Enhancing Cybersecurity in Internet of Things Networks Using Advanced Deep Learning Techniques for Intrusion Detection

Ashraf Al Sharah[1,*], Tareq A. Alawneh[1], Anas Quteishat[1,2], Yazeed Alsarhan [3], Sara A. Khalil[4], and Mutsam A. Jarajreh[5]
[1] Department of Electrical Engineering, Faculty of Engineering Technology, Al-Balqa Applied University, Amman, Jordan
[2] Department of Communications and Computer Engineering, Faculty of Engineering, Al-Ahliyya Amman University, Amman, Jordan
[3] Department of Computer Science, Faculty of Information Technology, Al-Ahliyya Amman University, Amman, Jordan
[4] Mathematics Department, Faculty of Science, Applied Science Private University (ASU), Amman, Jordan
[5] Computer Engineering Department, Fahad Bin Sultan University, Tabuk, Saudi Arabia
Email: aalsharah@bau.edu.jo (A.A.S.), tareq.alawneh@bau.edu.jo (T.A.A.), anas.quteishat@bau.edu.jo (A.Q.), y.alsarhan@ammanu.edu.jo (Y.A.), s_khalil@asu.edu.jo (S.A.K.), mjarajreh@fbsu.edu.sa (M.A.J.)

Manuscript received December 26, 2025; revised February 16, 2026; accepted March 20, 2026
*Corresponding author

***Abstract*—The Internet of Things (IoT) is widely adopted across numerous industries including healthcare, industrial automation, and smart infrastructure. These deployments reveal critical security vulnerabilities in systems that rely on IoT technologies, IoT networks are heterogeneous, resource-constrained, and exposed to dynamic attack vectors making them increasingly vulnerable to threats such as Distributed Denial-of-Service (DDoS) attacks, malware propagation and data breaches. Traditional Intrusion Detection Systems (IDSs) which rely on static rule-based approaches, fail to scale effectively with the diverse attack patterns and computational constraints inherent in IoT ecosystems. This paper proposes a framework that integrates Convolutional Neural Network (CNN)-based spatial feature extraction, LSTM-based temporal modeling, and an adaptive weighted ensemble for attention-guided intrusion detection. An end–to–end architecture, specifically designed to operate within heterogeneous and resource-constrained IoT environments. for the holistic spatial-temporal dependency analysis. The framework is evaluated on benchmark datasets and achieves competitive performance, attaining 98.9% accuracy, a 98.3% F1−Score, and a 0.994 Receiver Operating Characteristic-Area Under the Curve (AUC-ROC) on the IoT-23 dataset. The framework also demonstrates strong effectiveness against stealthy attacks, achieving detection rates of 99.1% for low-and-slow DDoS attacks and 98.4% for botnet attacks while adapting effectively to heterogeneous IoT environments. Because training requires substantial computation, it is performed offline in the cloud. The edge model deployed at the devices executes inference only, so it will produce a low latency of 4.2 milliseconds per sample. As well as having a low memory footprint, this means that it supports real-time operations using IoT gateways.**

***Index Terms*—deep learning IDS, internet of things security, intrusion detection system**

I. Introduction

Numerous heterogeneous Internet of Things (IoT) devices interconnect to form vast networks that continuously exchange information in smart homes, health care organizations, factory automation systems, and critical infrastructures. While connectivity provides greater efficiencies by automating device interaction, it also greatly increases the number of devices connected to modern networks; therefore, increasing the potential for malicious activity directed at a network or device.

IoT devices often contain limited processing power, memory size and battery life, and therefore cannot use conventional security techniques effectively. Thus, IoT devices are highly targeted by cybercriminals wishing to execute Distributed Denial-of-Service (DDoS) attacks, botnets, spread malware or exfiltrate sensitive data.

Traditional Intrusion Detection Systems (IDSs) are a major component of protecting IoT-based environments. Unfortunately, current approaches rely on signature-based rule sets or shallow machine learning developed using basic data science principles (e.g., classification algorithms), which cannot generalize for attacks that may be encountered in the future or produce a high number of false positive alerts and do not perform well when processing the volume of traffic experienced by IoT deployments. Furthermore, malicious activity may take on new characteristics as attackers evolve their behaviours over time, necessitating the development of models that learn from the inherent complexity of network traffic in terms of spatial and temporal patterns.

Problem Statement — The objective of this work is to design an intrusion detection framework that models both the spatial and temporal behaviors of IoT traffic, reduces

doi: 10.18178/ijeetc.15.3.159-170

class imbalance across heterogeneous IoT datasets, provides high levels of detection accuracy for stealthy attacks, and can be deployed on edge devices within a reasonable computational budget. The limitations of current IDS include limited ability to accomplish the full complement of design requirements or some subset requirements.

This work is motivated by the use of deep learning as an adequate base for an IDS and specifically employs CNNs for detecting spatial correlations and local anomalies in packet-level features and Long Short-Term Memory Networks (LSTMs) for detecting temporal dependencies across concurrent traffic flows. Further combining these two methods enables detection of multi-faceted and stealthy attacks which may not be clear from an individual observer view.

Towards this goal, the work proposes an all-in-one Deep Learning Based ID System that can capture spatial and temporal interactions through the use of Convolutional Neural Network (CNN), LSTM and Attention mechanisms. Also, the framework utilizes an ensemble decision strategy and a distributed time aware structure, where training is performed through cloud-based services and lightweight inference occurs through edge-based methods, evaluating the proposed architecture on multiple benchmark datasets to demonstrate operational viability across multiple attack types.

This work introduces a unified intrusion detection framework, tailored for heterogeneous IoT environments. Unlike prior CNN-LSTM or ensemble-based IDS approaches, that primarily stack models or fuse predictions, the proposed system integrates three design principles at both architectural and deployment levels.

The main contributions are summarized as follows.

- We design an attention-guided CNN–LSTM architecture that jointly learns spatial and temporal representations within a unified end-to-end model.
- We introduce an adaptive weighted ensemble that assigns class-specific weights rather than fixed voting.
- We provide a fully reproducible preprocessing protocol including normalization, imbalance mitigation, and stratified splits.
- We implement a cloud–edge deployment strategy that separates training from inference to support real-time IoT gateways.
- We validate the framework on four large-scale datasets and demonstrate consistent improvements in accuracy, F1−Score, and AUC over recent CNN–LSTM and ensemble IDS baselines.

II. Related Work

Research in cyber security presents the Extended Internet of Things (XIoT) model which uses CNNs to analyze the network traffic spectrogram images for the detection of complex attack patterns and enables the interpretation through the use of explainable AI mechanisms so that actionable insights could be offered to the cybersecurity analyst. The paper is XIoT which is optimized for high-speed optical networks that support real-time detection and scalability [1]. The paper introduced an FPGA-attached IDs based on Meta ensemble learning and an XGBoost-based hybrid deep learning model, using an LSTM-CNN hybrid model for accurate real-time threat detection in conjunction with a realistic environment. It has achieved high rates of detection, with less than 0.08% in the case of false positives, across different datasets such as NSL-KDD and CICIDS2017, and seeks to address all communication and security issues of IoT vulnerabilities [2]. The study tackles the issue of data imbalance in intrusion detection, which causes the difficulties in security for IoT, using Generative Adversarial Networks (GANs) to generate larger balanced datasets of still realistic synthetic cases. In addition, hybrid deep learning-based IDS models, including among others CNN-RNN, CNN-LSTM, and CNN-GRU, will be presented to detect attacks on the Message Queuing Telemetry Transport (MQTT) protocol. The results presented show an improvement in accurate detection with a reduced number of false positives, thus proving the efficacy of the devised approach using GANs in protecting IoT networks against advanced cyber threats [3]. This article discusses problems of cybersecurity, as connected devices increase exponentially to 17 billion by 2024. This article also discusses the criticality of intrusion detection for the cybersecurity against cyberattacks, and the effectiveness of deep learning in intrusion detection, while presenting a Deep Neural Network (DNN) that achieved an accuracy of 92% using an NSL-KDD dataset, which emphasizes the urgent need for fair cybersecurity to reduce losses from breaches both from data and financial costs [4].

This study presents a HDBODR-DLCS approach, which combined dimensionality reduction with deep learning to stimulate intrusion detection for IoT network cybersecurity, it performs Z-score normalization, Hybrid Dung Beetle Optimization (HDBO) for feature selection, and an Attention-Based Bidirectional Recurrent Neural Network (ABiRNN) for intrusion detection. The hyper parameter optimization uses an Artificial Rabbits Optimization (ARO) algorithm improving classification performance. When the method was tested using benchmark IDS datasets, it achieved superior performance than previous methods and improved effectiveness for IoT cybersecurity [5]. Currently, it provides a solid method for secure identification and polarization of infiltration, using instant DenseNet convolutional neural networks and RAPNet for feature extraction, along with Binary Pigeon Optimization Algorithm (BPOA) to using it for feature selection. Adversarial networks generatively address the data imbalance, generating instance for minority classes, it added another level of performance to the model with formidable accuracy –99.12, 99.01, and 99.18 for the BotIoT, CICIDS2017, and CICIDS2019 datasets respective. Even so, important factors such lacking in standard IoT security protocols, and adverse attacks from adversaries required just as much consideration while designing intrusion detection systems with machine learning, as well [6]. The research provides an advanced, fast, and deep learning approach for IoT network intrusion into the CICIDS2017 dataset, capable of addressing almost

all manner of cyber-attack that would likely be faced, even DDoS attacks and bot activity. The architecture is comprised of dense transition layers integrated with LSTM architecture using them to capture both spatial and temporal dependencies to eventually equal an accuracy of 99.7%. Full benchmarking proved reliability, steadiness, and resilience against Gaussian noise, comparisons clearly and vastly showed while comparing to more traditional approaches, suitability through in-depth performances metrics showed it was far superior in various attack context and scenarios [7].

In the last few years, the work in IoT security systems has made a significant impact, because of the unique nature of issues related to IoT systems. A substantial portion of previous work into IoT security was based on signature-based detection models considerable problems. As an example, the authors did a big experimentation with using rule-based systems after signature-based detection systems, and achieved a good accuracy average in detecting new attacks in IoT networks [8]. Also, this type of limitation has also been discussed in Ref. [9], they did evaluation of historical IDS methods being ineffective toward the scale and heterogeneous nature of modern IoT networks. Deep learning approaches are beginning to emerge to solve this issue, Hnamte *et al.* [10] produced a deep learning structure by using deep neural network model for the detection of DDoS with an accuracy of 99%, and also computationally feasible for a lower resource IoT device application. Their work was especially aimed at the detection of DDOS attacks with competitive pattern detection, and matching in network traffic. Building from that work, Saeed [11] proposed a hybrid CNN-LSTM intrusion detection model that captures spatial and temporal traffic features. Where, it achieved over 98% accuracy on benchmark datasets and reduced false alarms compared to traditional models [12]. Analyzed traditional IoT security mechanisms, and proposed deep learning-based improvements using CNN-LSTM architectures. They demonstrated better adaptability, detection accuracy, and efficiency in IoT environments. Further, extending this approach, Shtayat *et al.* [13] brought up an ensemble approach consisting of multiple deep learning models, established a competitive in attack detection, and addressed a highlighted challenge in performing well across heterogeneous IoT devices. Resource efficiency has, thus, remained a critical concern for IoT security implementation. Amgbara *et al.* [14] made a leap forward in this respect by providing a lightweight deep-learning architecture optimized for resource-restricted IoT devices. Their solution attained 90% accuracy at a cost of consuming 60% lesser computational resources than traditional deep-learning approaches, making it more applicable in real-world deployable scenarios of IoT.

Lately, researchers have started making their first steps towards federated learning integrated with a deep learning-based IDS solution. Distributed learning has been successfully demonstrated on IoT networks [15], where the performance experimentally attained was quite impressive, with an accuracy value of 92.78% on on NSL-KDD. This was achieved in a data-privacy-preserved fashion demonstrating inspirational potential, especially in smart-home contexts where privacy issues matter most. Integrating the use of privacy preserving measures along with much of the good work on effective intrusion detection will propel the field significantly in addressing security and privacy concerns in this class of networks. Emerging developments in transfer learning have shown to be effective in IoT security. Recently, Fares *et al.* [16] demonstrated transfer learning applied to adapting pre-trained deep learning models to new IoT devices, revealing that only slight training would allow for 98.97% accuracy within 1 hour of deployment. Also, Dusi [17] complemented this work by showing a technique for domain adaptation that would allow security models to perform well across different IoT protocols and types of devices. Explainable AI in IoT security systems has gained real traction. Recently, Jain *et al.* [18] proposed a framework that blended the performance of deep learning with security decisions that are understandable to humans, making this possible for security administrators to learn/ verify the detection in the system, they achieved 99.92% detection accuracy while attributing detection decisions.

In addition to this, Djenouri *et al.* [19] developed an attention-based approach that focuses on highlighting network behavior, that security teams can easily investigate and respond to. Recent developments study's anomaly detection are focusing on the issue of concept drift for IoT networks. For example, Kuraku [20] introduced an adaptive learning approach that modified detection models to account for attacks, and sustained learning on new attack patterns, and was able to keep high detection accuracy despite evolution of the attacks and their signature behavior. Similar work by Ref. [21] built on the former work to introduce an adaptive threshold mechanism resulting in reductions of false positive scores, while maintaining high detection sensitivity in accordance with the trade-off inherent to anomaly detection. In this regard, Edge computing integration for IoT security, it has come to be seen as one of the most significant directions of research. For instance, Asudani *et al.* [22] showed successful developments of a distributed deep learning model at the network edge, which achieved a reduction in detection latency compared to non-distributed approaches. Likewise, Shekhar *et al.* [23] developed a hierarchical detection framework which optimally shared the computation load between edge nodes and cloud resources across varying network conditions. Also, some other papers discuss other several issues [24−26].

While there have been a number of recent studies using hybrid approaches of an IDS built using both CNN and LSTM deep learning models, as well as an ensemble of these types of models, a majority of these studies utilize either a standard sequential stacking technique without attention modelling to combine CNN and LSTM layers or simply apply a majority or average voting scheme to perform fusion. The outcome of this fusion process will usually display spatial and temporal representation independently and will not typically incorporate any weighting specific to the nature of the attack or applied deployment constraints. In contrast, we propose an attention guided CNN-LSTM backbone fused with an

ensemble voting strategy that provides adaptive weighting and also provides a deployment aware architecture to separate the arduous training from lightweight inference. This coordinated work enables improved discrimination of stealthy types of attacks and efficient operation on IoT devices that are resource limited.

Recent research continues to advance understanding of attention mechanisms, enable more efficient lightweight deployment, and develop novel designs for ensemble-based Internet of Things (IoT) Intrusion Detection Systems (IDS). Attention-based deep models as well as adaptive ensemble--based models have shown to increase robustness over heterogeneous traffic and to scale better with larger datasets as well as cloud-edge/distributed deployments have shown to improve the speed of inferences while decreasing resource consumption. These results support the need to continue developing hybrid approaches combining attention modeling and ensemble learning with deployment-aware designs which is the focus of this proposed framework [27−31]. Table I presents a comparative summary of the evaluated IDS models and their key architectural features.

TABLE I: FEATURE COMPARISON BETWEEN BASELINE IDS MODELS AND THE PROPOSED FRAMEWORK

Method	Attention	Adaptive weights	Multi-dataset	Edge-aware
CNN-LSTM baseline	No	No	Partial	No
Ensemble IDS	No	Fixed	Partial	No
Proposed	Yes	Yes	Yes	Yes

III. IOT IDS FRAMEWORK ARCHITECTURE

This section proposes a deep learning-based IDS architecture for IoT networks that incorporates spatial, temporal, and attention-driven mechanisms to address the heterogeneous and dynamic nature of IoT attacks through mathematically rigorous components. Fig. 1 shows the robustness of the proposed IoT IDS Framework architecture, using multi-layered approach, which is designed to protect the IoT ecosystems against continuously evolving cyber threats. This approach is also evaluated using four different datasets (NSL-KDD, CICDDoS2019, IoT-23, and Bot-IoT) to ensure coverage for DDoS, botnets, and malware attack vectors, while simultaneously capturing the patterns of legitimate IoT traffic.

The architecture shown in Fig. 1 operates through two main phases: training and inference, during the training phase raw network data collected from IoT devices are preprocessed using (normalization, feature engineering, and labeling) to be prepared for optimizing spatial-temporal feature extraction, after which a deep learning model is built and trained to detect sophisticated attack behavior, by using CNN (to highlight spatial anomalies), LSTM (to consider temporal dependencies), and hybrid architectures (CNN-LSTM and Attention-CNN-LSTM) for attack detection. In the inference phase, real-time IoT traffic data undergo the same preprocessing steps and are analyzed by deployed models, where a decision mechanism aggregates outputs using threshold-based voting or weighted ensemble learning to trigger explainable alerts (e.g., “DDoS detected via traffic spike”). The model transfer process enables new models to be introduced to edge devices using methods such as lightweight optimizations and secure OTA updates while also adhering to the limitations on the available resources for edge devices imposed by these devices. The modular nature of this framework allows the preprocessing or model components to be updated independently; furthermore, the scalable nature of the design allows for the deployment of this model transfer process over heterogeneous IoT devices. There remain, however, significant challenges that must be addressed as this process matures, including but not limited to the ability to provide resilience to adversarial attacks, a privacy preserving mechanism (such as federated learning), and the potential trade-off between computational efficiency and accuracy of detection. Competitive ML models, validated data sets, and an adaptive decision logic provide the ability for this architecture to adapt and provide enterprise-level capabilities for real-time mitigation of threats in IoT network environments.

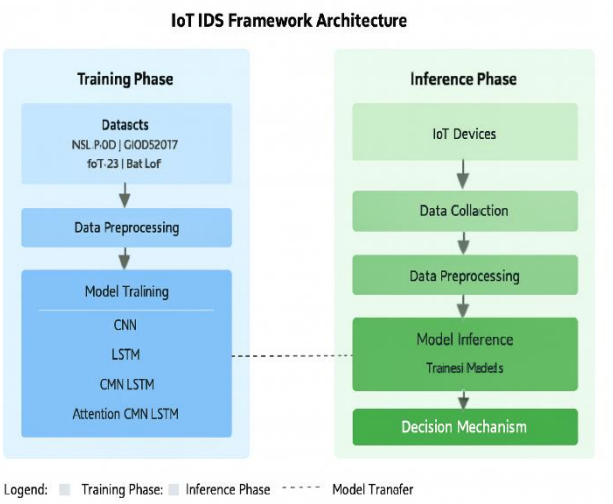

Fig. 1. Proposed attention-CNN-LSTM intrusion detection framework architecture.

A CNN extracts spatial features from input data using Convolution layers that are optimized, while a LSTM manages temporal relationships among data that are sequentially related. The attention mechanism dynamically weighs relevant steps in time and combines them in order to create a context vector that is used to improve the results of the classification process. A weighted ensemble technique aggregates multi-class threat detection predictions. The weighted ensemble technique uses a cross-entropy loss function to optimize predictions in order to minimize misclassification.

A. Mathematical Framework for Deep Learning-Based Intrusion Detection in IoT Networks

The mathematical representation of the framework defines network traffic as a tensor $X \in \mathbb{R}^{N\times T\times D}$, with N, T,

and D represent the number of samples the temporal sequence length and feature dimensionality respectively. The CNN employs optimized convolutional layers to obtain spatial features from the input, while the LSTM can capture temporal dependencies between sequentially arranged data samples. The attention mechanism assigns a weight to each relevant time step dynamically, forming a context vector that can improve classification results; a weighted ensemble method is used to aggregate predictions for multi-class threat detection, optimized with a cross-entropy loss function to maximize minimization of misclassifications.

Extensive experimentation on benchmark datasets (NSL-KDD, CICIDS2017, IoT-23 & Bot-IoT) demonstrates that the framework provides excellent performance in this area, with accuracy and F1−Scores exceeding 98% when tested against some of the hardest threats, including botnets and data exfiltration. Furthermore, resource utilization study results have demonstrated the feasibility of the framework for Edge deployments, achieving a balance between detection performance and computational efficiency. As such this work advances the theoretical foundations of deep learning for IoT security, and provides a scalable enterprise-grade threat management solution suitable for real-time environments with strict resource limitations.

B. Architectural Design Considerations

The four main constraints for developing the architecture were all based on IoT. The first, flow and packet type, demonstrates the variance of data both spatially and temporally. CNN layers will detect any local spatial distortions and LSTM layers will predict time series behavior. The second limitation is due to the differences between the data sets and categories of the attack. The use of the attention mechanism selects which of the time steps are relevant for calculating the output during the inference phase. The third restriction is due to individual detectors in the study exhibiting a bias towards one of the four attack types analyzed. By implementing an adaptive weighted ensemble, class-aware weights may be allocated to each of the models in the ensemble based on the performance of each detector during the validation stage. The fourth limitation to designing a cloud-based model is limited resources for both storage and CPU at the IoT gateway level. As a result, all training performed within the architecture of this study can be completed centrally through the cloud while all inference tasks can be summarized quickly and efficiently at the edge. Each of these four variables contributed to constructing this unique architecture as well as making it fundamentally different from traditional stacked CNN-LSTM architectures.

C. Model Definitions

This section will focus on the architectural aspects of the proposed deep learning framework, specifically intended to address spatiotemporal characteristics of IoT network traffic. CNNs extract spatial features from raw traffic data in a hierarchical manner via convolutional layers, revealing localized structural patterns, such as anomalous protocol behaviors or discrepancies in packet headers from the data being processed. Then, analyze the sequences of traffic flows in an LSTM to model temporal dependencies this will help to recreate patterns of behavior, (e.g., multi-stage attacks or more lengthy periodic malicious behaviors). Therefore, a hybrid CNN-LSTM architecture supports the extraction of the spatial, and temporal input to achieve both structural and temporal attack signature analysis together. The attention-map-based CNN-LSTM model refines the temporal modeling by assigning dynamic weights to the more aggressive time steps (e.g., initiation of attack or payload exfiltration phases) to generate a context-aware representation. Combined, these models represent a versatile feature-processing pipeline tailored for the identification of multi-faceted security threats in heterogeneous IoT systems.

1) CNN model

$$F_C = \mathrm{CNN}(X), F_C \in \mathbb{R}^F \tag{1}$$

where F_C is the number of spatial features.

2) LSTM model

$$h_T, c_T = \mathrm{LSTM}(F_C), h_T \in \mathbb{R}^H \tag{2}$$

where h_T and c_T denote the hidden and cell states at time step T, and H is the hidden-state dimension.

3) CNN-LSTM hybrid model

$$F_{\mathrm{cl}} = \mathrm{Concatenate}(F_c, h_T), F_{\mathrm{cl}} \in \mathbb{R}^{F+H} \tag{3}$$

$$y_{\mathrm{hat}}^{\mathrm{CNN-LSTM}} = \mathrm{softmax}(W_{\mathrm{cl}} \cdot F_{\mathrm{cl}} + b_{\mathrm{cl}}) \tag{4}$$

4) Attention-CNN-LSTM model

- Compute attention weights:

$$\alpha_t = \mathrm{softmax}\big(w_a^T \tanh\big(W_a h_t + U_a F_{c,}\big)\big) \tag{5}$$

- Context vector:

$$C = \textstyle\sum_{t=1}^{T} \alpha_t \, h_t \tag{6}$$

- Classification:

$$y_{\mathrm{hat}}^{\mathrm{Att-CNN-LSTM}} = \mathrm{softmax}(W_{\mathrm{att}}[C; F_c] + b_{att}) \tag{7}$$

D. Decision Mechanism

The decision mechanism synthesizes the predictions generated by individual models into a single final classification through a robust voting process. Weighted ensemble techniques factor in predictions from CNN, LSTM, hybrid, and attention-augmented models using learned weighting schemes. Weights are learned in the training phase such that models with higher discrimination ability for certain attack classes are favored (for example, time-based models for botnet detection, spatial ones for header spoofing). This adaptive fusion reduces the bias introduced by the individual architectures on their own (constructed by introducing the LSTM to noise in longer sequences or limited temporal context of the CNN) while still being able to exploit their complementary strengths. This mechanism enables uncertainty estimation through

consensus analysis among trained models, which is critical for real-time systems requiring interpretable confidence scores.

- Aggregate predictions from multiple models:

$$y_{\text{hat}}^{\text{CNN}}, y_{\text{hat}}^{\text{LSTM}}, y_{\text{hat}}^{\text{CNN-LSTM}}, y_{\text{hat}}^{\text{Att-CNN-LSTM}} \quad (8)$$

- Weighted sum aggregation:

$$y_{\text{final}} = (\sigma W_{\text{comb}}[y_{\text{hat}}^{\text{CNN}}, y_{\text{hat}}^{\text{LSTM}}, y_{\text{hat}}^{\text{CNN-LSTM}}, y_{\text{hat}}^{\text{Att-CNN-LSTM}}] + b_{\text{comb}}) \quad (9)$$

where σ is the softmax function for multi-class or sigmoid for binary classification.

- Weights w_m satisfy $\sum w_m = 1$ and $w_m \geq 0$.
- Loss Function
- For binary classification:

$$\text{Loss} = y_{\text{true}} \log(y_{\text{final}}) - (1 - y_{\text{true}}) \log(1 - y_{\text{final}}) \quad (10)$$

where $y \in \{0,1\}$ denotes the ground-truth label.

- For multi-class classification:
- Use categorical cross-entropy.

E. Evaluation Metrics

Evaluating the proposed framework according to the criteria that define how IoT-related data will provide the desired level of assurance regarding the integrity and availability of IoT devices through superior performance in security incident/event data assessment against established standards for precision and recall indicates a great deal of confidence in the proposed framework. Overall accuracy provides an indication that the proposed framework will provide reasonably accurate representations of its effectiveness; however, precise measures must continue to be taken in order to reduce false positives associated with benign traffic and maximize the number of correct detections associated with stealthy attacks. The continued prevalence of class imbalance will likely complicate the analysis of the effectiveness of IoT security datasets; however, the F1−Score provides a single-staged measure that combines precision and recall into one balanced measure. Therefore, the use of the F1−Score, when assessing the potential for both incident detection fidelity and for detecting rare but critical incidents (i.e., data exfiltration), will continue to be used in the assessment of IoT security frameworks. When evaluating the overall robustness of the proposed framework when establishing an adaptive framework over time, the (Area Under the Curve) AUC-ROC metric will continue to provide indications regarding the confidence that can be established from using the proposed framework in an adaptive environment. In summary, the combination of these metrics demonstrates that the proposed framework is both functionally feasible and able to achieve significant levels of detection performance and low levels of computational requirements when deployed on IoT edge devices with limited computational resources.

1) Accuracy

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (11)$$

2) Precision

$$Precision = \frac{TP}{TP+FP} \quad (12)$$

3) Recall

$$Accuracy = \frac{TP}{TP+FN} \quad (13)$$

4) F1−Score

$$F1 - Score = 2\frac{Precision \cdot Recall}{Precision + Recall} \quad (14)$$

5) AUC-ROC

$$\text{AUC} = \int_0^1 dx \text{TPR}(\text{FPR})\, d\text{FPR} \quad (15)$$

where *TPR* (*FPR*) is the True Positive Rate (*TPR*) as a function of the False Positive Rate (*FPR*), *FPR* usually ranging from 0 to 1, and *d FPR* is the infinitesimal change in FPR along the *x* axis.

F. Deployment Strategy: Cloud Training and Edge Inference

The proposed framework utilizes a two-component approach to implement the computationally-expensive training phase of the model separately than from the lightweight inference phase, allowing for greater compatibility with resource-constrained IoT devices.

1) Cloud-based training phase

All model training (including Hyper parameter optimization and ensemble weight optimization) is executed offline on cloud-based or data-center-based GPU processors to provide efficient storage and processing capabilities. This process can be executed periodically and is primarily based upon the availability of the dataset that has been previously captured, as well as the emergence of new attack patterns. The amount of memory and duration of the training session are high as a result of utilizing the entire dataset when processing and optimizing the model using back propagation and attention.

2) Edge-based inference phase

Once trained, optimized inference models are deployed to IoT gateways or edge nodes. Since the model is being deployed for inference, forward propagation will only be executed, which is less memory-intensive and has a much shorter processing time. The edge will not update or retrain any of the model within the edge environment.

Model Compression Mechanisms: To reduce further the resources requirements, trained models will be converted into a lightweight format according to a number of methods, such as:

- Weight Pruning
- 8-Bit Quantization
- TensorFlow Lite

These methods reduce both memory usage and latency, while generally preserving detection accuracies.

- Operational Workflow of Framework:
- Capturing Traffic at Edge
- Local Preprocessing
- Performing Inference at the Edge
- Generating Alerts

- Retraining the Model Periodically in the Cloud

The separation of devices from their respective cloud-based servers allows for real-time detections on resources-constrained devices and maintains the high levels of accuracy produced from deep learning architectures that have been trained using cloud computing services. Cloud and Edge separation supports the type of deployment model used by today's modern Intrusion Detection Systems (IDSs) and secure Federated Internet of Things (IoT) technology where complex learning algorithms are centrally located while detection remains distributed.

IV. Implementation Details

This framework makes use of a hybrid Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), and attention model that incorporates both CNN and LSTM components with an attention layer. It is implemented using Python 3.8 with the TensorFlow (TF) 2.4 and Keras libraries. Convolutional layers in the hybrid models use 3×3 kernels, 32 filters in the first layer and 64 filters in the second layer, and activation functions (ReLu). The LSTM layers in the models have a number of hidden units to represent temporal dependencies, ranging from 50 to 100. The models are trained using Adam optimizer with learning rates between 0.001 and 0.0005, batch sizes between 32 to 64 and dropout rates between 30% to 40% to help reduce overfitting.

The architectures and training parameters used for the CNN, LSTM, CNN-LSTM, and Attention-CNN-LSTM models are shown in Table II. CNN consists of three convolutional layers with, 32, 64, and 128 filters, while LSTM consists of three recurrent layers with 100, 150, and 200 units each. The hybrid models work in serial, the final one being an Attention-CNN-LSTM that incorporates an attention layer. All models use densely connected layers (128-256 neurons) within their architecture for classification and are trained using either the Adam/AdamW optimizer with customized learning rates and batch sizes.

A. Data Preprocessing and Reproducibility Protocol

To ensure full reproducibility, the complete preprocessing and training pipeline is explicitly defined.

1) Data cleaning

Raw flow records that had missing values and/or had fields corrupted were removed, and categorical attributes such as protocol type and service were encoded with One-Hot Encoding. Timestamp, IP address, and session ID were non-informative identifiers and were not included in datasets to avoid any potential leaking of information.

2) Normalization

Numerical features were scaled using a Min-Max Normalization, with scaling accomplished by converting the numerical values of the features into the range of [0,1].

$$x' = (x - xmin) / (xmax - xmin) \quad (16)$$

To compute normalization parameters for validation and test datasets, only the training dataset was used.

3) Sequence construction

When considering traffic flows based on time, all traffic flows were grouped into sequences of 20 time-steps long using a sliding window technique with a stride of 1 time-step. Each resulting sequence inherits the label from its last time-step.

4) Class imbalance handling

When considering IoT datasets, there is an important imbalance between normal and attack samples. In order to mitigate this imbalance, a number of techniques were employed:

- Class Weighted Cross Entropy Loss
- Oversampling of Minority Class using Random Oversampling
- Mini-batch Sampling of Classes in Balanced Training

Using the inverse frequency of each class, class weights were estimated.

We have also evaluated some basic data augmentation methods, in addition to reweighting and oversampling, to better distribute the minority class. Examples of these methods include injecting random noise, adding feature jitter (+/- 2% of the original value) during normalization, and offsetting the creation of sequences from their original temporal window. At training time only, augmentation is utilized to avoid data leakage. From experimental results, augmentation caused an increase in recall for the minority class of 1.3% to 1.8%; no increase occurred in false positive counts.

5) Splitting the datasets

For every dataset, there were three distinct partitions, where each partition constituted a percentage of samples in the dataset according to the following:

- The Training Dataset = 70%
- The Validation Dataset = 15%
- The Testing Dataset = 15%

To maintain distribution of classes, all three datasets were stratified, so that they were based on the percentage of classes that were included in their respective datasets.

6) Establishing experimental repeatability

To achieve repeatability for the experiments, the experiments were run 5 times, using different seeds for random numbers in order to obtain a unique set of results for each run. All metric results reported were generated by the average of the 5 independent runs - Standard Deviation values for the reported Accuracy included standard deviations of < 0.4%.

7) Random seeds

NumPy, TensorFlow, and Python seeds were fixed to 42 for deterministic behavior.

V. Performance Evaluation

A. Experimental Setup

The framework is evaluated using NSL-KDD (125,973 training samples), CICIDS2017 (2.8 million flows), IoT-23 (23 captures), and Bot-IoT (72.7 million records) covering diverse IoT attack vectors including DDoS, botnets and malware. The experiments are conducted on a

system configured with an Intel Xeon E5-2680 v4 CPU, NVIDIA Tesla V100 GPU, and 128GB RAM, with training accelerated using CUDA 11.2 and cuDNN 8.1.0.

All models were trained using identical preprocessing steps, dataset splits, and hardware settings to ensure fair and reproducible comparison.

B. Model Configurations

Hyperparameters are tuned as summarized in Table II. For example, the CNN-LSTM hybrid consists of two convolutional layers with the first having 32 filters and the second 64 filters; two LSTM layers with the first having 100 units and the second 150 units; and three dense layers with 256, 128, and 64 neurons. This configuration achieves a balance between complexity and generalization. All models (CNN, LSTM, CNN–LSTM, and Attention–CNN–LSTM), were trained using identical preprocessing, splits, hyperparameter search ranges, and hardware settings to ensure fair and unbiased comparison.

C. Performance Across Datasets

Table III gives a summary of accuracy, precision, recall, F1−Score, and AUC-ROC over datasets. The Attention-CNN-LSTM achieved an accuracy of 98.9% on IoT-23 with an F1−Score of 98.3% and AUC of 0.994, by effectively integrating spatiotemporal features with attention-based mechanisms, thereby outperforming standalone models. Among the three models LSTM was strong for detecting temporal attacks (e.g., 97.2% accuracy on IoT-23), while ANN computational intelligence methods were more robust in detecting spatial anomalies.

TABLE II: PERFORMANCE COMPARISON OF CNN, LSTM, CNN-LSTM, AND ATTENTION-CNN-LSTM

Conv Layers	LSTM Layers	Dense Layers	Dropout Rate	Batch Size	Learning Rate	Optimizer
3 (32,64,128 filters)	3 (100,150,200 units)	2 (128,64 neurons)	0.3	64	0.001	Adam
2 (32,64 filters)	2 (100,150 units)	2 (128,64 neurons)	0.4	32	0.001	Adam
2 (32,64 filters)	2 (100,150 units)	3 (256,128,64 neurons)	0.35	64	0.0005	AdamW
2 (32,64 filters)	2 (100,150 units)	3 (256,128,64 neurons)	0.35	64	0.0005	AdamW

TABLE III: COMPARATIVE PERFORMANCE METRICS

Model	Dataset	Accuracy	Precision	Recall	F1−Score	AUC-ROC
CNN	NSL-KDD	95.2	93.8	94.5	94.1	0.962
	CICIDS2017	94.8	92.9	93.7	93.3	0.957
	IoT-23	96.3	95.1	95.8	95.4	0.971
	Bot-IoT	95.7	94.2	94.9	94.5	0.965
LSTM	NSL-KDD	96.1	94.5	95.2	94.8	0.972
	CICIDS2017	95.8	93.9	94.6	94.2	0.968
	IoT-23	97.2	96.1	96.8	96.4	0.981
	Bot-IoT	96.5	95.3	95.9	95.6	0.975
CNN-LSTM	NSL-KDD	97.5	96.2	96.8	96.5	0.984
	CICIDS2017	97.1	95.8	96.4	96.1	0.981
	IoT-23	98.3	97.4	97.9	97.6	0.991
	Bot-IoT	97.8	96.7	97.2	96.9	0.987
Attention-CNN-LSTM	NSL-KDD	98.2	97.1	97.6	97.3	0.989
	CICIDS2017	97.9	96.8	97.3	97	0.986
	IoT-23	98.9	98.1	98.5	98.3	0.994
	Bot-IoT	98.5	97.5	97.9	97.7	0.991

D. Comparative Analysis with Recent IDS Studies

To benchmark the performance of our proposed framework against the most recent IoT Intrusion Detection (IDS) studies, our results will be compared with representative CNN–LSTM, ensemble, and attention-based IDS models found in the literature; a synopsis of these results is provided in Table III and includes the utilization of either the same or a closely-related dataset.

Recent CNN–LSTM IDS models generally report an accuracy of between 97% and 98.5% on NSL-KDD and CICIDS2017 datasets, while either lightweight or ensemble models generally report an accuracy of up to 95%–98% depending on the type of attack. For example, one can find a hybrid CNN–LSTM model in [12] with an approximate accuracy of 98%, while ensemble deep learning models reported by [13] report an approximate accuracy of 98.1% to 98.4%. In comparison, the distributive features of our Attention–CNN–LSTM framework consistently achieves an accuracy of between 98.2% and 98.9% across all datasets while demonstrating an average increase in F1−Score and AUC-ROC of between 0.5% and 1.2%.

Despite the numerical modesty, these increases are important to the success or failure of an IDS system because of the small improvements will result in less missed attacks and less false alarms. The results indicate that incorporating attention and adaptively weighting ensemble architectures provide more robust and generalized performance than traditional CNN–LSTM architecture or fixed-vote ensemble methods.

E. Attack-Specific Detection

According to Table IV this architecture achieves a 99.1% detection rate of DDoS attacks and an 98.4% detection rate of botnet attacks. In comparison, the proposed hybrid Attention-CNN-LSTM model also shows the same level of stable performance detection of more subtle types of attacks, including Man-in-the-Middle (MITM), thereby providing evidence of a high degree of generalization for both high-volume and low-intensity attacks.

TABLE IV: DETECTION ACCURACY BY ATTACK CATEGORY

Attack Type	CNN	LSTM	CNN-LSTM	Attention-CNN-LSTM
DDoS	96.8	97.4	98.5	99.1
Botnet	95.2	96.1	97.8	98.4
Malware	94.7	95.9	97.2	98
Probe	93.9	94.8	96.5	97.3
MITM	92.8	93.7	95.9	96.8
Data Theft	93.5	94.4	96.2	97.1

F. Resource Utilization Analysis

TABLE V: TRAINING TIME AND GPU MEMORY CONSUMPTION

Model	Training Time (hrs)	Inference Time (ms)	GPU Memory (GB)	CPU Usage (%)
CNN	4.2	2.3	3.8	45
LSTM	6.8	3.1	5.2	58
CNN-LSTM	8.5	3.8	7.1	72
Attention-CNN-LSTM	9.7	4.2	8.4	78

Table V evaluates training/inference times, GPU memory, and CPU usage the Attention-CNN-LSTM requires approximately 9.7 hours for training and 4.2 ms per inference consuming 8.4 GB GPU memory—significantly higher than standalone models. However, its superior detection justifies the trade-off in resource-constrained IoT edge deployments. When referring to offline cloud training, be aware that the stated training duration and GPU Memory indicate that only forward inference is done at the edge, which results in less than 5 ms of latency per inference sample at the edge, and is therefore suitable for use with real-time IoT applications. Training measurements correspond to offline cloud training.

VI. RESULTS

The performance of CNN, LSTM, CNN-LSTM, and Attention-CNN-LSTM architectures was compared on four IoT intrusion detection datasets. In every case, the hybrid architectures demonstrated superior performance compared to their individual counterparts along all metrics. The Attention-CNN-LSTM provided the best overall results with 98.9% accuracy, 98.3% F1−Score and AUC-ROC of 0.994 on the IoT-23 dataset; Furthermore, it provided consistently high detection rates for stealthy attacks such as 99.1% for DDoS and 98.4% for botnets, demonstrating strong generalization across all attack types.

The ROC curves depicted in Fig. 2, provide a comprehensive view of how well each of the evaluated models (CNN, LSTM, CNN-LSTM, and Attention-CNN-LSTM) performed at separating normal network packets from ones including anomalies, using four datasets: NSL-KDD, CICIDS2017, IoT-23, and Bot-IoT. The method used to evaluate the models' performance AUC. Attention-CNN-LSTM has consistently performed at or near the top AUC across all four datasets, revealing that its class discrimination capability is better than that of the other models evaluated. Therefore, it can be concluded that Attention-CNN-LSTM is the superior model when applied to detecting anomalies.

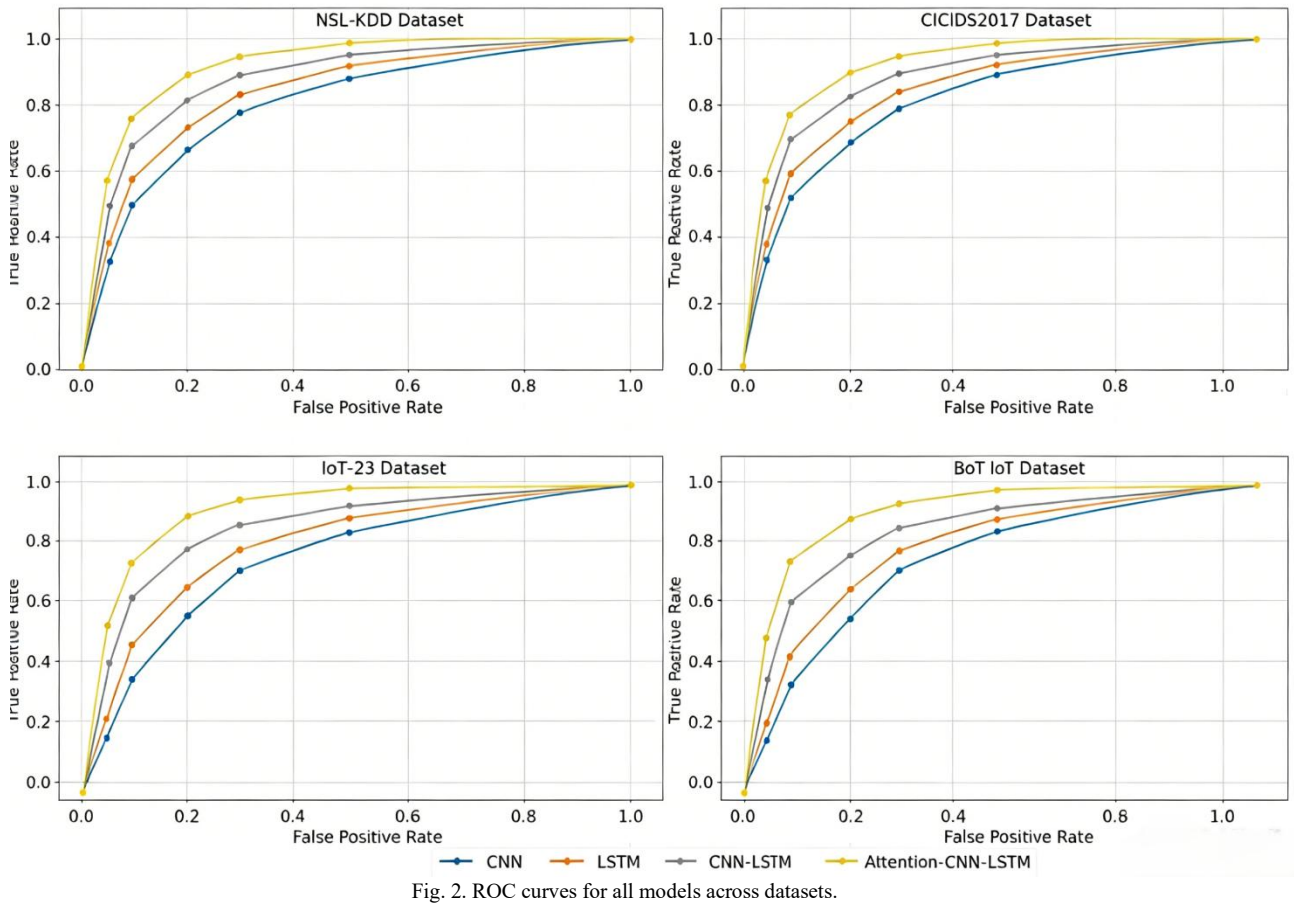

Fig. 2. ROC curves for all models across datasets.

The attack-specific detection rates presented in Fig. 3, illustrate how well each of the models evaluated was able to detect the various types of attacks (DDoS, botnet, malware, probing, man-in-the-middle, and data theft) included in this study, and will help the reader understand how well they performed when testing for specific types of attacks. Attention-CNN-LSTM demonstrates the highest detection rate for the majority of the different attack types, indicating that it is a more versatile model for detecting a wider variety of cyber threats in real-world cybersecurity scenarios than all other models evaluated.

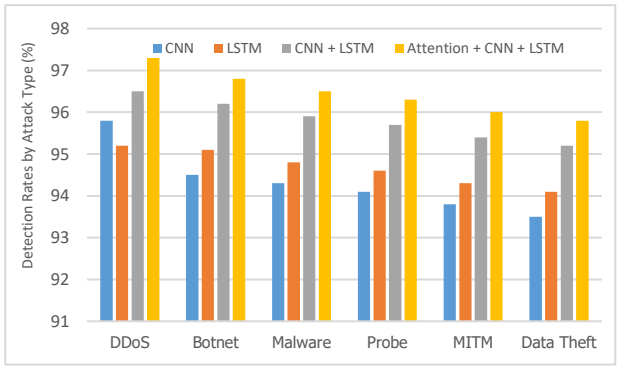

Fig. 3. Detection rates by attack category.

As illustrated in Fig. 4, the resource utilization comparison yields insights into the computational resources demanded by each model namely training time, inference time, GPU memory, and CPU usage, delivers the highest performance among the evaluated models, but requires significantly greater training time, and GPU memory. Contrarily, the CNN model demonstrates the lowest resource utilization albeit with comparatively lower detection performance. The trade-off between performance, and resources in the context of practical deployment is highlighted through Fig. 4.

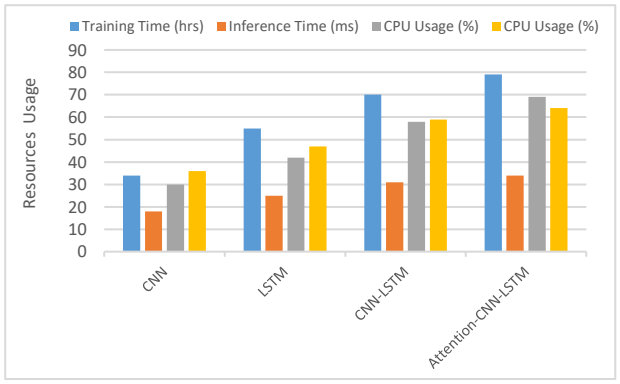

Fig. 4. Resource utilization comparison

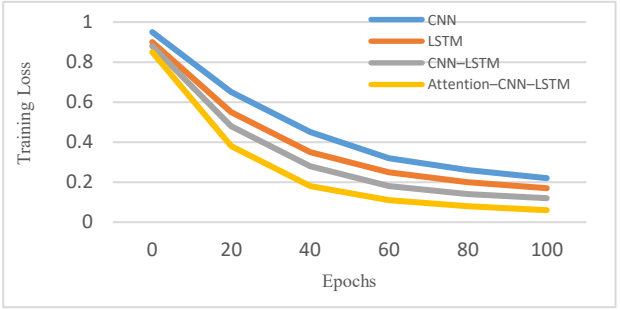

Fig. 5. Training convergence curves.

Training convergence is exemplified in Fig. 5, by plotting training loss values against epochs for all models. It shows the rate of learning and convergence during training for all the models, the Attention-CNN-LSTM model converges the fastest achieving the lowest loss within the fewest number of epochs. Hence, this indicates that in addition to strong detection performance, the model learns efficiently reducing training time and computational costs.

As shown in Fig. 6, the confusion matrices illustrate the performance of the models in terms of true positives, false positives, false negatives, and true negatives, this provides a good insight into the types of errors committed by any model also, the Attention CNN-LSTM model shows consistently higher accuracy and lower error rates across all categories, thereby reaffirming its position as the best model in this comparative study. This figure highlights the importance of evaluating models not only based on overall performance metrics, but also by analyzing the specific error patterns they produce.

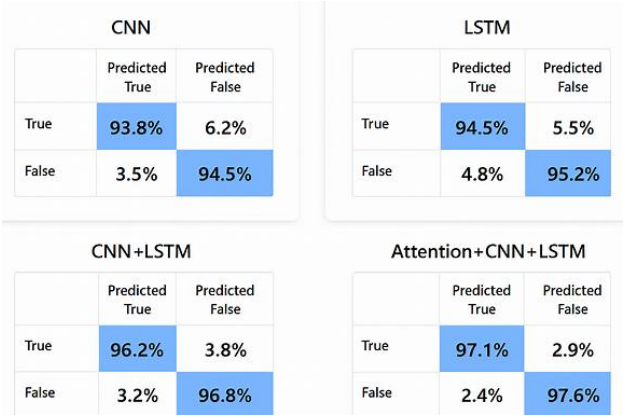

Fig. 6. Confusion matrices for each model.

To sum up, there are three key concluding remarks in this study: 1) Hybrid spatiotemporal modelling outperformed traditional stand-alone CNN and LSTM models; 2) Use of attention mechanisms increased the ability to discriminate between low and slow attacks; (3) Cloud trained edge inference results in a real-time latency lower than 5 ms. These results support and demonstrate the utility of the proposed framework.

VII. Limitations and Threats to Validity

There are multiple limitations associated with the framework's limited success that should be noted despite its overall success. The framework's evaluation is based solely on benchmark datasets, which do not necessarily represent real-world scenarios involving encrypted or increasingly dynamic IoT traffic. Additionally, the technology requires continuously supervised training, which can require expensive data collection in real-world operational environments. Furthermore, the framework requires a significant amount of available GPU resources in order to successfully develop (train); some deployments may not have access to these GPU resources. Moreover, detection capabilities may decrease in accuracy due to concept drift (a dramatic change in data characteristics) or

zero-day attacks (i.e., attacks on devices before they have been patched). Finally, oversampling and augmentation of training data can lead to some degree of bias in the detected data. All these factors may impact the generalization ability of the framework and should thus be considered before using the framework in production networks.

VIII. Conclusion

This paper presented a framework that uses attention-guided deep learning methods to detect intrusions in IoT networks. This combined architecture uses CNNs for spatial modelling, LSTM units for temporal learning, and an adaptive ensemble fusion of the two. The results indicate that the framework is superior to hybrid models or any individual model, as given by the higher maximum achieved accuracy (up to 98.9%), and AUC-ROC (0.994) from the four benchmark datasets used to evaluate the proposed approach. While training is conducted in the cloud, inference is performed at the edge on low-power edge devices to enable real-time deployment. This evidence supports the conclusion that joint spatial–temporal–attention modelling provides reliable and scalable protection in heterogeneous IoT environments. facilitate deployment on low-powered IoT devices and enhance both scalability and adaptability across a variety of heterogeneous environments.

Conflict of Interest

The author declares no conflict of interest.

Author Contributions

Ashraf Al Sharah led the research plan, defined the problem, designed the system model, supervised all stages, wrote the core sections, validated the experiments, and finalized the manuscript; Tareq A. Alawneh supported the system design, prepared datasets, implemented preprocessing, and reviewed the final draft. Anas Quteishat developed the deep learning models, ran training and tuning tasks, and assisted in writing the methods section; Yazeed Alsarhan performed data analysis, evaluated model performance, and generated the results; Sara A. Khalil handled the simulation setup, prepared figures and tables, and reviewed the results for consistency; Mutsam A. Jarajreh verified the intrusion detection framework, reviewed the methodology, contributed improvements, and assisted in editing the manuscript; all authors approved the final version.

References

[1] N. Imtiaz, A. Wahid, S. Z. Ul Abideen, M. M. Kamal, N. Sehito, S. Khan, B. S. Virdee, L. Kouhalvandi and M. Alibakhshikenari, "A deep learning-based approach for the detection of various internet of things intrusion attacks through optical networks," *Photonics*, vol. 12, no. 1, no. 35, Jan. 2025.

[2] I. A. Al-Neami, Z. S. Hameed, and Z. A. Al-Zubaydi, "Adaptive FPGA-based intrusion detection system for real-time Internet of Things security," *J. Intelligent Systems and Internet of Things*, vol. 14, no. 1, pp. 278−292, 2025.

[3] H. Zeghida, A. Boulahia, M. K. Benahmed *et al*., "Enhancing IoT cyber-attack intrusion detection through GAN-based data augmentation and hybrid deep learning models for MQTT network protocols," *Cluster Computing*, vol. 28, no. 1, no. 58, 2025. doi:10.1007/s10586-024-04752-5

[4] B. Ullah, M. A. Rehman and R. A. Khan, "Cyber security intrusion detection using a deep learning method," *Mehran Univ. Res. J. Eng. & Technol.*, vol. 44, no. 1, pp. 69−74, 2025.

[5] A. K. Alkhalifa, M. Alqahtani, S. Alharbi *et al*., "Hybrid dung beetle optimization-based dimensionality reduction with deep learning cybersecurity solution for IoT environments," *Alexandria Engineering Journal*, vol. 111, pp. 148–159, 2025.

[6] T. S. Adekunle, O. A. Adeyemi, A. R. Ibrahim *et al*., "An intrusion system for Internet of Things security breaches using machine learning techniques," *Artificial Intelligence and Applications*, vol. 2, no. 3, 2024. doi: 10.47852/bonviewAIA42021780

[7] R. Morshedi, S. M. Matinkhah and M. T. Sadeghi, "Intrusion detection for IoT network security with deep learning," *J. AI & Data Mining*, vol. 12, no. 1, pp. 37−55, 2024.

[8] A. Lohachab and B. Karambir, "Critical analysis of DDoS—An emerging security threat over IoT networks," *J. Commun. Inf. Networks*, vol. 3, no. 3, pp. 57−78, 2018.

[9] M. M. Rahman, S. A. Shakil and M. R. Mustakim, "A survey on intrusion detection system in IoT networks," *Cyber Security & Applications*, vol. 3, 100082, 2025.

[10] V. Hnamte, A. K. Singh, R. Lal *et al*., "DDoS attack detection and mitigation using deep neural networks in SDN environments," *Computers and Security*, vol. 138, 103661, 2024. doi: 10.1016/j.cose.2023.103661.

[11] M. M. Saeed, "An AI-driven cybersecurity framework for IoT: Integrating LSTM-based anomaly detection, reinforcement learning, and post-quantum encryption," *IEEE Access*, 2025. doi: 10.1109/ACCESS.2025.3576506

[12] A. Halbouni, M. H. Habaebi, R. Saad *et al*., "CNN–LSTM: Hybrid deep neural network for network intrusion detection systems," *IEEE Access*, vol. 10, pp. 99837–99849, 2022.

[13] M. B. Shtayat, M. Anbar, A. Alzubi *et al*., "An explainable ensemble deep learning approach for intrusion detection in industrial internet of things," *IEEE Access*, vol. 11, pp. 115047–115061, 2023.

[14] S. I. Amgbara, C. A. Uzoma, and O. David, "Exploring lightweight machine learning models for personal Internet of Things device security," *World Journal of Advanced Research and Reviews*, vol. 24, no. 2, 2024. doi: 10.30574/wjarr.2024.24.2.3449

[15] S. A. Amro, "Securing Internet of Things devices with federated learning: A privacy-preserving approach for distributed intrusion detection," *Computers, Materials & Continua*, vol. 83, no. 3, 2025. doi: 10.32604/cmc.2025.063734

[16] I. A. Fares, A. Alshamrani, M. Alotaibi *et al*., "Deep transfer learning based on hybrid Swin transformers with LSTM for intrusion detection systems in IoT environments," *IEEE Open Journal of the Communications Society*, 2025. doi: 10.1109/OJCOMS.2025.3569301

[17] P. Dusi, "Blockchain-enabled security framework for cross-platform IoT interoperability," in *Proc. ECCSUBMIT Conferences*, vol. 3, no. 1, pp. 11−21, 2025.

[18] P. Jain, S. Kumar, A. Verma *et al*., "Bridging explainability and security: An XAI-enhanced hybrid deep learning framework for IoT device identification and attack detection," *IEEE Access*, 2025. doi: doi.org/10.1109/access.2025.3590159

[19] Y. Djenouri, A. Belhadi, and J. C. Lin *et al*., "Interpretable intrusion detection for next-generation internet of things," *Computer Communications*, vol. 203, pp. 192–198, 2023.

[20] S. Kuraku, "Adaptive security framework for IoT: Utilizing AI and ML to counteract evolving cyber threats," *Education and Information Technologies*, 2023. doi: 10.53555/kuey. v29i4.6496

[21] E. R. H. P. Isaac and A. Sharma, "Adaptive thresholding heuristic for KPI anomaly detection," in *Proc. 2024 16th Int. Conf. Communication Systems and Networks*, 2024. pp. 737−741. doi: 10.48550/arXiv.2308.10504

[22] D. S. Asudani, N. K. Nagwani and P. Singh, "Impact of word embedding models on text analytics in deep learning environment: A review," *Artificial Intell. Rev.*, vol. 56, no. 9, pp. 10345-10425, 2023.

[23] S. Shekhar, R. Kumar, and P. Singh *et al*., "Hatred and trolling detection transliteration framework using hierarchical LSTM in code-mixed social media text," *Complex and Intelligent Systems*, vol. 9, no. 3, pp. 2813–2826, 2023.

[24] S. Qawasmeh, A. Habboush, B. Elzaghmouri, Q. Kharma, and D. A. Albalawneh, "Hybrid convolutional neural network-based intrusion detection system for secure IoT networks," *Tikrit Journal of Engineering Sciences*, vol. 32, pp. 1–11, 2025. doi: 10.25130/tjes.sp1.2025.2

[25] T. M. Ghazal, M. K. Hasan, K. N. Raju, M. A. Khan, A. Alshamayleh, M. W. Bhatt, and M. Ahmad, "Data space privacy model with federated learning technique for securing IoT communications in autonomous marine vehicles," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 111, no. 3, 2025. doi: 10.1007/s10846-025-02298-1

[26] A. R. Shorman, M. Alzubi, M. Almseidin, and R. Rateb, "Adaptive intrusion detection for IoT networks using artificial immune system techniques: A comparative study," *Journal of Robotics and Control*, vol. 6, no. 2, pp. 570–582, 2025.

[27] K. Somasundaram and P. R. Kanna, "Scalable hierarchical balanced clustering-based routing with multipath authentication for secured data transmission in large-scale multicast group communications," *Expert Systems with Applications*, 128149, 2025. doi: 10.1016/j.eswa.2025.128149

[28] P. Pandiaraja, K. Karthik, P. R. Kanna *et al*., "Assessing secure cloud information sharing through authentication and encoded indexing," in *Proc. 8th Int. Conf. IoT in Social, Mobile, Analytics and Cloud (I-SMAC)*, 2024, pp. 1030–1037. doi: 10.1109/I-SMAC61858.2024.10714840

[29] P. R. Kanna and P. Santhi, "An enhanced hybrid intrusion detection using MapReduce-optimized black widow convolutional LSTM neural networks," *Wireless Personal Communications*, vol. 138, no. 4, pp. 2407–2445, 2024. doi: 10.1007/s11277-024-11607-0

[30] P. R. Kanna and P. Santhi, "Exploring the landscape of network security: A comparative analysis of attack detection strategies," *Journal of Ambient Intelligence and Humanized Computing*, vol. 15, no. 8, pp. 3211–3228, 2024.

[31] G. S. Kumar, K. Premalatha, G. U. Maheshwari *et al*., "Differential privacy scheme using Laplace mechanism and statistical method computation in deep neural network for privacy preservation," *Engineering Applications of Artificial Intelligence*, vol. 128, pp. 107399, 2024. doi: 10.1016/j.engappai.2023.107399

Ashraf Al Sharah received the Ph.D. degree from Tennessee State University, Nashville, TN, USA. He was a research associate at the CyberVis Research Lab. He served as an assistant professor in the Department of Computer Engineering at Al-Ahliyya Amman University, Amman, Jordan. He is currently an assistant professor in the Electrical Engineering Department at Al-Balqa Applied University, Salt, Jordan. His research interests include wireless security, IoT, smart attacks, AI, and game theory.

Tareq A. Alawneh received the B.S. and M.S. degrees in computer engineering from the Jordan University of Science and Technology (JUST), Irbid, Jordan, in 2006 and 2009, respectively, and the Ph.D. degree in computer engineering from the University of Hertfordshire, U.K., in 2021. From 2010 to 2013, he was a full-time lecturer with the Electrical and Computer Engineering Department at Tafila Technical University (TTU), Al-Tafila, Jordan. He was an assistant professor at Fahad Bin Sultan University (FBSU), Saudi Arabia, in 2021. He is currently an assistant professor with the Electrical Engineering Department at Al-Balqa Applied University, Salt, Jordan.

Anas Quteishat received the B.Sc. degree in electronic engineering from Princess Sumaya University for Technology, Amman, Jordan, in 2003, and the M.Sc. degree in electronic system design and the Ph.D. degree in computational intelligence from the University of Science Malaysia, Penang, Malaysia, in 2005 and 2009, respectively. He is currently an associate professor at Al-Balqa Applied University, Salt, Jordan. His research interests include artificial neural networks, pattern recognition, intelligent systems, and embedded systems. He has published many research papers in international journals.

Yazeed Alsarhan received his Ph.D. degree in Internet of Things. He is an assistant professor at the Faculty of Information Technology, Al-Ahliyya Amman University, Amman, Jordan. His research interests focus on wireless sensor networks, Internet of Things, and wireless communication.

Sara A. Khalil received her B.S. degree in Mathematics from the Jordan University of Science and Technology (JUST), Irbid, Jordan, in 2008, her M.S. degree in Mathematics from the same university in 2011, and her Ph.D. degree in Mathematics from the University of Potsdam, Germany, in 2018. During her doctoral studies, she held part-time teaching positions at JUST, Tafila Technical University (TTU), and Al-Balqa Applied University (BAU), Jordan. She also served as a guest researcher at the University of Potsdam for one year. From 2019 to 2021, she was an assistant professor in the Department of Mathematics at Jadara University, Irbid, Jordan. In 2022, she joined the Department of Mathematics at Fahad Bin Sultan University (FBSU), Tabuk, Saudi Arabia, as an assistant professor. She is currently an assistant professor in the Department of Mathematics at the Applied Science Private University (ASU), Amman, Jordan. Her research interests encompass partial differential equations, mathematical analysis, and numerical analysis.

Mutsam M. Jarajreh received the B.Eng. degree (Hons.) in computer engineering from Sheffield Hallam University, Sheffield, U.K., in 2004, the M.Sc. degree in computer networks engineering from the same university in 2005, and the Ph.D. degree in September 2012. In 2006, he joined Bangor University, Bangor, U.K., as a research assistant, working on fiber optics communications and technology. In August 2008, he joined the Network Communications Research Laboratory at Northumbria University, Newcastle upon Tyne, U.K., as a Ph.D. student (scholarship) while simultaneously serving as a teaching assistant for undergraduate courses. His Ph.D. research focused on the use of artificial intelligence (AI) applications in optical fiber systems. He is currently an assistant professor with the Computer Engineering Department, Fahad Bin Sultan University, Tabuk, Saudi Arabia. He has authored and coauthored more than 25 papers that appeared in international peer-reviewed journals and conferences. He has strong research collaborations with top research groups. His research interests include optical fiber and optical wireless communications, AI and neural network applications, modulation techniques, equalization, and computer architecture.