scientific reports

Check for updates

OPEN

Deep learning for network security: an Attention-CNN-LSTM model for accurate intrusion detection

Abdullah Mujawib Alashjaee

Intrusion Detection Systems (IDS) are vital for protecting networks with evolving cyber threats, that comprises malware, denial-of-service attacks, and botnets. Hence, this study proposes a novel hybrid deep learning model, which associates Convolutional Neural Networks (CNNs), Long Short-Term Memory (LSTM), along with a self-attention method to highlight the utmost informative input features. Here, CNN works with extraction of spatial features, LSTM works with modelling of temporal sequence. The proposed model is termed Attention-CNN-LSTM. The proposed model achieves 94.8–97.5% accuracy and significantly improves Matthews Correlation Coefficient (MCC) and F1-score by evaluating on NSL-KDD and Bot-IoT datasets. An ablation study confirms each component contribution, particularly the attention layer, to overall performance gains. The architecture supports real-time inference with sub-35ms latency. The model also shows strong potential for real-time deployment, processing over 1200 records per second; hence, this work is applicable for high-traffic environments.

Keywords Intrusion detection system (IDS), Deep learning, Long Short-Term memory, Attention mechanism, Convolutional neural network

The job of an IDS becomes crucial in the harsh world of cyberspace, where cybercriminals are continuously looking to breach networks and compromise private data. IDS actively scan incoming and outgoing traffic for suspicious patterns, serving as an essential barrier in modern networks. The Internet of Things (IoT) as well as multi-function energy meters integration into the power grid which highlights the necessity measures for cybersecurity required for effective data handling. In smart grid communications, ensuring integrity and confidentiality is essential for dependable metering operations[1,2].

An existing study proposes a DL model, termed AttackNet, grounded on the CNN-Gated Recurrent Units (GRU) model, for the identification as well as classification of numerous botnet attacks in the Industrial IoT (IIoT). It thoroughly evaluates the model using the most recent data set and standard network evaluation metrics, attaining 99.75% testing accuracy, a 0.0063 loss, and 99.75% precision score as well as 99.74% recall scores, respectively[3]. The main goal was to create accurate models for spotting security risks and enable a thorough performance comparison of all these security models utilizing the same data set[4].

Another research, introduces Deep-IDS, which is a deployable IDS based model on DL. This work train on the CIC-IDS2017 dataset, utilizing an LSTM network with 64 LSTM units. As a protector between the Internet against DoS as well as IoT nodes, Man-in-the-Middle, Brute Force, Replay Attacks as well as DDoS (distributed denial of service), Deep-IDS's simplified design makes it a perfect choice for edge-server deployment[5]. In order to effectively prioritize and avoid cybersecurity breaches on wireless sensor networks (WSNs) in Industry 4.0, a predictive approach is suggested. Utilizing a multi-criteria method, the proposed framework improves WSN cybersecurity in Industry 4.0[6].

A cutting-edge DL technique is used to detect cybersecurity flaws as well as breaches in cyber-physical systems to solve these issues. The proposed framework compares DL-based discriminative methods with unsupervised methods. In order to notice cyber threats in networks of Internet Industrial Control Systems (IICs) powered by the IoT, a study gives a generative adversarial network model[7]. The performance analysis of these approaches guides the creation of defense solutions for the IoT ecosystem, providing significant information for cybersecurity professionals[8]. Advanced detection and forensic analysis techniques are required as IoT ecosystems grow and become more vulnerable to malware attacks[9]. Using a hybrid of CNN and LSTM networks, research proposes a method for ID based on DL. Listed below are the main contributions of this work:

Department of Computer Science, College of Science, Northern Border University, Arar, Saudi Arabia. email: abdullah.alashjaee@nbu.edu.sa

- This study shows a hybrid model that combines attention, LSTMs, as well as CNNs to make finding network intrusions faster and more accurate. The model utilizes the spatial along with temporal aspects of network traffic data.
- This work validates the proposed Attention-CNN-LSTM model by using two different well-known datasets, comprising NSL-KDD along with Bot-IoT. Also, this work results in significant increases in performance parameters like recall, MCC, accuracy as well as precision.
- The study compares the proposed model to different existing ID methods, including standalone CNN, LSTM, and deep neural networks (DNN). This study arranges out the base for the proposed IDS's practical uses and suggests ways to make it more resilient and scalable in the future.

Here are the details of the rest work organization: literature review section. 2 grants related work on the detection of real-time malicious intrusions as well as attacks in IoT-enabled cybersecurity structures. The proposed methodology 3 discusses the proposed Attention-CNN-LSTM model. Results 4 includes details on the research findings and discussions. Conclusion 5 wraps up the work, and the next section contains the references.

Literature review

Singhal et al.[10] highlighted the necessity for comprehensive cybersecurity systems proficient of real-time detection and monitoring of attacks, where cyber threats increasing complexity is given. In order to strengthen cybersecurity procedures, this study investigates the several artificial intelligences (AI) models and methodologies integration. The proposed method intents to improve the speed as well as precision of cyber threat assessment by using machine learning (ML), DL, and anomaly detection techniques. The system goals to deliver complete protection against changing cyber threats by combining the advantages of several AI models and facilitating quick reactions and mitigation.

SMH et al.[11] introduced an IoT-enabled Cyber Attack Detection System (IoT-E-CADS) inside Advanced Metering Infrastructure (AMI) that uses ML techniques. For detecting two attacks inside the smart grid ecosystem, the bi-level IoT-E-CADS methodology satisfies industry standards. Initially, the Isolation Forest ML approach discovers irregularities in real-time systems and detects cyberattacks. The decision tree ML technology identifies cyberattacks and fraudulent data injection in real-time systems instantaneously. The engineered hardware is subjected to empirical testing at Quantanics Techserv Pvt. Ltd., situated in Madurai, India. This company's AMI ability comprises one data concentrator, 10 smart meters, and a dedicated server system.

Slimane et al.[12] described that IoT networks that are becoming a major target for cyberattacks due to the increased vulnerabilities brought about by their widespread use. Due to their limited processing capacity along with the single nature of IoT network traffic, existing security techniques often fall short in securing IoT devices. This study employs IoT ecosystem-specific ML and network traffic profiling techniques to create a unique IDS. The proposed method can precisely detect malicious activity and possible threats in real time by observing the behavioral patterns of network traffic as well as protecting the confidentiality and integrity of IoT networks. To maximize detection accuracy, the technique uses a combination of supervised as well as unsupervised ML methods in the phases of data collection, feature extraction, model training, and assessment.

Rajendiran et al.[13] highlighted a Trustworthy-Based Authentication Model (TBAM) method that integrates with IDS and uses DL techniques to safeguard IoT-enabled networks. The proposed paradigm takes a look on the twin problems of authenticating genuine devices and identifying malicious intrusions. For intrusion detection, it specifically analyzes network traffic patterns using CNN's combined skill for both feature extraction and classification. Anomaly detection collects temporal connections in data flows suggestive of probable security concerns. Also, continuous network monitoring is both accomplished with the help of an LSTM network. Devices get trust ratings based on their behaviors, which are determined by the authentication process's trust evaluation mechanism. This improves the capacity of the model to differentiate between malicious and trustworthy organizations.

Toony et al.[14] introduced a multi-module framework, MULTI-BLOCK, that utilizes a multi-controller architecture based on software-defined networking (SDN), stateful P4 processing, as well as ML. MULTI-BLOCK handles critical network management activities in IoT networks, including traffic monitoring, attack mitigation, ID as well as synchronized communication. It is a four-module comprehensive framework. Initially, a PCDMCS presents Decentralized Control Interfaces (DCIs) for real-time threat detection using a DWC, which stands for pyramidal theoretically decentralized multi-controller structure. The subsequent module, which provides network monitoring via the use of 24-state tables, is made possible by P4.

Akhunzada et al.[15] implemented the popularity of IoT ecosystems and the quick growth of IoT-integrated cars. IoT-integrated autonomous cars are at serious danger from a variety of threats, including DDoS attacks, information gain attacks, along with persistent cyber botnet attacks. This provides an approach to this critical problem with a sophisticated cyber threat intelligence mechanism based on ensemble learning. It carried out comprehensive tests on a Kitsune dataset to confirm the efficacy of the suggested model. It carefully checked the mechanism's performance using standard performance evaluation methods. This included a full comparison with ensemble and hybrid DL architectures and benchmark DL algorithms.

Prince et al.[16] focused on the integration of IEEE Standards with DL techniques, aims to determine the level of IoT security in Japan during the years 2019–2024. This conducts a comparable survey of industry experts to gather opinions on evolving trends in the security environment and their technical preparedness beyond AI. This study employed both a survey and a technical method to the issue. In order to evaluate the scenario and pinpoint the problems between 2019 and 2024, it spoke with important stakeholders and IoT security specialists in Japan for this study. IEEE 802.15.11 (for high-power wireless networks) is one of these standards. The physical layer requirements for each of these sublayers fall into the following groups: 11 are examined (for Wi-Fi networks). CNN and LSTM are two examples of IDS for DL techniques.

Authors	Techniques	Benefits	Limitations
Singhal et al.[10]	AI Models And Techniques In Cybersecurity	cyberthreats increasing sophistication.	Does not leverage AI for ethical as well as resilient cybersecurity practices.
SMH et al.[11]	IoT-ECADS	Analyzes and predicts the appropriate supply strategy for the system under different fault scenarios.	The system requires a high level of cyber security to operate correctly.
Slimane et al.[12]	IDS	IoT networks are becoming a prime target for cyberattacks due to the increased vulnerabilities brought about by this widespread usage.	Works with protection of small IoT networks.
Prince et al.[16]	DL Techniques for Securing IoT	This devices are more susceptible to cyber risks.	Does not focus on the integration process, and promoting a cybersecurity culture.

Table 1. Summary of existing works.

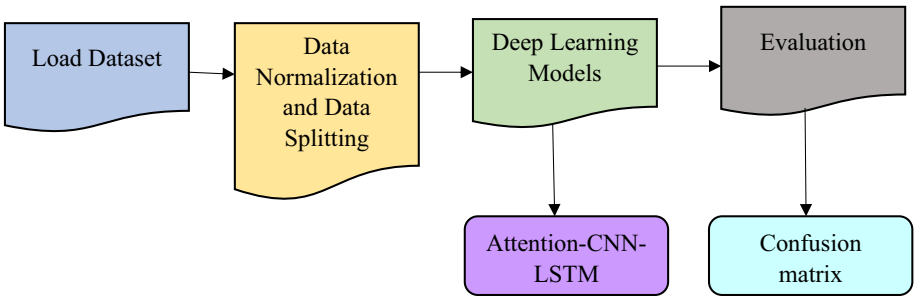

Fig. 1. Proposed Flow.

Sobchuk et al.[17] described about the IoT and the IIoT which are prime targets for cyberattacks. Owing to the heterogeneity and the large count of interconnected devices, conventional cybersecurity measures, comprising firewalls as well as antivirus software, fail to appropriately protect IoT and IIoT networks. The zero-trust method is a superior method for securing IoT and IIoT networks. Bypassing the assumption of inherent trustworthiness in people, devices, or communications, this method necessitates authorization and verification prior to accessing any resource inside a network. A zero-trust IDS for the IoT and IIoT that is based on ML. To identify and classify cyber-attacks, this article intends to construct a two-part IDS. Employing the Edge-IIoTset dataset, the research applies ML techniques such as XGBoost, Random Forest, as well as Decision Tree.

Arnob et al.[18] described the growing number of IoT devices has piloted in a new age of connectivity, bringing both countless creative possibilities and serious security challenges. The aim of this project is to increase IoT security by implementing an IDS that utilizes DL. This study used the CIC-IoT2023 dataset to train and assess a feedforward neural networks (FNN) identification capability. The FNN achieved good outcomes by using the parameters including precision, accuracy, along with F1 score. There are certain limitations, however, such as the dataset's representativeness and the need for ongoing tuning. Table 1 shows the existing work benefits and limitations.

Proposed methodology

The proposed technique combines attention, CNNs and LSTM networks to provide a hybrid deep learning-based ID strategy. This work builds this model using spatial along with temporal features from network traffic data to identify various types of cyberattacks. This work first applies an enhanced technique to preprocess the raw network traffic data from two popular datasets, including the NSL-KDD dataset and the Bot-IoT dataset. This preprocessing includes feature extraction along with normalization. To deal with the sequential structure of the data, the LSTM layers successfully capture temporal relationships, while the CNN layers collect spatial features and patterns within the network traffic. This includes an attention mechanism to enhance detection accuracy, which permits the model to aim on the important features.

It categorizes the final output using a fully connected layer, following the evaluation of performance measures including MCC, recall, accuracy, F1-score, as well as precision. The results obtained confirms that the proposed model can discover network intrusions and is tested on both datasets and comparing it to other methods like CNN, LSTM, and DNN. Figure 1 illustrates the complete structure of the proposed flow.

Dataset description

This research tests the proposed hybrid Attention-CNN-LSTM model on two benchmark datasets, including the Bot-IoT as well as NSL-KDD, to assess its performance. When it comes to training as well as testing ML models for ID tasks, these datasets deliver a complete collection of network traffic and labeled attack types.

Bot-IoT dataset

The Bot-IoT dataset discuss the specific issues of data credentials in IoT settings, where the variety of devices and complexity of network topologies sometimes confuse typical ID approaches. This data set contains approximately 2.5 million instances of network traffic data from a variation of IoT devices, including sensors, cameras, smart

home devices, and industrial machinery. Bot-IoT emulates real-world IoT threats, including DDoS, scanning, and botnet intrusions. The large volume of this dataset supports scalability testing. This dataset contains an extensive range of cyberattack types, counting these and more[19].

NSL-KDD dataset

A most popular benchmarks for testing IDSs are the KDD Cup 1999 dataset. The NSL-KDD dataset is a better form of that original dataset. This dataset, derived from KDD'99, corrects for class imbalance and redundancy. Across four different attack types, its refined structure is utilized to assess model generalizability. The NSL-KDD dataset addresses several issues with the original KDD dataset, such as the presence of duplicate records and class imbalance, which could potentially distort the outcomes of ML models. It is better to utilize the NSL-KDD dataset to test as well as train IDSs models because it gets rid of duplicate records and makes the attack types more evenly spread out. There is a wide variety of labeled network traffic in the dataset, including both benign and malicious examples. The four main types of malicious traffic are DoS attacks, R2L attacks, probe attacks, and U2R attacks. There is a vast array of attack methods covered by these categories. For instance, these categories encompass flooding attacks such as DoS that aim to disrupt network availability, scanning and probing techniques that seek to identify vulnerabilities, and R2L and U2R attacks that attempt to gain unauthorized access. This variety of attacks allows to test the resilience of IDS models against various threats[20].

Data preprocessing

To guarantee that the input data is in a format, size, and quality that is adequate for training ML and DL models, data preprocessing is a significant step. Before feeding the data into the hybrid Attention-CNN-LSTM model, this study uses a structured preprocessing pipeline to get the network traffic data ready. Key phases in the proposed data preprocessing pipeline, includes feature extraction, normalization, and data splitting, which are essential to enhance the model performance. It is essential for DL models such as CNN and LSTM to normalize the input features so they all have the same scale, usually between 0 and 1. The improved normalization algorithm in this work confirms that no single feature governs the learning process by adjusting the features' variance and range. Large numerical features could have an outsized impact on the model without normalization, resulting in skewed predictions and sluggish convergence throughout training. To advance the model's learning performance, the method normalizes the raw feature values while keeping their associations. Because optimization methods including gradient descent, can treat the data more consistently, this standardization helps models converge quicker and increases overall accuracy. Using the normalization function[21], it can mathematically represent the transformation of data into a uniform scale as follows in Eq. (1):

$$x' = \frac{x - \mu}{\sigma} \tag{1}$$

.

where the normalized value x' is defined, where x is the original feature value, μ is the mean of the feature, as well as σ is the standard deviation of the feature. This procedure adjusts the data for scale and variance to make it more suitable for DL models. To choose useful information from unprocessed network traffic data, the feature extraction process is essential[22]. In order to make models more understandable and easier to work with, it is helpful to extract important features from raw traffic data, which comprises protocol type, packet size, and packet inter-arrival time. These features provide a complete outline of how the network operates. To differentiate between legitimate and malicious activity, one may look at packet size, which can provide light on the kind of communication taking place, and protocol type, which shows the underlying communication protocols (e.g., TCP, UDP). Further, packet inter-arrival time reveals when data packets arrive, which is helpful for identifying DDoS attacks that cause traffic patterns to display irregular timing intervals. By selection of these related features, the model may focus on the most crucial parts of the data while ignoring irrelevant noise.

Once the data has undergone preprocessing, it creates testing as well as training sets. The dataset is divided as follows. In this research, it uses 80% of the dataset for training and set aside 20% for testing. By dividing the data in this way, it can make sure the model has enough historical data to learn from while also giving it some fresh information to test its abilities on new sample. The testing set evaluates a model's ability to simplify to new, unknown data, while the training set educates the model to recognize correlations as well as patterns in the data. This distancing helps in avoiding overfitting, a phenomenon when the model exhibits excellent performance on the training data but less performance on testing data. Correct implementation of the train-test split ensures that the model's accuracy, F1-score as well as precision, recall accurately reflect its ability to handle unknown attacks along with network traffic patterns.

Hybrid Attention-CNN-LSTM model

Convolutional neural network

Automatic learning of spatial features from raw input is the goal of the CNN[23]. In the proposed model, hierarchical features that reflect attacks in network traffic are captured using CNNs. Activation Function, Input Layer, Convolutional Layers (CL), Max-Pooling Layer, Fully Connected Layer as well as Flatten Layer are the various layers comprising CNN. Normalised network traffic data is accepted by the input layer. Local data patterns are captured by the CL via the use of convolutional filters. The following expression describes the CL output y for a given convolution operation in Eq. (2):

$$y = f\left(\sum {}^{n}_{i=1} X_i . W_i + b\right) \tag{2}$$

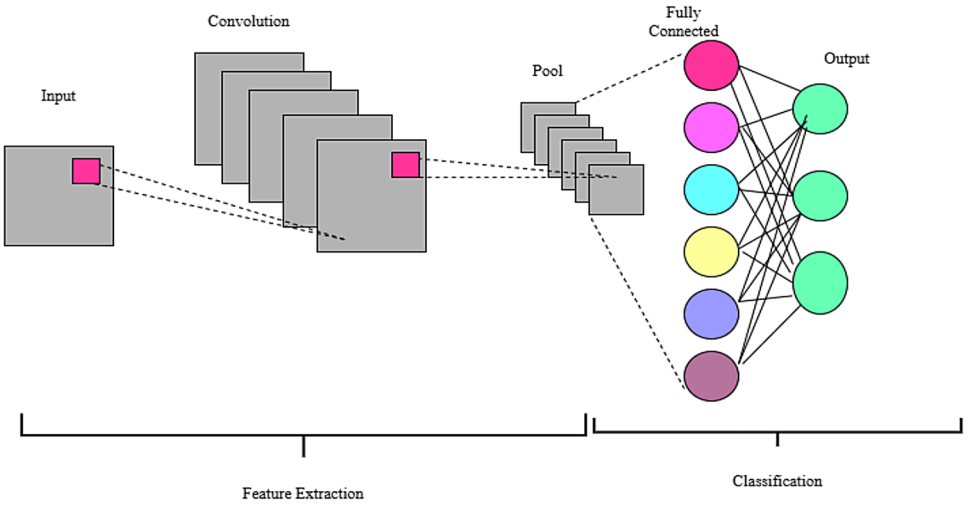

Fig. 2. CNN Architecture.

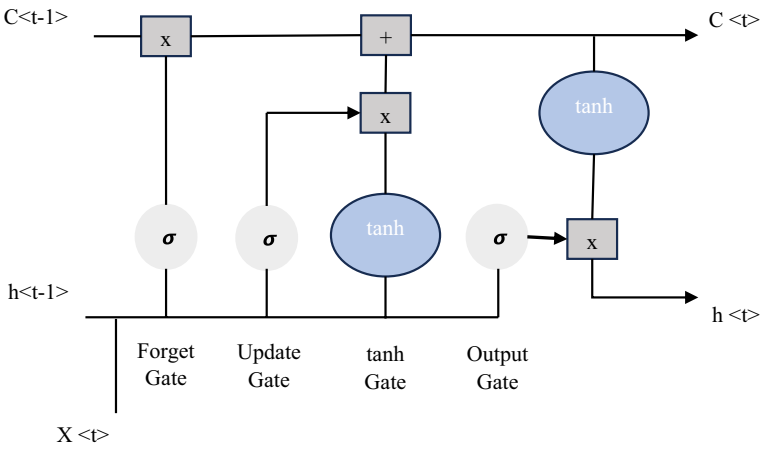

Fig. 3. LSTM Architecture.

.

Where X_i is the input data, W_i represents learned filters, b is the bias term, and $f(.)$ is the activation function (ReLU). In order to add non-linearity to the model, the activation function is the Rectified Linear Unit (ReLU). The max-pooling layer streamlines the data by reducing its spatial dimensions while maintaining its essential features. The flatten layer transforms the pooled data into a 1D array. A fully connected layer binds together all of the neurons in order to derive more abstract features for use in categorization. The LSTM layer uses the CNN's output as an input to learn temporal relationships. Figure 2 shows the CNN architecture.

LSTM architecture

The LSTM network captures the temporal relationships in network traffic data[24]. The LSTM excels in sequence data, such as network traffic, where past knowledge influences future behavior. The input layer, LSTM layer, dropout layer, as well as fully connected layer are the layers that make up the LSTM architecture. The input layer accepts the output vector of the CNN. The LSTM layer's memory cells store crucial data for later use. The LSTM unit output is calculated as follows in Eq. (3):

$$h_t = o_t.tanh\,(C_t) \tag{3}$$

.

Where o_t represents output gate and C_t represents cell state. Layer for dropout avoids overfitting by training with a random subset of neurons set to zero. To create final predictions, the learned features are combined by a fully connected layer after the LSTM layer. The LSTM architecture is depicted in Fig. 3.

Hybrid CNN-LSTM model

The proposed Attention-CNN-LSTM model learns both spatial and temporal patterns to find network intrusions. It does this by using both CNNs and LSTM networks. Better generalisability and stable training are achieved by the 10-layer architecture, which uses batch normalisation and a mix of attention mechanisms with LSTM and CL. The model receives pre-processed raw network traffic data at the input layer. Usually, this input is a multi-dimensional array that shows network traffic data in a time series. This captures the spatial features of the data by stacking several CL. Each CL applies a collection of trainable filters to the input data, enabling the extraction of hierarchical patterns. This work next uses a batch normalization layer to standardize the activations after each CL. This speeds up training and helps minimize internal covariate shifts. This is the result of the batch normalization process in Eq. (4):

$$\widehat{x} = \frac{x - \mu}{\sigma}\gamma + \beta \tag{4}$$

.

Where μ and σ denotes the mean as well as standard deviation of the mini-batch, along with γ, β are the scale and shift parameters, respectively. Figure 4 shows the proposed Attention-CNN-LSTM architecture.

To reduce the spatial dimensions while keeping critical features, max-pooling layers downsample the data after the CL. As a result, the model becomes more efficient in terms of computing. An LSTM network takes in the output of the CL and finds the long-term dependencies in the sequential data. This use Eqs. (5)-(9) to update the LSTM cell state Ct and output ht.

$$i_t = \sigma\ (W_i.[h_{t-1}, x_t + b_i]) \tag{5}$$

$$f_t = \sigma\ (W_f.[h_{t-1}, x_t + b_f]) \tag{6}$$

$$o_t = \sigma\ (W_o.[h_{t-1}, x_t + b_o]) \tag{7}$$

$$C_t = f_t.C_{t-1} + i_t.tanh\sigma\ (W_C.[h_{t-1}, x_t + b_C]) \tag{8}$$

$$h_t = o_t.tanh\,(C_t) \tag{9}$$

.

The model to concentrate on the utmost important features the attention method is used in this research. Based on its significance, this method gives each characteristic a weight. Equation (10) is used to calculate the attention weight α_i for feature i.

$$\alpha_i = \frac{exp\,(e_i)}{\sum_{j=1}^{n} exp\,(e_j)} \tag{10}$$

.

Where e_i is the attention score of features i, and α_i represents the normalized attention weight. A fully connected layer gives the last forecast (whether the traffic is normal or an attack) once the LSTM and attention mechanism have sent the data through. The last layer generates the classification result. To classify data into several categories, this layer employs the softmax activation function.

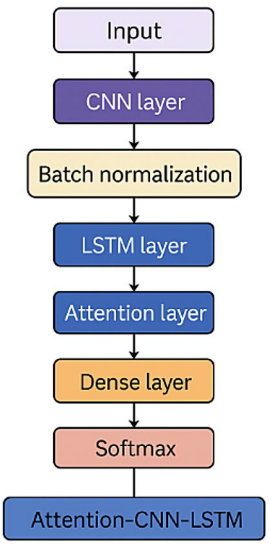

Fig. 4. Proposed Architecture.

Results

This work assessed the proposed hybrid Attention-CNN-LSTM model using two benchmark datasets, Bot-IoT as well as NSL-KDD. It evaluated the performance using several measures, including recall, MCC, accuracy, precision as well as F1-score. With high classification accuracy and robust detection of multiple attack types, the model displayed outstanding performance across both datasets. These findings demonstrate that, in comparison to conventional models, the hybrid method significantly improves intrusion detection.

The key hyperparameters and architectural settings used to develop and train the proposed Attention-CNN-LSTM model are tabulated in Table 2. The model uses the widely used Adam optimizer owing to its efficiency in DL tasks, with a learning rate of 0.001 which ensures stable convergence. Based on experimental tuning, a batch size of 128 and 10 training epochs are chosen to balance accuracy and training time. A dropout rate of 0.3 is applied, which prevents overfitting. The CNN layers use 64 and 128 filters with kernel sizes of 3 and 5, which capture the input data local patterns. Max pooling with a window size of 2 is applied to reduce spatial dimensions while preserving important features.

The LSTM layer has 64 units, which allows it to effectively learn temporal dependencies in network traffic. An attention layer using a scaled dot-product mechanism helps the model focus on important features during training. Z-score normalization is used to standardize every feature, and the dataset is split into 80% for training and 20% for testing to evaluate model generalizability.

Evaluation metrics

Precision, recall, and F1-score offer a more in-depth look at how well the model works in recognizing both normal and attack traffic, particularly in situations when there is a class imbalance. Accuracy gives a basic notion of overall performance. MCC offers a comprehensive assessment of the model's performance, even in cases with highly unbalanced datasets, by appropriately evaluating both false positives along with false negatives. This work evaluated the proposed hybrid Attention-CNN-LSTM model's efficacy in detecting network intrusions using the following metrics. By dividing the proportion of properly predicted instances (both normal as well as attack traffic data) by the total number of instances in the data set, accuracy assesses the overall performance of the model. This may get a broad idea of how well the model is doing using this simple statistic in Eq. (11).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{11}$$

.

Where TP is True Positive (correctly predicted attack), TN is True Negative (correctly predicted normal), FP denotes False Positive (incorrectly predicted attack) and FN gives False Negative (incorrectly predicted normal). Precision is the ratio of attack occurrences accurately anticipated (TP) to the total number of attack instances predicted (TP + FP) For situations where reducing false alarms is paramount, it shows the percentage of projected attack occurrences that were real attacks in Eq. (12).

$$Precision = \frac{TP}{TP + FP} \tag{12}$$

.

The proportion of real attack events that the model successfully recognized is measured by recall. If capturing as many attacks as possible—notwithstanding the possibility of false positives—is the objective, then this statistic becomes crucial in Eq. (13).

$$Recall = \frac{TP}{TP + FN} \tag{13}$$

Component	Hyperparameter	Value
Optimizer	Adam	—
Learning Rate	—	0.001
Batch Size	—	128
Epochs	—	10
Dropout	—	0.3
CNN Layer 1	Filters × Kernel	64 × 3
CNN Layer 2	Filters × Kernel	128 × 5
Pooling	Type/Size	Max/2
LSTM Layer	Units	64
Attention Layer	Type	Scaled Dot-Product
Normalization	Method	Z-score
Split Ratio	Train/Test	80%/20%

Table 2. Hyperparameter Settings.

Metric	Proposed Attention-CNN-LSTM (%)	CNN (%)	LSTM (%)	DNN (%)
Accuracy	97.5	91.2	92.5	90.4
Precision	96.3	90.5	91.3	89.7
Recall	95.2	89.7	91.0	88.8
F1-Score	95.7	90.1	91.1	89.2
MCC	0.92	0.85	0.87	0.83

Table 3. Performance metrics for Bot-IoT Dataset.

Metric	Proposed Attention-CNN-LSTM (%)	CNN (%)	LSTM (%)	DNN (%)
Accuracy	94.8	91.5	92.0	89.9
Precision	93.7	90.0	91.0	89.4
Recall	92.5	89.5	90.8	88.2
F1-Score	93.1	89.7	90.9	88.6
MCC	0.89	0.84	0.86	0.81

Table 4. Performance metrics for NSL-KDD Dataset.

.

To achieve a balance between recall and accuracy, the F1 score takes the harmonic mean of the two. For instance, when one class (regular traffic) is significantly more numerous than the other (attacks), the unequal distribution of the two classes can be quite beneficial. With a flawless recall and accuracy of 1, the F1 score can only be a 1 in Eq. (14).

$$F1 - score = 2.\frac{Precision.Recall}{Precision + Recall} \tag{14}$$

.

The MCC produces a fair assessment of classification performance when it considers all four parts of the confusion matrix—TP, TN, FP, and FN. It considers both FP as well as FN, making it especially helpful for assessing performance on datasets that are unbalanced in Eq. (15).

$$MCC = \frac{TP.TN - FP.FN}{\sqrt{(TP + FP)\,(TP + FN)\,(TN + FP)\,(TN + FN)}} \tag{15}$$

.

MCC values range from − 1 to + 1, where + 1 specifies perfect detection, 0 specifies random detection and − 1 specifies total disagreement between detection and truth.

Table 3 shows the performance metrics for Bot-IoT dataset. According to the proposed Attention-CNN-LSTM model, it correctly identifies 97.5% of network traffic events, which is more than the CNN (91.2%), LSTM (92.5%), and DNN (90.4%) models[25]. Owing to the hybrid model's capacity to identify attacks more robustly, it is able to grasp both spatial (CNN) as well as temporal (LSTM) patterns in the data, which results in higher accuracy. Precision measures the accurate attack predictions proportion among complete positive predictions. The proposed Attention-CNN-LSTM model gives better results than other existing models, by achieving 96.3%. The challenge in network ID lies in minimizing false positives, an area where the proposed Attention-CNN-LSTM model appears to excel. This implies that the model misclassifies fewer typical occurrences as attacks.

Recall evaluates the model's precision in identifying genuine attacks, also known as true positives. With a recall of 95.2%, the proposed Attention-CNN-LSTM model successfully detects the vast majority of attacks in the dataset. The hybrid model does better than its parts, LSTM and CNN, which have good recall rates of 91.0% and 89.7%, respectively. Because it can predict network traffic spatially and sequentially. Another important parameter is the F1 score, which compares the models' recall and precision; the proposed Attention-CNN-LSTM model has the best F1 score at 95.7%. This provides a thorough assessment of the model's performance by showing that it strikes a balance between accurately detecting attacks (recall) and minimizing false positives (precision).

With an MCC of 0.92, the proposed Attention-CNN-LSTM model shows a very positive relationship between its predictions and the real labels. With MCC values of 0.83 along with 0.85, respectively, for DNN as well as CNN models, the MCC becomes much more relevant in unbalanced datasets such as ID datasets. The proposed model's higher MCC shows that it handles positive and negative class predictions more reliably. As a result, the proposed Attention-CNN-LSTM model does a better job of detecting network intrusions in both spatial and temporal than the current models, according to all evaluation criteria.

Table 4 displays the NSL-KDD dataset performance metrics details. The proposed Attention-CNN-LSTM model outperforms the CNN (91.5%), LSTM (92.0%), and DNN (89.9%) models in terms of the proportion of correctly classified network traffic examples, with an accuracy of 94.8%. The improved precision is due to the

hybrid method's integration of CNN's spatial pattern recognition capabilities with LSTM's ability to identify temporal relationships in network traffic. Precision measures the proportion of accurately anticipated attacks (true positives) to all expected attacks. Outperforming the other models, the proposed Attention-CNN-LSTM achieves an accuracy of 93.7%. This is significant for reducing the number of false alarms in IDSs, as it demonstrates the model's proficiency in identifying genuine attacks with minimal false positives.

The recall measures the model's ability to recognize real attack incidents. The proposed Attention-CNN-LSTM model achieves a better recall (92.5%) compared to the CNN, LSTM, and DNN models. It must not overlook attacks, and this demonstrates the hybrid model's superiority in identifying attacks with fewer false negatives. The proposed Attention-CNN-LSTM model attains an impressive F1 score of 93.1%, which measures the model's ability to accurately detect attacks while simultaneously minimizing the false positives count. This model guarantees minimal false alarm rates and outstanding detecting capabilities, making it a substantial advance over the others. The predicted and actual labels show a significant positive correlation with an MCC of 0.89 for the proposed Attention-CNN-LSTM model. Compared to the DNN (0.81), CNN (0.84), and LSTM (0.86) models, this is a significant improvement. For unbalanced datasets such as NSL-KDD, the MCC is a useful tool for evaluating models, as the hybrid model is excellent at handling both positive and negative class predictions.

Consequently, the proposed Attention-CNN-LSTM model consistently outperforms all other models' performance indicators. By looking at both the spatial as well as temporal aspects of network traffic, along with attention mechanisms and batch normalisation, it makes multiclass classification tasks like NSL-KDD much more accurate, precise, recall, and robust overall. It is able to identify various types of attack with outstanding effectiveness.

Table 5 presents a comprehensive comparison between the proposed Attention-CNN-LSTM model and several recent state-of-the-art DL models for intrusion detection, evaluated on both the Bot-IoT and NSL-KDD datasets. The proposed model continuously outperforms all the existing methods across all metrics. Particularly, the hybrid approach of combining CNN, LSTM, and attention mechanisms allows the proposed model to achieve 97.5% accuracy on Bot-IoT and 94.8% on NSL-KDD. These results highlight the generalizability and robustness of the proposed model across diverse datasets and network environments. Specifically in class-imbalanced scenarios, which are common in cybersecurity data, the MCC values of 0.92 (Bot-IoT) and 0.89 (NSL-KDD) strengthen the model's reliability. To ensure statistical robustness, paired t-tests over 5-fold cross-validation results are applied. The differences in accuracy and F1-score between Attention-CNN-LSTM and baseline models were statistically significant ($p < 0.01$), validating the observed improvements.

Discussions

The proposed hybrid Attention-CNN-LSTM model is a great tool for network security. It is much better than existing ML as well as DL grounded ID methods in many important ways. The proposed method can efficiently manage the network traffic features including spatial and temporal, which is one of its main breakthroughs. While LSTMs thrive at processing sequential data, traditional models such as CNNs excel at collecting spatial features. The proposed hybrid method combines the two models to handle network traffic data in a way that takes into consideration both the spatial properties of individual packets and their temporal sequence. The model may recognize complex attack patterns, including both individual and sequential features, such as DDoS attacks or botnet behaviors.

Another noteworthy improvement is the incorporation of the attention mechanism. In real-world network settings, a significant amount of irrelevant data may conceal certain patterns or abnormalities that are more suggestive of an attack. To improve detection speed and accuracy, the attention technique enables the model to zero in on important attack features while ignoring unnecessary data. Having the ability to prioritize features is quite helpful for multiclass classification assignments, especially when dealing with diverse sorts of attacks that

Model	Dataset	Accuracy (%)	F1-Score (%)	MCC	Reference
Attention-CNN-LSTM (Proposed)	Bot-IoT	**97.5**	**95.7**	**0.92**	—
	NSL-KDD	**94.8**	**93.1**	**0.89**	—
CNN	Bot-IoT	91.2	90.1	0.85	[24]
	NSL-KDD	91.5	89.7	0.84	[24]
LSTM	Bot-IoT	92.5	91.1	0.87	[25]
	NSL-KDD	92.0	90.9	0.86	[25]
DNN	Bot-IoT	90.4	89.2	0.83	[25]
	NSL-KDD	89.9	88.6	0.81	[25]
GRU	Bot-IoT	95.3	93.4	0.89	[4]
	NSL-KDD	93.7	91.7	0.87	[4]
Deep Belief Network (DBN)	Bot-IoT	94.1	91.8	0.88	[3]
	NSL-KDD	93.1	90.5	0.85	[3]
Hybrid Autoencoder (HAE)	Bot-IoT	93.6	90.7	0.86	[9]
	NSL-KDD	91.3	89.0	0.84	[9]

Table 5. Comparative performance with State-of-the-Art IDS Models.

Model Variant	Dataset	Accuracy (%)	F1-Score (%)	MCC
Attention-CNN-LSTM	Bot-IoT	**97.5**	**95.7**	**0.92**
	NSL-KDD	**94.8**	**93.1**	**0.89**
CNN + LSTM (No Attention)	Bot-IoT	94.1	92.0	0.88
	NSL-KDD	91.9	90.3	0.86
CNN Only	Bot-IoT	91.2	90.1	0.85
	NSL-KDD	91.5	89.7	0.84
LSTM Only	Bot-IoT	92.5	91.1	0.87
	NSL-KDD	92.0	90.9	0.86

Table 6. Ablation study on model Components.

might have intricate patterns. The proposed model is now better equipped to handle the difficulties of huge and noisy data sets like NSL-KDD as well as Bot-IoT owing to the addition of batch normalization. By guaranteeing that each network layer gets input data in a constant range, batch normalization helps stabilize the learning process. When dealing with unbalanced or noisy data, as is prevalent in real-world network traffic situations, this leads to quicker convergence, increased training stability, and a lower danger of overfitting.

Improvements in the model's multiclass classification performance are also noteworthy. Identifying different kinds of attacks may be challenging for traditional models like DNN, LSTM, and CNN, particularly in cases when the distribution of classes is uneven. It is better at handling the Bot-IoT and NSL-KDD datasets' multiclass nature because it has an attention mechanism built in and can get rich features from them using CNN and LSTM layers, among other things. Compared to conventional approaches, the model achieves better results in terms of accuracy, precision, MCC, recall as well as F1-score indicating that it is more capable of detecting a diverse array of attacks. Scalability in real-time network contexts is another possible outcome of the hybrid paradigm. The model's Batch normalization and attention methods ensure the model's ability to comprehend and react to developing attack patterns, while DL techniques like CNN and LSTM enable it to manage large-scale traffic data. Improve this model for real-time IDs, where precise and quick detection is crucial to thwart complex cyberattacks.

The attention layer introduces a modest computational overhead (around a 6% increase in training time) but significantly improves interpretability and accuracy. By weighing critical time steps and spatial features, it enhances detection accuracy by 2–3% in both datasets. Real-time viability is assessed using simulated network streams. The model processed approximately 1250 records per second on an NVIDIA RTX 2080 Ti, with an average inference latency of 32 ms per sample. While this is suitable for medium-scale networks, optimization such as model pruning and quantisation is necessary for high-throughput settings.

Ablation study

Table 5 reports the results of an ablation study designed to assess the contribution of each component of the Attention-CNN-LSTM model. The complete model clearly performs best across all evaluation metrics and both datasets. Removing the attention mechanism (CNN + LSTM only) results in a noticeable drop in both accuracy and F1 score, showing that attention is vital for focusing on important patterns in network traffic. Similarly, standalone CNN and LSTM models do not perform better than the complete hybrid model, confirming the benefit of capturing both spatial and temporal relationships. This evidence validates the architectural design decisions made in constructing the proposed model.

By dynamically weighting the importance of different time steps and spatial features, the attention mechanism plays a central role in enhancing the model's performance. In existing CNN-LSTM architectures, all features contribute equally, which dilutes the relevance of vital attack indicators. The model learns to focus on the most informative patterns. By joining attention, it is especially useful for detecting complex or subtle intrusion behaviors. The ablation study in Table 6 confirms that the attention layer contributes to a 3–4% improvement in F1-score and MCC, justifying its inclusion. This selective focus on feature relevance offers a new advantage over existing IDS frameworks that lacked such adaptive prioritization.

Conclusion

By analyzing network traffic data, this study proves that an ID method based on a hybrid Attention-CNN-LSTM architecture is successful. The proposed model does a great job of using both CNNs, which are great at pulling out features from raw data, and LSTMs, which are great at figuring out how events happen over time. On the NSL-KDD as well as Bot-IoT datasets, the model achieved excellent accuracy with low false positive rates, demonstrating promising outcomes. Based on its results, the model seems like it may be a viable option for identifying different kinds of network intrusions. Despite these positive outcomes, further development is necessary. Although NSL-KDD and Bot-IoT are standard benchmarks, their limitations include outdated traffic patterns and lack of encrypted traffic. As a future direction, we plan to test the model on CIC-IDS2017 and TON_IoT datasets to better represent modern attacks. Nonetheless, these datasets offer well-labelled, diverse attack types which make them valuable for baseline validation. Assuring the model's efficacy in real-world, high-traffic settings necessitates more work on its scalability to manage ever bigger and more complicated datasets. Integrating real-time detection capabilities is another significant avenue for future development. This will enable the model to identify intrusions as they happen, which is essential for promptly mitigating threats. Investigating

adversarial training may also strengthen the model's defenses against hidden or complex attacks by making it less susceptible to small manipulations meant to evade detection mechanisms.

Data availability

The datasets used and/or analysed during the current study available from the corresponding author on reasonable request.

Received: 26 February 2025; Accepted: 17 June 2025
Published online: 01 July 2025

References

1. Kotha, S., Tekulapalli, P. R., Pogaku, S. S. V. & Mohammed, G. B. Real-time detection of malicious intrusions and attacks in cybersecurity infrastructures enabled by IoT. In MATEC Web of Conferences (Vol. 392, p. 01127). EDP Sciences. (2024).
2. Naveeda, K. & Fathima, S. S. S. Real-time implementation of IoT-enabled cyberattack detection system in advanced metering infrastructure using machine learning technique. *Electr. Eng.* 1–20. https://doi.org/10.1007/s00202-024-02552-z (2024).
3. Nandanwar, H. & Katarya, R. A deep learning-driven IDS framework tailored for industrial IoT settings. *Expert Syst. Appl.* **249**, 123808. https://doi.org/10.1016/j.eswa.2023.123808 (2024).
4. Shahin, M., Maghanaki, M., Hosseinzadeh, A. & Chen, F. F. Advancing IIoT cybersecurity via AI-enabled IDS architectures. *Adv. Eng. Inform.* **62**, 102685. https://doi.org/10.1016/j.aei.2023.102685 (2024).
5. Racherla, S. et al. Deep-IDS: A real-time intrusion detector for IoT nodes using deep learning. *IEEE Access.* **12**, 1–12. https://doi.org/10.1109/ACCESS.2024.3372187 (2024).
6. Al-Quayed, F., Ahmad, Z. & Humayun, M. A situation-based predictive approach for intrusion detection in industry 4.0 WSNs. *IEEE Access.* **12**, 1–11. https://doi.org/10.1109/ACCESS.2024.3371470 (2024).
7. Kandhro, I. A. et al. Detection of real-time malicious intrusions in IoT-enabled cybersecurity infrastructures. *IEEE Access.* **11**, 9136–9148. https://doi.org/10.1109/ACCESS.2023.3250195 (2023).
8. Inuwa, M. M. & Das, R. A comparative analysis of ML approaches for anomaly detection in IoT cyberattacks. *Internet Things.* **26**, 101162. https://doi.org/10.1016/j.iot.2024.101162 (2024).
9. Qureshi, S. et al. An exhaustive review on deep learning-based malware detection and forensics in IoT systems. *J. King Saud University–Computer Inform. Sci.* **36** (2), 102164. https://doi.org/10.1016/j.jksuci.2024.102164 (2024).
10. Singhal, S. Real-time detection and tracking using multiple AI models and techniques in cybersecurity. *Transactions Latest Trends Health Sector*, **16**(16). https://ijsdcs.com/index.php/TLHS/article/view/462/182 (2024).
11. SMH, S. S. F. Real-time implementation of IoT enabled cyber attack detection system (IoT-E-CADS) in advanced metering infrastructure (AMI) using machine learning technique (MLT). Electrical Engineering (in review), (2024).
12. Slimane, J. B., Abd-Elkawy, E. H. & Maqbool, A. Intrusion detection using network traffic profiling and machine learning for IoT. *J. Electr. Syst.* **20** (3S), 2140–2149 (2024).
13. Rajendiran, M., Rani, M. J. & Vimalnath, S. Trust-based authentication model with intrusion detection for IoT-enabled networks using deep learning. *Journal Cybersecur. & Inform. Management*, **14**(2), 198. (2024).
14. Toony, A. A., Alqahtani, F., Alginahi, Y. & Said, W. MULTI-BLOCK: A novel ML-based intrusion detection framework for SDN-enabled IoT networks using pyramidal structure. *Internet Things.* **26**, 101231. https://doi.org/10.1016/j.iot.2024.101231 (2024).
15. Akhunzada, A., Al-Shamayleh, A. S., Zeadally, S., Almogren, A. & Abu-Shareha, A. A. Design and performance of an AI-enabled threat intelligence framework for IoT-based autonomous vehicles. *Comput. Electr. Eng.* **119**, 109609. https://doi.org/10.1016/j.compeleceng.2024.109609 (2024).
16. Prince, N. U. et al. IEEE standards and deep learning techniques for Securing IoT devices against cyberattacks. *J. Comput. Anal. Appl.* **33** (7), 1270–1289 (2024).
17. Sobchuk, V., Pykhnivskyi, R., Barabash, O., Korotin, S. & Omarov, S. Sequential intrusion detection system for zero-trust cyber defense in iot/iiot networks. *Adv. Inform. Syst.* **8** (3), 92–99 (2024).
18. Arnob, A. K. B. & Jony, A. I. Enhancing IoT security: A deep learning approach with feedforward neural network for detecting cyberattacks. *Malaysian J. Sci. Adv. Technol.* **2024**, 413–420 (2024).
19. Zeeshan, M. et al. Protocol-aware deep learning for detecting dos/ddos intrusions using UNSW-NB15 and Bot-IoT datasets. *IEEE Access.* **10**, 2269–2283. https://doi.org/10.1109/ACCESS.2021.3050854 (2021).
20. Abrar, I., Ayub, Z., Masoodi, F. & Bamhdi, A. M. ML-based approach for intrusion detection on NSL-KDD dataset. In 2020 Int. Conf. on Smart Electronics and Communication (ICOSEC) (pp. 919–924). IEEE. (2020). https://doi.org/10.1109/ICOSEC49089.2020.9215282
21. Siddiqi, M. A. & Pak, W. An agile strategy for identifying normalization techniques to enhance ML-based network IDS. *IEEE Access.* **9**, 137494–137513. https://doi.org/10.1109/ACCESS.2021.3118533 (2021).
22. Di Mauro, M., Galatro, G., Fortino, G. & Liotta, A. Supervised feature selection techniques in network intrusion detection: A critical review. *Eng. Appl. Artif. Intell.* **101**, 104216. https://doi.org/10.1016/j.engappai.2021.104216 (2021).
23. Kim, J., Kim, J., Kim, H., Shim, M. & Choi, E. CNN-based network intrusion detection against denial-of-service attacks. *Electronics* **9** (6), 916. https://doi.org/10.3390/electronics9060916 (2020).
24. Halbouni, A. et al. CNN-LSTM: hybrid deep neural network for network intrusion detection system. *IEEE Access.* **10**, 99837–99849. https://doi.org/10.1109/ACCESS.2022.3202237 (2022).
25. Liu, Z. et al. Deep learning approach for IDS: using DNN for network anomaly detection. In Fourth Int. Congress on Information and Communication Technology (ICICT 2019), Vol. 1, 471–479. Springer Singapore. https://doi.org/10.1007/978-981-15-3284-7_40 (2020).

Acknowledgements

The author extends his appreciation to the Deanship of Scientific Research at Northern Border University, Arar, KSA, for funding this research work through the project number "NBU-FFR-2025-1576-03".

Author contributions

The corresponding author 'Abdullah Mujawib Alashjaee' is responsible for completion of the entire manuscript starting from methodology, results, visualization etc.

Declarations

Competing interests
The authors declare no competing interests.

Additional information
Correspondence and requests for materials should be addressed to A.M.A.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.