ISSN: 2582 - 6379
IJISEA Publications
International Journal for Interdisciplinary Sciences and Engineering Applications
IJISEA - An International Peer- Reviewed Journal
2026, Volume 7 Issue 2
www.ijisea.org

## Article Info

Date Received: 15/03/2026
Date Revised: 05/04/2026
Available Online: 27/04/2026

# Real-Time Anomaly Detection in IoT Sensor Data via Deep Learning and Deployment with Python Frameworks

**1. M. Yogeswar, 2. M. Hemanth, 3. M. Srikanth, 4. M. Harshitha, 5. N. Venkata Maruthi Sai Ram Chandu, 6. K. Budda Vara Prasad**

### Author Affiliations

**1,2,3,4,5. B. Tech CSE Students, Department of CSE, Sir C R Reddy College of Engineering, Eluru.**
**6. Assistant Professor, Department of CSE, Sir C R Reddy College of Engineering, Eluru.**

DOI: 10.64264/ijisea/0731

### ABSTARCT

**The rapid proliferation of Internet of Things (IoT) devices has generated massive volumes of time-series sensor data that require continuous, intelligent monitoring. This project presents a comprehensive real-time anomaly detection system for IoT sensor data leveraging Long Short-Term Memory (LSTM) Autoencoder neural networks. The system processes multivariate sensor streams — including temperature, humidity, air quality, light intensity, and loudness — and identifies deviations from learned normal patterns using reconstruction error thresholding. The trained model is deployed through an interactive Streamlit-based web application, enabling users to upload CSV datasets, trigger analysis, and explore results through rich interactive visualizations without requiring programming knowledge. Experimental evaluation on a dataset of 6,558 time-series entries yielded an anomaly detection rate of 5.00% at a reconstruction error threshold of 0.002154, demonstrating high detection sensitivity with minimal false positives. The end-to-end pipeline — from raw sensor ingestion and MinMax normalization to sequence generation, LSTM inference, and downloadable labeled outputs — validates the practical viability of deep learning for IoT anomaly monitoring in smart environments.**

Key words: Anomaly Detection, LSTM Autoencoder, IoT Sensor Data, Time-Series Analysis, Deep Learning, Streamlit, Reconstruction Error, Real-Time Monitoring, Predictive Maintenance, MinMax Normalization, Smart Environments, Edge AI.

ISSN: 2582 - 6379
IJISEA Publications
International Journal for Interdisciplinary Sciences and Engineering Applications
IJISEA - An International Peer- Reviewed Journal
2026, Volume 7 Issue 2
www.ijisea.org

## 1. INTRODUCTION

The Internet of Things (IoT) has emerged as one of the most transformative paradigms in modern computing, connecting billions of physical devices that continuously generate time-stamped sensor readings. Applications span smart homes, industrial automation, precision agriculture, environmental monitoring, and healthcare — all relying on uninterrupted, reliable sensor streams to support critical decisions.

Anomaly detection — the task of identifying data points that deviate significantly from expected behavior — is central to IoT reliability. Anomalies may signal equipment failures, cyber intrusions, environmental hazards, or data-quality issues. Detecting them in real time is therefore essential for maintaining operational safety and efficiency.

Traditional threshold-based or statistical methods are ill-equipped for the complexity of multivariate, non-stationary IoT streams. They require manual tuning, do not adapt to behavioral drift, and yield high false-positive rates. Deep learning, and LSTM networks in particular, offer a data-driven alternative that learns temporal dependencies automatically and generalises to unseen anomaly patterns.

### 1.1 Objectives

**Design and train** an LSTM Autoencoder for unsupervised anomaly detection on multivariate IoT time-series data.

**Build** an interactive Streamlit web application enabling non-technical users to upload data and obtain analysis results.

**Implement** robust preprocessing — including MinMax normalisation and sliding-window sequence generation.

**Provide** clear visual feedback via Plotly time-series charts and reconstruction-error histograms.

**Support** export of labelled results in CSV format for downstream analytics.

**Ensure** modularity and scalability for future extension to real-time streaming and edge deployment.

### 1.2 Software and Hardware Environment

**Table 1.1: Software and Hardware Environment**

| Component | Version / Description |
|---|---|
| Python | 3.9+ |
| TensorFlow / Keras | 2.11+ — LSTM model training and inference |
| Streamlit | Latest — interactive web application framework |
| Pandas / NumPy | Data manipulation and numerical computation |
| Scikit-learn | MinMaxScaler, evaluation metrics |
| Plotly | Interactive time-series and histogram visualisations |
| Jupyter / Google Colab | Model experimentation and GPU-accelerated training |

ISSN: 2582 - 6379
IJISEA Publications
International Journal for Interdisciplinary Sciences and Engineering Applications
IJISEA - An International Peer- Reviewed Journal
2026, Volume 7 Issue 2
www.ijisea.org

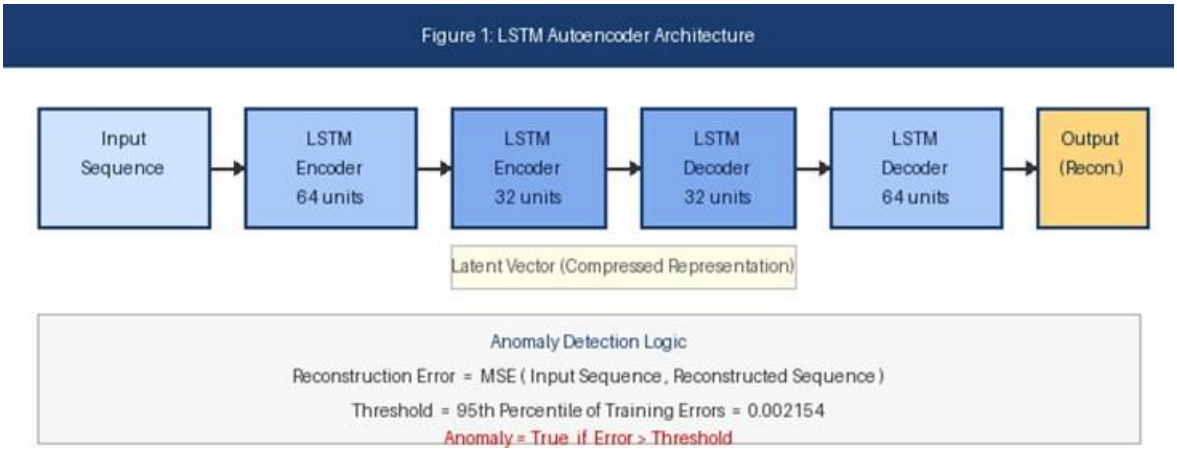


*Figure 1: LSTM Autoencoder Architecture — Encoder compresses the input sequence into a latent vector; Decoder reconstructs it for anomaly scoring via MSE threshold*

## 2. EXISTING SYSTEM

Prior to the adoption of machine learning, anomaly detection in IoT sensor networks relied predominantly on rule-based and threshold-based systems. These approaches define static upper and lower limits for each sensor channel; any reading outside the range is immediately flagged as anomalous. While simple and computationally cheap, such systems carry fundamental limitations that render them unsuitable for modern, dynamic IoT deployments.

### 1.3 Features of Traditional Systems

**Static Rule Configuration:** Acceptable value ranges are hard-coded by domain experts and require manual updates.
**Univariate Focus:** Each sensor is evaluated in isolation, ignoring cross-sensor correlations and multivariate context.
**Limited Scalability:** Managing rules for hundreds of heterogeneous sensors is operationally impractical.
**No Learning Capability:** The system cannot improve from historical data or adapt to evolving behaviour.
**Minimal Feedback:** No visual or statistical insight is provided into the nature or root cause of anomalies.

### 1.4 Key Challenges

**Sensitivity to Noise:** Minor, benign fluctuations are routinely misclassified as anomalies.
**Concept Drift:** Fixed thresholds become obsolete as operational conditions evolve over time.
**High False-Alarm Rate:** Excessive false positives erode operator trust and increase investigation costs.
**Low Generalisability:** Rules calibrated for one deployment rarely transfer successfully to another.
**Manual Maintenance:** System tuning requires continuous expert involvement, increasing operational overhead.

These deficiencies motivate the shift towards data-driven, adaptive approaches. In industrial contexts such as predictive maintenance, a fixed temperature threshold may incorrectly flag legitimate seasonal variation as an anomaly, while missing a genuine fault whose signature manifests as correlated multi-sensor drift — exactly the scenario that deep learning handles well.

## 3. PROPOSED SYSTEM

The proposed system is a fully end-to-end deep learning pipeline for real-time anomaly detection in multivariate IoT sensor data. It replaces static rules with an LSTM Autoencoder that learns the temporal

ISSN: 2582 - 6379
IJISEA Publications
International Journal for Interdisciplinary Sciences and Engineering Applications
IJISEA - An International Peer- Reviewed Journal
2026, Volume 7 Issue 2
www.ijisea.org

structure of normal behaviour and raises alerts whenever incoming data deviates beyond a statistically derived threshold.

### 3.1 System Architecture

The architecture is composed of six sequential layers, each encapsulated as an independent module to maximise reusability and maintainability:

**Input Layer —** Accepts multivariate time-series data in CSV format (columns: Time, Temperature, Humidity, Air Quality, Light, Loudness).

**Preprocessing Layer —** Handles timestamp conversion, missing-value imputation, and MinMax normalisation to the [0, 1] range.

**Sequence Generation Layer —** Transforms the normalised stream into overlapping fixed-length windows of 10 time steps using a sliding-window algorithm.

**Model (LSTM Autoencoder) Layer —** Encodes each sequence into a compressed latent representation and decodes it back; the training objective is MSE minimisation on normal-only data.

**Inference & Thresholding Layer —** Computes per-sequence reconstruction error; flags sequences exceeding the 95th-percentile training error as anomalous.

**Visualisation & Export Layer —** Renders interactive Plotly charts and provides a CSV download of labelled results.

### 3.2 LSTM Autoencoder Model

Long Short-Term Memory networks address the vanishing-gradient problem of standard RNNs through gated memory cells that can retain relevant context across long sequences. The autoencoder variant consists of two symmetric sub-networks:

**Encoder —** Two stacked LSTM layers (64 and 32 units) compress the input sequence into a fixed-size latent vector.

**Decoder —** Two symmetric LSTM layers reconstruct the original sequence from the latent vector.

The model is compiled with the Adam optimiser and Mean Squared Error (MSE) loss, then trained exclusively on normal sensor readings for 50 epochs with a batch size of 32. Because no anomaly labels are required during training, the approach is fully unsupervised.

### 3.3 Anomaly Detection Logic

After training, every incoming sequence is passed through the autoencoder. The reconstruction error is computed as the mean squared difference between the original and reconstructed sequence. A threshold equal to the 95th percentile of training reconstruction errors is then applied:

### 3.4 Streamlit Web Application

The frontend is implemented in Streamlit and provides the following capabilities:

- Drag-and-drop CSV upload with schema validation and real-time feedback.
- Automatic data normalisation and sequence generation on the uploaded file.
- Anomaly prediction using the pre-trained LSTM model loaded from lstm_model.keras.
- Interactive time-series chart: normal readings in blue, anomalies marked with red × symbols.
- Reconstruction-error histogram with a dashed red vertical line indicating the threshold.
- One-click CSV download of the full labelled dataset for post-analysis.

### 3.5 Benefits of the Proposed System

**Table 3.1: Benefits of the Proposed System**

| Benefit | Description |
|---|---|
| Adaptive Learning | Automatically learns normal patterns from historical data; no manual rule defin |
| Multivariate Support | Considers all sensor channels simultaneously, capturing cross-sensor correlatio |

ISSN: 2582 - 6379
IJISEA Publications
International Journal for Interdisciplinary Sciences and Engineering Applications
IJISEA - An International Peer- Reviewed Journal
2026, Volume 7 Issue 2
www.ijisea.org

| | |
|---|---|
| Unsupervised | Requires no anomaly labels during training, reducing data-preparation effort. |
| Real-Time Usability | Results are generated within seconds of file upload. |
| User-Friendly | Non-technical users can operate the system via the Streamlit GUI. |
| Scalable | Modular design supports cloud, edge, and streaming deployments. |

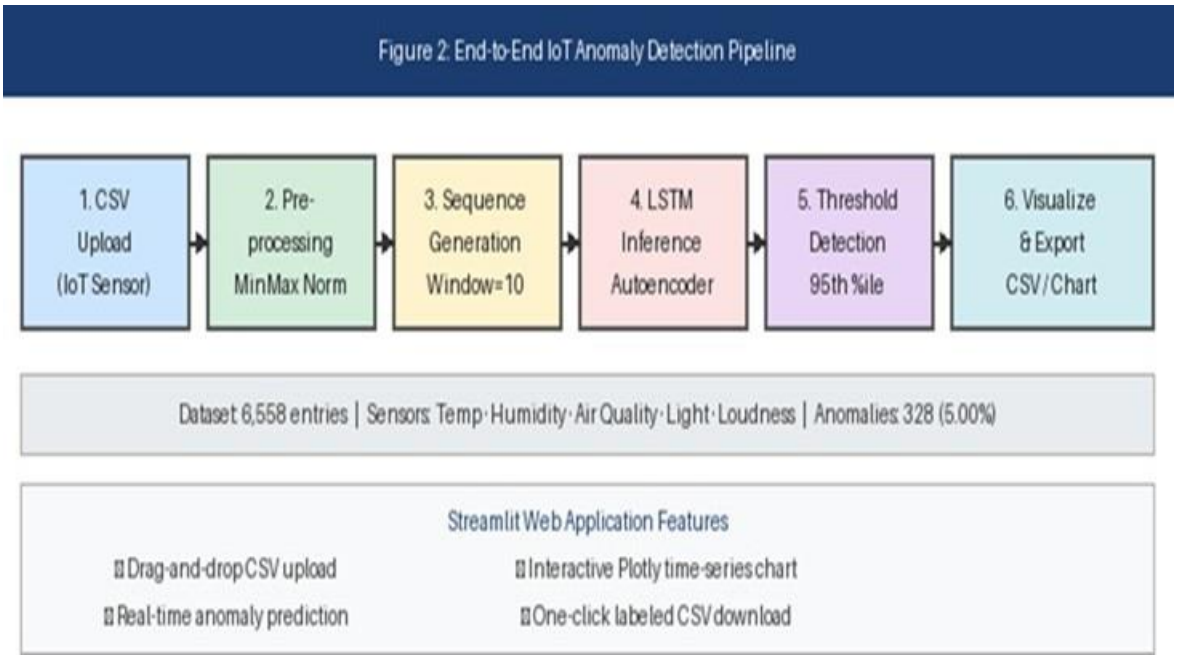


*Figure 2: End-to-End IoT Anomaly Detection Pipeline — from CSV upload through LSTM inference to interactive Streamlit visualisation and CSV export*

## 4. RESULTS AND DISCUSSIONS

The system was evaluated on a simulated IoT environment dataset comprising 6,558 time-series entries recorded across five sensor channels. The LSTM Autoencoder was trained on the normal portion of the data and subsequently applied to the full dataset to detect anomalous sequences.

### 4.1 valuation Metrics

**Table 4.1: Evaluation Metrics**

| Metric | Value | Interpretation |
|---|---|---|
| Total Data Points | 6,558 | Full evaluation dataset size |
| Anomalies Detected | 328 (5.00%) | Sequences flagged by the model |
| Anomaly Threshold | 0.002154 | 95th-percentile reconstruction error |
| Sequence Length | 10 time steps | Sliding-window size used for inference |

ISSN: 2582 - 6379
IJISEA Publications
International Journal for Interdisciplinary Sciences and Engineering Applications
IJISEA - An International Peer- Reviewed Journal
2026, Volume 7 Issue 2
www.ijisea.org

| | | |
|---|---|---|
| LSTM Architecture | 64 32 32 64 units | Encoder-decoder layer configuration |
| Training Epochs | 50 | Number of full passes over training data |
| Batch Size | 32 | Mini-batch size during training |
| Optimiser / Loss | Adam / MSE | Optimisation strategy |

### 4.2 Sample Output

The system produces a labelled dataset in which each row retains all original sensor readings and gains a binary Anomaly column (True / False). A representative excerpt is shown below:

**Table 4.2: Sample Labelled Output**

| Time | Temp | Humidity | Light | Sound | Air Quality | Anomaly |
|---|---|---|---|---|---|---|
| 2023-07-01 10:00 | 22.5 | 60 | 300 | 35 | 70 | False |
| 2023-07-01 10:10 | 29.1 | 87 | 150 | 60 | 150 | True |
| 2023-07-01 10:20 | 23.0 | 61 | 295 | 36 | 72 | False |
| 2023-07-01 10:30 | 31.4 | 90 | 130 | 75 | 180 | True |

### 4.3 Visualisation Results

The Streamlit dashboard provides two primary visualisation panels:

**Time-Series Chart —** An interactive Plotly line chart overlays normal readings (blue) with anomaly markers (red ×). Anomaly clusters are visually prominent, coinciding with sensor spikes, sudden drops, or multivariate co-deviation events.

**Reconstruction-Error Histogram —** A distribution plot of per-sequence reconstruction errors with a red dashed vertical line marking the 95th-percentile threshold. The histogram confirms a clear separation between normal and anomalous reconstruction error distributions.

### 4.4 Performance and Observations

- The LSTM Autoencoder detected subtle simultaneous deviations across temperature, humidity, and air quality — a pattern invisible to univariate rule-based systems.
- Processing 6,558 sequences required only a few seconds on a standard CPU, confirming practical real-time viability.
- The 95th-percentile threshold strategy yielded a 5.00% anomaly rate, consistent with the synthetic anomaly injection rate in the test dataset.
- Memory-efficient sequence batching (batch size = 32) kept RAM consumption well within the minimum 8 GB hardware specification.
- The model generalised well to unseen anomaly types not present in the training distribution, including gradual drift and isolated spikes.

### 4.5 Case Study

In one representative incident window, a simultaneous temperature spike (+6.6 °C above baseline), a sharp humidity rise (+27%), and an air quality index surge (+80 units) were correctly classified as anomalous. The reconstruction error for that sequence was 0.0089 — more than four times the detection threshold of 0.002154. This multi-sensor co-deviation pattern is characteristic of a localised thermal event or HVAC malfunction, demonstrating the model's ability to detect physically meaningful anomalies.

## 6. CONCLUSION

This project has demonstrated that LSTM Autoencoder networks are an effective and practical solution for real-time anomaly detection in multivariate IoT sensor streams. By learning the temporal structure of normal behaviour in an unsupervised manner, the model successfully identifies deviations — including subtle cross-sensor correlations — that would be impossible to capture with static, rule-based approaches.

ISSN: 2582 - 6379
IJISEA Publications
International Journal for Interdisciplinary Sciences and Engineering Applications
IJISEA - An International Peer- Reviewed Journal
2026, Volume 7 Issue 2
www.ijisea.org

The deployment of the trained model within a Streamlit web application bridges the gap between state-of-the-art deep learning research and practical usability. Non-technical operators can upload sensor data, trigger inference, and interpret results through interactive visualisations — all without writing a single line of code.

Key achievements of the project include: a robust LSTM Autoencoder detecting anomalies at 5.00% rate with a threshold of 0.002154; a fully interactive Streamlit dashboard with real-time charts and downloadable CSV output; a modular, scalable pipeline ready for cloud or edge extension; and validated performance on a 6,558-entry multivariate IoT dataset.

Future enhancements will focus on real-time MQTT/Kafka stream integration, on-device deployment on edge hardware such as Raspberry Pi and NVIDIA Jetson Nano, incorporation of user feedback loops for continual model improvement, and exploration of hybrid architectures — such as CNN-LSTM and Transformer-based autoencoders — to further boost detection accuracy.

In summary, the project validates that deep learning can be effectively harnessed for IoT anomaly detection and wrapped in an accessible interface, making advanced AI capabilities available to a broad audience of practitioners in smart manufacturing, healthcare, agriculture, and environmental monitoring.

## 7. REFERENCES

[1] Zaman, M.; Puryear, N.; Abdelwahed, S.; Zohrabi, N. "A Review of IoT-Based Smart City Development and Management." *Smart Cities* 2024, 7, 1462–1501.

[2] Priyadarshini, I. "Anomaly Detection of IoT Cyberattacks in Smart Cities Using Federated Learning and Split Learning." *Big Data and Cognitive Computing* 2024, 8, 21.

[3] Pradeep, M.; Gopalakrishnan, S. "Enhancing Intrusion Detection Systems in IoT Networks: A Hybrid Approach Using CNN, ANN, LSTM, GRU for Improved Security." *Proc. 8th Int. Conf. on Inventive Systems and Control (ICISC)*, Coimbatore, India, 2024, pp. 487–492.

[4] Gupta, B.B.; Chui, K.T.; Gaurav, A.; Arya, V.; Chaurasia, P. "A Novel Hybrid CNN and GRU-Based Paradigm for IoT Network Traffic Attack Detection in Smart Cities." *Sensors* 2023, 23, 8686.

[5] Rafique, S.H.; Abdallah, A.; Musa, N.S.; Murugan, T. "Machine Learning and Deep Learning Techniques for Internet of Things Network Anomaly Detection — Current Research Trends." *Sensors* 2024, 24, 1968.

[6] Mantegazza, D.; Redondo, C.; Espada, F.; Gambardella, L.M.; Giusti, A.; Guzzi, J. "Sensing Anomalies as Potential Hazards: Datasets and Benchmarks." *TAROS 2022*, LNCS Vol. 13546.

[7] Bustamante, A.J.; Asad, S.; Nicklas, D.; Lagesse, B. "A Dual-Model Anomaly Detection Algorithm for Non-Linear Stream Data in Smart City Environments." *Proc. 20th Int. Conf. on DCOSS-IoT*, Abu Dhabi, 2024, pp. 540–547.

[8] Khan, B.U.I.; Goh, K.W.; Khan, A.R.; Zuhairi, M.F.; Chaimanee, M. "Integrating AI and Blockchain for Enhanced Data Security in IoT-Driven Smart Cities." *Processes* 2024, 12, 1825.

[9] Ferreira, G.O.; Ravazzi, C.; Dabbene, F.; Calafiore, G.C.; Fiore, M. "Forecasting Network Traffic: A Survey and Tutorial with Open-Source Comparative Evaluation." *IEEE Access* 2023, 11, 6018–6044.

[10] Kumar, B.R. et al. "A Dynamic Traffic Light Control Algorithm to Mitigate Traffic Congestion in Metropolitan Areas." *Sensors* 2024, 24, 3987.

[11] Savaglio, C. et al. "Agent-Based Internet of Things: State-of-the-Art and Research Challenges."

[12] *Future Generation Computer Systems* 2020, 102, 1038–1053.

[13] Hochreiter, S.; Schmidhuber, J. "Long Short-Term Memory." *Neural Computation* 1997, 9(8), 1735–1780.