

TinyML-Optimised Multimodal Lstm for Unsupervised Anomaly Detection in Smart Home IoT Systems

*Osaretin Edith Okoro, Nurudeen Mahmud Ibrahim, Prema Kirubakaran, and Suleiman Aliyu Muhammad

Department of Computer Science, Nile University of Nigeria, FCT Abuja, Nigeria.

*Corresponding authors' email: 20233004@nileuniversity.edu.ng

ABSTRACT

The rapid rise in IoT smart home devices generates extensive time-series datasets, while anomaly detection solutions remain largely dependent on cloud-based processing, resulting in latency, privacy concerns, and connectivity requirements. Existing approaches often rely on a single modality and insufficiently address concept drift and edge-device constraints. This study proposes a TinyML-optimised multimodal LSTM model for unsupervised anomaly detection in smart home applications deployed on a Raspberry Pi edge device. Temperature, humidity, and power consumption modalities were combined to capture cross-modal relationships. A simulated dataset represented normal diurnal and seasonal behaviour and incorporated three anomaly types: heat spikes, power outages, and midnight appliance usage. Model training used only normal-value sequences within a 250-timestep sliding window. TinyML optimisation reduced the stored LSTM model size to 196 KB, which represents the model storage size and should not be interpreted as total runtime memory or direct evidence of real-time or low-power operation. An online learning strategy was employed to address concept drift through periodic retraining. On the simulated test set, the model achieved 98.97% recall, 8.79% precision, 8.86% F1-score, 0.294 Matthews correlation coefficient (MCC), and an AUC-ROC of 0.9984. The confusion matrix comprised 83,416 true negatives, 3,994 false positives, 4 false negatives, and 385 true positives. Although the model detected nearly all injected anomalies, its low precision indicates a substantial false-alarm burden and limits claims regarding deployment readiness. The findings indicate the feasibility of local anomaly detection using a TinyML-optimised multimodal LSTM on an edge platform without cloud inference, while real-time performance, energy efficiency, and privacy benefits require further validation.

Received: 18 Sep. 2026

Accepted: 05 Oct. 2026

Published: 08 Oct. 2026

Keywords:

Anomaly detection, LSTM model, TinyML, Smart home IoT, Edge Computing, Multimodal fusion

INTRODUCTION

The convergence of IoT technologies with residential automation has produced smart home environments capable of monitoring and controlling a wide array of appliances and environmental conditions. Smart homes utilise networked sensors and actuators to enhance user comfort, energy efficiency, and safety, particularly for elderly and disabled residents (Ezugwu *et al.*, 2025; Hong *et al.*, 2022). However, this proliferation of interconnected devices also enlarges the attack surface for faults, intrusions, and anomalous behaviours that can escalate into safety critical events (Kanimozhi, 2026).

Anomaly detection; the identification of data patterns deviating significantly from established norms is a cornerstone of smart home security and operational reliability (Fahad & Tahir, 2021).

Standard cloud-based approaches provide computational ability but introduce intolerable latency for real-time response, generate privacy issues by sending private household data to off-premises and are dependent on persistent internet connectivity (Katib *et al.*, 2025; Rajaan *et al.*, 2025a). In contrast, early edge-based solutions tend to use shallow statistical or rule-based techniques that fail to capture the inherent nonlinear, temporal dependencies in Multimodal sensor streams (Pradhan & Gangashetty, 2026). Deep learning architectures, particularly Long Short-Term Memory (LSTM) networks and Models, have demonstrated

superior capability in modelling sequential anomalies in IoT time-series data (Kumar *et al.*, 2024; Shanmuganathan & Suresh, 2023). However, their deployment on resource-constrained edge hardware has been impeded by high memory and computational requirements. The execution of optimised machine learning models on microcontrollers and low power embedded devices, addresses this gap through techniques such as quantisation and pruning (Abadade *et al.*, 2023; Dutta & Bharali, 2021).

It is important to distinguish conventional edge computing from microcontroller-oriented TinyML. Conventional edge computing may employ relatively capable devices such as Raspberry Pi computers, gateways, or industrial edge systems, whereas TinyML generally focuses on executing machine-learning workloads under highly constrained memory, computational, and energy conditions, particularly on microcontroller-class devices. In this study, the Raspberry Pi 4 Model B is used as the experimental edge-computing platform, while model compactness and INT8 quantisation are adopted as TinyML-oriented optimisation strategies. Therefore, the present work should be interpreted as an evaluation of a TinyML-optimised model on a Raspberry Pi edge platform rather than as a demonstration of microcontroller deployment.

Several notable limitations persist in the current literature. First, the majority of TinyML-based anomaly detection works employ unimodal sensor inputs (Arciniegas *et al.*, 2025;

Hammad et al., 2023), neglecting the cross-modal correlations that substantially improve detection robustness. Second, static models trained on historical data are vulnerable to concept drift, gradual shifts in normal behaviour caused by seasonal changes, new appliances, or occupancy variations which degrades detection performance over time (Arora et al., 2024a; Litayem, 2024a). Third, comprehensive resource benchmarking (inference time, memory footprint, energy consumption) on embedded platforms remains rare in the anomaly detection literature (da Costa Filho et al., 2025; Rajaan et al., 2025).

This study addresses these gaps by proposing the TinyML-Optimised Multimodal LSTM Model (AM-LSTM), a lightweight, unsupervised anomaly detection framework that fuses temperature, humidity, and power-load modalities within a single LSTM Model, quantises for Raspberry Pi deployment, and employs an online learning strategy for adaptation to concept drift.

The proposed framework was evaluated using measurable criteria including precision, recall, F1-score, AUC-ROC, MCC, false-positive rate, balanced accuracy, model size, and, where directly measured, inference latency, memory consumption, CPU utilisation, and energy consumption. These criteria are used to assess both anomaly-detection performance and the practical resource characteristics of the proposed model.

Related Work

Anomaly Detection in IoT and Smart Homes

The IoT anomaly detection has evolved from rule-based threshold methods to sophisticated deep learning pipelines (Kumari et al., 2024; Morshedi & Matinkhah, 2025).

Statistical approaches such as Gaussian models and fixed thresholds are computationally efficient and interpretable, but they depend on static data distributions and therefore are insufficient for dynamically smart home environments (Jiang et al., 2020). By learning typical behaviour from data, classical machine learning techniques like Random Forests (RF), Isolation Forests, and One-Class Support Vector Machines (OC-SVM) perform better than statistical baselines, but they have trouble generalising to new usage patterns and heterogeneous sensor streams (Bouazza et al., 2026).

In raw IoT data, deep learning models including RNNs, LSTMs, GRUs, CNNs, Models have shown remarkable ability in capturing nonlinear temporal dependencies (Naik et al., 2026). Using an RNN-LSTM on a Kaggle smart home dataset, (Muthunambu et al., 2024) obtained 83.25% accuracy and an AUC of 0.9896; however, their method was cloud-only and ignored edge deployment limitations. A CNN-LSTM model that increased F1-score by 12–15% on IoT time series was proposed by Yin et al., (2022), but it came with set input lengths and a high computational cost.

TinyML and Edge Deployment

Arciniegas et al., (2025) reported 96.5% accuracy for vibration-based TinyML anomaly detection, but with 300 ms latency deemed excessive for fast faults; Antonini et al., (2023) demonstrated unsupervised TinyML anomaly detection on ESP32 with detection latency below 16 ms, but limited to industrial unimodal sensor streams; Shubham Gupta & Shiv Naresh Shivhare, (2025) presented a 1D CNN

with a memory footprint of less than 100 KB; (Litayem, 2024a) deployed a CNN for activity classification on an ESP32-S3, covering only six activity classes.

Multimodal and Adaptive Approaches

By learning cross-modal relationships, multimodal fusion frameworks with spatiotemporal models and attention processes have consistently improved detection accuracy (Srilakshmi et al., 2025; Zeng et al., 2023). According to Reis & Serôdio, (2025), these methods usually require server-grade or cloud computing capabilities. It has been demonstrated that concept drift adaptation via online and incremental learning may maintain model performance under distributional shifts (Shang et al., 2025). However, solutions aimed at microcontrollers are still restricted because of memory and energy limitations (Disabato & Roveri, 2024).

Research Gap

Despite previous studies (Hu et al., 2025; Sun et al., 2024; Zhao et al., 2024) have investigated individual aspects of multimodal temporal modelling, TinyML optimisation, unsupervised anomaly detection, and adaptive edge intelligence, these capabilities have not necessarily been evaluated together under identical datasets, hardware platforms, and experimental protocols. Therefore, direct claims of superiority over existing approaches require controlled comparative experiments. The present study addresses this gap by integrating multimodal sensing, attention-based temporal modelling, unsupervised reconstruction, INT8 optimisation, and an adaptive mechanism within a single experimental framework.

MATERIALS AND METHODS

Research Design

This study proposes an experimental and design-oriented research to develop and evaluate a TinyML-optimized Adaptive Multimodal Long Short-Term Memory (AM-LSTM) framework for unsupervised anomaly detection in smart-home IoT systems. The experimental process includes multimodal smart-home data preparation, training the AM-LSTM model with normal behaviour sequences, introducing controlled anomalies, and evaluating detection performance. To evaluate the practical edge-computing feasibility, model optimisation and deployment experiments on a Raspberry Pi 4 Model B was conducted. The independent variables are the AM-LSTM architecture and relevant hyperparameters, while the dependent variables are Precision, Recall, F1-Score, AUC-ROC, MCC, inference time, and memory footprint (Ferrag et al., 2022; Xiao et al., 2024). Standardised preprocessing, data partitioning, sequence construction, training configuration, hardware, software, and evaluation procedures guarantee experimental consistency.

The workflow consists of dataset preparation, preprocessing, temporal sequence construction, model development, normal-only training, anomaly scoring, threshold determination, evaluation, TinyML optimisation, and edge deployment. An online-learning component then evaluates adaptation to behavioural changes. The design is generally developmental and evaluative in nature, allowing systematic assessment, practical deployment and reproducibility.

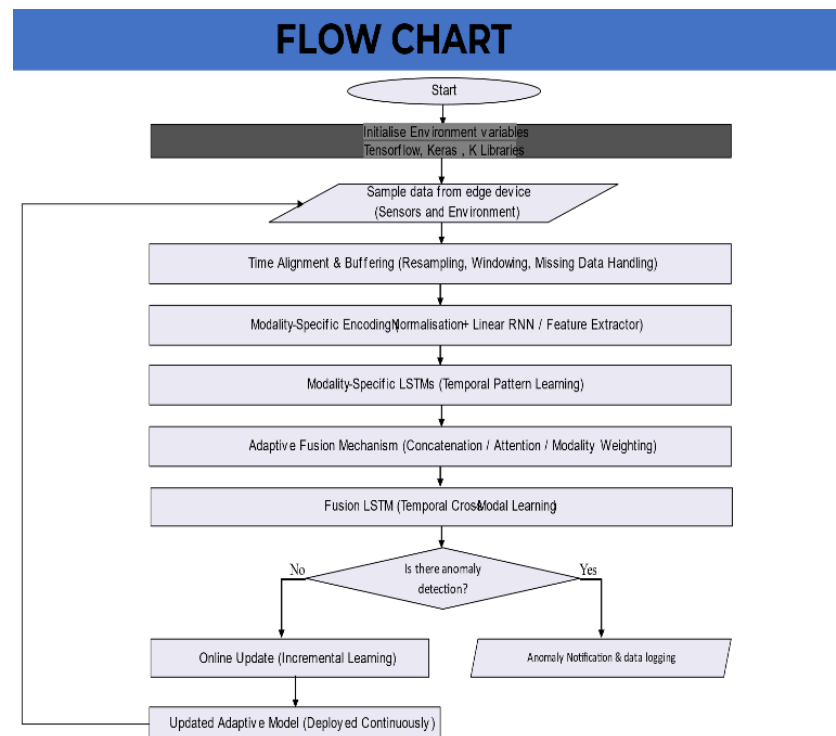


Figure 1: Research Design Chart

Conceptual Framework

The conceptual framework suggests an integrated, adaptive, edge-based architecture for anomaly detection in smart-home IoT systems. It establishes logical relations between multimodal sensing, temporal representation, anomaly detection, TinyML optimisation, edge inference and online learning. The framework posits that reliable anomaly detection relies on analysing not just single sensor readings, but also temporal interactions and trends across various sensor modalities.

The framework starts with multimodal sensing, where complementary environmental and household-activity information are obtained in the forms of temperature, humidity, and power-consumption data. Preprocessing and normalisation of the data collected is performed and temporal sequences are created by applying overlapping sliding windows. This enables the model to learn behavioural context rather than isolated observations

The processed sequences are then input into the Adaptive Multimodal LSTM (AM-LSTM) model that learns time-based patterns using LSTM and fuses multimodal data using attention to learn relationships between sensor streams. The model reconstructs the input sequences and the Mean Squared Error (MSE) between original and reconstructed sequences is used as a continuous anomaly score. A threshold derived from normal validation data is then used to separate normal from anomalous behaviour without requiring anomaly labels for primary training. After training, TinyML optimisation through quantisation is used to reduce computational and memory requirements while maintaining useful detection performance. The optimised model is deployed on a Raspberry Pi 4 Model B for local low-latency inference reducing dependence on cloud connectivity and supporting privacy.

Lastly, the anomaly threshold is periodically recalibrated and freshly observed normal behaviour is included into online learning and concept-drift adaptation. The resulting framework creates a closed-loop adaptive system in which

new normal behavioural data feeds back into the learning process while sensor observations provide anomalous choices. The suggested architecture may therefore continuously adjust to shifting smart-home settings, in contrast to static detection techniques.

Theoretical Foundations of the Proposed Framework

Instead of relying on a single machine-learning theory, the suggested methodology is based on the integration of multiple complementary theoretical principles. The framework's integration of multimodal sensor fusion, temporal deep learning, reconstruction-based unsupervised detection, attention mechanisms, edge intelligence, TinyML optimisation, and adaptive learning is explained by these foundations.

Multimodal Representation and Information-Fusion Foundation

Heterogeneous streams are produced by smart home environments, where various modalities record various facets of domestic life. Humidity offers supplementary environmental information, temperature reflects the state of the environment, and power consumption indicates household and appliance activity.

Multimodal fusion is theoretically justified by the fact that complementing sensor streams can yield information not possible with a single modality. Depending on the contemporaneous ambient temperature and humidity circumstances, for instance, an extraordinary increase in power consumption may be interpreted differently. Therefore, instead of depending just on individual sensor thresholds, the model may learn cross-modal correlations thanks to the integration of modalities. The application of cross-modal interactions to enhance representation and detection performance is supported by research on multimodal fusion and attention-based techniques (Srilakshmi et al., 2025; Zeng et al., 2023).

Temporal Sequence-Modelling Foundation

IoT observations from smart homes create sequential data, where each observation's importance is determined by its temporal context. Because LSTM networks' recurrent architecture allows for the depiction of dependencies over several time steps, they are suitable. Instead of treating every sensor observation as a separate sample, this foundation supports the usage of fixed-length temporal windows. The 250-timestep sequence enables the AM-LSTM to recognise aberrations that might only become significant over time and to learn temporal properties linked to typical household behaviour. Previous research supports the use of LSTM and related recurrent architectures for IoT time-series anomaly detection (Gøthesen et al., 2023; Shanmuganathan & Suresh, 2023).

Reconstruction-Based Unsupervised Anomaly-Detection Foundation

The suggested methodology learns the statistical and temporal structure of normal behaviour instead of requiring exhaustive anomaly labels during training. The encoder learns a representation of normal multimodal sequences, while the decoder attempts to reconstruct those sequences; reconstruction error is then used as the anomaly score. Theoretically, a model exposed primarily to normal behavioural patterns will reconstruct similar normal patterns effectively, but will experience greater reconstruction difficulty when presented with previously unseen abnormal behaviour. This principle has been successfully applied in deep-learning-based IoT anomaly detection (Fu et al., 2024; Hammad et al., 2023).

Attention-Based Adaptive Fusion

All sensor modalities are assumed to contribute equally at all times in simple concatenation. In dynamic smart-home environments, this assumption might not be true. The model can reduce the impact of less informative or momentarily unreliable signals while giving more weight to informative modalities thanks to attention-based fusion. As a result, the attention mechanism offers an adaptive mechanism that allows the contribution of power, humidity, and temperature signals to change based on the observed temporal context. This helps achieve the goal of enhancing robustness against sensor data that is noisy, fleeting, or partially unreliable (Rahim et al., 2023; Shaheen et al., 2022).

TinyML and Resource-Constrained Learning

The computational basis for implementing machine-learning models on edge devices with limited resources is provided by TinyML. For small IoT systems, conventional deep-learning architectures may impose prohibitive memory, compute, and energy needs. The suggested framework uses post-training INT8 quantisation to overcome this constraint. Quantisation can significantly lower memory needs and computing costs while decreasing the numerical precision of model parameters. The goal of the resulting compact model is to maintain adequate anomaly-detection capacity while supporting local inference.

Edge Intelligence

The architectural basis for conducting inference near the data source is provided by edge intelligence. The suggested method uses the Raspberry Pi to carry out preprocessing, inference, reconstruction-error computation, and anomaly determinations locally rather than continuously sending raw smart-home sensor data to a distant cloud server. This design can lessen reliance on continuous cloud connectivity,

inference latency, and communication overhead. Because sensor data can stay in the local area, it also facilitates smart-home applications that are sensitive to privacy.

Concept Drift and Adaptive Learning

Smart home settings are ever-changing. Over time, daily activities, ambient conditions, household occupancy, and appliance usage may all change. As a result, a pattern that is deemed typical at one time may become abnormal at another, or vice versa. To solve this issue, the framework integrates online education. Periodically, new normal observations can be added to the model, and the updated reconstruction-error distribution is used to compute the anomalous threshold. This enables the system to adjust to shifting home habits and concept drift (Arora et al., 2024a; Yang & Shami, 2021).

Integrated Theoretical Foundation

In addition, the theoretical underpinnings provide the justification for the suggested architecture. The foundation for integrating heterogeneous sensor data is provided by multimodal learning; temporal dependencies are captured by LSTM sequence modelling; anomaly identification is made possible by reconstruction-based detection without the need for detailed anomaly labels; adaptive modality weighting is provided by attention mechanisms; computational and memory constraints are addressed by TinyML; local real-time inference is made possible by edge intelligence; and concept drift and behavioural changes are addressed by online learning. A closed-loop architecture that can transition from multimodal sensing to temporal representation, anomaly scoring, resource-efficient deployment, and continuous adaptation is created by combining these foundations. The development of an effective, unsupervised, multimodal, temporal, and resource-aware anomaly-detection system for smart-home IoT contexts is thus theoretically in line with the main research challenge.

Hardware and Software Platform

Edge deployment was performed on a Raspberry Pi 4 Model B. This platform was selected for its higher performance relative to microcontroller alternatives (ESP32, ESP8266, ATmega, STM32) and native Python/TensorFlow Lite support, which is cost-effective edge hardware in the target 100 KB to 3 MB model size range. The software stack comprised Python 3.9, TensorFlow 2.x and Keras for model development, TensorFlow Lite (TFLite) for post-training optimisation, NumPy and Pandas for data handling, Scikit-learn for preprocessing and evaluation metrics, and Matplotlib for visualisation. The Raspberry Pi 4 Model B represents an edge-computing platform rather than a microcontroller-class TinyML device. Accordingly, the use of the term TinyML in this study refers primarily to the compact model design and quantisation strategy rather than to demonstrated deployment on a microcontroller.

Dataset Generation

A Kaggle dataset for smart homes was utilized to offer ground-truth labels for anomalies that are missing in entirely unlabelled real-world data, while still maintaining realistic sensor behavior. The dataset was generated as a simulated smart-home time series representing one-minute observations of temperature, humidity, and electrical power consumption over one year. The resulting dataset contained 527,040 timestamped observations. Normal observations were generated within the ranges and statistical characteristics reported in this section, while controlled anomaly patterns

were subsequently introduced to enable quantitative evaluation.

Temperature (°C): Exhibits a diurnal sinusoidal trend with an average value of 27°C, a standard deviation of 2.43°C, and a range from 23.5°C to 31°C.

Humidity (%): Shows an inverse relationship with temperature, with a range of 44% to 77.5% and a standard deviation of 10.43%.

Power load (W): Characterized by a time-of-day usage pattern, with a mean of 403W, a standard deviation of 233W, and a range of 150W to 650W under typical conditions

To create realistic safety event scenarios and provide ground-truth labels for evaluation while maintaining anomaly-free training data, three types of anomalies were introduced:

- i. Heat Spike: The temperature was artificially raised from 33.5°C to 38.5°C alongside a concurrent power surge (from 900 W to 1,400 W), which was implemented on the 5th, 15th, and 25th of each month at 14:00.

- ii. Power Outage: The power load was reduced to nearly zero values (0–15 W), which was implemented over a span of three months.

- iii. Appliance Usage at Midnight: Unusual power consumption of 800–1,100 W was recorded at 02:00 AM over three months. The comprehensive dataset presented 526,956 regular samples and 84 anomalous samples (anomaly ratio: approximately 0.016% at the raw sample level; 0.44% at the sequence level following windowing).

The anomaly patterns were introduced according to predetermined temporal and sensor-value conditions. Because the anomalies were synthetically injected, the resulting labels were used only for validation and testing; the model was trained using normal sequences without anomaly labels. Thus, the learning procedure is unsupervised with respect to anomaly detection, while the evaluation uses known anomaly labels to quantify detection performance.

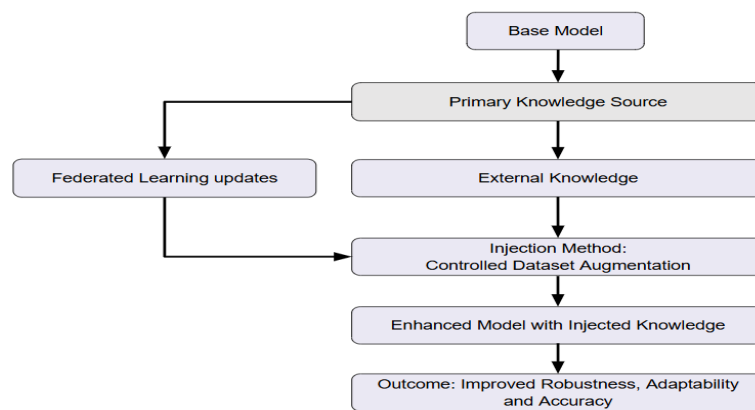


Figure 2: The Knowledge Injection

Data Preprocessing

All three sensor readings namely, temperature, humidity, and power load; were scaled to values within the range of [0, 1] by using MinMaxScaler before the training process. Feature scaling is considered an essential pre-processing task for training models based on LSTM neural networks because features with a wider absolute range of values (like the power load with the maximum value of 1,400 W) will have a stronger impact than features with a narrower range of values (such as the temperature, which lies within the range of 23.5 to 38.5 °C). The normalisation formula applied is:

$$x_{norm} = (x - x_{min}) / (x_{max} - x_{min}) \quad (1)$$

The data collected from sensors, after being scaled, was segmented into fixed-sized overlapping sequences using sliding windows, thus turning unstructured sensor information into structured input to feed into the LSTM model. Sequences of 250 timesteps with 3-timestep strides were created in order to capture sequences containing approximately 250 minutes' worth of multimodal sensor data, where the sliding window moves forward by 3 timesteps before extracting another sequence. The use of 250 timesteps gives sufficient temporal context to the LSTM to distinguish between abnormal trends such as a heat spike that lasts long enough or appliances operating through the night from regular fluctuations in sensor data.

Because the use of overlapping windows can introduce information leakage when windows from the same temporal

region are randomly distributed across training and testing subsets, the temporal partitioning was performed before final window construction. This ensured that future observations used for evaluation were not included in the training data.

Model Architecture

The architecture of the AM-LSTM, which includes the design of the encoder-decoder architecture, the multimodal input structure, and the layer-wise structure used for efficient temporal anomaly detection under the computational constraints of the Raspberry Pi system. The model was trained to reconstruct normal temporal patterns rather than to directly classify labelled anomalies. Anomaly scores were obtained from the reconstruction error between the observed and reconstructed sequences. The model was trained to reconstruct normal temporal patterns rather than to directly classify labelled anomalies. Anomaly scores were obtained from the reconstruction error between the observed and reconstructed sequences. Anomaly detection operates on the mean squared error (MSE) between the original and reconstructed sequences:

$$MSE = (1/N) \sum (x_i - \hat{x}_i)^2 \quad (2)$$

An attention-based fusion layer modulates the contribution of each modality during inference, improving robustness to noisy or transiently missing sensor signals (Rahim et al., 2023; Shaheen et al., 2022).

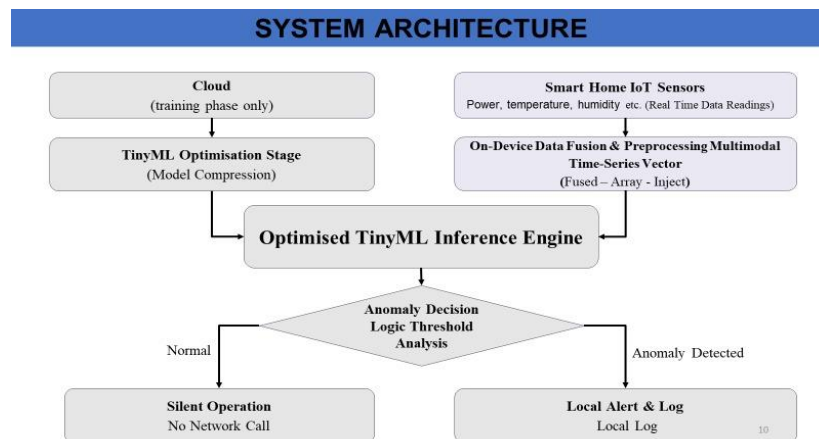


Figure Error! No text of specified style in document.: The General Architecture of the Proposed Adaptive Multimodal LSTM (AM-LSTM) Framework. (Not adopted but rather designed by myself)

TinyML Optimisation

Post-training INT8 quantisation was applied after converting the trained Keras model to TFLite format. Quantisation reduces weight precision from 32-bit float (FP32) to 8-bit integer (INT8), achieving a theoretical 75% size reduction and approximately 4× inference speedup with a typical

accuracy penalty of 2–5%. The resulting AM-LSTM model size is 196 KB, within the target 100 KB to 3 MB range and comparable to recent TinyML works (Litayem, 2024b; Shubbham Gupta & Shiv Naresh Shivhare, 2025). The quantisation techniques assessed and their trade-offs are presented in Table 1

Table 1: Power Reduction analysis

Quantization Type	Bit Width	Size Reduction	Accuracy Impact	Power Reduction
FP32 (baseline)	32	0%	0%	0%
FP16	16	50%	<1%	20%
INT8 (adopted)	8	75%	2–5%	50%
INT4	4	87.5%	5–10%	65%
Binary (1-bit)	1	96.9%	15–25%	80%

Edge Inference Pipeline

The on-device inference pipeline on the Raspberry Pi 4B operates as follows: (i) raw sensor readings are sampled at one-minute intervals; (ii) incoming readings are normalised using the pre-fitted MinMaxScaler and appended to a rolling buffer of 250 timesteps; (iii) once the buffer is complete, a sequence is passed to the TFLite interpreter; (iv) the AM-LSTM- produces a reconstructed sequence; (v) MSE between original and reconstructed sequences is computed; and (vi) the MSE is compared to the pre-established threshold, any exceedance triggers an anomaly detection event. The entire pipeline runs natively on the Raspberry Pi without cloud communication, supporting local processing without requiring cloud-based inference. However, low-latency and privacy claims require direct latency measurements and dedicated privacy/security evaluation, respectively. The 196 KB model size alone does not establish real-time performance. End-to-end latency should include preprocessing, sequence buffering, model inference, reconstruction-error calculation, and classification.

Online Learning and Concept Drift Adaptation

To address concept drift, an online learning strategy periodically retrains the AM-LSTM on newly accumulated normal sensor data and recalculates the anomaly detection threshold as the 95th percentile of reconstruction errors on the updated validation buffer. This approach adapts the model to gradual changes in household behaviour, seasonal patterns, or new appliance additions without full retraining from scratch (Arora et al., 2024b; Yang & Shami, 2021). The proposed updating mechanism represents a strategy for responding to

potential distributional changes rather than evidence that concept drift has been completely resolved. Its effectiveness should be evaluated using controlled drift scenarios in which the underlying sensor distribution changes over time.

Anomaly Detection Threshold

The anomaly detection threshold was determined using the 95th percentile of the reconstruction-error distribution, measured as mean squared error (MSE), obtained from the normal validation sequences. Thus, the threshold corresponds to the error value below which 95% of normal validation sequences fall, while the upper 5% defines the boundary for potential anomalous behaviour. This percentile-based approach is data-driven and unsupervised because the threshold is derived exclusively from normal validation data without requiring anomaly labels. Importantly, the 95th-percentile threshold does not imply that exactly 5% of normal sequences in the independent test set will be classified as anomalous, since the reconstruction error distributions of the validation and test sets may differ. Consequently, the actual false-positive rate is determined empirically from the confusion matrix obtained on the independent test set.

The threshold formula is given by:

$$\theta = \text{Percentile}_{95}\{MSE(x_i, \hat{x}_i) : x_i \in \text{Normal sequences}\} \quad (3)$$

Evaluation Framework

Model performance was evaluated in three aspects: detection metrics, False-positive rate (FPR) and resource efficiency such as Precision, Recall, F1-Score, AUC-ROC and Matthews Correlation Coefficient (MCC). AUC-ROC was

also calculated on raw MSE reconstruction scores instead of binary predictions, which is threshold-independent and robust to class imbalance (anomaly ratio: 0.44%). MCC was chosen as a balanced scalar metric because it accounts for all four elements of the confusion matrix while False-positive rate

(FPR) was additionally calculated because the substantial class imbalance in the test set makes accuracy alone insufficient for assessing anomaly-detection performance, while balanced accuracy was calculated as the mean of sensitivity and specificity:

$$\text{Precision} = TP / (TP + FP) \quad (4)$$

$$\text{Recall} = TP / (TP + FN) \quad (5)$$

$$F1 - \text{Score} = 2 \times (\text{Precision} \times \text{Recall}) / (\text{Precision} + \text{Recall}) \quad (6)$$

$$MCC = (TP \times TN - FP \times FN) / \sqrt{[(TP + FP)(TP + FN)(TN + FP)(TN + FN)]} \quad (7)$$

$$FPR = FP / (FP + TN)$$

RESULTS AND DISCUSSION

Dataset Characteristics

Table 2 presents the created dataset's basic statistics. There were no missing values in the entire dataset of 527,040 measurements, demonstrating the dataset's integrity. The temperature showed a bimodal distribution with a mean of 27.55°C (SD = 2.43°C) and a primary cluster at 28–30°C (normal operation) and a secondary cluster at 32–38.5°C

matching to injected heat spike anomalies, validating the statistical separability of anomalies. Realistic environmental variability was reflected in the humidity range of 43.1% to 77.5% (mean = 63.76%, SD = 10.43%). Power load exhibited the greatest variability (mean = 403.2 W, SD = 233.2 W), with heat spike power surges accounting for the maximum of 1,400 W and power outage anomalies for the minimum of 0 W.

Table 2: Basic Statistics of the Simulated Smart Home Dataset (n = 527,040)

Statistic	Temperature (°C)	Humidity (%)	Power Load (W)
Count	527,040	527,040	527,040
Mean	27.55	63.76	403.21
Std Dev	2.43	10.43	233.20
Min	23.5	43.1	0
25th Pct.	25.6	56.9	207
Median	27.7	68.4	283
75th Pct.	29.2	72.8	597
Max	38.5	77.5	1,400

Model Training Convergence

Figure 4 illustrates how the AM-LSTM was trained on normal-only training sequences across 15 epochs with a batch size of 32. The MSE training loss decreased from roughly 0.0225 to 0.003, while the validation loss closely followed from roughly 0.0135 to 0.003 by epoch 14. Both MSE loss and MAE converged smoothly. The model's ability to

generalise typical behaviour without overfitting is confirmed by the near-coincidence of training and validation curves across both metrics. This crucial feature ensures that anomalous sequences result in noticeably higher reconstruction errors. A brief optimisation disruption that was resolved by epoch 6 is reflected in a little inflection in training loss at epochs 3–4.

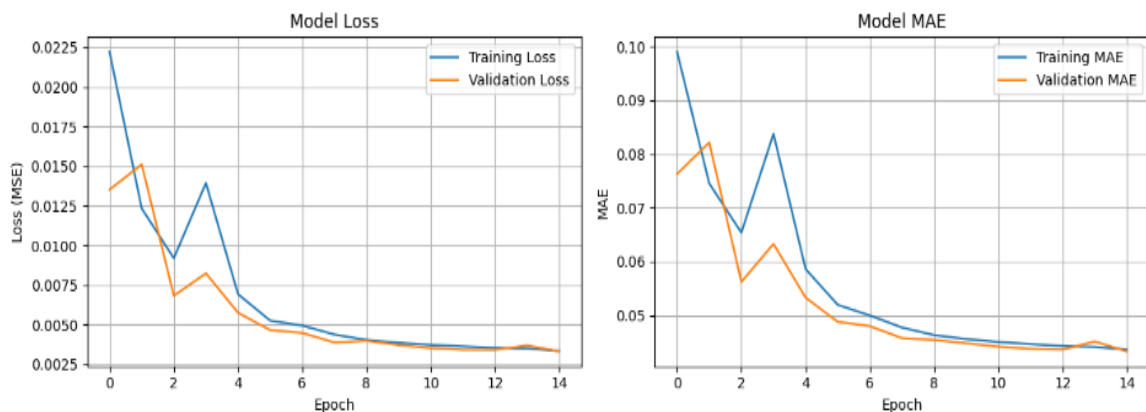


Figure 4: Training Loss and Validation Graph (Model Loss and Model MAE)

Detection Performance

Evaluation on the hold-out test set (87,799 sequences; anomaly ratio 0.44%) yielded the results presented in Table 3

Table 3: Model Performance Results

Metric	Value
Accuracy	95.45%
Precision	8.79%
Recall	98.97%
F1-score	89.86%
AUC-ROC	0.9984
MCC	0.294
False-Positive Rate	4.57%
Balanced Accuracy	97.20%

Confusion Matrix Analysis

Table 4 presents the confusion matrix model that detect 385 of 389 true anomalies, missing only 4 (False Negatives). The essential trade-off of applying a 95th-percentile threshold to a large test set roughly 5% of the 83,410 normal sequences surpass the barrier by design is reflected in the False Positive count of 3,994. For safety-critical anomaly detection, where

missing a real occurrence (heat spike, power failure) is considerably more crucial than creating an unnecessary warning, this result demonstrates a trade-off between high anomaly sensitivity and a substantial false-alarm burden. The suitability of this operating point therefore depends on the intended application and requires further threshold optimisation.

Table 4: Confusion matrix for AM-LSTM on the hold-out test set

	Predicted Negative	Predicted Positive
Actual Negative	83,416 (TN)	3,994 (FP)
Actual Positive	4 (FN)	385 (TP)

Reconstruction Error Separation

In addition to complementing the measurements, the graphs in Figure 5 offer a more thorough understanding of how the model functions when identifying abnormalities. The reconstruction-error distribution shows a detection threshold of 0.005 and strong separation between normal and anomalous sequences, resulting in an AUC-ROC of 0.9984. However, the selected threshold produced 3,994 false positives, demonstrating that a high AUC does not necessarily translate into optimal threshold-specific precision. The confusion matrix recorded 83,416 true negatives, 3,994 false positives, 385 true positives, and 4 false negatives. These results indicate high sensitivity to anomalies but also a considerable false-positive burden. Given the pronounced class imbalance, accuracy alone is insufficient for evaluating performance. Therefore, precision, recall, F1-score, specificity, false-positive rate, and precision-recall analysis should be considered when assessing the practical effectiveness of the selected operating threshold.

The model's performance on an example of a normal sequence is shown in the Normal Sample Reconstruction graphic, which displays the original load signal in blue and its reconstruction in orange. Throughout the 250 timesteps, the reconstructed pattern is consistent with the original one, effectively reproducing both the initial high-activity period and the subsequent lower phase.

Thus, the reconstruction error is low and is even below the detection threshold. On the contrary, the Anomaly Sample Reconstruction plot, where index 328 denotes the anomalous type of anomaly, presents the distinct behavior of the model on the example of an anomaly. The original load signal, plotted in red, is rather constant and low for the first 200 timesteps but then abruptly rises above 1.2 normalized units,

matching the midnight appliance usage and heat spike anomaly types mentioned in (Appendix B). Meanwhile, the reconstructed signal, plotted in orange, plateaus at roughly 0.55 after around 200 timesteps. Therefore, the model cannot effectively recreate such anomalous patterns, resulting in increased reconstruction error and thus a successful anomaly detection. The combination of the four plots in Figure 5 provides validation for the reconstruction-based anomaly detection framework proposed, proving the efficacy of the method's application on the Raspberry Pi edge device.

The almost complete separation of the distributions of normal and anomalous reconstruction errors with a threshold value of 0.005, which effectively divides the two categories, explains the area under ROC curve value of 0.9984 achieved and parallels the reconstruction error separation reported by Hammad *et al.*, (2023). This research revealed that CNN-LSTM Model models, trained solely on normal time-series datasets of IoT, display a clear inductive bias towards reconstructing normal examples with low error while achieving higher errors when reconstructing anomalous instances of data.

The reconstruction graph of a normal sample, which displays an almost perfect correlation between the reconstructed and the initial data point by point along all the 250 timesteps, proves the ability of the model to effectively learn the patterns associated with normal behavior in smart home appliance use. It is consistent with the findings by Xiao *et al.*, (2024) who stated that unsupervised models for anomaly detection trained solely on normal behavioral logs achieve a deep enough understanding of the patterns to be able to reliably reconstruct normal instances and fail at anomalous ones. The anomaly reconstruction graph of instance number 328 displays a flat reconstruction graph.

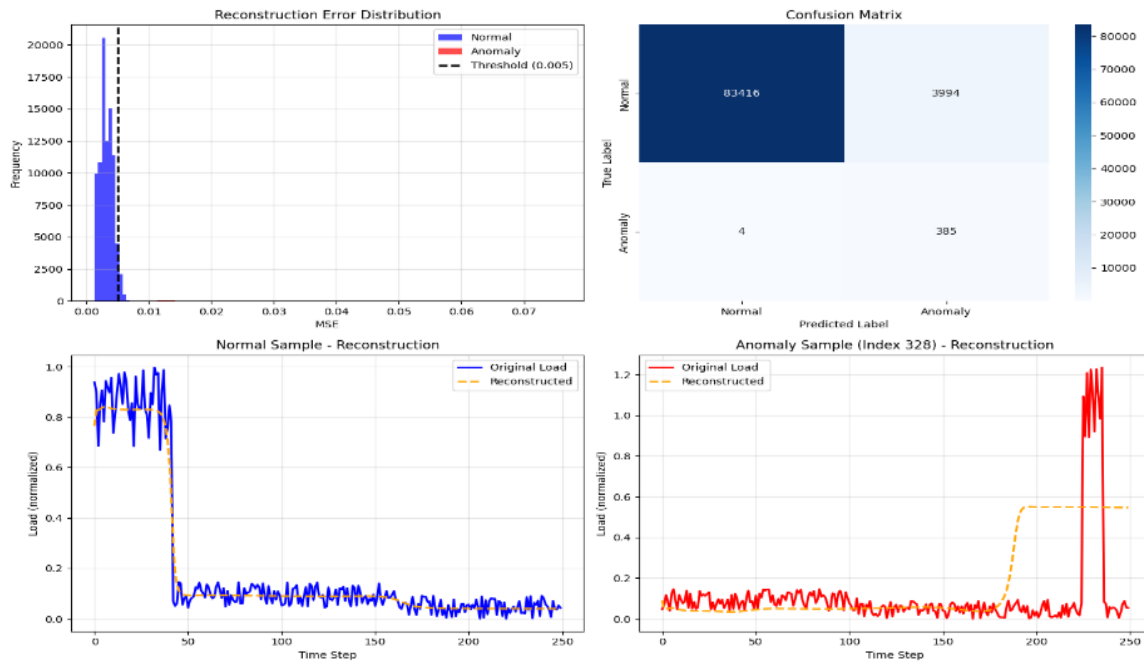


Figure 5: Reconstruction Error Distribution, Confusion Matrix, Normal Sample - Reconstruction and Anomaly Sample–Reconstruction

Comparative Analysis

Table 5 presents several prominent differences observed in the comparative analysis. The proposed TinyML-optimized Multimodal LSTM framework combines several characteristics, including high recall, high AUC-ROC, compact model size, edge deployment, multimodal sensing, temporal sequence modelling, and unsupervised anomaly detection. Under the experimental conditions of this study, the proposed model achieved these characteristics within a single framework. In comparison, Nimmy *et al.* (2023) reported a recall rate of 99.2%; however, their FFNN method does not operate at the edge and does not incorporate

multimodal sensing or unsupervised anomaly detection. Similarly, Shubham *et al.* (2025) reported a model size below 100 KB, but their approach relies on vibration data obtained from a single sensor and does not incorporate temporal sequence modelling or unsupervised detection. These differences indicate that the proposed framework integrates TinyML optimization, multimodal sensing, temporal sequence modelling, and unsupervised anomaly detection within an edge-deployment architecture. The comparison therefore highlights the specific combination of characteristics incorporated by the proposed approach under the experimental conditions considered in this study.

Table 5: Comparative Analysis with Selected Prior Works

S/N	Study	Approach	Recall	AUC-ROC	Model Size	Edge Deployed	Multimodal	Temporal Modelling	Unsupervised
1	Yamauchi <i>et al.</i> , (2020)	Behaviour sequence modelling	N/R	N/R	N/R	No	No	No	No
2	Manirho <i>et al.</i> , (2020)	Hybrid RF + feature selection	N/R	N/R	N/R	No	No	No	No
3	Fahad & Tahir, (2021)	MQ sensors + BLE	N/R	N/R	N/R	Yes	No	No	No
4	Antonini <i>et al.</i> , (2023b)	TinyML on ESP32	N/R	N/R	<16ms inf.	Yes	No	No	Yes
5	Zakariah & Almazyad, (2023)	Active learning + RF	N/R	N/R	N/R	No	No	No	No
6	(Atassi, 2023)	Federated Learning	N/R	>0.95	N/R	No	No	No	Yes
7	Atassi, (2023)	Energy profiling + DT	N/R	N/R	N/R	Yes	No	No	No
8	Hammad <i>et al.</i> , (2023)	TinyML on ESP32	N/R	N/R	N/R	Yes	No	No	No
9	Nimmy <i>et al.</i> , (2023)	FFNN power consumption	99.2%	N/R	N/R	No	No	No	No
10	Hammad <i>et al.</i> , (2023)	Unsupervised temporal detection	N/R	N/R	N/R	No	No	Yes	Yes
11	Shubham Gupta & Shiv Naresh Shivhare, (2025)	1D CNN + TinyML	N/R	N/R	<100KB	Yes	No	No	No
12	(Katib <i>et al.</i> , 2025)	DL + TinyML predictive maintenance	N/R	N/R	N/R	Yes	No	No	No
13	Proposed Model	TinyML Multimodal LSTM Model	98.97%	0.9984	196KB	Yes	Yes	Yes	Yes

N/R = Not Reported

Discussion

The AM-LSTM achieved an AUC-ROC of 0.9984, indicating strong discrimination between normal and anomalous sequences across thresholds. This is consistent with reconstruction-based anomaly detection, where models trained on normal data generally produce lower reconstruction errors for normal samples and higher errors for anomalous patterns (Xiao et al., 2024). At the selected operating point, the model achieved 98.97% recall, detecting 385 of 389 anomalies across heat-spike, power-outage, and midnight-usage patterns, indicating that multimodal sensor information captured complementary anomaly characteristics.

However, precision was 8.79%, with 3,994 normal sequences incorrectly classified as anomalous. Thus, the high recall was accompanied by substantial false alarms. Although the AUC-ROC demonstrates strong overall discrimination, it does not reflect precision at a specific threshold. The 95th-percentile validation threshold also did not produce exactly 5% false positives on the independent test set; the observed false-positive rate was 4.57%, reflecting differences between validation and test distributions. Threshold calibration is therefore an important consideration for practical deployment.

The INT8-quantised model occupied 196 KB, demonstrating a compact model representation, but model size alone cannot establish real-time performance, energy efficiency, or suitability for resource-constrained hardware. These properties require direct measurements of latency, peak memory, CPU utilisation, and energy consumption on the target platform. Similarly, the online-learning component provides a mechanism for adapting to changing sensor distributions, but controlled concept-drift experiments are required to establish its effectiveness under evolving operating conditions. Local inference may reduce dependence on cloud processing and external transmission of raw sensor data, although comprehensive privacy and end-to-end latency assessments remain necessary.

CONCLUSION

This study developed AM-LSTM, a TinyML-optimised multimodal LSTM model for unsupervised anomaly detection in smart-home IoT systems, and evaluated it on a Raspberry Pi edge-computing platform using temperature, humidity, and power-load modalities. INT8 quantisation reduced the model size to 196 KB, while adaptive learning addressed concept drift. Under the experimental conditions, AM-LSTM achieved 98.97% recall, 0.9984 AUC-ROC, and 95.45% accuracy on the simulated dataset. However, the 8.79% precision obtained at the selected threshold indicates a substantial false-alarm burden, while the simulated dataset limits generalisability. The 196 KB model size alone also does not establish real-time performance, energy efficiency, or microcontroller suitability. Future work should therefore evaluate the model on real-world datasets, optimise thresholds and false-alarm rates, assess multimodal fusion and quantisation through ablation studies, validate concept-drift adaptation under controlled distribution shifts, and directly measure latency, memory, CPU utilisation, and energy consumption on representative edge and microcontroller platforms.

REFERENCES

Abadade, Y., Temouden, A., Bamoumen, H., Benamar, N., Chtouki, Y., & Hafid, A. S. (2023). A Comprehensive Survey on TinyML. *IEEE Access*, 11. <https://doi.org/10.1109/ACCESS.2023.3294111>

Antonini, M., Pincheira, M., Vecchio, M., & Antonelli, F. (2023a). An Adaptable and Unsupervised TinyML Anomaly Detection System for Extreme Industrial Environments †. *Sensors*, 23(4). <https://doi.org/10.3390/s23042344>

Antonini, M., Pincheira, M., Vecchio, M., & Antonelli, F. (2023b). An Adaptable and Unsupervised TinyML Anomaly Detection System for Extreme Industrial Environments †. *Sensors*, 23(4). <https://doi.org/10.3390/s23042344>

Arciniegas, S., Rivero, D., Piñan, J., Diaz, E., & Rivas, F. (2025). IoT device for detecting abnormal vibrations in motors using TinyML. *Discover Internet of Things*, 5(1). <https://doi.org/10.1007/s43926-025-00142-4>

Arora, S., Rani, R., & Saxena, N. (2024a). A systematic review on detection and adaptation of concept drift in streaming data using machine learning techniques. In *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* (Vol. 14, Number 4). John Wiley and Sons Inc. <https://doi.org/10.1002/widm.1536>

Arora, S., Rani, R., & Saxena, N. (2024b). A systematic review on detection and adaptation of concept drift in streaming data using machine learning techniques. In *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* (Vol. 14, Number 4). John Wiley and Sons Inc. <https://doi.org/10.1002/widm.1536>

Atassi, R. (2023). Anomaly Detection in IoT Networks: Machine Learning Approaches for Intrusion Detection. *Fusion: Practice and Applications*, 13(1). <https://doi.org/10.54216/FPA.130110>

Bouazza, A., Kodad, M., Azizi, M., Oulahyane, A., & Chafik, H. (2026). Federated Learning in Smart Cities: A Comprehensive Review of Privacy, Security, and Scalable AI Integration. *Lecture Notes in Networks and Systems*, 1641 LNNS. https://doi.org/10.1007/978-3-032-06061-7_13

da Costa Filho, P. E., Marques, L. A. de A., Lima, I. da S. F. de, Sousa, E. L. de, Kreutz, M. E., Neto, A. V., Cunha, E. N., & Vieira, D. (2025). Machine-Learning-Based Classification of Electronic Devices Using an IoT Smart Meter. *Informatics*, 12(2). <https://doi.org/10.3390/informatics12020048>

Disabato, S., & Roveri, M. (2024). Tiny Machine Learning for Concept Drift. *IEEE Transactions on Neural Networks and Learning Systems*, 35(6). <https://doi.org/10.1109/TNNLS.2022.3229897>

Dutta, D. L., & Bharali, S. (2021). TinyML Meets IoT: A Comprehensive Survey. In *Internet of Things (Netherlands)* (Vol. 16). <https://doi.org/10.1016/j.iot.2021.100461>

Ezugwu, A. E., Taiwo, O., Egwuche, O. S., Abualigah, L., Van Der Merwe, A., Pal, J., Saha, A. K., Alzahrani, A. I., Alblehai, F., Greeff, J., & Olusanya, M. O. (2025). *Smart Homes of the Future*. Wiley Online Library/AE Ezugwu, O Taiwo, OS Egwuche, L Abualigah, A Van Der Merwe, J Pal, AK SahaTransactions on Emerging Telecommunications Technologies, 2025•Wiley Online Library, 36(1). <https://doi.org/10.1002/ETT.70041>

Fahad, L. G., & Tahir, S. F. (2021). Activity recognition and anomaly detection in smart homes. *Neurocomputing*, 423, 362–372. <https://doi.org/10.1016/j.neucom.2020.10.102>

- Ferrag, M. A., Friha, O., Hamouda, D., Maglaras, L., & Janicke, H. (2022). Edge-IIoTset: A New Comprehensive Realistic Cyber Security Dataset of IoT and IIoT Applications for Centralized and Federated Learning. *IEEE Access*, 10. <https://doi.org/10.1109/ACCESS.2022.3165809>
- Fu, R., Shi, S., Guo, H., Wang, T., Qiang, C., Wen, Z., Tao, J., Qi, X., Lu, Y., Wang, X., Wang, Z., Liu, Y., Liu, X., Zhang, S., & Li, G. (2024). MINT: a Multi-modal Image and Narrative Text Dubbing Dataset for Foley Audio Content Planning and Generation. <http://arxiv.org/abs/2406.10591>
- Gøthesen, S., Haddara, M., & Kumar, K. N. (2023). Empowering homes with intelligence: An investigation of smart home technology adoption and usage. *Internet of Things (Netherlands)*, 24. <https://doi.org/10.1016/j.iot.2023.100944>
- Hammad, S. S., Iskandaryan, D., & Trilles, S. (2023). An unsupervised TinyML approach applied to the detection of urban noise anomalies under the smart cities environment. *Internet of Things (Netherlands)*, 23. <https://doi.org/10.1016/j.iot.2023.100848>
- Hong, Y. K., Wang, Z. Y., & Cho, J. Y. (2022). Global Research Trends on Smart Homes for Older Adults: Bibliometric and Scientometric Analyses. *International Journal of Environmental Research and Public Health*, 19(22). <https://doi.org/10.3390/ijerph192214821>
- Hu, L., Lu, Y., & Feng, Y. (2025). Concept Drift Detection Based on Deep Neural Networks and Autoencoders. *Applied Sciences (Switzerland)*, 15(6). <https://doi.org/10.3390/app15063056>
- Jiang, J. C., Kantarci, B., Oktug, S., & Soyata, T. (2020). Federated learning in smart city sensing: Challenges and opportunities. In *Sensors (Switzerland) (Vol. 20, Number 21)*. <https://doi.org/10.3390/s20216230>
- Kanimozhi, R. (2026). Real-Time Cyber Security Risk Assessment in Smart Home IoT Networks Using a Hybrid Fuzzy-Neural Soft Computing Framework. *Transactions on Emerging Telecommunications Technologies*, 37(7). <https://doi.org/10.1002/ett.70467>
- Katib, I., Albassam, E., Sharaf, S. A., & Ragab, M. (2025). Safeguarding IoT consumer devices: Deep learning with TinyML driven real-time anomaly detection for predictive maintenance. *Ain Shams Engineering Journal*, 16(2). <https://doi.org/10.1016/j.asej.2025.103281>
- Kumar, S. V., Aloy Anuja Mary, G., Chohan, J. S., & Kalita, K. (2024). Efficient sensor anomaly detection using Markov-LSTM architecture for methane sensing. *Journal of Autonomous Intelligence*, 7(3). <https://doi.org/10.32629/jai.v7i3.1285>
- Kumari, S., Prabha, C., Karim, A., Hassan, M. M., & Azam, S. (2024). A Comprehensive Investigation of Anomaly Detection Methods in Deep Learning and Machine Learning: 2019–2023. In *IET Information Security (Vol. 2024, Number 1)*. <https://doi.org/10.1049/2024/8821891>
- Litayem, N. (2024a). Scalable Smart Home Management with ESP32-S3: A Low-Cost Solution for Accessible Home Automation. 2024 International Conference on Computer and Applications, ICCA 2024. <https://doi.org/10.1109/ICCA62237.2024.10927887>
- Litayem, N. (2024b). Scalable Smart Home Management with ESP32-S3: A Low-Cost Solution for Accessible Home Automation. 2024 International Conference on Computer and Applications, ICCA 2024. <https://doi.org/10.1109/ICCA62237.2024.10927887>
- Maniriho, P., Niyigaba, E., Bizimana, Z., Twiringiyimana, V., Mahoro, L. J., & Ahmad, T. (2020). Anomaly-based Intrusion Detection Approach for IoT Networks Using Machine Learning. *CENIM 2020 - Proceeding: International Conference on Computer Engineering, Network, and Intelligent Multimedia* 2020. <https://doi.org/10.1109/CENIM51130.2020.9297958>
- Morshedi, R., & Matinkhah, S. M. (2025). A Comprehensive Review of Deep Learning Techniques for Anomaly Detection in IoT Networks: Methods, Challenges, and Datasets. In *Engineering Reports (Vol. 7, Number 9)*. John Wiley and Sons Inc. <https://doi.org/10.1002/eng2.70415>
- Muthunambu, N. K., Prabakaran, S., Kavin, B. P., Siruvangur, K. S., Chinnadurai, K., & Ali, J. (2024). A Novel Eccentric Intrusion Detection Model Based on Recurrent Neural Networks with Leveraging LSTM. *Computers, Materials and Continua*, 78(3). <https://doi.org/10.32604/cmc.2023.043172>
- Naik, R., Devaraddi, R., Kutte, V., Patil, P., & Dalavi, V. (2026). Real-Time Vibration Detection and Signal Analysis of Machines. *Proceedings of 3rd International Conference on Electronics, Computing, Communication and Control Technology: AI-Driven Smart Electronics and Next-Generation Communication Technologies, ICECCC 2026*. <https://doi.org/10.1109/ICECCC70334.2026.11633152>
- Nimmy, K., Dilraj, M., Sankaran, S., & Achuthan, K. (2023). Leveraging power consumption for anomaly detection on IoT devices in smart homes. *Journal of Ambient Intelligence and Humanized Computing*, 14(10). <https://doi.org/10.1007/s12652-022-04110-6>
- Pradhan, S. K., & Gangashetty, S. V. (2026). An Intelligent Voice-Based Authentication and Anomaly Detection Framework for Secure Smart-Home Environments. *Sci*, 8(7). <https://doi.org/10.3390/sci8070162>
- Rahim, A., Zhong, Y., Ahmad, T., Ahmad, S., Pławiak, P., & Hammad, M. (2023). Enhancing Smart Home Security: Anomaly Detection and Face Recognition in Smart Home IoT Devices Using Logit-Boosted CNN Models. *Sensors*, 23(15). <https://doi.org/10.3390/s23156979>
- Rajaan, R., Kumar, L., Choudhary, N., Sharma, A., & Butwall, M. (2025a). Anomaly Detection in Smart Home IoT Systems Using Machine Learning Approaches. *International Journal of Engineering, Business and Management*, 9(2), 43–47. <https://doi.org/10.22161/ijebm.9.2.5>
- Rajaan, R., Kumar, L., Choudhary, N., Sharma, A., & Butwall, M. (2025b). Anomaly Detection in Smart Home IoT Systems Using Machine Learning Approaches. *International Journal of Engineering, Business and Management*, 9(2), 43–47. <https://doi.org/10.22161/ijebm.9.2.5>

- Reis, M. J. C. S., & Serôdio, C. (2025). Edge AI for Real-Time Anomaly Detection in Smart Homes. *Future Internet*, 17(4). <https://doi.org/10.3390/fi17040179>
- Shaheen, M., Farooq, M. S., Umer, T., & Kim, B. S. (2022). Applications of Federated Learning; Taxonomy, Challenges, and Research Trends. *Electronics (Switzerland)*, 11(4). <https://doi.org/10.3390/electronics11040670>
- Shang, D., Zhang, G., & Lu, J. (2025). Concept drift detection based on radial distance. *Neurocomputing*, 653. <https://doi.org/10.1016/j.neucom.2025.131190>
- Shanmuganathan, V., & Suresh, A. (2023). LSTM-Markov based efficient anomaly detection algorithm for IoT environment. *Applied Soft Computing*, 136. <https://doi.org/10.1016/j.asoc.2023.110054>
- Shubham Gupta, & Shiv Naresh Shivhare. (2025). Embedded TinyML for Predictive Maintenance: Vibration Analysis on ESP32 with Real-Time Fault Detection in Industrial Equipment. *International Journal on Computational Modelling Applications*, 2(2), 1–17. <https://doi.org/10.63503/j.ijcma.2025.114>
- Srilakshmi, V., Babu Veesam, S., Shiva Rama Krishna, M., Kumar Munaganuri, R., & Devesi Sivaprasad, D. (2025). Design of an Improved Model for Anomaly Detection in CCTV Systems Using Multimodal Fusion and Attention-Based Networks. *IEEE Access*, 13, 27287–27309. <https://doi.org/10.1109/ACCESS.2025.3536501>
- Sun, W., Cao, L., Guo, Y., & Du, K. (2024). Multimodal and multiscale feature fusion for weakly supervised video anomaly detection. *Scientific Reports*, 14(1). <https://doi.org/10.1038/s41598-024-73462-0>
- Xiao, J., Xu, Z., Zou, Q., Li, Q., Zhao, D., Fang, D., Li, R., Tang, W., Li, K., Zuo, X., Hu, P., Jiang, Y., Weng, Z., & Lyu, M. R. (2024). Make your home safe: Time-aware unsupervised user behavior anomaly detection in smart homes via loss-guided mask. *DL.Acm.OrgJ Xiao, Z Xu, Q Zou, Q Li, D Zhao, D Fang, R Li, W Tang, K Li, X Zuo, P Hu, Y Jiang, Z Weng* Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and, 2024•dl.Acm.Org, 3551–3562. <https://doi.org/10.1145/3637528.3671708>
- Yamauchi, M., Ohsita, Y., Murata, M., Ueda, K., & Kato, Y. (2020). Anomaly Detection in Smart Home Operation from User Behaviors and Home Conditions. *IEEE Transactions on Consumer Electronics*, 66(2), 183–192. <https://doi.org/10.1109/TCE.2020.2981636>
- Yang, L., & Shami, A. (2021). A Lightweight Concept Drift Detection and Adaptation Framework for IoT Data Streams. *IEEE Internet of Things Magazine*, 4(2), 96–101. <https://doi.org/10.1109/IOTM.0001.2100012>
- Yin, C., Zhang, S., Wang, J., & Xiong, N. N. (2022). Anomaly Detection Based on Convolutional Recurrent Autoencoder for IoT Time Series. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 52(1). <https://doi.org/10.1109/TSMC.2020.2968516>
- Zakariah, M., & Almazyad, A. S. (2023). Anomaly Detection for IOT Systems Using Active Learning. *Applied Sciences (Switzerland)*, 13(21). <https://doi.org/10.3390/app132112029>
- Zeng, F., Chen, M., Qian, C., Wang, Y., Zhou, Y., & Tang, W. (2023). Multivariate time series anomaly detection with adversarial transformer architecture in the Internet of Things. *Future Generation Computer Systems*, 144, 244–255. <https://doi.org/10.1016/j.future.2023.02.015>
- Zhao, Z., Guo, H., & Wang, Y. (2024). A multi-information fusion anomaly detection model based on convolutional neural networks and AutoEncoder. *Scientific Reports*, 14(1). <https://doi.org/10.1038/s41598-024-66760-0>

