explanation rather than binary classification alone.

Overall, the obtained results show that combining Digital Twin physical-state simulation with neural reconstruction mechanisms forms an effective baseline for context-aware anomaly detection in IoT environments. The approach addresses limitations related to noise, missing data, and temporal complexity, offering both improved performance and higher interpretability compared to traditional machine learning models.

7. Conclusions

This work presents a hybrid anomaly detection framework that integrates Digital Twin simulation with AI-driven telemetry analysis for IoT systems. The proposed approach leverages state-space modelling to generate expected behaviour patterns and compute deviation vectors, which are further processed by an LSTM Autoencoder to identify abnormal states through reconstruction error. Experimental results demonstrate that the hybrid DT+AI architecture outperforms conventional anomaly detection methods, achieving higher accuracy and robustness under noisy and incomplete data conditions. The system reduced false-negative rates, improved recognition of gradual and contextual anomalies, and maintained stability under high noise levels compared to classical baselines.

The visual and metric-based evaluation confirms the applicability of the approach for real-time monitoring scenarios, where early detection of system degradation is essential for failure prevention. The proposed method is particularly relevant for industrial IoT deployments, predictive maintenance

systems, and cyber-physical infrastructure with constrained computational resources. The modular architecture allows flexible integration with alternative preprocessing or AI models and supports future extensions.

Future work will focus on expanding the Digital Twin model with multi-physics simulation elements, incorporating federated learning for distributed analytics, and evaluating deployment feasibility on edge hardware such as microcontrollers and embedded gateways. Additional research may explore reinforcement learning for adaptive thresholding and encrypted telemetry pipelines for secure anomaly inference in mission-critical environments.

Acknowledgements

This research was funded by the Ministry of Education and Science of Ukraine under grant 0123U100270.

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

[1] S. H. Abdulhussain, B. M. Mahmmod, A. Alwhelat, D. Shehada, Z. I. Shihab, H. J. Mohammed, A. Hussain, A comprehensive review of sensor technologies in iot: Technical aspects, challenges, and future directions, Computers 14 (2025) 342.

[2] G. Vidyalakshmi, S. Gopikrishnan, W. Boulila, A. Koubaa, G. Srivastava, Digital twins and cyber-physical systems: A new frontier in computer modeling, Computer Modeling in Engineering & Sciences 143 (2025) 51.

[3] S. Adibi, A. Rajabifard, D. Shojaei, N. Wickramasinghe, Enhancing healthcare through sensor-enabled digital twins in smart environments: A comprehensive analysis, Sensors 24 (2024) 2793.

[4] A. Yarmilko, I. Rozlomii, S. Naumenko, Dependability of embedded systems in the industrial internet of things: Information security and reliability of the communication cluster, in: Proceedings of the International Scientific-Practical Conference "Information Technology for Education, Science and Technics", Springer Nature Switzerland, Cham, Switzerland, 2024, pp. 235–249.

[5] V. Larin, et al., Prediction of the final discharge of the UAV battery based on fuzzy logic estimation of information and influencing parameters, in: 2022 IEEE 3rd KhPI Week on Advanced Technology (KhPIWeek), 2022, pp. 1–6.

[6] A. S. AlSalehy, M. Bailey, Improving time series data quality: Identifying outliers and handling missing values in a multilocation gas and weather dataset, Smart Cities 8 (2025) 82.

[7] M. Mayilsamy, Intelligent anomaly detection in real-time big data engineering, Journal of Engineering and Computer Sciences 4 (2025) 577–589.

[8] S. Zhyla, et al., Practical imaging algorithms in ultra-wideband radar systems using active aperture synthesis and stochastic probing signals, Radioelectronic and Computer Systems 1 (2023) 55–76.

[9] Z. Amiri, A. Heidari, N. J. Navimipour, M. Unal, Resilient and dependability management in distributed environments: A systematic and comprehensive literature review, Cluster Computing 26 (2023) 1565–1600.

[10] I. Rozlomii, A. Yarmilko, S. Naumenko, P. Mykhailovskyi, Hardware encryptors and cryptographic libraries for optimizing security in iot, in: Proceedings of the 12th International Conference on Information Control Systems & Technologies (ICST 2024), Odesa, Ukraine, 2024, pp. 99–109.

[11] M. Abd Elaziz, L. Abualigah, I. Attiya, Advanced optimization technique for scheduling iot tasks in cloud-fog computing environments, Future Generation Computer Systems 124 (2021) 142–154.

[12] C. Lin, B. Du, L. Sun, L. Li, Hierarchical context representation and self-adaptive thresholding for multivariate anomaly detection, IEEE Transactions on Knowledge and Data Engineering 36 (2024) 3139–3150.

[13] F. Tusa, S. Clayman, A. Buzachis, M. Fazio, Microservices and serverless functions—lifecycle, performance, and resource utilisation of edge-based real-time iot analytics, Future Generation Computer Systems 155 (2024) 204–218.
[14] I. Rozlomii, S. Naumenko, P. Mykhailovskyi, V. Monarkh, Resource-saving cryptography for microcontrollers in biomedical devices, in: Proceedings of the IEEE 5th KhPI Week on Advanced Technology, IEEE, 2024, pp. 1–5.
[15] N. Jeffrey, Q. Tan, J. R. Villar, A review of anomaly detection strategies to detect threats to cyber-physical systems, Electronics 12 (2023) 3283.
[16] Y. V. Voievodin, I. O. Rozlomii, Advanced software framework for comparing balancing strategies in container orchestration systems, in: Proceedings of an International Conference on Distributed Systems, 2024, pp. 60–69.
[17] Y. Voievodin, I. Rozlomii, Application security optimization in container orchestration systems through strategic scheduler decisions, in: Proceedings of the CPITS-2024: Cybersecurity Providing in Information and Telecommunication Systems, volume 3654 of *CEUR Workshop Proceedings*, 2024, pp. 471–478.
[18] N. L. Rane, M. Paramesha, S. P. Choudhary, J. Rane, Machine learning and deep learning for big data analytics: A review of methods and applications, Partners Universal International Innovation Journal 2 (2024) 172–197.
[19] S. Oswal, S. Shinde, M. Vijayalakshmi, A survey of statistical, machine learning, and deep learning-based anomaly detection techniques for time series, in: International Advanced Computing Conference, Springer Nature Switzerland, Cham, Switzerland, 2022, pp. 221–234.
[20] S. A. Bkheet, J. I. Agbinya, G. S. M. Khamis, Advanced deep learning approach for smart home appliance identification using recurrent neural networks with lstm, IoT 5 (2024) 835–851.
[21] M. Tayebi, S. El Kafhali, Performance analysis of recurrent neural networks for intrusion detection systems in industrial internet of things, Franklin Open 12 (2025) 100310.
[22] H. Farhat, A. Altarawneh, Physics-informed machine learning for intelligent gas turbine digital twins: A review, Energies 18 (2025) 5523.
[23] S. S. Reza, C. M. K. Uddin, M. F. Rabbi, Bridging virtual and physical worlds: Ai in digital twin development for mechanical systems, Scientia: Technology, Science and Society 2 (2025) 16–31.
[24] X. Bampoula, N. Nikolakis, K. Alexopoulos, Condition monitoring and predictive maintenance of assets in manufacturing using lstm-autoencoders and transformer encoders, Sensors 24 (2024) 3215.
[25] H. Li, Y. Li, Anomaly detection methods based on gan: A survey, Applied Intelligence 53 (2023) 8209–8231.
[26] S. Baimukhanov, H. Ali, A. Yazici, Enhancing ml-based anomaly detection in data management for security through integration of iot, cloud, and edge computing, Expert Systems with Applications (2025) 128700.
[27] Q. Feng, Y. Zhang, B. Sun, X. Guo, D. Fan, Y. Ren, Z. Wang, Multi-level predictive maintenance of smart manufacturing systems driven by digital twin: A matheuristics approach, Journal of Manufacturing Systems 68 (2023) 443–454.
[28] Y. Zhang, Q. Feng, D. Fan, Y. Ren, Y. Song, M. Liu, Z. Wang, Predictive control for operation and maintenance in smart manufacturing systems with multiple operating modes, Computers & Industrial Engineering 207 (2025) 111196.
[29] Y. Wang, E. Zhang, A. Yang, K. Du, J. Gao, Mixed reality-based multi-scenario visualization and control in automated terminals: A middleware and digital twin driven approach, Buildings 15 (2025) 3879.