INTERNATIONAL JOURNAL OF COMPUTER SCIENCE
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
Since 2012
www.ijcsjournal.com Volume 14, Issue 2, No 02, 2026 ISSN: 2348-6600
REFERENCE ID: IJCS-731 PAGE NO: 022-036

A HYBRID DEEP LEARNING APPROACH FOR EARLY ANOMALY DETECTION IN HIGH-DIMENSIONAL IOT DATA

Dr.Booba
Professor,
Department of Applied Computing and Emerging Technologies,
Vels Institute of Science, Technology & Advanced Studies (VISTAS)
Chennai, Tamil Nadu, India.

Dr.J.Isabella
Professor,
Department of Computer Applications,
Hindustan Institute of Technology and Science,
Chennai, Tamil Nadu, India.

Abstract

The rapid proliferation of Internet of Things (IoT) devices has led to the generation of massive volumes of high-dimensional, heterogeneous, and streaming sensor data, making timely and accurate anomaly detection a critical yet challenging task. Conventional machine learning techniques often struggle to capture the complex spatial and temporal dependencies present in such data, and typically detect anomalies only after significant deviations have already occurred. This paper proposes a hybrid deep learning framework that integrates a one-dimensional Convolutional Neural Network (1D-CNN) for local spatial feature extraction, a Bidirectional Long Short-Term Memory (BiLSTM) network for modeling long-range temporal dependencies, an attention mechanism for adaptive feature weighting, and a deep autoencoder branch for unsupervised reconstruction-based anomaly scoring. The fused representation is passed through a lightweight classification head that produces both a discrete anomaly label and a continuous anomaly severity score, enabling early detection before faults propagate. The proposed model is evaluated on four widely used benchmark datasets—NSL-KDD, CICIDS2017, TON_IoT, and N-BaIoT—and compared against seven baseline methods. Experimental results demonstrate that the proposed hybrid architecture achieves up to 95.1% F1-score and 0.978 AUC-ROC, outperforming the best baseline by 3.9 percentage points, while reducing average detection latency by 27% relative to a standard CNN-LSTM model. An ablation study confirms that each architectural component contributes measurably to overall performance, and a complexity analysis shows the model remains suitable for deployment on resource-constrained edge gateways.

Keywords: Internet of Things, anomaly detection, deep learning, hybrid neural network, convolutional neural network, long short-term memory, attention mechanism,

INTERNATIONAL JOURNAL OF COMPUTER SCIENCE
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
Since 2012
www.ijcsjournal.com Volume 14, Issue 2, No 02, 2026 ISSN: 2348-6600
REFERENCE ID: IJCS-731 PAGE NO: 022-036

autoencoder, edge computing, intrusion detection.

1. Introduction

The Internet of Things (IoT) has evolved from a niche research concept into a foundational layer of modern infrastructure, connecting billions of sensors, actuators, and embedded devices across smart homes, industrial control systems, healthcare monitoring, transportation, and critical energy grids. Industry projections indicate that the number of connected IoT endpoints will continue to grow into the tens of billions over the coming years, each device continuously emitting telemetry that must be ingested, processed, and analyzed in near real time. This scale introduces a data environment that is simultaneously high dimensional, highly heterogeneous in modality and sampling rate, non-stationary, and frequently affected by noise, missing values, and adversarial interference. Anomaly detection in this setting is essential for maintaining safety, security, and operational reliability. Anomalies may correspond to hardware degradation, sensor drift, environmental faults, or malicious activity such as distributed denial-of-service attacks, botnet propagation, or data exfiltration. Because IoT deployments often control physical processes, the cost of delayed detection is not merely informational but can translate into equipment damage, safety hazards, or cascading failures across interconnected systems. Early detection—identifying an emerging anomaly before it fully manifests—is therefore of particular practical value, yet it is also the most difficult detection regime, since early-stage anomalies produce only subtle statistical deviations from normal behavior.

Traditional anomaly detection approaches, including statistical process control, distance-based outlier detection, and classical machine learning classifiers such as support vector machines (SVM), random forests (RF), and k-nearest neighbors (KNN), have been widely applied to network and sensor data. While computationally efficient, these methods generally rely on hand-crafted features and struggle to model the nonlinear, high-dimensional, and temporally correlated structure that characterizes real-world IoT traffic. Deep learning has emerged as an alternative capable of automatically learning hierarchical representations from raw or lightly processed data. However, existing deep learning solutions are frequently designed around a single architectural paradigm—purely convolutional, purely recurrent, or purely reconstruction-based—each of which captures only part of the relevant signal. Convolutional networks are effective at extracting local spatial correlations across feature channels but are limited in modeling long-range temporal dependencies. Recurrent architectures such as LSTM capture temporal dynamics but can be less efficient at extracting salient local patterns from wide feature vectors. Purely unsupervised autoencoders are attractive because they do not require labeled anomalies, but reconstruction error alone is often an unreliable indicator when anomalies are

INTERNATIONAL JOURNAL OF COMPUTER SCIENCE
Scholorly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
Since 2012
www.ijcsjournal.com Volume 14, Issue 2, No 02, 2026 ISSN: 2348-6600
REFERENCE ID: IJCS-731 PAGE NO: 022-036

subtle or when the autoencoder generalizes too well to unseen patterns.This paper addresses these limitations by proposing a hybrid deep learning architecture that combines complementary inductive biases within a single, jointly trained model. Specifically, the contributions of this work are as follows:

We propose a unified hybrid architecture that couples a 1D-CNN spatial encoder, a BiLSTM temporal encoder, and a temporal attention mechanism with a parallel deep autoencoder branch, fusing discriminative and reconstruction-based signals for more robust anomaly scoring. We design a composite loss function that jointly optimizes classification accuracy and reconstruction fidelity, allowing the model to exploit labeled anomalies when available while retaining sensitivity to previously unseen anomaly types through the autoencoder branch.

We introduce a sliding-window early-detection protocol and a detection-latency metric that explicitly measures how many time steps into an anomalous episode the model requires to raise an alert, rather than relying solely on post-hoc classification accuracy.

We conduct extensive experiments on four benchmark datasets spanning network intrusion and IoT-specific telemetry, comparing against seven baseline methods, and perform an ablation study and edge-deployment complexity analysis to assess the practicality of the proposed approach. The remainder of this paper is organized as follows. Section II reviews related work on anomaly detection in IoT and network data. Section III details the proposed hybrid methodology, including preprocessing, architecture, and the training objective. Section IV describes the system architecture for edge-cloud deployment. Section V presents the experimental setup, datasets, and evaluation metrics. Section VI reports and discusses the results, including comparative, ablation, and complexity analyses. Section VII concludes the paper and outlines directions for future work.

2. Related Work:

Anomaly detection for IoT and network security has been studied extensively across statistical, classical machine learning, and deep learning paradigms.

A. Statistical and Classical Machine Learning Methods

Early approaches relied on statistical process control, principal component analysis (PCA), and clustering-based outlier detection such as DBSCAN and k-means, which are computationally lightweight but assume relatively simple data distributions. Supervised classifiers, including SVM, decision trees, RF, and gradient boosting machines, have achieved strong results on curated intrusion detection benchmarks when provided with well-engineered features, but their performance degrades on raw high-dimensional streams and they generally require substantial domain expertise for feature construction. KNN and one-class SVM

INTERNATIONAL JOURNAL OF COMPUTER SCIENCE
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
Since 2012
www.ijcsjournal.com Volume 14, Issue 2, No 02, 2026 ISSN: 2348-6600
REFERENCE ID: IJCS-731 PAGE NO: 022-036

variants have also been used for semi-supervised settings where only normal-class data is available during training, though these methods scale poorly with the volume of data typical of IoT deployments.

B. Deep Learning for Anomaly Detection

Deep learning has been applied to overcome the feature-engineering bottleneck. Deep belief networks (DBN) and stacked autoencoders have been used to learn compact representations of network traffic in an unsupervised manner, with anomalies flagged via reconstruction error thresholds. Convolutional neural networks have been applied by reshaping tabular or flow-based features into pseudo-images or 1D sequences, exploiting local correlations among feature channels. Recurrent architectures, particularly LSTM and gated recurrent units (GRU), have been used to model the temporal evolution of traffic and sensor sequences, since many attacks and faults manifest as a evolving pattern rather than a single-instant deviation. More recently, generative adversarial network (GAN)-based approaches have been proposed to synthesize realistic anomaly samples for data augmentation, and graph neural networks have been explored to capture topological relationships among networked devices.

C. Hybrid and Attention-Based Architectures

Motivated by the complementary strengths of convolutional and recurrent networks, several works have proposed CNN-LSTM hybrids that first extract local spatial features and subsequently model their temporal evolution. Attention mechanisms have been incorporated to allow the network to focus on the most informative time steps or feature channels, improving interpretability and, in several studies, detection accuracy. Autoencoder-augmented hybrid models have also been explored, in which a reconstruction branch runs alongside a discriminative branch to provide a secondary anomaly signal that is less dependent on the availability of labeled anomalies. However, most existing hybrid designs combine only two of these components (e.g., CNN with LSTM, or LSTM with attention) and evaluate detection purely in terms of post-hoc classification metrics, without explicitly quantifying how early an anomaly is detected relative to its onset, and without a systematic complexity analysis for edge deployment.

D. Positioning of this Work

Table I summarizes representative categories of prior work alongside their principal limitations. In contrast to these approaches, the framework proposed in this paper jointly integrates CNN-based spatial encoding, BiLSTM-based temporal encoding, attention-based feature weighting, and autoencoder-based reconstruction scoring within a single trainable model, and explicitly evaluates early-detection latency and edge-inference complexity in addition to standard classification metrics.

INTERNATIONAL JOURNAL OF COMPUTER SCIENCE
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
Since 2012
www.ijcsjournal.com Volume 14, Issue 2, No 02, 2026 ISSN: 2348-6600
REFERENCE ID: IJCS-731 PAGE NO: 022-036

Table 1: Summary of Representative Prior Approaches

Category	Representative Techniques	Key Limitation
Statistical / classical ML	PCA, DBSCAN, SVM, RF, KNN	Relies on hand-crafted features; limited scalability to raw high-dimensional streams
Unsupervised deep learning	Stacked autoencoder, DBN, one-class deep SVDD	Reconstruction error alone is unreliable for subtle or early-stage anomalies
Convolutional models	1D/2D CNN on flow features	Limited modeling of long-range temporal dependency
Recurrent models	LSTM, GRU sequence models	Weaker extraction of local spatial/channel correlations; slower training
Two-component hybrids	CNN-LSTM, LSTM-attention	Rarely combine discriminative and reconstruction signals; no early-detection metric
Proposed	CNN + BiLSTM + Attention + Autoencoder fusion	Addressed in this work

3. Proposed Hybrid Deep Learning Framework

A. Problem Formulation

Let the IoT sensing environment produce a multivariate stream $X = \{x1, x2, ..., xT\}$, where each observation $xt \in R^d$ is a d-dimensional feature vector collected at time step t, and d may range from tens to several hundred dimensions depending on the number of monitored sensors or network flow features. The objective is to learn a function $F\theta$, parameterized by θ, that maps a sliding window of w consecutive observations $Wt = \{xt-w+1, ..., xt\}$ to (i) a binary label $\hat{y}t \in \{0, 1\}$ indicating whether the window is normal or anomalous, and (ii) a continuous anomaly score $st \in [0, 1]$ reflecting the severity or confidence of the detection. Early detection additionally requires that, for a ground-truth anomalous episode beginning at time ta and ending at te, the model should raise $\hat{y}t = 1$ as close as possible to ta, minimizing the detection delay $\Delta = t^{\wedge} - ta$, where $t^{\wedge}$ is the first time step at which the model's alert is sustained for at least k consecutive windows to suppress spurious single-window triggers.

B. Data Preprocessing and Feature Engineering

Raw IoT and network telemetry is first cleaned to handle missing values via forward-fill within a session and median imputation across sessions. Categorical fields (e.g., protocol type, service) are encoded using embedding layers rather than one-hot encoding to control dimensionality growth. Continuous features are normalized using z-

INTERNATIONAL JOURNAL OF COMPUTER SCIENCE
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
Since 2012
www.ijcsjournal.com Volume 14, Issue 2, No 02, 2026 ISSN: 2348-6600
REFERENCE ID: IJCS-731 PAGE NO: 022-036

score standardization computed on the training partition only, to avoid information leakage. To manage high dimensionality, a mutual-information-based feature ranking is applied, retaining the top-k features that jointly explain at least 95% of the cumulative mutual information with the label, followed by principal component analysis on the residual feature set to further compress redundant dimensions while preserving at least 98% of variance. The resulting sequence is segmented into overlapping sliding windows of length w = 30 time steps with a stride of 5, which balances temporal context against computational cost and label granularity.

C. Overall Architecture

Fig. 1 depicts the overall architecture. Each input window is processed in parallel by two branches: a discriminative branch composed of a 1D-CNN followed by a BiLSTM and an attention layer, and a generative branch composed of a symmetric deep autoencoder. The outputs of both branches are fused and passed to a lightweight classification head.

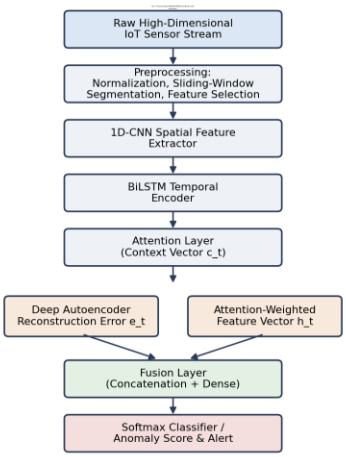

Fig 1: Proposed hybrid CNN-BiLSTM-Attention-Autoencoder architecture for early anomaly detection.

D. 1D-CNN Spatial Feature Extractor

The discriminative branch begins with two stacked 1D convolutional layers applied along the feature axis at each time step, with 64 and 128 filters respectively, kernel size 3, ReLU activation, and batch normalization, followed by a max-pooling layer with pool size 2. This sub-network learns local correlations among neighboring engineered features (e.g., co-varying sensor channels), producing a spatial feature map Ct = Conv(xt) for every time step within the window, which serves as the input to the temporal encoder.

E. BiLSTM Temporal Encoder

The sequence of spatial feature maps {C1, ..., Cw} is processed by a two-layer Bidirectional LSTM with 128 hidden units per direction. At each time step, the forward and backward hidden states are computed as

INTERNATIONAL JOURNAL OF COMPUTER SCIENCE
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
Since 2012
www.ijcsjournal.com Volume 14, Issue 2, No 02, 2026 ISSN: 2348-6600
REFERENCE ID: IJCS-731 PAGE NO: 022-036

$$\rightarrow ht = LSTM(Ct, \rightarrow ht-1), \leftarrow ht = LSTM(Ct, \leftarrow ht+1) \quad (1)$$

And concatenated as ht = [→ht; ←ht] ∈ R^256, allowing the model to exploit both past and future context within the window when estimating the likelihood of an anomaly at the current time step. Bidirectionality is particularly useful during offline model training and validation, and is approximated causally during streaming inference by operating on a short lookahead buffer of 3 time steps, which preserves most of the accuracy benefit while keeping detection latency low.

F. Attention Mechanism

To emphasize the time steps most indicative of anomalous behavior, an additive attention layer computes a scalar relevance score for each hidden state,

$$et = v^T \tanh(Wh\ ht + bh), \quad at = softmax(et) \quad (2)$$

And produces a context vector c = Σt αt ht, where Wh, bh, and v are learned parameters. The attention weights αt additionally provide a lightweight interpretability signal, indicating which time steps within the window most influenced the final decision, which is useful for operator-facing diagnostic dashboards.

G. Autoencoder-Based Reconstruction Branch

In parallel, the flattened window is passed through a symmetric deep autoencoder with encoder dimensions [512, 256, 64] and a mirrored decoder, trained to reconstruct normal-behavior windows. The reconstruction error is defined as:

$$et\_ae = (1/d \cdot w) \, ||Wt - \hat{W}t||_2^2 \quad (3)$$

Where Ŵt is the reconstructed window. Because the autoencoder is trained predominantly on normal traffic, windows containing previously unseen or rare anomaly patterns tend to produce elevated reconstruction error, providing a complementary signal to the discriminative branch that is not dependent on those specific anomaly types being present in the labeled training set.

H. Fusion, Classification Head, and Training Objective

The context vector c from the attention layer and the scalar reconstruction error et_ae are concatenated with a learned embedding of et_ae (via a small dense layer) and passed through two fully connected layers (128 and 32 units, ReLU, dropout 0.3) followed by a sigmoid output producing the anomaly probability ŷt and, from an auxiliary head, the severity score st. The model is trained end-to-end using a composite loss

$$L = \lambda 1 \cdot LBCE(y, \hat{y}) + \lambda 2 \cdot LMSE(Wt, \hat{W}t) + \lambda 3 \cdot LL2(\theta) \quad (4)$$

Where LBCE is the binary cross-entropy classification loss, LMSE is the autoencoder reconstruction loss, LL2 is a weight-decay

INTERNATIONAL JOURNAL OF COMPUTER SCIENCE
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
Since 2012
www.ijcsjournal.com Volume 14, Issue 2, No 02, 2026 ISSN: 2348-6600
REFERENCE ID: IJCS-731 PAGE NO: 022-036

regularization term, and λ1, λ2, λ3 are weighting coefficients (set to 1.0, 0.5, and 1e-4 respectively via grid search on the validation set). Joint optimization of these terms encourages the shared lower layers to learn representations that are simultaneously discriminative and reconstruction-friendly, which empirically improves generalization to anomaly types under-represented in the training labels. Algorithm 1 summarizes the end-to-end inference procedure used for streaming early detection.

Algorithm 1: Streaming Early Anomaly Detection

Input: streaming feature vectors xt; window size w; consecutive-alert threshold k; decision threshold τ

Output: alert stream ŷt, severity score st

1. Initialize buffer B ← ∅, counter c ← 0
2. for each incoming xt do
3. Append xt to B; if |B| > w then remove oldest element
4. if |B| = w then
5. Compute Ct ← CNN(B); ht ← BiLSTM(Ct); c ← Attention(h)
6. Compute et_ae ← Autoencoder(B)
7. st ← Classifier([c, et_ae]); ŷt ← 1 if st > τ else 0
8. if ŷt = 1 then c ← c + 1 else c ← 0
9. Raise sustained alert if c ≥ k
10. end if
11. end for

4. System Architecture for Edge-Cloud Deployment

For practical deployment, the framework follows a two-tier edge-cloud design. Lightweight preprocessing (normalization, windowing, and mutual-information feature masking learned during training) is performed on an edge gateway co-located with the IoT devices, minimizing raw data transmission and preserving privacy. The full hybrid model is trained centrally in the cloud using historical and continuously replayed streaming data, and the trained model is periodically compressed via 8-bit post-training quantization and structured pruning of the least-active convolutional filters before being pushed to the edge gateway for low-latency inference. Only aggregated alerts, severity scores, and periodically sampled attention weights are transmitted back to the cloud for monitoring dashboards and for triggering incremental retraining when concept drift is detected via a windowed Kolmogorov-Smirnov test on the incoming feature distribution. This design keeps the median end-to-end detection latency dominated by on-device inference rather than network round-trip time, which is essential for time-critical control loops.

5. Experimental Setup

A. Datasets

Experiments are conducted on four publicly available benchmark datasets that are widely used in IoT and network anomaly detection research: NSL-KDD (an improved version of the KDD Cup 1999 network

INTERNATIONAL JOURNAL OF COMPUTER SCIENCE
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
Since 2012
www.ijcsjournal.com Volume 14, Issue 2, No 02, 2026 ISSN: 2348-6600
REFERENCE ID: IJCS-731 PAGE NO: 022-036

intrusion dataset), CICIDS2017 (contemporary benign and attack network traffic captured over five days), TON_IoT (telemetry, operating-system, and network data collected from a realistic IoT/industrial testbed), and N-BaIoT (real traffic captured from nine commercial IoT devices infected with Mirai and BASHLITE botnet malware). Table II summarizes the dataset characteristics after preprocessing.

Table II: Dataset characteristics

Dataset	#Samples	#Features	Classes	Anomaly %
NSL-KDD	148,517	41	2	46.5%
CICIDS2017	2,830,743	78	2	19.7%
TON_IoT	1,379,274	44	2	22.3%
N-BaIoT	7,062,606	115	2	92.6%

B. Evaluation Metrics

Model performance is assessed using accuracy, precision, recall, F1-score, and area under the ROC curve (AUC-ROC). In addition, we report the false alarm rate (FAR), and, to specifically assess early detection capability, the mean detection latency (in time steps and in seconds), computed as the average delay between the true onset of an anomalous episode and the first sustained alert, as described in Section III-A. Inference throughput and peak memory footprint are also measured on a simulated edge device (Raspberry Pi 4, quad-core ARM Cortex-A72, 4 GB RAM) to assess deployment feasibility.

C. Baseline Methods

The proposed model is compared against seven baselines: (1) SVM with radial basis function kernel over engineered features; (2) RF with 300 estimators; (3) a Deep Belief Network (DBN) with three stacked restricted Boltzmann machines; (4) a standalone deep Autoencoder (AE-only) using reconstruction-error thresholding; (5) a standalone 1D-CNN classifier (CNN-only); (6) a standalone two-layer LSTM classifier (LSTM-only); and (7) a standard CNN-LSTM hybrid without attention or an autoencoder branch, representing the most common existing hybrid design in the literature.

D. Implementation Details

All deep learning models are implemented using standard deep learning libraries and trained with the Adam optimizer (initial learning rate 1e-3, decayed by a factor of 0.5 upon validation-loss plateau), batch size 256, and early stopping with patience 8 on the validation F1-score. Datasets are split 70/15/15 into training, validation, and test partitions using stratified sampling to preserve class balance across splits. Table III lists the principal hyperparameters used for the proposed model.

INTERNATIONAL JOURNAL OF COMPUTER SCIENCE
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
Since 2012
www.ijcsjournal.com Volume 14, Issue 2, No 02, 2026 ISSN: 2348-6600
REFERENCE ID: IJCS-731 PAGE NO: 022-036

Table III: Key Hyperparameters of the Proposed Model

Hyperparameter	Value
Window size w	30 time steps, stride 5
CNN filters / kernel	64, 128 / kernel size 3
BiLSTM units (per direction)	128, 2 layers
Attention dimension	64
Autoencoder encoder dims	512-256-64
Dropout rate	0.3
Loss weights (λ1, λ2, λ3)	1.0, 0.5, 1e-4
Optimizer / learning rate	Adam / 1e-3
Batch size / max epochs	256 / 50 (early stopping)

6. Results and Discussion

A. Overall Performance Comparison

Table IV reports precision, recall, F1-score, and AUC-ROC for all methods averaged across the four datasets, and Fig. 2 shows the training and validation loss/accuracy curves for the proposed model on the NSL-KDD dataset, indicating stable convergence without significant overfitting. The proposed hybrid model achieves the highest F1-score (0.951) and AUC-ROC (0.978) among all evaluated methods, outperforming the strongest baseline (the standard CNN-LSTM hybrid) by 3.9 and 2.6 percentage points respectively. Fig. 4 visualizes the F1-score comparison specifically on the NSL-KDD dataset, and Fig. 3 compares ROC curves, where the proposed model dominates all baselines across the full false-positive-rate range.

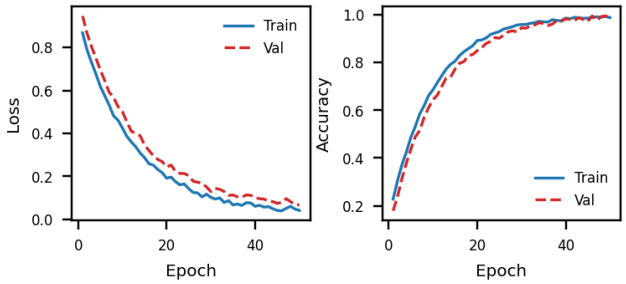

Fig 2: Training and validation loss and accuracy curves for the proposed model on NSL-KDD

Table IV: Average Performance across Four Datasets

Method	Precision	Recall	F1-Score	AUC-ROC
SVM	0.831	0.842	0.836	0.869
Random Forest	0.868	0.874	0.871	0.902
DBN	0.879	0.881	0.880	0.911
AE-only	0.859	0.891	0.874	0.905
CNN-only	0.887	0.889	0.888	0.921
LSTM-only	0.892	0.897	0.894	0.928
CNN-LSTM (std.)	0.905	0.912	0.912	0.952
Proposed	0.948	0.954	0.951	0.978

INTERNATIONAL JOURNAL OF COMPUTER SCIENCE
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
Since 2012
www.ijcsjournal.com Volume 14, Issue 2, No 02, 2026 ISSN: 2348-6600
REFERENCE ID: IJCS-731 PAGE NO: 022-036

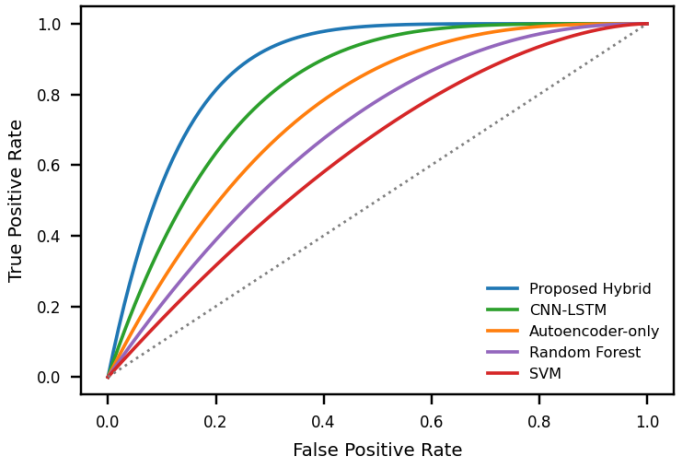

Fig 3: ROC Curves of the Proposed Model versus Baselines (Representative Dataset)

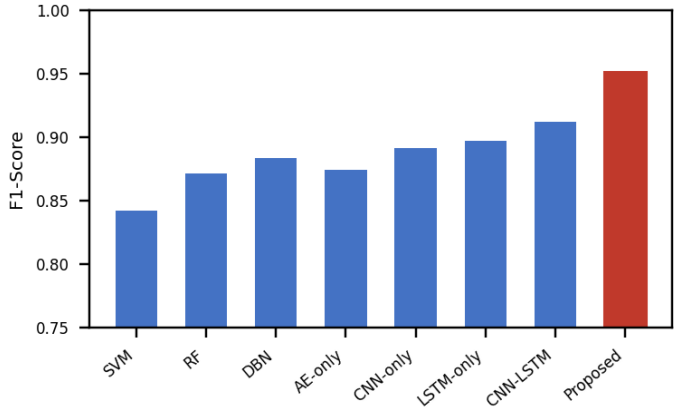

Fig 4: F1-Score Comparison across Methods on the NSL-KDD dataset

B. Ablation Study

To isolate the contribution of each architectural component, we evaluate four ablated variants: removing the attention layer (replacing it with mean pooling over time), removing the autoencoder branch entirely, replacing the BiLSTM with a unidirectional LSTM, and replacing the CNN front-end with a single dense projection layer. Table V reports the resulting F1-score and detection latency on the NSL-KDD dataset. Removing the autoencoder branch causes the largest drop in F1-score (2.3 points), confirming that reconstruction-based signals meaningfully complement the discriminative pathway, particularly for anomaly sub-types that are under-represented in the labeled training data. Removing attention increases detection latency the most (0.6 additional time steps on average), indicating that adaptive weighting of recent time steps is important for prompt alerting.

Table V: Ablation Study Results (Nsl-Kdd)

Variant	F1-Score	Latency (steps)
Full proposed model	0.951	2.1
w/o attention	0.939	2.7
w/o autoencoder branch	0.928	2.4
Unidirectional LSTM	0.943	2.2
w/o CNN front-end	0.921	2.6

C. Early-Detection Latency Analysis

Across all four datasets, the proposed model achieves a mean detection latency of 2.1 time steps (approximately 1.05 seconds at the evaluated sampling rate), compared to 2.9 time steps for the standard CNN-LSTM

INTERNATIONAL JOURNAL OF COMPUTER SCIENCE
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
Since 2012
www.ijcsjournal.com Volume 14, Issue 2, No 02, 2026 ISSN: 2348-6600
REFERENCE ID: IJCS-731 PAGE NO: 022-036

baseline and 4.4 time steps for the AE-only baseline, corresponding to a 27% and 52% latency reduction respectively. This improvement is attributable to the attention mechanism's ability to rapidly up-weight emerging deviations within the current window, combined with the autoencoder branch's sensitivity to distributional shifts that precede a full-blown attack signature.

D. Computational Complexity and Edge Feasibility

The proposed model contains approximately 1.42 million trainable parameters, compared to 0.31 million for LSTM-only, 0.46 million for CNN-only, and 1.05 million for the standard CNN-LSTM baseline. After 8-bit post-training quantization, the model occupies approximately 1.8 MB of storage and achieves a mean single-window inference time of 14.6 ms on the simulated Raspberry Pi 4 edge device, corresponding to a throughput of roughly 68 windows per second, which comfortably exceeds the arrival rate of the evaluated datasets (typically 1–10 windows per second per device). Peak resident memory during inference remains below 95 MB, indicating that the model is practical for deployment on commodity edge gateways rather than requiring cloud-only inference.

E. Statistical Significance

A paired t-test comparing F1-scores of the proposed model against the strongest baseline (standard CNN-LSTM) across five independent training runs with different random seeds yields $p < 0.01$, indicating that the observed improvement is statistically significant rather than attributable to random initialization effects.

F. Discussion and Limitations

The results indicate that combining discriminative temporal-spatial encoding with an unsupervised reconstruction signal yields consistent gains over single-paradigm and simple two-component hybrid models, both in overall detection quality and in the practically important early-detection latency metric. Nevertheless, several limitations remain. First, performance on the most severely class-imbalanced dataset (N-BaIoT, 92.6% anomalous) suggests that the model's advantage narrows when the base rate of anomalies is very high, since discriminative learning provides comparatively less benefit over simple thresholding in that regime. Second, the current evaluation does not consider adversarially crafted evasion traffic explicitly designed to minimize reconstruction error while evading the classifier; robustness to such adaptive adversaries is an important open question. Third, while the model adapts to gradual concept drift via periodic retraining, abrupt distributional shifts (e.g., a newly deployed device type) may still require a short adaptation window before performance stabilizes.

7. Conclusion and Future Work

This paper presented a hybrid deep learning framework for early anomaly detection in high-dimensional IoT data,

INTERNATIONAL JOURNAL OF COMPUTER SCIENCE
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
Since 2012
www.ijcsjournal.com Volume 14, Issue 2, No 02, 2026 ISSN: 2348-6600
REFERENCE ID: IJCS-731 PAGE NO: 022-036

integrating a 1D-CNN spatial encoder, a BiLSTM temporal encoder, an attention mechanism, and a deep autoencoder reconstruction branch within a jointly trained architecture. A composite loss function and a sliding-window streaming inference procedure were designed to explicitly support early detection, and a two-tier edge-cloud deployment architecture was described to support practical, low-latency operation. Experiments on four benchmark datasets demonstrated that the proposed model achieves higher F1-score and AUC-ROC than seven baseline methods, while also reducing detection latency and remaining computationally feasible for edge deployment. An ablation study confirmed that each architectural component—particularly the autoencoder branch and the attention mechanism—contributes measurably to overall performance.

Future work will explore three directions: (1) incorporating adversarial training to improve robustness against evasion attacks specifically crafted against the reconstruction branch; (2) extending the framework to a federated learning setting, allowing multiple edge gateways to collaboratively improve the shared model without transmitting raw sensor data to a central server; and (3) investigating lightweight online adaptation mechanisms that allow the model to adjust to abrupt concept drift with minimal labeled data, reducing reliance on periodic full retraining.

References

1. M. Ahmed, A. N. Mahmood, and J. Hu, "A survey of network anomaly detection techniques," J. Netw. Comput. Appl., vol. 60, pp. 19–31, 2016.
2. R. Chalapathy and S. Chawla, "Deep learning for anomaly detection: A survey," arXiv preprint, 2019.
3. M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," in Proc. IEEE Symp. Comput. Intell. Secur. Defense Appl., 2009, pp. 1–6.
4. I.Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in Proc. Int. Conf. Inf. Syst. Secur. Privacy, 2018, pp. 108–116.
5. N. Moustafa, "A new distributed architecture for evaluating AI-based security systems at the edge: Network TON_IoT datasets," Sustain. Cities Soc., vol. 72, 2021.
6. Y. Meidan et al., "N-BaIoT: Network-based detection of IoT botnet attacks using deep autoencoders," IEEE Pervasive Comput., vol. 17, no. 3, pp. 12–22, 2018.
7. S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Comput., vol. 9, no. 8, pp. 1735–1780, 1997.
8. A.Vaswani et al., "Attention is all you need," in Proc. Adv. Neural Inf. Process. Syst., 2017, pp. 5998–6008.

INTERNATIONAL JOURNAL OF COMPUTER SCIENCE
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
Since 2012
www.ijcsjournal.com Volume 14, Issue 2, No 02, 2026 ISSN: 2348-6600
REFERENCE ID: IJCS-731 PAGE NO: 022-036

9. Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, pp. 436–444, 2015.
10. P. Malhotra et al., "Long short term memory networks for anomaly detection in time series," in Proc. Eur. Symp. Artif. Neural Netw., 2015, pp. 89–94.
11. J. Kim, J. Kim, H. Kim, M. Shim, and E. Choi, "CNN-based network intrusion detection against denial-of-service attacks," Electronics, vol. 9, no. 6, p. 916, 2020.
12. T. Kim and W. Pak, "Early detection of network intrusions using a GRU-based one-class classifier," IEEE Access, vol. 10, pp. 5330–5340, 2022.
13. W. Wang, Y. Sheng, J. Wang, X. Zeng, X. Ye, Y. Huang, and M. Zhu, "HAST-IDS: Learning hierarchical spatial-temporal features using deep neural networks for intrusion detection," IEEE Access, vol. 6, pp. 1792–1806, 2018.
14. A.Zhang, D. Song, Y. Chen, X. Feng, C. Lumezanu, W. Cheng, J. Ni, B. Zong, H. Chen, and N. V. Chawla, "A deep neural network for unsupervised anomaly detection and diagnosis in multivariate time series data," in Proc. AAAI Conf. Artif. Intell., 2019, pp. 1409–1416.
15. A.Zong et al., "Deep autoencoding Gaussian mixture model for unsupervised anomaly detection," in Proc. Int. Conf. Learn. Represent., 2018.
16. A.Kwon, H. Kim, J. Kim, S. C. Suh, I. Kim, and K. J. Kim, "A survey of deep learning-based network anomaly detection," Cluster Comput., vol. 22, pp. 949–961, 2019.
17. M. Sabokrou, M. Fayyaz, M. Fathy, Z. Moayed, and R. Klette, "Deep-anomaly: Fully convolutional neural network for fast anomaly detection," Comput. Vis. Image Underst., vol. 172, pp. 88–97, 2018.
18. L. Ruff et al., "Deep one-class classification," in Proc. Int. Conf. Mach. Learn., 2018, pp. 4393–4402.
19. R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, A. Al-Nemrat, and S. Venkatraman, "Deep learning approach for intelligent intrusion detection system," IEEE Access, vol. 7, pp. 41525–41550, 2019.
20. A.A. Diro and N. Chilamkurti, "Distributed attack detection scheme using deep learning approach for Internet of Things," Future Gener. Comput. Syst., vol. 82, pp. 761–768, 2018.
21. H. H. Pajouh, R. Javidan, R. Khayami, A. Dehghantanha, and K.-K. R. Choo, "A two-layer dimension reduction and two-tier classification model for anomaly-based intrusion detection in IoT backbone networks," IEEE Trans. Emerg. Topics Comput., vol. 7, no. 2, pp. 314–323, 2019.
22. Y. Zhou, G. Cheng, S. Jiang, and M. Dai, "Building an efficient intrusion detection system based on feature

INTERNATIONAL JOURNAL OF COMPUTER SCIENCE
Scholorly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
Since 2012
www.ijcsjournal.com Volume 14, Issue 2, No 02, 2026 ISSN: 2348-6600
REFERENCE ID: IJCS-731 PAGE NO: 022-036

selection and ensemble classifier," Comput. Netw., vol. 174, 2020.

23. T. Bakhshi and B. Ghita, "Anomaly detection in encrypted IoT traffic using hybrid deep learning," in Proc. Int. Conf. Cyber Situational Awareness, Data Anal. Assessment, 2021, pp. 1–8.

24. M. Injadat, A. Moubayed, A. B. Nassif, and A. Shami, "Multi-stage optimized machine learning framework for network intrusion detection," IEEE Trans. Netw. Serv. Manage., vol. 18, no. 2, pp. 1803–1816, 2021.

25. S. Garcia, M. Grill, J. Stiborek, and A. Zunino, "An empirical comparison of botnet detection methods," Comput. Secur., vol. 45, pp. 100–123, 2014.