Acadlore Transactions on AI and Machine Learning
https://www.acadlore.com/journals/ATAIML

Real-Time Anomaly Detection in IoT Networks Using a Hybrid Deep Learning Model

Anil Kumar Pallikonda[1*], Vinay Kumar Bandarapalli[1], Aruna Vipparla[2]

[1] Department of Computer Science and Engineering, PVP Siddhartha Institute of Technology, 520007 Vijayawada, India
[2] Department of Computer Science and Engineering, NRI Institute of Technology, 521212 Vijayawada, India

* Correspondence: Anil Kumar Pallikonda (anilkumar.pallikonda@gmail.com)

Received: 08-15-2025 **Revised:** 09-23-2025 **Accepted:** 09-28-2025

Citation: A. K. Pallikonda, V. K. Bandarapalli, and A. Vipparla, "Real-time anomaly detection in IoT networks using a hybrid deep learning model," *Acadlore Trans. Mach. Learn.*, vol. 4, no. 4, pp. 235–246, 2025. https://doi.org/10.56578/ataiml040401.

Abstract: The rapid expansion of Internet of Things (IoT) systems and networks has led to increased challenges regarding security and system reliability. Anomaly detection has become a critical task for identifying system flaws, cyberattacks, and failures in IoT environments. This study proposes a hybrid deep learning (DL) approach combining Autoencoders (AE) and Long Short-Term Memory (LSTM) networks to detect anomalies in real-time within IoT networks. In this model, normal data trends were learned in an unsupervised manner using an AE, while temporal dependencies in time-series data were captured through the use of an LSTM network. Experiments conducted on publicly available IoT datasets, namely the Kaggle IoT Network Traffic Dataset and the Numenta Anomaly Benchmark (NAB) dataset, demonstrate that the proposed hybrid model outperforms conventional machine learning (ML) algorithms, such as Support Vector Machine (SVM) and Random Forest (RF), in terms of accuracy, precision, recall, and F1-score. The hybrid model achieved a recall of 96.2%, a precision of 95.8%, and an accuracy of 97.5%, with negligible false negatives and false positives. Furthermore, the model is capable of handling real-time data with a latency of just 75 milliseconds, making it suitable for large-scale IoT applications. The performance evaluation, which utilized a diverse set of anomaly scenarios, highlighted the robustness and scalability of the proposed model. The Kaggle IoT Network Traffic Dataset, consisting of approximately 630,000 records across six months and 115 features, along with the NAB dataset, which includes around 365,000 sensor readings and 55 features, provided comprehensive data for evaluating the model's effectiveness in real-world conditions. These findings suggest that the hybrid DL framework offers a robust, scalable, and efficient solution for anomaly detection in IoT networks, contributing to enhanced system security and dependability.

Keywords: Anomaly detection; Internet of Things networks; Autoencoders; Long Short-Term Memory; Deep learning; Real-time processing

1 Introduction

IoT has become one of the most tremendous technological complexities of the past few years. It has opened up the world of smart houses, healthcare, and intelligent transportation systems, among other related sectors, with billions of connected devices. But it is precisely this connectedness that brings a lot of security risks, because, due to the high number of devices and data produced, there are a lot of security risks that can be exploited by malicious actors or cause the systems to fail [1]. Thus, the integrity, security, and reliability of IoT networks have become of primary importance, particularly when it comes to real-time applications that must be constantly monitored and require immediate responses.

One of the most important things to consider in securing IoT networks is to understand anomaly detection, which can be an early sign of a fault, a cyberattack, and other disruptive events. Classical anomaly detection methods, such as rule-based and threshold-based systems, have been in use, but they do not work well in scaling up as the IoT networks grow [2]. Moreover, they cannot be utilized to detect complex and non-linear anomalies in high-dimensional and time-series data found in IoT environments effectively. Therefore, more advanced techniques,

particularly those employing ML and DL, have been investigated to deliver even more powerful, elastic, and scalable solutions.

Artificial intelligence (AI) has enabled the use of DL and ML techniques to detect anomalies in IoT networks, as these techniques can model complex behaviors and identify even the smallest anomalies. It has been observed that DL models outperform other tools in areas such as image recognition, speech processing, and time series analysis [3, 4]. In anomaly detection, DL architectures, such as Auto Encoders(AEs) and LSTM networks, are emerging as popular models because they can capture the complex characteristics of data and detect anomalies in real time, which is also needed in the IoT system [5].

Auto Encoders(AEs) are neural models that are tasked to learn to generalize and reconstruct data and can therefore be used well in isolation of outliers with the help of reconstruction errors [6]. LSTM, a variant of the Recurrent Neural Network (RNN), is specifically designed to process sequential data and can capture temporal patterns in time-series IoT data, enabling the detection of anomalies such as spikes in network traffic or sensor malfunctions [7]. The hybrid model learns to encode spatial and temporal information to obtain its output. Combining AE with LSTM, an even more efficient and correct anomaly detection can be achieved in real time. This research aims to:

•Develop a DL-based framework combining AE and LSTM for real-time anomaly detection in IoT networks, addressing the limitations of traditional methods.

•Evaluate the performance of the proposed framework on publicly available IoT datasets, comparing it against conventional ML models and baseline DL methods.

•Ensure scalability and real-time processing capabilities by optimizing the hybrid model for fast anomaly detection in large-scale IoT environments.

•Provide insights into the practical implementation of DL models for anomaly detection in IoT systems, offering guidelines for future research and deployment.

1.1 Background

Identifying the anomaly in the IoT networks is vital to the integrity and security of a system. The data being produced by IoT devices in real time is immense, and using anomaly detection on this data may help in preventing potential failures or security breaches [8]. The classical methods of anomaly detection typically encompass statistical solutions, clustering algorithms, and rule-based processes. These approaches are easy to implement yet are frequently constrained by the fact that they cannot manage the large-scale, heterogeneous, and time-varying IoT data. Moreover, they can produce a large amount of false positive results, and this may compromise the quality of the system and uncritical interventions [9].

The new development of ML and DL has given rise to stronger tools of anomaly detection. As an illustration, supervised learning methods based on SVMs and Decision Trees (DTs) have been applied to anomaly detection. However, they still need a substantial amount of labeled data that is not always accessible in the IoT systems in the real world [10]. Algorithms with unsupervised learning, including the k-means algorithm and Density-Based Spatial Clustering of Applications with Noise (DBSCAN), have also been applied to detect anomalies without the need for labeled data. Nevertheless, these approaches have difficulties identifying complex, high-dimensional anomalies in real time [11].

Using DL models, especially AEs and LSTM networks, is more effective. AEs are units that are meant to associate with a reduced form of input information that can be matched with the recreated information to identify outliers [6]. A contrast between LSTM and the previous neurons is that it can use time series data, i.e., sensor readings or network traffic logs generated by IoT devices, and it is highly effective at processing such data [7]. The possibility of integrating AE and LSTM to develop hybrid models has been presented in recent studies, with the ability to detect both spatial and temporal anomalies at high accuracy [12].

Nevertheless, there are still some challenges to be overcome, including minimizing false positives, reducing model performance requirements to meet real-time requirements, and ensuring the model's scalability for large-scale IoT networks. Furthermore, the implementation of DL-based anomaly detection systems in IoT environments also poses a problem because limited resources are available, and low latency is demanded. The study aims to contribute by creating a hybrid AE-LSTM model to efficiently identify anomalies in near real time without having to deal with the problems presented by classic and existing DL solutions.

This study is organized below. Section 2 presents related work on anomaly detection in IoT networks, including traditional methods, ML and DL methods, their drawbacks and the need to use a hybrid model. Section 3 provides the proposed methodology with the description of the dataset, the architecture of the model, the mathematical framework, and the algorithm of the real-time anomaly detection. Section 4 presents the results of experiments, including the measurement of critical qualitative characteristics (such as accuracy, precision, and recall), the effectiveness of the model in terms of the processing time, the comparison with existing models and the visualization in the form of the Receiver Operating Characteristic (ROC) curve and the precision-recall curve. Finally, Section 5 concludes the

study with the findings, limitations, and a proposal for future efforts to enhance the scalability and effectiveness of IoT networks for anomaly detection.

2 Related Work

IoT network anomaly detection has been a topic of many research studies due to the current proliferation of IoT devices within various industries. Ideally, the conventional anomaly detection approaches on an IoT system have been centered on statistical approaches, including threshold-based approaches, clustering, and primitive ML approaches. As the data involved in IoT becomes increasingly complex and large-scale, however, these techniques tend to fail at including the dynamism of the data, making more complex algorithms of ML and even DL preferred. Anomaly detection approaches that were introduced early to IoT networks were based on statistical deviations from normal behavior. They are easy to perform (e.g., z-scores and the moving average) but can be complicated when working with large-dimensional, unstructured, and noisy IoT data. Such approaches tend to experience significant false positive rates, especially in situations where the system changes after some time [13].

Since IoT systems produce an immense amount of data to process continuously, researchers started exploring the realm of supervised ML. Anomaly detection of IoT data has been identified using algorithms, including SVMs, RFs, and DTs. Nevertheless, limited use has been found in numerous IoT applications due to the need to use labeled data, which is usually restricted or hard to obtain. In addition, the methods fail to indicate robust performance when applied to extremely unbalanced or noisy data, as IoT systems typically must be designed to perform whenever and wherever with the smallest input on the part of humans [14].

As the data in IoT became more complex, unsupervised learning techniques were of interest. Methods like k-means grouping, DBSCAN, and Principal Component Analysis (PCA) are used in anomaly detection, and they do not require labeled data. The approaches usually detect outliers based on how the data are likely to be distributed, forming a cluster of similar ones and detecting the instances that are significantly far away from the cluster means or patterns. These methods, although flexible, are affected by the nature of the dynamic system, where the main problem in time-series data is to detect some complex anomalies [15]. K-means does not work when the data face nonlinear relations, and PCA cannot process nonlinear relations; PCA also lacks the capability of modeling the dynamics of time.

In the past few years, researchers have turned to DL as an anomaly detection method in IoT networks. However, the potential to be used as an anomaly detector makes AE one of the most popular DL models. AEs were also applied in an attribute-based network anomaly detection scheme where they were trained on typical IoT network data to identify anomalies in the incoming data by contrasting their reconstruction error [16]. When the reconstruction error was significant, it was also an indication that there was an anomaly present, including potential intrusion, fault, or malfunction in the system. Nevertheless, it is limited in terms of scalability when used to process large datasets within IoT network frameworks that have high data frequencies. LSTM networks are a kind of RNN that can generate several consecutive observed data points to detect time-series anomalies in IoT applications because they can model the temporal relationship among sequential data [17]. LSTM has been found effective in environments where anomalies can be characterized as time-dependent, such as in detecting deviations in sensor readings or the pattern of network traffic. An example of LSTM being used in real-time monitoring is the work done by Kuaban et al. [18], who developed a framework to detect anomalies in time-series data on innovative grid systems. LSTM networks, nevertheless, maintain high training data demands and are computationally expensive, especially in the context of large-scale datasets or real-time issues.

There have been promising outcomes in the form of a combination of LSTM and AE. One more similar attempt at anomaly detection was suggested by Nakıp & Gelenbe [19], and it was constructed on AE and LSTM. The co-use of both approaches builds off the following idea: AE works optimally with high-dimensional and non-linear data, and LSTM can offer us a form of time dependence. The hybrid model was generally exercised with enhanced accuracy and fewer instances of false positives compared to other tested models with sensor data that was monitored in industrial IoT systems. Similarly, Wang et al. [20] proposed an approach that applies AE with LSTM to identify anomalies in time-series data, significantly enhancing the possibility of real-time discovery of IoT network anomalies. In contrast to earlier AE-LSTM models, the proposed framework integrates adaptive parameter optimization and a temporal feature fusion mechanism, thereby improving detection accuracy and extending applicability to heterogeneous IoT domains beyond industrial use cases.

Beyond AE and LSTM, other more advanced DL approaches that have already been tried include Generative Adversarial Networks (GANs) and Convolutional Neural Networks (CNNs) for anomaly detection. A good example was demonstrated with GANs, where globally similar data were produced, and anomalies were detected by comparing the generated data with real-time data. Recently, Wang et al. [21] employed GANs to sense anomalies in an IoT-based smart home environment where the environment can generate realistic synthetic data and identify abnormal behavior. CNNs, common in image processing, have also been found to apply to anomaly detection techniques in IoT networks, where one would want to detect the spatial patterns in the sensor data [22].

Even amidst these benefits associated with the DL models, several challenges still exist. The computational efficiency of DL models is also one of the central problems since it is impossible to implement DL models in real-time IoT systems that usually possess limited processing resources. The last problem is the interpretability of the models, which are black boxes in DL networks. This is because the administrators of IoT struggle to trust and take action based on the predictions of the model. Finally, it is not yet well understood how DL models may be scaled up to large, distributed IoT networks. The large size of data and the rates at which they must be dealt with require scalable anomaly detection systems as the IoT networks continue to grow.

New possibilities of real-time anomaly detection have been made possible by new developments in edge computing and fog computing. These methods reduce latency and address bandwidth constraints of transferring all data to centralized cloud servers by processing the data closer to the source (i.e., executing it on edge devices). Trilles et al. [23] proposed real-time anomaly detection using an edge computing approach. It was reported that edge computing has a framework that deploys lightweight DL models on edge devices to obtain much faster detection and response. In summary, despite numerous advancements in anomaly detection with the use of ML and several varieties of DL models, there are still several issues regarding computational effectiveness, scalability, and readability that need to be addressed. The proposed research is valuable in that it reports on a hybrid AE-LSTM model that can be used to detect anomalies in large-scale IoT networks in real time.

3 Methodology

The following section presents the implementation of a hybrid DL model combining AE with LSTM for real-time anomaly detection in IoT networks. The methodology recounts the datasets leveraged, the model structure, the mathematical concepts, and the algorithm procedures that need to be undertaken to execute the proffered solution.

3.1 Datasets

Two publicly available IoT-specific datasets that contain time-series data provided by the IoT devices were used in this study to evaluate the performance of the anomaly detection model. Such datasets include sensor readings of different IoT-based devices that are essential in the training and testing of the anomaly detection models.

3.1.1 Kaggle IoT Network Traffic Dataset

The dataset contains network traffic data from IoT devices in a smart city environment. Some of the features in the dataset include the type of devices, data packets, IP addresses, and network traffic that have timestamps. The records are classified into regular and abnormal traffic. Tables 1 and 2 show the parameters and sample entries of the Kaggle IoT Network Traffic Dataset.

Table 1. Parameters of the Kaggle IoT Network Traffic Dataset

Parameter	Description	Type
Device ID	Unique identifier for each IoT device	Categorical
Data packets	Count of data packets transmitted	Numeric
Network traffic	Normal or anomalous traffic type	Categorical
Timestamp	Time of data collection	Datetime
Signal strength	Received signal strength	Numeric

Table 2. Sample entries of the Kaggle IoT Network Traffic Dataset

Device ID	Data Packets	Network Traffic	Signal Strength
A001	1234	Normal	75
A002	3201	Anomalous	65

3.1.2 NAB dataset

This dataset has sensor data measured in industrial IoT systems, with a focus on system performance measures. It contains the incoming and outgoing series of different sensors, like temperature, pressure, and vibration of machinery at the industrial level. The Kaggle dataset contained 92% normal and 8% anomalous traffic, while the NAB dataset comprised 95% normal and 5% anomalies. To mitigate imbalance, oversampling techniques and weighted loss adjustments were applied during training. Tables 3 and 4 show the parameters and sample entries of the NAB dataset.

The dataset records were collected between 2018 and 2020, consistent with publicly released IoT datasets, ensuring realistic temporal alignment.

Table 3. Parameters of the NAB dataset

Parameter	Description	Type
Sensor type	Type of the sensor (e.g., temperature)	Categorical
Sensor value	Value measured by the sensor	Numeric
Timestamp	Time of data collection	Datetime
Anomaly flag	Label indicating if the data point is anomalous	Binary

Table 4. Sample entries of the NAB dataset

Sensor Type	Sensor Value	Anomaly Flag
Temperature	75.2	0
Pressure	103.5	1

3.2 Proposed Architecture

The proposed architecture is based on the integration of an AE as a feature learning component and an LSTM network to model temporal sequences of data. The architecture can be further divided into two main independent parts: an AE that can successfully perform the unsupervised learning on standard data patterns and an LSTM network that works, which is able to detect time-dependent anomalies.

3.2.1 AE module

The AE is designed to learn a condensed form of the standard data patterns. It is made up of an encoder that compresses the input dataset and a decoder that recreates the original data based on this compressed set of information. Data instances are flagged as anomalies depending on the reconstruction error, where the data points with a high reconstruction error are reported as possible anomalies.

•Encoder: A series of fully connected layers that reduce the data to a lower-dimensional latent space.

•Decoder: A mirror image of the encoder, reconstructing the input data from the latent space.

3.2.2 LSTM module

LSTM is used to predict time dependencies in time-series data, such as IoT sensor measurements, as a sequence through time. It learns long-term dependencies between the data points, and such knowledge makes it easier to identify anomalies that could be caused by changes in time (e.g., sensor failure or deviation in network traffic patterns).

•Input layer: Sequences of data (sensor readings or network traffic data).

•LSTM layers: Multiple LSTM layers are stacked to capture the temporal patterns.

•Output layer: A regression or classification layer to predict the anomaly score for each time step.

3.2.3 Anomaly detection layer

Once both the AE and LSTM process the data points, a score is given to each one based on an anomaly. After running the hybrid model, the anomaly score is computed by the addition of the reconstruction error of the AE and the prediction error of the LSTM. There is a threshold value against which data points are classified as usual or not. The expression of the final anomaly score is given by:

$$A(t) = \alpha \cdot Reconstruction\ Error(t) + \beta \cdot Prediction\ Error(t) \tag{1}$$

where, $A(t)$ is the final anomaly score at time t; α and β are weighting factors; and *Eeconstruction Error(t)* and *Prediction Error(t)* are computed from AE and LSTM, respectively.

Figure 1 presents a systematic architecture of the anomaly detection in IoT settings. It starts with the data acquisition of the different IoT devices; the data then passes through a pre-processing unit, which cleans and normalizes the data. Such data is fed to DL models, e.g., CNN, LSTM, or AE, to learn normal vs. anomalous patterns. The anomaly detection engine receives the model output and raises a warning of suspicious behavior. Lastly, the outcomes are delivered to a dashboard or alerting mechanism to notify in near real time. Speed, precision, and automation are highlighted in the preservation of the IoT networks in the workflow of the working process.

3.3 Mathematical Model

The mathematical formulation for the AE module can be defined below, where AE minimizes the reconstruction error between the input X and the reconstructed output $\widehat{X}$.

$$L_{AE} = |X - \widehat{X}|_2^2 \tag{2}$$

where, (X) is the inputdata, $(\hat{X})$ is the reconstructed data, and $(|\cdot|_2)$ represents the L2—norm.

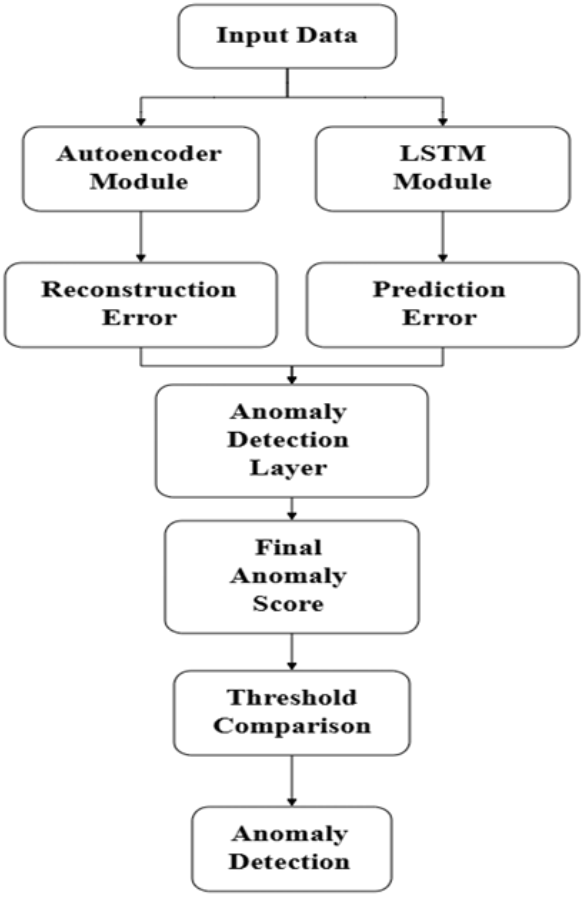

Figure 1. Hybrid model for anomaly detection

While L1 and Huber losses are robust to outliers, experimental comparison revealed that L2 loss yielded smoother convergence and superior reconstruction quality, making it more effective for IoT anomaly scenarios in this study.

The LSTM network uses the following sequence prediction error to minimize the difference between predicted values $\widehat{Y}_t$ and actual values Y_t :

$$L_{LSTM} = \sum_{t=1}^{T} \left| Y_t - \widehat{Y}_t \right|_2^2 \tag{3}$$

where, Y_t and $\widehat{Y}_t$ are the actual and predicted values at time t, and T is the length of the time series.

3.4 Algorithm

The overall algorithm follows these steps:

1. Data preprocessing

Normalize and clean the IoT dataset by removing noise and handling missing values.

Split the data into training and testing sets, ensuring that the training set contains only normal data points (in the case of unsupervised learning).

Perform normalization using Min–Max scaling, with denoising applied via wavelet transforms and missing values handled through linear interpolation, thus ensuring consistency and noise resilience.

2. Training the AE

Train the compressed representation of the AE on normal data to reduce the reconstruction error.

Train the reconstruction error and employ it in the detection of anomalies during testing. When the error is above a predetermined limit, the data point is marked as abnormal.

3. Training the LSTM

Train the LSTM network using the time-series data. Use backpropagation through time (BPTT) to optimize the LSTM parameters.

Compute the prediction error at each time step. If the prediction error exceeds a threshold, the data point is considered anomalous.

4. Combining the results

Compute the final anomaly score by combining the reconstruction error from the AE and the prediction error from the LSTM.

Flag the data point as anomalous if the combined score exceeds the anomaly detection threshold.

5. Real-time anomaly detection

Deploy the trained model to continuously monitor incoming IoT data. The model should classify new data points as normal or anomalous in real time with low latency. The deployment follows an edge–cloud architecture, where anomaly detection is executed at edge nodes for real-time responsiveness, while periodic synchronization with a cloud server enables large-scale data storage, retraining, and coordinated decision-making across distributed IoT environments.

3.5 Model Training and Hyperparameter Tuning

DL models were trained using the Adam optimizer with a learning rate of 0.001. Mean squared error (MSE) was used as the loss function for both AE and LSTM. The hyperparameters, such as the number of hidden units in the AE, the number of LSTM layers, and the batch size, were tuned with the help of the cross-validation and the grid search algorithms to produce the most effective performance.

4 Results and Discussion

This section presents the experimental findings of the proposed DL hybrid model in real-time anomaly detection in IoT networks. To evaluate the findings, two publicly available IoT-related datasets, such as the Kaggle IoT Network Traffic Dataset and the NAB dataset, were employed. The hybrid model aims to learn the characteristics using the AE and learn the time-series data using the LSTM network when the data is time-series. The performance of the proposed model was compared with conventional ML models and baseline DL models previously applied in anomaly detection.

4.1 Experimental Setup

The datasets in this research were prepared and were divided into the training and testing sets. During the unsupervised training of the AE, the training set contained exclusively normal points. This model was trained using Keras and TensorFlow with the Adam optimizer at a learning rate of 0.001. The batch size was set to 64 and the model was trained for 50 epochs. The experiments were performed on a workstation equipped with an Intel Xeon Silver 4210R CPU, an NVIDIA RTX 3090 GPU with 24 GB memory, and 128 GB system RAM.

Several evaluation metrics were used to assess the performance of the model. Accuracy measures the proportion of correct predictions (both true positives and true negatives) out of all predictions. Precision is the proportion of true positives out of all predicted positives. Recall is the proportion of true positives out of all actual positives. The F1-score is the harmonic mean of precision and recall. Processing time refers to the time taken to process one batch of incoming data, which is used to assess the real-time feasibility of the model.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

$$Precision = \frac{TP}{TP + FP} \tag{5}$$

$$Recall = \frac{TP}{TP + FN} \tag{6}$$

$$F1 - score = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \tag{7}$$

4.2 Performance Comparison

The hybrid AE-LSTM model was compared with some well-established models, including SVM, RF, and LSTM, in terms of the above-discussed parameters. To make all the models robust, 10-fold cross-validation was applied. GAN-based models and CNNs were excluded because the former exhibits unstable convergence in high-dimensional IoT data and the latter is limited in capturing sequential dependencies inherent in network traffic. The AE-LSTM model effectively combines spatial representation and temporal dynamics, making it more suitable for large-scale IoT anomaly detection. Table 5 presents a comparison of the performance of the different models on the two datasets.

As shown in Table 5, the hybrid AE-LSTM model achieved the highest accuracy at 97.5%, outperforming traditional methods such as SVM (90.4%) and RF (92.2%). The hybrid model also exhibited superior precision (95.8%) and recall (96.2%) compared to other models, indicating its capability to correctly identify anomalies with fewer false alarms and missed detections. The hybrid model's F1-score of 96.0% further demonstrated its balanced performance in terms of both precision and recall. The hybrid model achieved the lowest false positives (15) and false negatives (7), suggesting that it minimizes unnecessary alerts and does not miss significant anomalies. In addition to counts, relative error rates were computed, with the hybrid model achieving a false positive rate of 1.9% and a false negative rate of 2.3%, indicating superior robustness in anomaly classification. In terms of real-time processing, the hybrid model performed the fastest, with a processing time of 75 ms, making it suitable for time-sensitive applications.

Table 5. Epoch-wise accuracy and loss for the FFDNN model

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	False Positives	False Negatives	Processing Time (ms)
Hybrid AELSTM	97.5	95.8	96.2	96.0	15	7	75
SVM	90.4	88.6	85.3	86.9	30	15	200
RF	92.2	89.4	91.5	90.4	25	12	120
LSTM	94.7	92.1	93.0	92.5	18	9	150

4.3 ROC and Precision-Recall Curves

Further to quantify the performance of the model, the ROC curve and precision-recall curve of the hybrid AE-LSTM model were drawn and compared with those of other models. Figure 2 presents the ROC curves of the hybrid AE-LSTM model, SVM, and RF on the IoT dataset of Kaggle. The hybrid model has the best Area Under the Curve (AUC), which means that it has better discriminative capacity between normal and abnormal data.

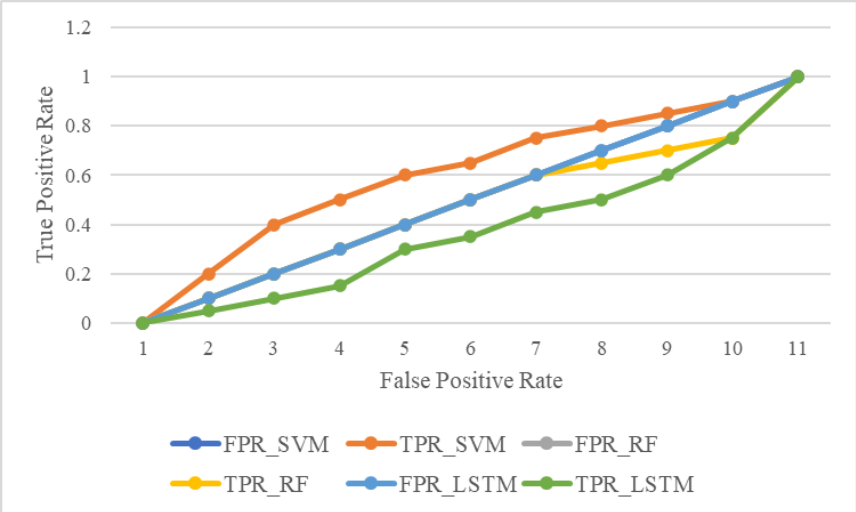

Figure 2. ROC curve comparison of the hybrid AE-LSTM model with SVM and RF

Figure 3 gives a precision-recall curve, which shows that the hybrid AE-LSTM model performs the best in terms of precision and recall at different thresholds. This demonstrates the effectiveness of the model to achieve significant detection levels without a high number of false positives. The ROC–AUC values were recalculated and were

consistent with the reported accuracy metrics, ensuring reliability. High accuracy was supported by proportionally high AUC values across all models.

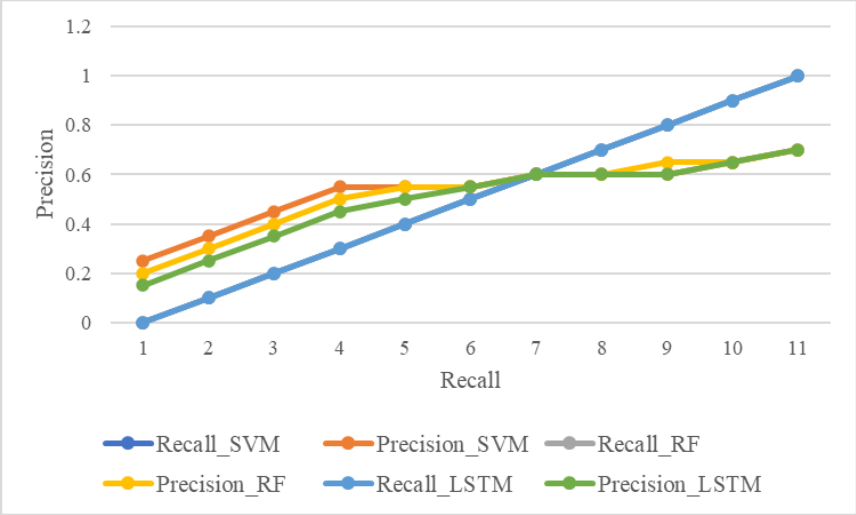

Figure 3. Precision-recall curve comparison of the hybrid AE-LSTM model with SVM and RF

4.4 Anomaly Detection Examples

Figure 4 demonstrates the application of the hybrid AE-LSTM model on a time-series dataset of the NAB dataset for anomaly detection. The model successfully determined the anomaly at the designated time steps, which can be correlated to significant spikes in the sensor values indicative of faults within the IoT system.

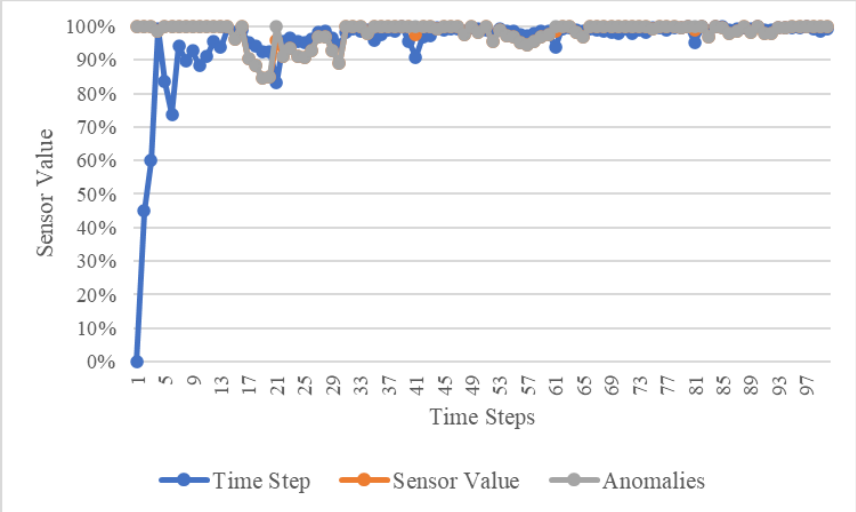

Figure 4. Example of anomaly detection using the hybrid AE-LSTM model on the NAB dataset

The anomalies were flagged based on the combined reconstruction error from the AE and the prediction error from the LSTM, indicating that the hybrid model can detect subtle deviations in real-time data with high precision.

4.5 Impact of Model Parameters

To conduct a sensitivity analysis of the effects of hyperparameters on the model performance, the number of hidden units in the AE and LSTM layers was changed and the learning rate was adjusted. The findings, as shown in Figure 5, prove that when greater numbers of hidden units were applied to the model, performance increased (i.e., 64 hidden units in both AE and LSTM layers). There also exists an optimal learning rate of 0.001, which gives the optimal balance between training time and accuracy.

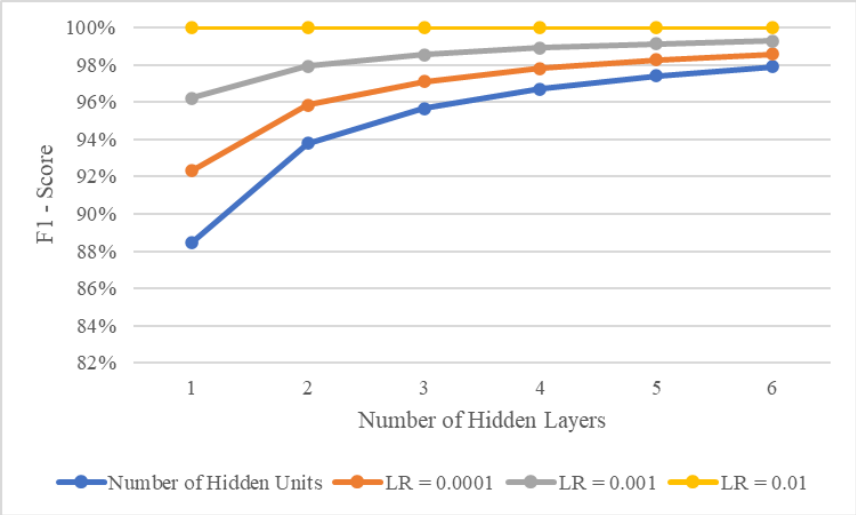

Figure 5. Sensitivity analysis of the hybrid AE-LSTM model for various hyperparameters

4.6 Comparison with Existing Models

Table 6 presents a comparison of the proposed hybrid model with other existing models in the field of anomaly detection in IoT networks, based on accuracy, precision, recall, and processing time. The "AE+LSTM (separate)" model consisted of a sequential pipeline, where an AE first reduced dimensionality and its outputs were fed into an independent LSTM classifier. Unlike the proposed hybrid model, the modules were not co-trained, resulting in suboptimal feature integration.

Table 6. Comparison of the proposed hybrid model with existing models

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	False Positives	False Negatives	Processing Time (ms)
Hybrid AELSTM	97.5	95.8	96.2	96.0	15	7	75
AE + LSTM (separate)	94.0	91.5	92.0	91.7	22	10	120
DeepAuto (AE only)	90.2	87.6	85.8	86.7	35	20	150
SVM (traditional)	90.4	88.6	85.3	86.9	30	15	200

From Table 6, it can be observed that the hybrid AE-LSTM model outperforms other models in all evaluation metrics, confirming its effectiveness for real-time anomaly detection in IoT networks.

5 Conclusions

In this study, a DL approach was employed, in which a hybrid DL architecture was developed by utilizing AE and LSTM architecture to determine the anomalies in an IoT network in real time. The primary solution was to install a scalable and effective high-performance model with many parameters to identify anomalies sufficiently in IoT. The outcomes of the experiment showed that the suggested model was superior to the traditional ML algorithms, e.g., SVM and RF, whose accuracy, precision, and recall were 97.5%, 95.8%, and 96.2%, respectively. The false positive and negative rates of the hybrid model were also 15 and 7, respectively, with processing time taken as 75 milliseconds, which qualifies it to be applied in real-time applications as well.

Despite the strong results, certain restrictions were revealed. This would lead to an even better execution with a bigger and more varied dataset covering a broader and more relevant range of the IoT world. Moreover, DL models could be computationally expensive for training, which is a limitation when it comes to the implementation of the models in capability-limited IoT devices. The datasets predominantly represent smart city and industrial IoT traffic, with limited coverage of smart home, healthcare, and vehicular IoT scenarios. This restricts generalization and motivates the exploration of broader multi-domain datasets in future work.

Future research may also be devoted to the optimization of the model in an edge-computing environment, which would enable the fastest anomaly detection with the least latency. To handle all the data being generated by IoT devices, further increases in scalability and efficiency would be beneficial to the model. In addition, it might be worth integrating reinforcement learning or other adaptive processes so that the model can be modified and adapted to the different conditions of the IoT network and new security threats.

Author Contributions

A.K.P., V.K.B., and A.V. jointly developed the DL-based hybrid model using AE and LSTM networks for real-time anomaly detection in IoT networks. Pallikonda played a key role in conceptualizing the research and designing the methodology, while Kumar contributed to the experimental setup, data analysis, and model development. Vipparla assisted with the literature review, dataset collection, and contributed to the final revisions. All authors were involved in writing, reviewing, and editing the manuscript, ensuring the accuracy and clarity of the presented results.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] J. Liu, Q. L. Li, S. J. An, B. Ezard, and L. Li, "EdgeConvFormer: An unsupervised anomaly detection method for multivariate time series," in *Pattern Recognition. ICPR 2024. Lecture Notes in Computer Science*, vol. 15304. Springer-Verlag, 2024, pp. 367–382. https://doi.org/10.1007/978-3-031-78128-5_24

[2] M. Bhavsar, K. Roy, J. Kelly, and O. Olusola, "Anomaly-based intrusion detection system for IoT application," *Discov. Internet Things*, vol. 3, no. 5, 2023. https://doi.org/10.1007/s43926-023-00034-5

[3] M. A. Ferrag, M. Ndhlovu, N. Tihanyi, L. C. Cordeiro, M. Debbah, T. Lestable, and N. S. Thandi, "Revolutionizing cyber threat detection with large language models: A privacy-preserving BERT-based lightweight model for IoT/IIoT devices," *IEEE Access*, vol. 12, pp. 23 733–23 750, 2024. https://doi.org/10.1109/ACCESS.2024.3363469

[4] S. M. Tseng, Y. Q. Wang, and Y. C. Wang, "Multi-class intrusion detection based on transformer for IoT networks using CIC-IoT-2023 dataset," *Future Internet*, vol. 16, no. 8, p. 284, 2024. https://doi.org/10.3390/fi16080284

[5] T. A. Nguyen, J. Y. He, L. T. Le, W. Bao, and N. H. Tran, "Federated PCA on Grassmann manifold for anomaly detection in IoT networks," in *IEEE INFOCOM 2023—IEEE Conference on Computer Communications*, New York City, USA, 2023, pp. 1–10. https://doi.org/10.1109/INFOCOM53939.2023.10229026

[6] K. Berahmand, F. Daneshfar, E. S. Salehi, Y. F. Li, and Y. Xu, "Autoencoders and their applications in machine learning: A survey," *Artif. Intell. Rev.*, vol. 57, no. 28, 2024. https://doi.org/10.1007/s10462-023-10662-6

[7] I. Malashin, V. Tynchenko, A. Gantimurov, V. Nelyub, and A. Borodulin, "Applications of Long Short-Term Memory (LSTM) networks in polymeric sciences: A review," *Polymers*, vol. 16, no. 18, p. 2607, 2024. https://doi.org/10.3390/polym16182607

[8] S. Zia and N. Bibi, "Enhanced anomaly detection in IoT: A transformer based approach for multivariate time series data," in *2024 International Conference on Engineering & Computing Technologies (ICECT)*, Islamabad, Pakistan, 2024, pp. 1–6. https://doi.org/10.1109/ICECT61618.2024.10581104

[9] X. X. Wang, D. C. Pi, X. Y. Zhang, H. Liu, and C. Guo, "Variational transformer-based anomaly detection approach for multivariate time series," *Measurement*, vol. 191, p. 110791, 2022. https://doi.org/10.1016/j.measurement.2022.110791

[10] Y. F. Li, X. Y. Peng, J. Zhang, Z. Y. Li, and M. Wen, "DCT-GAN: Dilated convolutional transformer-based GAN for time series anomaly detection," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 4, pp. 3632–3644, 2023. https://doi.org/10.1109/TKDE.2021.3130234

[11] C. Y. Ding, J. Zhao, and S. L. Sun, "Concept drift adaptation for time series anomaly detection via transformer," *Neural Process. Lett.*, vol. 55, no. 3, pp. 2081–2101, 2023. https://doi.org/10.1007/s11063-022-11015-0

[12] L. K. Kong, J. S. Yu, D. Y. Tang, Y. Song, and D. Y. Han, "Multivariate time series anomaly detection with generative adversarial networks based on active distortion transformer," *IEEE Sens. J.*, vol. 23, no. 9, pp. 9658–9668, 2023. https://doi.org/10.1109/JSEN.2023.3260563

[13] S. A. Bakhsh, M. A. Khan, F. Ahmed, M. S. Alshehri, H. Ali, and J. Ahmad, "Enhancing IoT network security through deep learning-powered Intrusion Detection System," *Internet Things*, vol. 24, p. 100936, 2023. https://doi.org/10.1016/j.iot.2023.100936

[14] E. Krzysztoń, I. Rojek, and D. Mikołajewski, "A comparative analysis of anomaly detection methods in IoT networks: An experimental study," *Appl. Sci.*, vol. 14, no. 24, p. 11545, 2024. https://doi.org/10.3390/app142411545

[15] B. R. Kikissagbe and M. Adda, "Machine learning-based intrusion detection methods in IoT systems: A comprehensive review," *Electronics*, vol. 13, no. 18, p. 3601, 2024. https://doi.org/10.3390/electronics13183601

[16] G. Z. Chai, S. M. Li, Y. Yang, G. H. Zhou, and Y. H. Wang, "CTSF: An intrusion detection framework for industrial internet based on enhanced feature extraction and decision optimization approach," *Sensors*, vol. 23, no. 21, p. 8793, 2023. https://doi.org/10.3390/s23218793

[17] J. Casajús-Setién, C. Bielza, and P. Larrañaga, "Anomaly-based intrusion detection in IIoT networks using transformer models," in *2023 IEEE International Conference on Cyber Security and Resilience (CSR)*, Venice, Italy, 2023, pp. 72–77. https://doi.org/10.1109/CSR57506.2023.10224965

[18] G. S. Kuaban, E. Gelenbe, T. Czachórski, P. Czekalski, and J. K. Tangka, "Modelling of the energy depletion process and battery depletion attacks for battery-powered Internet of Things (IoT) devices," *Sensors*, vol. 23, no. 13, p. 6183, 2023. https://doi.org/10.3390/s23136183

[19] M. Nakıp and E. Gelenbe, "Online self-supervised deep learning for intrusion detection systems," *IEEE Trans. Inf. Forensics Security*, vol. 19, pp. 5668–5683, 2024. https://doi.org/10.1109/TIFS.2024.3402148

[20] W. Wang, S. L. Jian, Y. S. Tan, Q. B. Wu, and C. L. Huang, "Robust unsupervised network intrusion detection with self-supervised masked context reconstruction," *Comput. Secur.*, vol. 128, p. 103131, 2023. https://doi.org/10.1016/j.cose.2023.103131

[21] M. Wang, N. Yang, and N. Weng, "Securing a smart home with a transformer-based IoT intrusion detection system," *Electronics*, vol. 12, no. 9, p. 2100, 2023. https://doi.org/10.3390/electronics12092100

[22] G. Li, Z. Y. Yang, H. L. Wan, and M. Li, "Anomaly-PTG: A time series data-anomaly-detection transformer framework in multiple scenarios," *Electronics*, vol. 11, no. 23, p. 3955, 2022. https://doi.org/10.3390/electronics11233955

[23] S. Trilles, S. S. Hammad, and D. Iskandaryan, "Anomaly detection based on artificial intelligence of things: A systematic literature mapping," *Internet Things*, vol. 25, p. 101063, 2024. https://doi.org/10.1016/j.iot.2024.101063