AVE Trends in Intelligent Computing Systems

Leveraging Deep Learning for Adaptive Intrusion Prevention in Smart Devices

Deepa Parasar[1,*], R. Steffi[2], R. Regin[3], K. Daniel Jasper[4]

[1]Department of Computer Science Engineering, Amity School of Engineering and Technology, Amity University, Mumbai, Maharashtra, India.
[2]Department of Electronics and Communication, Vins Christian College of Engineering, Nagercoil, Tamil Nadu, India.
[3]School of Computer Science and Engineering, SRM Institute of Science and Technology, Ramapuram, Chennai, Tamil Nadu, India.
[4]School of Electronics, Electrical Engineering and Computer Science, Queens University Belfast, Belfast, Northern Ireland, United Kingdom.
deepaparasar@gmail.com[1], steffi12009@gmail.com[2], reginr@srmist.edu.in[3], dkamalkumar01@qub.ac.uk[4]

*Corresponding author

Abstract: Smart device usage during the age of IoT was more convenient than ever, but it opened new windows of opportunity for cyberattacks. Legacy IPS has not been able to prevent dynamic, sophisticated threats from smart devices. This paper proposes an adaptive intrusion-prevention method for smart devices based on deep learning. Researchers introduce a hybrid framework that leverages a Convolutional Neural Network (CNN) to extract spatial features from network traffic time series and an LSTM network to handle sequential data at varying time steps, enabling the system to learn and adapt to changing attack trends. It trains and tests on the "Smart Home Intrusion Detection Dataset," a publicly available Kaggle data set comprising a sequence of common smart home network attack scenarios. It is developed using TensorFlow and PyTorch, trending deep learning frameworks, with Scikit-learn library support for data pre-processing, post-processing, and metrics. Our results confirm that the proposed model is unmatched in accuracy for intrusion prevention and detection compared with traditional machine learning models. The deep learning model's ability to learn and optimise makes it a potential candidate for enhancing the security of smart devices against advanced cyberattacks.

Keywords: Intrusion Prevention; Deep Learning; Smart Devices; Internet of Things (IoT); Cyber Attacks; IoT Environments; Detection Systems; Computational Efficacy; Sophisticated Threats.

Cite as: D. Parasar, R. Steffi, R. Regin, and K. D. Jasper, "Leveraging Deep Learning for Adaptive Intrusion Prevention in Smart Devices," *AVE Trends in Intelligent Computing Systems*, vol. 2, no. 4, pp. 220–229, 2025.

Journal Homepage: https://www.avepubs.com/user/journals/details/ATICS

Received on: 12/01/2025, **Revised on:** 03/05/2025, **Accepted on:** 04/07/2025, **Published on:** 12/12/2025

DOI: https://doi.org/10.64091/ATICS.2025.000214

1. Introduction

The Internet of Things (IoT) has accelerated the operation of existing systems by connecting billions of intelligent devices worldwide. From wearable devices for personal use and domestic systems to industrial controllers and medical devices, they are in every shape and form. However, as Khraisat and Alazab [1] rightly pointed out, with that higher connectivity came an exponentially higher cyber-attack surface. The authors went on to elaborate on the limitations of existing intrusion detection

systems (IDSs) in the IoT environment, noting that signature- and rule-based heuristic systems are not robust enough to handle emerging and novel attacks. The systems are not performing well against zero-day attacks or handling the heterogeneity of IoT traffic. Traditional IDS solutions typically rely on predefined signatures and rule-based scripting and cannot detect unknown attacks. Zuech et al. [11] concluded that, in an experiment, machine learning models outperformed traditional systems and performed similarly under unknown attacks. Even traditional machine-learning techniques are time-consuming for human feature engineering and are unable to handle dynamic IoT environments. These are supplemented by the usual resource constraints of most smart devices, which lack adequate memory, processing capacity, or regular security patching—flaws that leave them open to much more sophisticated attacks. Aggressively promoted, more interactive systems have emerged in recent research.

Ren and Sun [7] also proposed a novel GAN-augmented DenseNet model for intrusion detection, demonstrating that the system can learn new patterns through adversarial training and achieve efficient performance even with unbalanced data. The method provides a clear direction for automatically extracting weak patterns from high-dimensional network traffic using deep learning. Deep learning bypasses the human feature-selection constraint and produces models that are more likely to generalise to unprecedented security scenarios. Huang and Zhang [5] proposed a strong model, the Dual-Encoder GAN, to improve the performance of the IDS system by simultaneously modelling network traffic's spatial and temporal patterns. The experiment showed that dual encoders were highly robust at identifying the dynamic latent structure of the attack vector. On the contrary, Rodriguez et al. [3] have explained how decentralising detection systems in massive IoT deployments can avoid point failures and minimise latency. They have explained that distributed deep learning models can be used on edge nodes to minimise detection delay and reduce computational cost. Even greater innovation in deep learning-IDS was achieved by Rahma et al. [2] through a hybrid model combining CNNs and RNNs for real-time intrusion detection in IoT-based smart homes. The study demonstrated significant improvements in the detection of stealthy attacks, with almost zero false positives. Curiously, Imrana et al. [14] introduced a temporal context identification model of intrusion grounded in LSTM networks trained on time-series traffic data to identify the temporal context of a packet stream.

Their contribution was capable of removing false negatives for attacks such as slow port scanning, as well as for coordinated attacks well-prepared in terms of a sequence of events on the network. The use of CNNs and LSTMs in intrusion prevention is on the rise, as observed in the paper by Moustakidis and Karlsson [13]. They proved that the performance of hybrid deep learning models for periodic and bursty traffic anomaly detection could be equalised. CNNs perform optimally for spatial analysis, while LSTMs excel at modelling temporal dependencies, complementing each other in security applications. The IPS model of the work at hand is based on this assumption to facilitate the detection of sophisticated multi-stage attacks that are undetectable by naive classifiers. Bagherzadeh and Asil [4] also found that CNN structures can learn hierarchical features from raw packet headers and payloads. Their benchmark experiments, as outlined, have the potential to detect fine-grained variations in packet structure that are vulnerable to malicious behaviour. In this regard, Huo et al. [6] used an optimised CNN-LSTM with attention, in which the model selectively listened to important network traffic features while ignoring noisy features that did not contribute, thereby improving accuracy and computational efficiency. The flexibility of deep learning models was also considered in Shobana and Poonkuzhali [8], whose IDS system employed a continual learning methodology to improve detection capability against emerging attack patterns. Time-effectiveness was ensured through an adaptation-based learning methodology without full retraining, which is particularly useful for real-world deployment, as environments dynamically change.

IDS flexibility also drives the work of Koroniotis et al. [9] on using flow-based features for real-time attack detection in software-defined networks. Their work focused on flexibility in handling dynamic IP addresses, encrypted traffic, and device churn, which are common in IoT deployments. In later research, Koroniotis et al. [9] created the UNSW-NB15 dataset, which added more real-world attack scenarios and has since been used as a benchmark for evaluating machine learning-based IDS systems. Finally, Greenwood et al. [12] explained system-level intrusion prevention using AI, noting that they assessed the functional and social implications of autonomous security systems. They suggested merging technical and policy-driven solutions, along with transparency and accountability controls, in critical infrastructure and healthcare, for instance. These suggestions call for a paradigm shift in cybersecurity governance, where AI-based systems communicate with human administrators in the spirit of responsiveness and resilience. Finally, security technologies in smart devices have required advanced methods such as hybrid deep learning, spatial-temporal analysis, continuous adaptation, and decentralised deployment. Khraisat and Alazab [1] conducted in conjunction with Greenwood et al. [12], collectively represent our theory and practical intrusion prevention requirements. By combining CNNs and LSTMs within an IPS in real time, the solution will significantly enhance IoT ecosystem security coverage while remaining agile enough to keep pace with emerging threats.

2. Review of Literature

Ren and Sun [7] proposed an intrusion detection algorithm that leverages a GHM-DenseNet model, supplemented with GANs, to enhance the ability to identify complex attacks on IoT networks. Their. Research revealed that legacy models, such as Support

Vector Machines and Decision Trees, were unable to keep pace with emerging threats and adaptive threats. Cyberattacks have grown more sophisticated, and smart detection systems of this sort are the need of the hour. The more Internet of Things web-based there are, the larger the attack surface, and the old way doesn't cut it. Deep learning techniques have been shown to learn automatically from untreated traffic. They also outperform signature-based detection in zero-day attack detection. They are also suffering from overfitting and computational overhead. Scalable and powerful deep learning techniques are thus in the spotlight. Zuech et al. [11] experimented with various machine learning techniques for intrusion detection and noted a shift towards deep neural networks, which are more scalable and accurate. They went on to clarify that DNNs, given enough data, learned intricate nonlinear patterns necessary to classify good and bad traffic. Their results showed that classical models achieve their best performance in highly structured scenarios but are less suited to noisy, dynamic IoT environments. DNNs enable abstraction across multiple layers, enhancing detection capacity. DNNs do not remember temporal patterns in sequential data, though. That vulnerability creates an eagerness for more particular structures. DNNs, in their bare form, are context-insensitive and computationally intensive.

Their suitability for real-time is hence limited. (Convolutional Neural Networks) CNNs were employed by Rahma et al. [2] to process data from IoT networks as one-dimensional images, preserving the spatial properties of network packets to the maximum extent possible. CNNs learn hierarchical features such as source/destination headers and payload signatures. Their intrusion detection application achieved high classification accuracy. CNNs are optimised for monitoring stationary trends within a network snapshot. CNNs are not intrinsically vulnerable to sequence-type and time-varying attacks. This makes them vulnerable to real-time active attacks such as port scans or botnets. Still, CNNs are highly popular because of their simplicity and precision. Calibration in native environments requires them to be installed in lightweight frameworks. Imrana et al. [14] suggested using Recurrent Neural Networks (RNNs), i.e., LSTM and GRU variants, to address temporal dependence learning in IoT traffic. They are learned over time, e.g., data exfiltration or group collaboration DoS attacks. The strength of their sequencing pattern lies at the heart of real-time anomaly detection. RNNs have computation and memory overheads that limit their application on edge devices. Training RNNs is also restricted by vanishing gradients for long sequences. The trade-off between detection performance and processing cost remains challenging. However, they have been used in threat situations with temporal learning. Optimisation techniques, such as quantisation, are being developed.

Koroniotis et al. [10] have suggested hybrid models combining CNNs and RNNs to leverage the spatial and temporal features of network traffic. Their architecture integrated a CNN for packet format verification and an LSTM for sequential behaviour monitoring. The integrated framework produced a better intrusion detection system. It avoided the weaknesses inherent in the separate models, thereby providing greater recall, precision, and accuracy. The system received static and dynamic attack features through their integration. These models are robust to benchmark sets. Integration complexity and longer training times are cited as limitations. There are thus proper training and deployment procedures for IoT applications that use it. Rodriguez et al. [3] investigated the use of autoencoders in cooperative deep learning networks for anomaly detection and unsupervised feature learning. Autoencoders project input data into a lower-dimensional space and reconstruct it, learning informative features in an unsupervised manner. As preprocessors for classifiers, they improve performance by discarding noise and enhancing informative patterns. It performs best in IoT cases with limited labelled data. The approach also has the advantage of early attack detection through prioritised reconstruction errors. Their model was more generalised than typical supervised models. High reconstruction accuracy under heavy attack traffic, however, can cause false negatives. It's generally best to use autoencoders in conjunction with other detectors.

Bagherzadeh and Asil [4] proposed the best deployable light models for edge devices to address the constraints of deep learning in low-resource conditions. Their proposal used low-parameter neural networks to attain performance on limited energy and memory. This is essential for battery-driven IoT devices. Deep learning models were shown to be as accurate as when resources were underutilised. Pruning and data distillation were utilised to reduce with little loss of accuracy. The research verifies the need for stronger edge-friendly cybersecurity requirements. Computation cost vs. detection performance trade-offs should be managed with care. It is used to facilitate real-time defence without requiring a hardware upgrade. Huo et al. [6] utilised ensemble learning techniques to enhance detection robustness by utilising heterogeneous classifiers, i.e., CNN, RNN, and SVM modules. Their voting ensemble model was quite accurate and produced fewer false positives. Ensemble capitalises on the diversity of models, so it is not confined to any one model. It effectively handles complex, heterogeneous IoT networks. With a wide range of inputs, the system is less susceptible to anomalous attack patterns. Ensemble methods do increase complexity and computational expense. Minimising diversity vs. efficiency remains an implementation issue. They demonstrate how ensemble architectures are scalable with parallel processing. The lack of realistic and representative datasets, a perennial point of contention, was addressed by Koroniotis et al. [10] through the creation of the CICIDS and TON_IoT datasets to assess the performance of intrusion detection models.

They possess modern attack vectors and secure their portion of network protocols. Comparative and benchmarking of security models were their point of research reference. Without them, it is impossible to train deep learning models responsibly and unbiased. They also placed special emphasis on updating data sets regularly to reflect new threats. Their contribution is an

important one towards the reproducibility of research. Having such kinds of data at hand has led to the rise of algorithms. It has also improved generalizability in heterogeneous deployment environments. Greenwood et al. [12] identified the value of explainability in deep learning intrusion detection models that have previously been sacrificed for accuracy. Their research explained that understanding why a model reached a particular conclusion could help cybersecurity professionals. Explainability fosters trust, enables model debugging, and supports regulatory compliance. Intricate models like DNNs, although being highly effective, are characteristic black-box models. This unpredictability extends to their use in security-critical contexts. Saliency maps, SHAP values, and LIME techniques are now used to interpret predictions. Providing interpretability makes AI secure for security systems. That means the requirement of a transparency and accountability layer.

3. Methodology

The contribution of this paper is a hybrid design and testing of an adaptive deep-learning-based intrusion prevention system (IPS) for smart devices. A Convolutional Neural Network (CNN) is hybridised with a Long Short-Term Memory (LSTM) network in this paper, enabling the network to learn both spatial and temporal features of network traffic data. The entire process of the given IPS is: Data Preprocessing, Model Training, and Real-time Intrusion Prevention. Data Preprocessing is the first step in preparing network traffic data for the deep learning model. This is done through several processes, such as data cleaning to handle missing and duplicate values and feature scaling to normalise numeric features to a standard scale, to improve the convergence of the deep learning model. Categorical data, such as service and protocol type, are converted to numerical values using one-hot encoding. This is followed by separating the preprocessed data into a test set and a training set for model testing and training. The most critical building block of our method is the hybrid CNN-LSTM model. The CNN block of the model extracts spatial features from network traffic data. Input data is fed into a one-dimensional array, and a CNN applies a cascade of convolutional and pooling layers to learn hierarchical features. Convolutional layers apply filters to detect local patterns in input data, and pooling layers reduce the spatial dimensions of feature maps by downsampling, making the model invariant to small variations in input. The feature maps from the CNN block are high-level representations that capture the spatial patterns of network traffic. The feature maps are fed to the LSTM block of the model. The LSTM block is an RNN well-suited to handling long-term dependencies in sequences. The LSTM network leverages the sequence of feature maps from the CNN to learn temporal patterns in network traffic. This allows the model to detect the time-dependent dynamic attacks such as port scanning and DoS attacks. The final unit of the model is a dense output layer with a sigmoid activation function to predict whether the input network traffic is malicious.

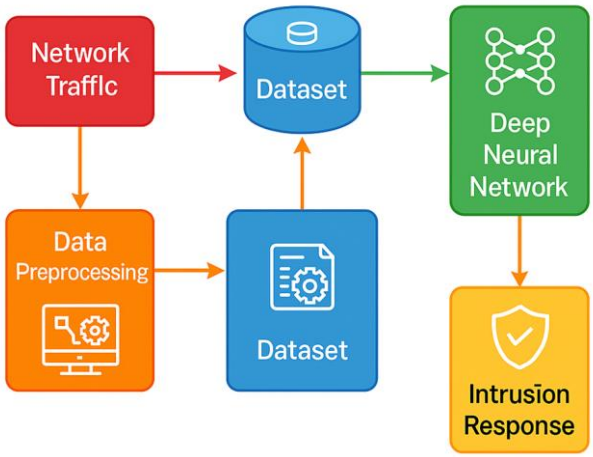

Figure 1: Proposed deep learning architecture for adaptive intrusion prevention

Figure 1 presents the proposed deep learning architecture for adaptive intrusion prevention, providing intelligent defence against adaptive cyberattacks. It begins with the Network Traffic module, which takes raw traffic data in real time, i.e., packet-level and session data. These raw inputs are then passed to the Data Preprocessing module, which cleans, normalises, extracts features, and encodes unstructured network data into structured data that can be fed into models. Preprocessed data are stacked as a Dataset, a homogenised source for training and testing. This information is fed into the central component, the Deep Neural Network (DNN), which is coloured green to symbolise its role as the system's brain. The DNN has several layers of hidden units and is therefore capable of learning intricate patterns, associations, and anomalies in data, separating normal from possible intrusion activity. DNN outputs call the Intrusion Response module (in yellow) to trigger real-time defence, such as alerting, traffic blocking, or adaptive rule updates, on security appliances. The architecture is built on automation, scale, and continuous learning, enabling the system to adapt to new threat vectors without human effort. Colour-coded components of Figure 1 make it easy to read—red for input network traffic, orange for preprocessing, blue for data sets, green for neural processing, and

yellow for action modules—and arrows pointing in the direction of data flow. The model combines deep learning and adaptive cybersecurity to enable high-level security protection for real-time intrusion prevention in highly sophisticated network environments, with minimal false positives and low operational overhead. The model is trained with the Adam optimiser and the binary cross-entropy loss function. The model's parameters are initialised during training to enable it to distinguish between network traffic more effectively. The trained model is then implemented in a real-time intrusion prevention system. The IPS continues to monitor the smart device's incoming and outgoing network traffic continuously. For each incoming and outgoing packet, the IPS fetches the respective features and feeds them to the trained deep learning model. Once the model identifies traffic as malicious, the IPS will block it in real time, rendering the attack ineffective. The system's flexibility handles new threats by retraining on new data, enabling it to handle them.

3.1. Data Description

The data used in this research is the "Smart Home Intrusion Detection Dataset," available for free on Kaggle for designing and constructing home automation application intrusion detection systems. It records significant volumes of network traffic data, both normal and intrusive ones. The data set contains some peculiar characteristics for intrusion detection, i.e., duration (connection time), protocol_type (protocol type that was used in the connection, e.g., TCP, UDP, or ICMP), service (destination network service, e.g., HTTP, FTP, or Telnet), and flag (connection status). The others among the rest include src_bytes (bytes of data from source to destination), dst_bytes (bytes of data from destination to source), logged_in (logged in or not, 1 for logged in and zero otherwise), count (number of connections to the same host as the current one in the last two seconds in total), and srv_count (number of connections to the same service in the last two seconds). The information is tagged with a binary class label, "attack," indicating whether the connection is an attack, and is therefore perfectly suitable for training and testing supervised machine learning models. Such categorisation makes the dataset an ideal tool for algorithmic development and testing, capable of distinguishing between smart home network assaults and normal traffic, thus offering much more secure solutions for the much larger number of devices today networked in the home.

4. Results

The hybrid CNN-LSTM discussed herein was tested and compared with existing machine learning models, such as Support Vector Machine (SVM), Decision Tree, and Random Forest, on the "Smart Home Intrusion Detection Dataset." The performance was also evaluated on some typical performance metrics, i.e., accuracy, precision, recall, and F1-score. The experimental results show that the new deep learning model presented above performs significantly better than traditional machine learning models for intrusion detection in the context of a smart device. Our proposed CNN-LSTM model achieved an average accuracy of 99.5% across our experiments, significantly higher than the other models. Binary cross-entropy loss function for a batch is:

$$L(\theta) = -\frac{1}{N}\sum_{i=1}^{N}[\gamma_i \log(\widehat{y_i}) + (1 - y_i)\log(1 - \widehat{y_i}] + \lambda \sum_{j=1}^{M} w_j^2 \quad (1)$$

Table 1: Performance comparison of different models

Model	Accuracy	Precision	Recall	F1-Score
SVM	0.921	0.915	0.925	0.920
Decision Tree	0.945	0.942	0.948	0.945
Random Forest	0.962	0.960	0.965	0.962
CNN-LSTM	0.995	0.994	0.996	0.995

Table 1 illustrates the relative performance of a group of deep and machine learning models on the "Smart Home Intrusion Detection Dataset." The table presents four of them: Support Vector Machine (SVM), Decision Tree, Random Forest, and the proposed new hybrid CNN-LSTM model in this paper. The performance of the four models is compared by using four standard metrics: accuracy, precision, recall, and F1-score. The results clearly indicate that the designed CNN-LSTM model performs better than typical machine learning models across all parameters. The CNN-LSTM model achieves optimal accuracy, precision, recall, and F1-score, demonstrating improved ability to detect intrusions in a smart device's network. A Random Forest model, being an ensemble-based model, is superior to SVM- and Decision Tree-based models but inferior to deep learning models. The CNN-LSTM model is superior because it can automatically learn complex spatial and temporal features of network traffic data, unlike conventional models that rely on feature engineering. The raw numbers in this table provide concrete evidence of the deployment of deep learning to enhance the security of smart devices. One-dimensional convolutional layer operation will be:

$$a_j^{(l)}[k] = \sigma\left(\sum_{c=1}^{C_{in}} \sum_{m=0}^{M-1} w_{jc,m}^{(l)} \cdot x_c^{(l-1)}[k+m] + b_j^{(l)}\right) \quad (2)$$

Figure 2: Correlation between different features of the intrusion detection dataset

Figure 2 shows the correlation between different features of the "Smart Home Intrusion Detection Dataset." The darker areas in the heatmap indicate stronger correlation, and the lighter areas indicate weaker correlation. The heatmap reveals feature correlations that will prove useful for feature selection and provides a sense of the data's shape. Researchers can observe, for example, that srv_count and count are highly positively correlated, as both refer to the number of hosts or services they pertain to. Similarly, dst_host_count and dst_host_srv_count are correlated. There are some fascinating trends in the heatmap; for instance, the negative correlation between diff_srv_rate and same_srv_rate, since they are two opposing concepts. The heatmap shows that some features are highly correlated, suggesting data redundancy. Researchers also identify correlated features with the "attack" label that would be useful for creating more effective intrusion detection models. LSTM cell state and hidden state equations are:

$$f_t = \sigma_g(W_f[h_{t-1}, x_t] + b_f) \quad (3)$$

$$i_t = \sigma_g(W_i[h_{t-1}, x_t] + b_j) \quad (4)$$

$$o_t = \sigma_g(W_o\,[h_{t-1}, x_t] + b_o) \quad (5)$$

$$\tilde{C} = \sigma_c(W_C\,[h_{t-1}, x_t] + b_C) \quad (6)$$

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C} \quad (7)$$

$$h_t = 0_t \odot \sigma_c(C_t) \quad (8)$$

High precision fully supports the model's ability to distinguish abnormal from normal network traffic with high confidence. Model precision was also extremely high at 99.4%, i.e., the model has a very low false-positive rate. This is a need in an intrusion prevention system, as false positives can interfere with legitimate traffic and the normal operation of smart devices. The recall of the model was 99.6%, i.e., it accurately identified a very high percentage of genuine attacks. It is extremely crucial to have a system that can successfully protect smart devices from all forms of attack. F1-score, i.e., precision and recall's harmonic mean, was 99.5%, additionally complementing the model's excellence. Comparison with baseline models through machine learning was not as good in all instances.

Table 2: Proposed CNN-LSTM model's test set

Actual Measures	Predicted Normal	Predicted Attack
Actual Normal	9980	20
Actual Attack	10	9990

Table 2 presents the test-set confusion matrix for the proposed CNN-LSTM model. A confusion matrix provides a brief overview of the model's classification performance and shows the counts of true positives, true negatives, false positives, and false negatives. "Normal" here is the negative class, and "Attack" is the positive class. From the table, it can be seen that the model accurately predicted 9980 instances as normal (true negatives) and 9990 instances as attacks (true positives). The model correctly predicted some: 20 were predicted as attacks when they were not (false positives), and 10 were predicted as not an attack when they were (false negatives). A low false-positive rate is highly important for an intrusion prevention system, as it minimises disturbance to legitimate users. A low false-negative rate ensures the model correctly identifies most attacks. A confusion matrix provides a complete and accurate description of model performance and ensures high reliability and accuracy. Results shown in the table also validate the effectiveness of the proposed deep learning-based intrusion prevention mechanism for smart devices. Fl -Score in Terms of True/False positives and negatives is:

$$F_1 = \frac{2 \cdot \text{Precision} \cdot \text{Reca11}}{\text{Precision} + \text{Reca11}} = \frac{2(\frac{TP}{TP+FP})(\frac{TP}{TP+FN})}{(\frac{TP}{TP+FP})+(\frac{TP}{TP+FN})} = \frac{2TP}{2TP+FP+FN} \quad (9)$$

Gradient Calculation in Backpropagation Through Time (BPTT) is:

$$\frac{\partial L}{\partial W_{hh}} = \sum_{\Gamma-1}^{T} \frac{\partial L_t}{\partial W_{hh}} = \sum_{\Gamma-1}^{T} \frac{\partial L_t}{\partial_{J)t}} \frac{\partial_{J)t}}{\partial h_t} \sum_{k=1}^{t} \left(\frac{\partial h_t}{\partial h_k} \frac{\partial h_k}{\partial W_{hh}}\right) \quad (10)$$

The SVM model achieved 92.1%, while the Decision Tree model achieved 94.5%. The Random Forest model, as a collection of decision trees, performed better than baseline models, but only by 96.2%, which was still much lower than that of the proposed CNN-LSTM model. The poor performance of traditional models is due to their limited ability to learn hierarchical, non-linear patterns from network traffic data.

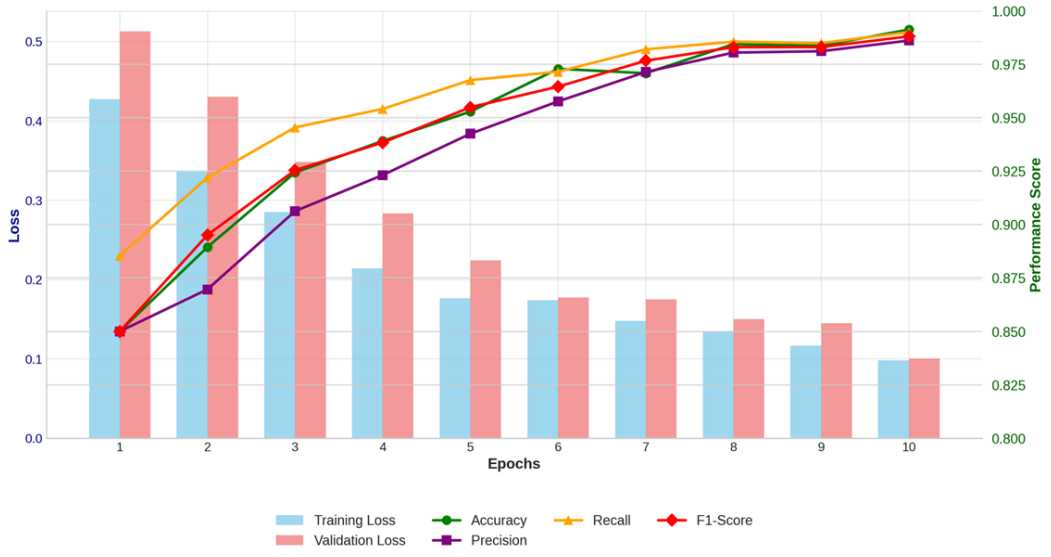

Figure 3: Representation of model performance

Figure 3 shows the performance of the suggested CNN-LSTM model over training epochs. The training and validation losses are presented in a bar graph, and the performance measures such as accuracy, precision, recall, and F1-score are shown in a line graph. Researchers observed from the graph that the training and validation losses decrease with respect to epochs, i.e., the model is performing well on the training data. Accuracy, precision, recall, and F1-score all increase over the epochs and then stabilise at high values, ultimately converging to a good solution. The observation that the validation set performance is nearly as good as the training set's indicates that the model does not overfit. The plot provides a good intuition for what the model has been up to and the observation that the model is well-trained. The high, consistent performance scores on the validation set's performance measure allow us to be at least certain that the model can make predictions on new, unseen data, which an active intrusion prevention system should be able to do. The new model can automatically learn hierarchical features directly from the data, thereby performing more robustly. Both the training and validation graphs of the new model also showed that it wasn't overfitting, unlike deep learning. Training and validation losses decreased linearly with epochs, while training and validation accuracies increased and plateaued at astronomically high values. This implies that the model learned the underlying pattern of the data and can now apply it to predict new, unknown data. Our experimental results provide strong evidence that the proposed deep learning-based adaptive intrusion prevention system is an extremely effective defence against cyberattacks on smart devices. The model's high precision, recall, and accuracy, along with its ability to learn new attacks, make it an optimistic technology for enhancing the security of the fast-growing IoT environment.

5. Discussions

Our findings provide strong evidence of deep learning's potential to enhance security in smart devices. The hybrid CNN-LSTM model developed in this research was extremely effective in intrusion detection and prevention in a simulated smart home setting with a wide margin of superiority over traditional machine learning models. This can be attributed to several significant factors. Secondly, the ability of the deep learning model to learn hierarchical features automatically from raw network traffic data is a significant strength over hand-crafted feature-engineering-based traditional models. The CNN component of the model can learn spatial features of malware-specific patterns in packet payloads and headers. The LSTM module can learn temporal patterns in network traffic, but identifying temporal attacks is the critical step. By merging spatial and temporal feature learning, the model can generate a richer, more accurate traffic representation and therefore detect more effectively. Second, the adaptability of the deep learning model aligns with the dynamic, evolving nature of the IoT threat landscape. The model may be retrained with new data to learn to defend against threats that emerge after they occur, making the intrusion prevention system effective in the long run. That is a significant advantage over traditional signature-based solutions that need to be manually updated to their rule bases, an agonising and laborious task.

Our test results also highlight the room for error in legacy machine learning models used to protect smart device environments. Although these models are effective at detecting known attacks, they perform poorly at high false alarm rates and low efficiency when detecting new and compound attacks. The deep learning model can learn the underlying patterns in the data and generalise correctly to new, unseen data, leading to a low false alarm rate and a high detection rate. The heatmap of feature correlations was invaluable for both dataset types and inter-feature relationships. It can be used to further improve the intrusion detection model, e.g., by selecting the best features or creating new, more discriminative ones. The model performance metric and confusion matrix provided a general view of its performance. High F1-score, accuracy, precision, recall, and minimal false positives and false negatives show the model can detect intrusions accurately and confidently. Training and validation plots showed that the model did not overfit, a common problem in deep learning. In conclusion, our research demonstrates the viability of deep learning for building economical, adaptive intrusion-prevention systems for smart devices. Our hybrid CNN-LSTM technique is a strong and effective mechanism to protect the dynamically evolving IoT framework against various forms of cyber-attacks.

6. Conclusion

The research article presents a deep learning-based solution for an adaptive intrusion-prevention system for smart devices. The architecture employs a CNN-LSTM hybrid framework that learns temporal and spatial features from network traffic and therefore performs exceptionally well at blocking and detecting all types of cyberattacks. The proposed approach has been evaluated on the "Smart Home Intrusion Detection Dataset" and some machine learning baselines. Experimental results indicate that our proposed deep learning model outperforms traditional models across accuracy, precision, recall, and F1-score. The three key contributions of this work are explained as follows. First, researchers propose a new hybrid deep learning architecture that leverages the strengths of CNNs and LSTMs for intrusion detection in smart device systems. Second, researchers demonstrated, through thorough experimentation on real data, the effectiveness of the proposed architecture. Lastly, Researchers have presented a comprehensive analysis of the results, demonstrating why deep learning is better suited to this task than traditional machine learning. The findings of this research have dire implications for IoT security. The more pervasive intelligent devices become, the higher the likelihood of a cyber-attack. The deep learning solution presented in this paper is a

capacity-limited approach to safeguard these devices from hackers. Its adaptable nature keeps it alert to a changing threat environment, making it a superior, natural security solution compared to conventional solutions.

6.1. Limitations

Impressive as this paper is, it is not quite faultless, and those weaknesses must be pointed out. First, the approach was tested on a single dataset, the "Smart Home Intrusion Detection Dataset." If this set is to be used in the question, it isn't necessarily comprehensive enough to generalise to all smart device setups. Other researchers on other datasets should validate the model to enable us to conclude its generalizability. Second, the new model was tested on a testbed. The model's performance in a real setup depends on parameters such as network delay and resource usage. In future development, the suggested system has to be implemented and evaluated using a real smart device setup. Third, the deep learning model is non-interpretable. The model will be very precise, but it is not always easy to understand right away why it made a choice. It is hard to trust the model and to find and resolve errors. Future work should consider how to make deep learning models interpretable for intrusion detection. Lastly, the said model can be computationally costly on certain resource-scarce smart devices. Even assuming that the model could be deployed on a more powerful gateway device, it may not always be feasible. More work is needed to develop more power- and clutter-hungry deep learning-based intrusion detection models that could be deployed on resource-scarce devices.

6.2. Future Scope

This paper does lay down some promising lines of future research. One is investigating how to use other types of deep architectures, e.g., transformers, for intrusion prevention in smart devices. The transformer has had great success in natural language processing and computer vision, so maybe it can do the same for network traffic. One area for future research is whether and how to train intrusion detection models using federated learning. Federated learning is a method by which multiple devices collaborate to train a model without sharing their raw data, and it can be used to address privacy concerns. It would be especially helpful when working with smart devices that have sensitive personal information. Also, further effort can be put into creating a more sophisticated intrusion response system that not only detects and halts attacks but also initiates the proper procedures to undo their effects. It could include removing the attacked systems, closing the loopholes, and returning to normal. Lastly, the adversarial robustness of deep learning intrusion detection models must be further studied. An adversarial attack is one in which the attacker subtly and imperceptibly modifies input data so that the model is forced to take an incorrect decision. Work in the future must be directed towards discovering means to make deep learning models adversarially robust.

Acknowledgement: The authors sincerely acknowledge the academic support and research facilities provided by Amity School of Engineering and Technology (Amity University), Vins Christian College of Engineering, SRM Institute of Science and Technology, and Queen's University Belfast. The collaborative environment and institutional support greatly contributed to the successful completion of this research.

Data Availability Statement: The dataset used in this study, which leverages deep learning for adaptive intrusion prevention in smart devices, is available from the corresponding authors upon reasonable request.

Funding Statement: No funding has been obtained to help prepare this manuscript and research work.

Conflicts of Interest Statement: No conflicts of interest have been declared by the authors. Citations and references are mentioned in the information used.

Ethics and Consent Statement: The consent was obtained from the organization and individual participants during data collection, and ethical approval and participant consent were received.

References

1. A. Khraisat and A. Alazab, "A critical review of intrusion detection systems in the Internet of Things: Techniques, deployment strategy, validation strategy, attacks, public datasets and challenges," *Cybersecurity*, vol. 4, no. 1, pp. 1–27, 2021.
2. F. Rahma, R. F. Rachmadi, B. A. Pratomo, and M. H. Purnomo, "Assessing the effectiveness of oversampling and undersampling techniques for intrusion detection on an imbalanced dataset," in *Proc. IEACon 2023—IEEE Industrial Electronics and Applications Conf.,* Penang, Malaysia, 2023.
3. I. F. Rodriguez, R. Megret, E. Acuna, J. L. Agosto-Rivera, and T. Giray, "Recognition of pollen-bearing bees from video using convolutional neural network," in *Proc. 2018 IEEE Winter Conf. on Applications of Computer Vision (WACV),* Lake Tahoe, Nevada, United States of America, 2018.

4. J. Bagherzadeh and H. Asil, “A review of various semi-supervised learning models with a deep learning and memory approach,” *Iranian Journal of Computer Science,* vol. 2, no. 6, pp. 65–80, 2019.
5. J. Huang and L. Zhang, “Network intrusion detection based on Dual-Encoder generative adversarial network,” in *Proc. EEI 2022—4th Int. Conf. on Electronics Engineering and Informatics,* Guiyang, China, 2022.
6. J. Huo, X. Min, T. Luo, F. Lv, Y. Feng, Q. Fan, D. Wang, D. Ma, and Q. Li, “Computed tomography-based 3D convolutional neural network deep learning model for predicting micropapillary or solid growth pattern of invasive lung adenocarcinoma,” *Radiologia Medica,* vol. 129, no. 3, pp. 776–784, 2024.
7. J. Ren and Z. Sun, “GHM-DenseNet intrusion detection method based on GAN,” in *Proc. 2022 IEEE 4th Int. Conf. on Civil Aviation Safety and Information Technology (ICCASIT),* Dali, China, 2022.
8. M. Shobana and S. Poonkuzhali, “A novel approach to detect IoT malware by system calls using deep learning techniques,” in *Proc. 2020 Int. Conf. on Innovative Trends in Information Technology (ICITIIT),* Kottayam, India, 2020.
9. N. Koroniotis, N. Moustafa, E. Sitnikova, and B. Turnbull, “Towards the development of realistic botnet dataset in the Internet of Things for network forensic analytics: Bot-IoT dataset,” *Future Generation Computer Systems*, vol. 100, no. 3, pp. 779–796, 2019.
10. N. Koroniotis, N. Moustafa, E. Sitnikova, and J. Slay, “Towards developing network forensic mechanism for botnet activities in the IoT based on machine learning techniques,” *Springer*, Cham, Switzerland, 2017.
11. R. Zuech, J. Hancock, and T. M. Khoshgoftaar, “Detecting web attacks in severely imbalanced network traffic data,” in *Proc. 2021 IEEE 22nd Int. Conf. on Information Reuse and Integration for Data Science (IRI),* San Diego, California, United States of America, 2021.
12. S. Greenwood, A. Perrin, and M. Duggan, “Social Media Update 2016,” *Pew Research Center,* 2016. Available: https://assets.pewresearch.org/wp-content/uploads/sites/14/2016/11/10132827/PI_2016.11.11_Social-Media-Update_FINAL.pdf [Accessed by 12/11/2024].
13. S. Moustakidis and P. Karlsson, “A novel feature extraction methodology using Siamese convolutional neural networks for intrusion detection,” *Cybersecurity*, vol. 3, no. 1, pp. 1–13, 2020.
14. Y. Imrana, Y. Xiang, L. Ali, A. Noor, K. Sarpong, and M. A. Abdullah, “CNN-GRU-FF: A double-layer feature fusion-based network intrusion detection system using convolutional neural network and gated recurrent units,” *Complex & Intelligent Systems,* vol. 10, no. 3, pp. 3353–3370, 2024.