International Journal of Innovation and Applied Studies
ISSN 2028-9324 Vol. 44 No. 4 Feb. 2025, pp. 1022-1029

http://www.ijias.issr-journals.org/

Hybrid 1DCNN+BiLSTM Architecture for Network Intrusion Detection Systems

Kamagaté Beman Hamidja[1], Kanga Koffi[1], Coulibaly Kpinnan Tiekoura[1], and Konaté Adama[1-2]

[1]Ecole Supérieure Africaine des Technologies de l'Information et de la Communication (ESATIC), Abidjan, Côte d'Ivoire

[2]Institut National Polytechnique Félix Houphouët Boigny (INP-HB), Yamoussokro, Côte d'Ivoire

ABSTRACT: The rapid growth of Internet of Things (IoT) networks, fuelled by advancements in Low-Power Wide-Area Networks (LPWANs) and 5G technologies, has transformed industries such as healthcare, smart cities, and manufacturing**.** However, this expansion has also exposed IoT systems to cybersecurity vulnerabilities, making them prime targets for network intrusions and cyberattacks**.** Addressing these threats requires effective Intrusion Detection Systems (IDS) capable of identifying and classifying malicious traffic patterns. This paper proposes a hybrid IDS framework that integrates a 1D Convolutional Neural Network (1D CNN) and a Bidirectional Long Short-Term Memory (BiLSTM) model. The 1D CNN serves as a feature extractor, capturing spatial patterns in network traffic, while the BiLSTM **l**everages temporal dependencies in both forward and backward directions to enhance classification accuracy**.** Experiments assess the model's performance in both binary and multi-class classification tasks**.** The results demonstrate that the 1D CNN+BiLSTM outperforms traditional methods, including SVM, XGBoost, and CNN+LSTM, achieving the highest accuracy (95.03%), recall (94.80%), and F1-score (94.90%). These findings highlight the model's ability to minimize false positives and false negatives, making it highly suitable for real-time intrusion detection in IoT environments.

KEYWORDS: Network security, Deep learning, anomaly detection, multi-class classification, IoT.

1 INTRODUCTION

Advancements in Internet of Things (IoT) networks, particularly in Low-Power Wide-Area Network (LPWAN) technologies such as LoRaWAN, along with the emergence of 5G Reduced Capability (Redcap) technology, have made these networks indispensable for modern infrastructures across various industries, including healthcare, manufacturing, agriculture, and smart cities. The proliferation of IoT devices and uses cases has led to their extensive use in real-time data communication, remote monitoring, and critical applications such as predictive maintenance, telemedicine, and automated supply chain management. Many of these applications rely on cloud computing environments to enhance processing power, data storage, and scalability, enabling seamless integration and efficient operation of IoT ecosystems [1].

However, the rapid adoption of IoT networks has introduced numerous and increasingly sophisticated cybersecurity threats. These threats exploit vulnerabilities in IoT devices and systems to carry out cyberattacks, compromising key properties such as confidentiality, integrity, and availability. Maintaining these properties is crucial for protecting sensitive IoT-generated data, including patient health metrics, industrial operation data, and urban traffic analytics. Data breaches can have severe consequences, ranging from jeopardizing patient safety in healthcare systems to disrupting essential services in smart cities and industrial operations [2]. One of the most significant threats to IoT networks is unauthorized network intrusion. Malicious actors exploit IoT-specific vulnerabilities to gain control over devices and systems, resulting in data theft, operational disruptions, and large-scale cyberattacks. For example, Distributed Denial of Service (DDoS) attacks often utilize compromised IoT devices as attack vectors, highlighting the urgent need for robust security measures in IoT networks.

To counter such threats, several solutions have been proposed, with Intrusion Detection Systems (IDS) standing out as one of the most effective [3]. An Intrusion Detection System (IDS) is a software or hardware solution designed to monitor a network or computing system to identify malicious activities or security policy violations. Depending on its scope, an IDS can operate at two levels. It can operate in network level by monitoring network to find malicious traffic, such IDS is called Network Intrusion Detection System (NIDS). When the IDS monitor a local process and system files of specific host it is called Host Intrusion Detection System (HIDS).

As intrusion attacks become increasingly sophisticated, Intrusion Detection Systems (IDS) have progressively embraced machine learning and deep learning approaches. These AI-driven technologies significantly enhance detection and monitoring capabilities by identifying emerging threats and previously uncatalogued attack signatures. Recent research highlights the pivotal role of these models in proactively detecting cyberattacks and fortifying system resilience against evolving threats.

In [4], an optimized IDS is proposed, utilizing Support Vector Machines (SVM) to efficiently detect both known and unknown attacks. The model incorporates feature selection and dimensionality reduction techniques to improve detection performance. Optimization strategies such as grid search and Particle Swarm Optimization (PSO) are employed to fine-tune the SVM-based IDS, delivering precise classification results.

Expanding on this, the study in [5] explores feature extraction methods combined with machine learning models to detect cyberattacks in Internet of Things (IoT) environments. With the growing prevalence of IoT devices managing sensitive data, effective detection mechanisms are critical. The study focuses on reducing data dimensions while preserving essential information to improve classification accuracy. Techniques such as image filters and transfer learning models, including VGG-16 and DenseNet, are evaluated. Additionally, machine learning algorithms like Random Forest, K-Nearest Neighbors (KNN), and SVM, as well as stacked models, are assessed. The combination of VGG-16 with stacking achieved the highest classification accuracy.

In [6], a network-based IDS is introduced to protect smart home devices, a frequent target of cyberattacks. This system uses eXtreme Gradient Boosting (XGBoost), an ensemble learning technique, to analyse network traffic and detect Distributed Denial-of-Service (DDoS) attacks. Cross-validation results reveal that the XGBoost model achieves up to 94% accuracy in classifying multiple attack types, outperforming traditional classifiers such as SVM, Decision Trees (DT), and Random Forests (RF) while improving network performance.

The study in [7] presents an IDS leveraging a deep learning model called Pearson-Correlation Coefficient - Convolutional Neural Networks (PCC-CNN). By combining linear feature extraction with CNNs, the system effectively handles both binary anomaly detection and multiclass classification for various attack types. Evaluated on three publicly available datasets, the PCC-CNN model demonstrates superior accuracy compared to traditional machine learning models like Logistic Regression, KNN, CART, and SVM, establishing it as a promising solution for advanced intrusion detection.

In [8], a one-Dimensional Convolutional Neural Network (1D CNN) is introduced, specifically designed for feature extraction from time-series data. This architecture comprises three 1D convolutional layers, each followed by a max-pooling layer to reduce computational complexity while retaining key features. The outputs are flattened and processed through two fully connected layers, with a dropout layer added to mitigate overfitting and enhance generalization. The model achieves excellent performance metrics, including accuracy, precision, recall, and F1-score, effectively detecting various traffic classes such as normal, DDoS_TCP, Ransomware, and MITM. Its hierarchical feature extraction capability enables the 1D CNN to transition from simple patterns to high-level abstract representations, making it stand out among other architectures.

In [9], the authors propose an IDS that integrates a Convolutional Neural Network (CNN) for extracting local features and a Recurrent Neural Network (RNN) for capturing sequential data patterns. The system's performance, evaluated using the CICIDS-2018 benchmark dataset, underscores its effectiveness and superiority over existing approaches. Similarly, the study in [10] introduces three deep learning-based models for intrusion detection in IoT networks: a CNN, a Long Short-Term Memory (LSTM) network, and a hybrid CNN + LSTM model.

These studies collectively highlight the significant advancements in intrusion detection systems achieved through the adoption of machine learning and deep learning techniques. From optimized traditional models to cutting-edge hybrid architectures, these approaches provide robust, accurate, and scalable solutions to address the growing complexity of cyber threats in various environments. Despite these advancements, it remains crucial to improve the performance of these models to reduce the risks of false positives and false negatives, which can pose a threat to network security. It is also noteworthy that hybrid models have achieved excellent results on various datasets. Therefore, in this work, a hybrid approach is proposed by combining a Convolutional Neural Network (CNN) and a Bidirectional Long Short-Term Memory (BiLSTM) model.

The remainder of this work is organized as follows: Section 2 offers an overview of the proposed approach along with a detailed description of the implementation framework and dataset. Section 3 focuses on the experimental setup, results, and discussion. Finally, Section 4 concludes the work.

2 METHODOLOGY

2.1 PROPOSED MODEL

In this section, we propose a method for detecting and classifying network traffic using two complementary architectures. A one-Dimensional Convolutional Neural Network (1D CNN) is utilized as a feature extractor to capture spatial patterns within the data. This is followed by a Bidirectional Long Short-Term Memory (BiLSTM) network, which models temporal dependencies in both forward and backward directions.

The proposed 1D CNN architecture consists of three one-dimensional convolutional layers (Conv1D), each followed by a max pooling layer (MaxPooling1D). The convolutional section concludes with a flatten layer for feature extraction (see figure 1). This architecture is designed to capture spatially localized features, such as source and destination relationships, by analysing attributes like IP addresses, ports and protocols. Furthermore, it extracts traffic flow characteristics, including metrics such as total bytes sent or received, packet transmission rates, and other relevant features essential for network traffic analysis.

For an input sequence $\boldsymbol{X}$ of length $\boldsymbol{n}$ wit $\boldsymbol{d}$ features per timestep, the Conv1D layer performs convolution operation to extract spatial features by using formula 1:

$$\boldsymbol{y[i] = \sum_{j=0}^{k-1} x[i+j] \,.\, w[j] + b} \quad (1)$$

Where, $\boldsymbol{x[i+j]}$ is an input values within the receptive field, $\boldsymbol{w[j]}$ is a filter weights or kernel, $\boldsymbol{b}$ is the bias term added to the output. The kernel size $\boldsymbol{k}$ is the number of weights in the filter and finally $\boldsymbol{y[i]}$ is the output value, the feature map at position $\boldsymbol{i}$.

The Conv1D layer is followed by a MaxPooling layer, which is commonly used in CNNs to reduce dimensionality while preserving the most significant features. It achieves this by selecting the maximum value within a specified pool size. Given an input sequence $\boldsymbol{X = [x_1, x_2, x_3, \ldots\ldots, x_n]}$ and a pool size $\boldsymbol{p}$, the output after applying max pooling is (formula 2):

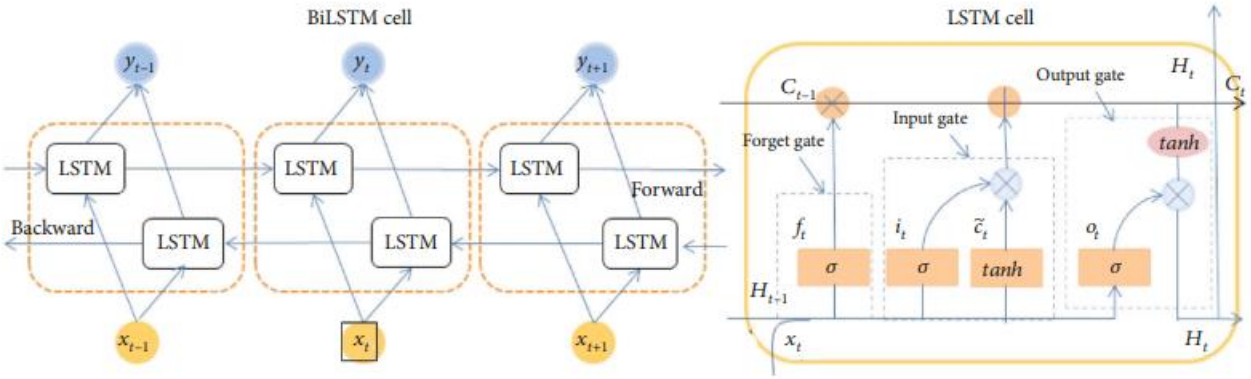
$$\boldsymbol{Y[i] = \max\,(X[i.s],\ X[i.s+1], \ldots\ldots, X[i.s+p-1]} \quad (2)$$

Where $\boldsymbol{Y[i]}$ is the output value at position $\boldsymbol{i}$. $\boldsymbol{X[i]}$ is the input value within the pooling size. the stride $\boldsymbol{s}$ is the step size for moving the pooling window and the pool size $\boldsymbol{p}$ is the width of the pooling window. $\boldsymbol{Max}$ refers to selecting the maximum value within a specified pooling region during the pooling operation. This process is used to highlight the most prominent features and reduce dimensionality while retaining important information from the input data.

The last step of 1DCNN section, consist of performing the flatten operation, which reorganizes multi-dimensional feature maps into a 1D vector, ensuring compatibility with dense layers. It retains all extracted features without modification, making it a crucial step in CNNs and models that process sequential data. Given an input tensor of dimensions $\boldsymbol{X \in \mathbb{R}^{(batch_{size},h,w,c)}}$, the flatten operation reshapes it into $\boldsymbol{Y \in \mathbb{R}^{(batch_{size},h.w.c)}}$ where $\boldsymbol{h}$ is the height of the feature map, $\boldsymbol{w}$ the number of filters and $\boldsymbol{batch_{size}}$, the number of input samples processed together.

At this stage the Bidirectional Long Short-Term Memory (BiLSTM) is used to capture long-term dependencies by using the output of 1DCNN as input. It captures past dependencies using forward LSTM and future dependencies using a backward LSTM and finally combine the outputs from both directions to produce richer feature representations.

Figure 1 inspired by [11] illustrates the architecture of a Bidirectional Long Short-Term Memory (BiLSTM) network and a detailed view of a LSTM cell.



.

Fig. 1. Architecture of BiLSTM and the internal structure of its LSTM cell unit

A BiLSTM consists of multiple LSTM cells, where each LSTM cell processes information through a set of gate**s** that regulate the flow of data. The right section zooms into the internal workings of a single LSTM cell, showing how it processes inputs and maintains memory states over time. At time t, the output of LSTM cell c_t depend on set of gates for state H_{t-1}. The forget gate f_t, determines the extent to which the information from the previous cell c_{t-1} will be discarded. The input gate i_t determines the proportion of new information that will be stored in the cell c_t. The output gate o_t controls the proportion of the internal state that will be passed to the next cell.

Let $Y\epsilon\ \mathbb{R}^m$ represent the output obtained from the 1D CNN phase. The LSTM receives as input a sequence vector $Y = (y_1, y_2, y_3, \dots y_m)$, and applies a linear transformation to produce a state vector $= (h_1, \dots, h_m)$ by computing following functions at each step $t \in \{1, \dots, m\}$

$$i_t = \sigma\left(W_{y_i} y_t + W_{h_i} h_{t-1} + W_{c_i} c_{t-1}\right); \quad (3.1)$$

$$f_t = \sigma\left(W_{y_f} y_t + W_{h_f} h_{t-1} + W_{c_f} c_{t-1}\right); \quad (3.2)$$

$$o_t = \sigma\left(W_{y_o} y_t + W_{h_o} h_{t-1} + W_{c_o} c_t\right)(6); \quad (3.3)$$

$$c_t = f_t c_{t-1} + i_t tanh\left(W_{y_c} x_t + W_{h_c} h_{t-1}\right); \quad (3.4)$$

$$h_t = i_t \tanh\left(c_t\right) \quad (3.5)$$

The parameters W_p, W_h, W_c are learned during the network training [49]. At each step, the cell receives an element y_t from the sequence and outputs h_t the state of that element. The final output of this LSTM layer is the output H_t from the last cell c_t. Such an LSTM layer processes the input in a single direction, from left to right, and can therefore only encode dependencies based on earlier elements in the sequence. The BiLSTM [12] processes the input in the two direction. This allows the network to detect dependencies based on earlier and later elements in the sequence by reading the input from right to left and from left to right.

The pseudo-code for processing the 1D CNN + BiLSTM model, as shown in Algorithm 1, describes the design and training of a hybrid neural network. This model integrates a 1D Convolutional Neural Network (1D CNN) for feature extraction and a Bidirectional Long Short-Term Memory (BiLSTM) network for sequence modeling, specifically aimed at classification tasks.

Algorithm 1 : 1DCNN+BiLSTM

```
Input : Train_X, Train_Y
Hyper-Parameters : optimizer, rate, feature_layers, poolsize, batchsize
Initialize () : Initializes the parameters, model weights, and architecture configurations.
Normalization (Train_X, Train_Y) : Scales the input data (Train_X) and target labels (Train_Y) to a normalized range
Convolution_1 = Sequential((Conv1D(filters, kernel_size, activation, name="Conv1D_1"), MaxPooling1D(pool_size), Dropout(rate)))
Convolution_1 = Sequential((Conv1D(filters, kernel_size, activation, name="Conv1D_1"), MaxPooling1D(pool_size), Dropout(rate)))
Convolution_1.compile(optimizer, loss function, metrics)
Convolution_1.fit(Train_X, Train_Y, epochs, batchsize)
Convolution_1_feature = Model(inputs, Convolution_1("Conv1D").output)
Features = Convolution_1_feature.predict(Train_X)
BiLstmModel = Sequential(Bidirectional(LSTM(units, activation='tanh', recurrent_activation='sigmoid')), Flatten())
BiLstmModel.compile(loss_function, optimizer)
BiLstmModel.fit(Features, Train_Y, batchsize, epochs)
```

2.2 FRAMEWORK AND DATASET

The proposed framework begins with data pre-processing, which involves cleaning, transforming, and organizing raw data to prepare it for modeling. In this framework, a pre-trained word embedding, called GloVe, is utilized to initialize the input layer of the 1D CNN.This step is followed by splitting the dataset into training, validation, and test sets. After that, we apply the hybrid 1D CNN + BiLSTM model to detect whether the traffic is normal or abnormal. Finally, abnormal traffic, identified as an attack, is classified into a specific attack type. Figure 2 illustrates the flowchart of this proposed framework.

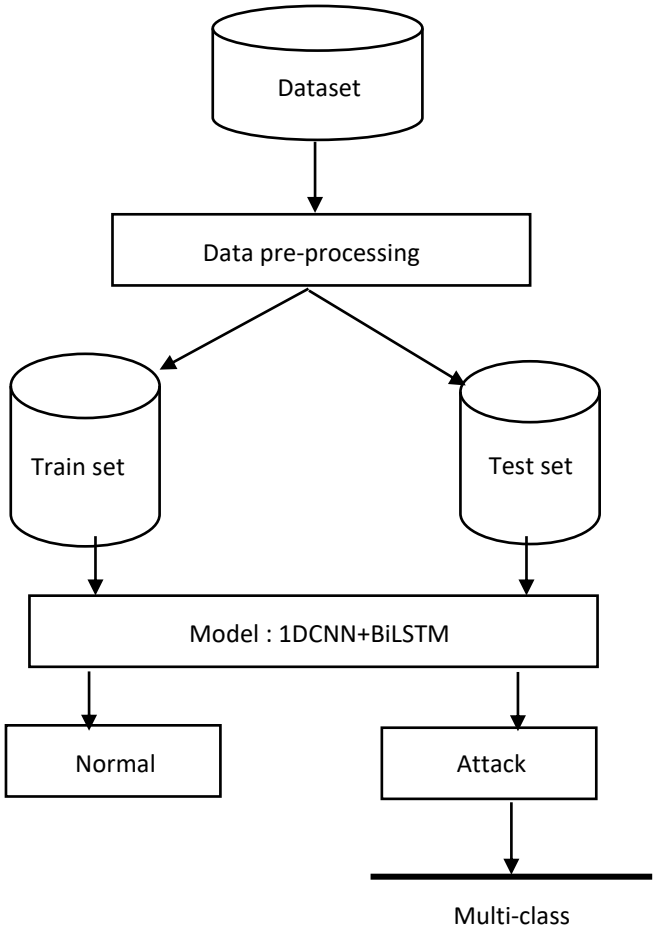


Fig. 2. Flowchart of attacks detection

The UNSW-NB15 dataset is a modern network intrusion detection dataset developed by the Australian Centre for Cyber Security (ACCS) to provide a realistic representation of network traffic and cyber-attacks. It is widely used to evaluate Intrusion Detection Systems (IDS) and machine learning algorithms for network security tasks. It 49 features, including basic connection details, traffic flow statistics, and content-based features. Figure 3 shows distribution of normal traffic and attacks. It includes eight categories of attacks: DoS, Exploits, Fuzzers, Reconnaissance, Analysis, Backdoor, Shellcode, and Worms.

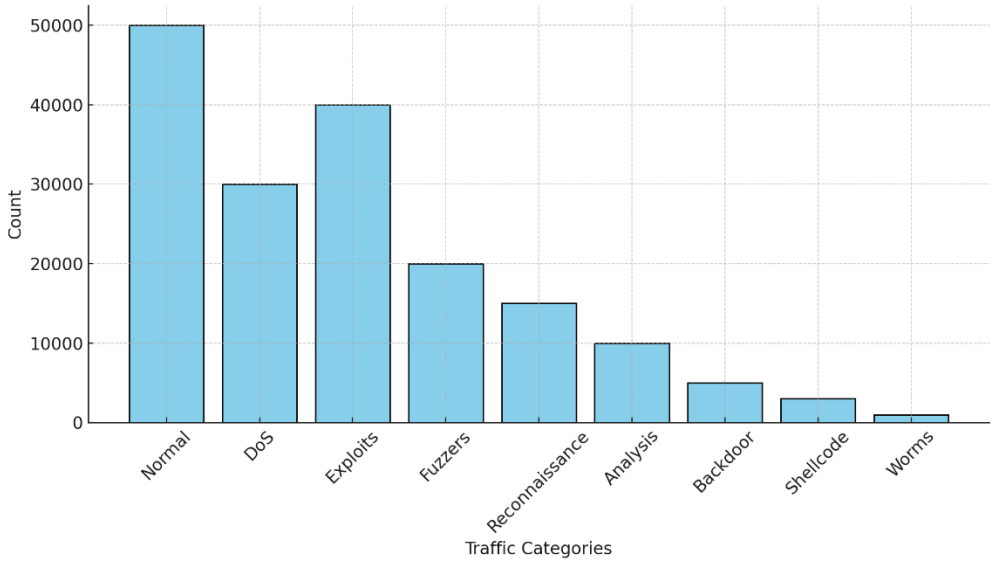


Fig. 3. Distribution of attacks and normal traffic in UNSW-NB15 dataset

3 EXPERIMENTATION, RESULTS AND DISCUSSIONS

3.1 EXPERIMENTATION

This is about the experimentation performed in this work in order to evaluate the performance of the proposed model compared to others models in literature. This section discusses the experiments conducted in this study to evaluate the performance of the proposed model in comparison to other models from the literature. The experiments were implemented using Python 3 and its machine learning libraries on a Core i7 computer with 16 GB of RAM.

The hybrid 1D CNN+BiLSTM model is compared with other intrusion detection models, including the hybrid CNN+LSTM model presented in [Z], the XGBoost algorithm described in [A], and the SVM classifier. The performance metrics used in this evaluation include accuracy, F1-score, and recall. The models are evaluated in both binary classification, which distinguishes normal from abnormal traffic, and multi-class classification, which identifies the specific types of attacks.

Prior to training the model, GridSearchCV, a hyperparameter tuning technique provided by Scikit-learn in Python is utilized to determine the optimal hyperparameters for the 1D CNN+BiLSTM architecture. The selected hyperparameters are presented in Table 1.

Table 1. model hyperparameters

Hyperparameter	1DCNN (value)	BiLSTM(value)
Dropout	0.3	0.2
Layers	2	3
Learning rate	0.01	0.01
Epoch	100	50
batchsize	80	64
Filter	64	64
optimizer	Gradient Descent	Gradient Descent
Kernel size	3	3
Reccurent activation function	-	sigmoid
Activation function in output layer	-	softmax

3.2 RESULTS AND DISCUSSION

Results of the binary and multi-class classification processes on performance metrics. Table 2 shows the result with the metric accuracy. The results in the table is the mean of both training and test accuracy.

Table 2. Results on accuracy of models in both binary and multi-class classification

Models	Binary classification	Multi-class classification
SVM	91.01	89.12
XGBOOST	92.05	90.01
CNN+LSTM	94.07	94.01
1DCNN+LSTM	95.03	95.00

The results clearly show that the 1D CNN+BiLSTM model outperforms all other approaches for both binary and multi-class classification tasks. Its ability to effectively model spatial and temporal features gives it an advantage in detecting both normal and specific types of attacks in network traffic. While traditional models like SVM and XGBoost offer decent performance, they fall short in handling complex sequential dependencies compared to deep learning architectures. The proposed model demonstrates state-of-the-art performance, making it highly suitable for intrusion detection systems (IDS) in modern network environments.

The study aims to evaluate the performance of both models by using recall as a metric to measure their ability to identify all positive samples. The results are in table 3.

Table 3. Results on recall of models in both binary and multi-class classification

Models	Binary classification	Multi-class classification
SVM	90.50	88.50
XGBOOST	91.70	89.80
CNN+LSTM	93.80	93.60
1DCNN+LSTM	94.80	94.70

Table 3 shows that 1D CNN+BiLSTM achieves the best recall performance in both binary and multi-class classification tasks, demonstrating its ability to accurately detect attacks and classify attack types with minimal false negatives**.** CNN+LSTM also performs well but is slightly outperformed by BiLSTM, which captures contextual dependencies more effectively**.** Traditional models (SVM and XGBoost), while performing reasonably well, are less suitable for complex and sequential data such as network traffic. For intrusion detection systems, where recall is critical to avoid missing attacks, the 1D CNN+BiLSTM model should be preferred due to its higher sensitivity and ability to handle sequential data efficiently.

Table 4 shows, the results achieved on F1-score metric. The F1-Score is a harmonic mean of precision and recall that balances the trade-off between false positives and false negatives**.** It is especially useful for imbalanced datasets, where relying solely on accuracy can be misleading.

Table 4. Results on F1-score of models in both binary and multi-class classification

Models	Binary classification	Multi-class classification
SVM	90.75	88.80
XGBOOST	91.85	89.90
CNN+LSTM	93.90	93.80
1DCNN+LSTM	94.90	94.85

The 1D CNN+BiLSTM model achieves the highest F1-scores across both binary and multi-class classification, making it the most reliable and effective model for intrusion detection systems. Its ability to capture spatial features through CNN and temporal dependencies using BiLSTM enables it to handle complex attack patterns with high sensitivity and precision. While SVM and XGBoost perform reasonably well, they lack the capability to model sequential dependencies, making them less suitable for tasks requiring deep feature extraction.

4 CONCLUSION

This study presented a hybrid Intrusion Detection System (IDS) combining a 1D Convolutional Neural Network (1D CNN) and a Bidirectional Long Short-Term Memory (BiLSTM) model to enhance the detection and classification of network traffic anomalies. The proposed framework effectively leverages 1D CNN for spatial feature extraction and BiLSTM for modeling temporal dependencies in both forward and backward directions, enabling comprehensive analysis of sequential data patterns.

Through experimentation using the UNSW-NB15 dataset, the proposed model was evaluated against traditional machine learning models, including SVM and XGBoost, as well as deep learning approaches like CNN+LSTM. The evaluation metrics—accuracy, recall, and F1-score—demonstrated that the 1D CNN+BiLSTM model consistently outperformed other methods in both binary classification (normal vs. abnormal traffic) and multi-class classification (specific attack types).

The 1D CNN+BiLSTM model achieved the highest accuracy (95.03% and 95.00%), recall (94.80% and 94.70%), and F1-scores (94.90% and 94.85%) across both classification tasks, proving its effectiveness in minimizing false positives and false negatives. These results validate its suitability for real-time intrusion detection systems in modern network environments where robustness and reliability are critical.

Despite the promising performance, further research is recommended to optimize the model's computational efficiency and address the dynamic nature of emerging threats in IoT and 5G networks. Future work will explore adaptive learning strategies, incremental training techniques, and the integration of reinforcement learning to enhance scalability and adaptability. Additionally, improving the handling of imbalanced datasets and real-time deployment scenarios remains a priority for advancing IDS solutions.

REFERENCES

[1] Lopez, Carlos Alberto, Luis Fernando Castillo, and Juan M. Corchado. «Discovering the value creation system in IoT ecosystems.» Sensors, vol. 21, no 2, p. 328 2021.

[2] Vallabhaneni, Rohith. «Effects of Data Breaches on Internet of Things (IoT) Devices within the Proliferation of Daily-Life Integrated Devices.», 2024.

[3] Moustafa, Nour, et al. «Explainable intrusion detection for cyberdefenses in the internet of things: Opportunities and solutions.» IEEE Communications Surveys & Tutorials, pp 1775-1807, Vol.no 25, Issue n.3, 2023.

[4] Yoheswari, S., «optimized intrusion detection model for identifying known and innovative cyber-attacks using Support Vector Machine (SVM) algorithms.», Vol., no. 5, Issue no.1, 2024.

[5] Musleh, Dhiaa, and al, «Intrusion detection system using feature extraction with machine learning algorithms in IoT.» Journal of Sensor and Actuator Networks, Vol., no 12 Issue no2,2023.

[6] Amru, Malothu, et al. «Network intrusion detection system by applying ensemble model for smart home.» International Journal of Electrical & Computer Engineering, Vol.14, no.3, 2024.

[7] Bhavsar, Mansi, et al. «Anomaly-based intrusion detection system for IoT application.» Discover Internet of things, Vol. no3, Issue no1, 2023.

[8] Arsalan, Muhammad, et al. «1D-CNN-IDS: 1D CNN-based Intrusion Detection System for IIoT.» 2024 29th International Conference on Automation and Computing (ICAC). IEEE, 2024.

[9] Qazi, Emad Ul Haq, Muhammad Hamza Faheem, and Tanveer Zia. «HDLNIDS: hybrid deep-learning-based network intrusion detection system.» Applied Sciences, Vol. 13, no 8, 2023.

[10] ALTUNAY, Hakan Can et ALBAYRAK, Zafer. A hybrid CNN+ LSTM-based intrusion detection system for industrial IoT networks. Engineering Science and Technology, an International Journal, vol. 38, p. 101322, 2023.

[11] PI, Maozheng, JIN, Ning, CHEN, Dongxiao, et al. Short-Term Solar Irradiance Prediction Based on Multichannel LSTM Neural Networks Using Edge-Based IoT System. Wireless Communications and Mobile Computing, 2022, vol., no 1, p. 2372748. 2022.

[12] GRAVES, Alex, FERNÁNDEZ, Santiago, et SCHMIDHUBER, Jürgen. Bidirectional LSTM networks for improved phoneme classification and recognition. In: International conference on artificial neural networks. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005. p. 799-804.