RESEARCH ARTICLE

DOI: 10.54392/irjmt26425

INTERNATIONAL RESEARCH JOURNAL OF MULTIDISCIPLINARY TECHNOVATION

Asian Research Association

CyberNet-IDS: An AI-Driven Framework for Real-Time Cyber Attack Detection Using Deep Learning

Ramu Moodu [a,*], B. Rajani [b], K. Swathi [c], Pesara Rajkumar [d], G. Bindu Madhavi [e]

[a] Department of Information Technology, Teegala Krishna Reddy Engineering College, Meerpet, Hyderabad, India.
[b] Department of Information Technology, TKR College of Engineering and Technology, Meerpet, Hyderabad, India.
[c] Department of Computer Science and Engineering (Cyber Security), CVR College of Engineering, Hyderabad, Telangana, India.
[d] Information Technology Engineer, Huron consulting group, Hyderabad, India.
[e] Department of CSE- AI&ML, Geethanjali College of Engineering and Technology, Hyderabad, India.
* Corresponding Author Email: ramuse2007@gmail.com

DOI: https://doi.org/10.54392/irjmt26425
Received: 23-04-2025; Revised: 29-06-2026; Accepted: 14-07-2026; Published: 30-07-2026

Abstract: Rising numbers of cyber threats require real-time, intelligent IDSs that are not only based on signatures or shallow ML models. Such systems often provide poor generalisation and high false alarms, especially in cases involving more complex temporal dependencies in network traffic. Although recent developments in deep learning offer new opportunities, many of these models struggle to capture the geographical and temporal characteristics of cyber threats, making them less applicable in real-time detection environments. In response to these concerns, this study presents CyberNet-IDS, an AI-based framework for detecting cyberattacks in real time, using a simulated deep-learning hybridisation approach. We integrate a 1-dimensional CNN to abstract spatial network data and BiLSTM networks to model temporal dependencies in network traffic in our framework. Stable preprocessing methods, such as normalisation, median imputation, and an XGBoost-based feature selection process, further enhance the model. CyberNet-IDS is deployed using Apache Kafka and TensorFlow Serving, supporting the streaming of real-time packet data and classification. Through extensive experimentation on the CIC-IDS2017 dataset, we demonstrate that CyberNet-IDS outperforms traditional and existing DL-based IDSs. A well-performing detection model with a reduced false alarm rate is obtained, achieving 97.30% classification accuracy, 96.80% precision, 96.20% recall, and 96.50% F1-score. Therefore, this proposed framework addresses the fundamental issues faced by current IDS solutions and provides a scalable, deployable architecture to secure modern networks against ever-evolving cyber threats instantly.

Keywords: Cyber Attack Detection, IDS, DL, CNN-BiLSTM, Real-Time Network Security

1. Introduction

The increasing number of cyberattacks has made network protection a top priority across industries, particularly amid the development of the Internet of Things (IoT), cloud computing, and intelligent infrastructure. Classical IDSs (Intrusion Detection Systems) typically employ rule-based procedures or shallow machine learning models, which are unable to handle emerging, complex, and dynamic cyber-attack behaviours. These models are generally challenging to model and infer nonlinear relationships and sequential dependencies in network traffic. They often suffer from high false-positive rates and limited generalisation capability. AI and ML have become an efficient component in intrusion detection, automatically identifying intricate trends from massive network data.

In recent literature, several DL approaches have been applied to IDS, including CNNs for recognising spatial patterns [1], RNNs and LSTM networks for modelling sequences [2], and hybrid models combining different architectures. Despite this tight-knit integration, numerous state-of-the-art models either neglect temporal dependencies, cannot process data in a continuous stream, or are assessed under stagnant conditions on small datasets. Therefore, the need of the hour is a scalable yet effective IDS framework that incorporates spatio-temporal feature learning and operates in real time.

To fill these gaps, this study suggests filling these gaps with CyberNet-IDS, an AI-based intrusion detection framework that utilises an HDL model comprising 1D CNN and BiLSTM networks. The main goal is to recognise real-time High-precision cyberattacks with minimal false-positive rates. Novel aspects of this work include the seamless integration of spatial and bidirectional temporal analyses, optimisation via feature selection with XGBoost, and deployment for

real-time inference with Kafka and TensorFlow Serving. Furthermore, this work provides a reproducible evaluation setting based on the CIC-IDS2017 dataset, a research-accepted IDS benchmark.

However, there are still significant limitations in the current DL-based IDS models. Most approaches were developed in the context of analysing single streams, neglecting to address cross-stream, joint spatial, and temporal-flow properties. Additionally, several models are trained and tested in a static, offline environment, lacking the ability to handle streaming and real-time evaluation. These limitations pose obstacles to deployment in operational cybersecurity frameworks, where issues such as low latency, high throughput, and applicability across various attack types are crucial. To address these problems, the CyberNet-IDS model is proposed by integrating a 1D-CNN and a BiLSTM into a unified architecture that can capture and learn from the spatial and sequential behaviours of traffic. Moreover, it interfaces with Kafka and TensorFlow Serving, enabling the deployment of real-time applications with stringent performance requirements.

The following are this work's primary contributions.

1. Proposed CyberNet-IDS – a new joint DL architecture based on 1D CNN and BiLSTM networks that complies with spatial flow-level patterns, and the temporal relations on the continuity of the traffic sequence, for accurate and adaptive intrusion detection.
2. A comprehensive and robust preprocessing pipeline employed feature normalisation, median-based missing-value imputation, and feature selection, with the essential features ranked by the XGBoost algorithm, as key methods for improving data quality, reducing dimensionality, and enhancing model learning efficiency.
3. Implemented a real-time cyberattack detection pipeline by serving the trained model through Apache Kafka for streaming ingestion, and TensorFlow Serving for real-time prediction, for low-latency and high-throughput traffic classification support.
4. Performed extensive experimental evaluations, conducting ablation studies to investigate the contributions of different components, and latency and resource usage profiling to validate the real-time capabilities of the proposed model, and performed a comparison with the state-of-the-art IDS models based on the CIC-IDS2017 dataset, achieving better accuracy, precision, recall, and F1-score.

The remainder of the document is structured as follows: Section 2 surveys the recent literature on AI-based IDS techniques and highlights the main research gaps. Section 3 describes the CyberNet-IDS framework proposed in this paper and its architecture, model design, and algorithm. The experimental setup and performance assessment are shown in Section 4. Section 5 presents the results in detail, assessing both their strengths and limitations. This work is eventually concluded in Section 6, which also outlines future directions for investigation.

2. Related Work

In the improvement approach, the combination of AI and Cybersecurity has lately been in high demand for real-time changes in IDS. De Assis *et al.* [1] focus on a CNN-based approach for IDS in SDN environments, which exhibits several advantages, including lower latency and superior threat detection capabilities. Garcia *et al*. [2] proposed a distributed AI approach for detecting encrypted SlowDoS attacks. Karande and Joshi proposed a real-time IDS for IoT devices focusing on a lightweight design [3]. Tsimenidis *et al.* For a comprehensive review of the application of deep learning techniques in IoT-based detection, please refer to [4]. Tran *et al.* [5] proposed a secure deep learning model for CNC machines and verified it through experiments. Abdalzaher *et al.* Integrating ML with intelligent IoT systems through data-driven security [6]. Lopez *et al.* [7] proposed a stream-based clustering approach to improve IDS towards sustainability. Hnamte *et al.* Introduced an LSTM-AE model using a two-stage deep learning technique [8]. Du *et al.* [9] proposed NIDS-CNNLSTM, and Yi *et al*. In [10], the authors examined deep learning applications for attack detection.

An RNN-based hierarchical deep learning framework for intrusion detection was proposed by Kasongo [11], which obtained good detection results in sequential traffic. For example, Abdelkhalek and Mashaly [12] utilised data resampling to address class imbalance in deep learning, achieving higher detection rates in imbalanced datasets. Hore *et al*. [13] proposed a sequential DL architecture that is more resilient to adversarial network conditions. Maddu and Rao [14] proposed using DL for ID to mitigate flow-based threats in an SDN environment. Aljehane et al. [15] proposed a DL-based hybrid model integrating golden jackal optimisation for an effective and accurate IDS. Roshan et al. Using a heuristic approach, a defence mechanism against adversarial attacks on deep learning-based IDS is proposed by [16]. An AI-based cooperative detection model is described by Sedjelmaci [17] for detecting in 5G networks. AlZubi *et al*. [18] Applied machine learning to detect cyberattacks in healthcare cyber-physical systems.

Table 1. Comprehensive Summary of Related Work and Research Gaps

Ref.	Authors & Year	Model / Approach	Application Domain	Key Contribution	Accuracy (%)	F1-Score (%)	Identified Research Gap
[1]	De Assis *et al.* (2020)	CNN-based IDS	SDN / IoT	Low latency intrusion detection	94.00	93.50	Limited temporal learning
[2]	Garcia *et al.* (2021)	Distributed AI Model	Encrypted Traffic	Detects SlowDoS attacks	93.80	93.20	Limited scalability
[3]	Karande & Joshi (2020)	Lightweight IDS	IoT	Real-time detection	92.50	91.90	Low accuracy for complex attacks
[4]	Tsimenidis *et al.* (2022)	DL Survey	IoT	Comprehensive DL review	—	—	No experimental validation
[5]	Tran *et al.* (2022)	Secure DL Model	CNC Systems	Real-time monitoring	94.20	93.80	Limited dataset usage
[6]	Abdalzaher *et al.* (2023)	ML-based IDS	Smart IoT	Data-driven security model	93.90	93.30	No deep temporal modeling
[7]	Lopez *et al.* (2023)	Stream Clustering	IoT	Sustainable IDS framework	94.10	93.70	Weak feature extraction
[8]	Hnamte *et al.* (2023)	LSTM-AE	Network IDS	Two-stage DL model	95.20	94.60	Limited spatial learning
[9]	Du *et al.* (2023)	CNN + LSTM	NIDS	Hybrid DL model	96.17	95.87	No optimization refinement
[10]	Yi et al. (2023)	DL Survey	Cybersecurity	DL application analysis	—	—	No real-time validation
[11]	Kasongo (2023)	RNN-based IDS	Network Traffic	Sequential learning	94.50	94.00	Limited spatial features
[12]	Abdelkhalek & Mashaly (2023)	Resampled DL	IDS	Handles imbalance	95.00	94.20	Not real-time ready
[13]	Hore *et al.* (2024)	Sequential DL	IDS	Robust detection	95.73	95.30	Not scalable for streaming
[14]	Maddu & Rao (2024)	DL-SDN IDS	SDN	Flow-based detection	94.86	94.25	Limited generalization
[15]	Aljehane *et al.* (2024)	GJO + DL	IDS	Optimization-based IDS	95.40	94.70	Lack of explainability
[16]	Roshan *et al.* (2024)	Heuristic DL Defense	IDS	Adversarial defense	94.60	93.90	Vulnerable to attacks
[17]	Sedjelmaci (2021)	AI Cooperative IDS	5G Networks	Distributed detection	93.70	93.10	Limited scalability
[18]	AlZubi *et al.* (2021)	ML-based Detection	Healthcare CPS	Cyberattack detection	93.50	92.90	No deep learning usage
[19]	Zaman *et al.* (2021)	AI Survey	IoT	Security countermeasures	—	—	No implementation
[20]	Iwendi *et al.* (2021)	AI Architecture	IoT Security	Sustainable IDS	94.00	93.40	Lacks real-time testing
[21]	Amaizu *et al.* (2021)	AI-based DDoS Detection	B5G Networks	Efficient detection	95.10	94.60	Limited generalization
[22]	Kuzlu *et al.* (2021)	AI in IoT Security	IoT	Anomaly detection	93.80	93.20	No hybrid modeling

[23]	Chaudhary *et al*. (2020)	AI Survey	Cybersecurity	Challenges review	—	—	No experimental work
[24]	Kocher & Kumar (2021)	ML/DL Survey	IDS	Performance gaps	—	—	No real-time deployment
[25]	Lansky *et al*. (2021)	DL Review	IDS	Dataset benchmarking	—	—	No new model
[26]	Lee *et al.* (2021)	DL Security Study	IDS	Adversarial defense	94.20	93.60	Limited robustness
[27]	Thakkar & Lohiya (2020)	IoT IDS Review	IoT	Security issues analysis	—	—	No implementation
[28]	Imrana *et al.* (2021)	BiLSTM IDS	IDS	Improved accuracy	95.30	94.80	Limited spatial features
[29]	Gamage & Samarabandu (2020)	DL Benchmark	IDS	Performance comparison	93.80	93.10	No novel model
[30]	Saranya *et al.* (2020)	ML Study	IDS	Algorithm comparison	92.50	91.80	Outdated methods
[31]	Yamin *et al.* (2021)	AI Threat Study	Cybersecurity	Weaponized AI	—	—	No detection model
[32]	Dhanushkodi & Thejas (2024)	AI Detection Framework	IDS	Real-time detection	95.80	95.10	Limited scalability
[33]	Sowmya & Mary Anita (2023)	AI Survey	IDS	Trends & challenges	—	—	No implementation
[34]	Alem *et al.* (2023)	Ensemble IDS	Industry 4.0	Bi-anomaly detection	95.00	94.50	Limited real-time support
[35]	Zeng *et al.* (2024)	AI Framework	Smart Cities	Anomaly detection	95.20	94.70	Conceptual only
[36]	Baker *et al*. (2023)	AI Detection	Smart Grid	Real-time anomaly detection	95.40	94.80	Limited dataset
[37]	Vishwakarma & Kesswani (2022)	DNN IDS	IoT	Real-time IDS	95.10	94.60	No hybrid model
[38]	Asaduzzaman & Rahman (2022)	Hybrid DL	IDS	Adversarial learning	95.30	94.70	High complexity
[39]	Zhang *et al.* (2022)	XAI Review	Cybersecurity	Explainable AI	—	—	No IDS implementation
[40]	Attkan & Ranga (2022)	AI + Blockchain	IoT	Secure framework	94.80	94.20	No real-time validation

Zaman *et al*. [19] surveyed AI countermeasures for IoT threats. Iwendi *et al.* [20] discussed sustainable AI architectures for IoT security.

Amaizu *et al.* [21] proposed a composite framework to detect DDoS attacks in B5G networks using an efficient AI-driven architecture. Kuzlu *et al.* [22] explored the role of AI in enhancing cybersecurity for IoT systems, focusing on anomaly detection and automation. Chaudhary *et al.* [23] reviewed AI-based solutions to key cybersecurity challenges, highlighting their relevance for modern threat landscapes. Kocher and Kumar [24] provided a survey on M and DL methods in IDS, identifying performance gaps in real-world deployment. Lansky *et al*. [25] systematically reviewed DL-based IDS, covering architecture types and benchmarking datasets. Lee *et al.* [26] analysed the security impact of DL in IDS and emphasised defence against adversarial threats. Thakkar and Lohiya [27] reviewed IoT-specific IDS techniques, focusing on DL algorithms and security issues. Imrana *et al.* [28] proposed a bidirectional LSTM-based IDS with improved detection accuracy. Gamage and Samarabandu [29] objectively compared DL-based IDS methods, while Saranya *et al*. [30] evaluated ML algorithms in IDS using performance metrics.

Yamin *et al.* [31] explored the concept of weaponised AI in cyberattacks, highlighting the dual role of AI as both an attack vector and a defence tool. Dhanushkodi and Thejas [32] proposed an AI-enabled framework for threat detection and cyber threat mitigation, demonstrating strong performance in real-time settings. Sowmya and Mary Anita [33] comprehensively reviewed AI-based IDS models, discussing trends, challenges, and deployment scenarios. Alem *et al.* [34] introduced a bi-anomaly-based IDS for Industry 4.0, utilising ensemble learning to enhance accuracy. Zeng *et al.* [35] developed a conceptual framework for AD using artificial intelligence in innovative IoT networks for smart cities. Baker *et al*. [36] proposed an AI-based anomaly detection system for power electronics in smart grids. Vishwakarma and Kesswani [37] designed a DNN-based real-time IDS for IoT environments. Asaduzzaman and Rahman [38] employed adversarial learning with hybrid DL models for ID. Zhang *et al.* [39] reviewed explainable AI in cybersecurity. Attkan and Ranga [40] examined the integration of AI and BC in IoT security systems. Table 1 outlines recent NIDS models, their performance metrics, and research gaps, highlighting limitations addressed by CyberNet-IDS. Luo *et al*. [41] proposed a DL-based AD framework for cyber–physical systems in Industry 4.0, enhancing real-time threat detection through adaptive spatiotemporal feature learning. While various deep learning approaches such as LSTM-AE [8], CNN-LSTM [9], and Resampled DL-NIDS [12] demonstrate notable performance improvements, they often struggle with one or more practical deployment issues. For instance, models like LSTM-AE exhibit strong recall but lack spatial feature abstraction, which is critical for detecting localised packet anomalies. Similarly, approaches such as CNN-LSTM, although effective at capturing spatial and sequential patterns, lack optimisation or architectural refinement, limiting their scalability in real-time contexts. Moreover, some of the models use fixed datasets and ignore issues like the imbalance of classes, on-the-fly flexibility, and latency, which are the essential needs of the IDS in operation. These are the gaps emphasised in Table 1, and they drive the design of CyberNet-IDS. It incorporates space modelling and bifurcation of time modelling and involves preprocessing schemes and real-time inference schemes, both of which deal with the limitations of the literature.

3. Proposed Framework

Two key components may be merged in the proposed CyberNet-IDS framework. It is based on a 1D-CNN and applies it to extract spatial features and temporal patterns via BiLSTM. A combination of these solutions allows for effective identification of cyberattacks in real time, since they focus on sequential and localised data. Help deploys effective processing, categorises evidence in the best way and evolves to a dynamic network.

3.1 Overview

The given model for utilising the CyberNet-IDS is an HDL model consisting of 1D CNNs that extract features and BiLSTMs that can identify live cyber-attacks. The initial stage of the system is preprocessing the dataset, which is then normalised, missing values are managed, and relevant dimension reduction is done to support classification. This confirmation also guarantees that high-quality, representative data was used to train the model. Then, the network traffic data that is input is handled by the 1D-CNN to extract spatial features. To learn about local trends in traffic flow, the CNN generates feature maps that describe different aspects of traffic, which are then transferred to the BiLSTM layer. The BiLSTM model can capture temporal dependencies in network traffic sequences, allowing it to remember patterns that evolve across different time steps. Unlike other architectures, this one can utilise the entire context instead of just a tiny portion: the LSTM neurons in this architecture are bidirectional. Thus, it can consider information before and after a specific time point, enhancing its ability to capture advances in complex temporal behaviours.

In Figure 1, the final classification is performed by fully connected layers that receive as input the product of the BiLSTM layer after the feature extraction and temporal learning phases. Softmax is applied to the outputs of the fully connected layers, encoding a probability distribution over the potential classes (regular and different types of attacks).

This makes the model capable of predicting the probability of each class for the input sample. The model is trained to minimise the difference between the expected rate and the actual labels, which is the cross-entropy loss of the samples. We use the AO to update the model parameters, enabling the model to learn optimal classification weights. Finally, the trained model is applied to the classification of network traffic in real time. The architecture is designed for robust feature extraction and sequential learning, and its performance is presented in Section 4. Such a system ensures high detection performance and the ability to process dynamic, real-time data streams, which are characteristic of most cybersecurity applications.

In our model, spatial dimensions represent the elemental characteristics of a network flow, which reflect immutable or instantaneous aspects (packet size, source/destination ports, flow duration, total bytes, flags, etc.). These are further addressed by the 1D-CNN layer, which identifies local patterns and relationships among features.

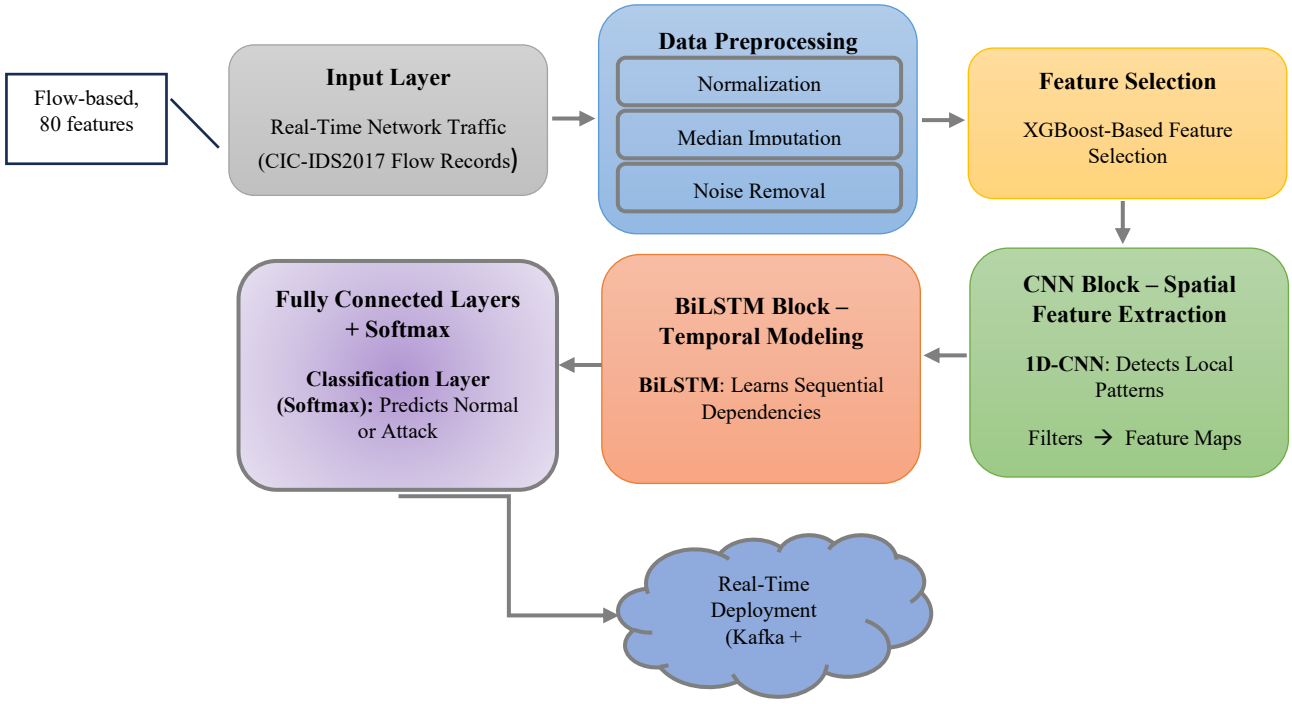

Figure 1. System Workflow of the CyberNet-IDS Framework for Real-Time Cyberattack Detection

Temporal elements include the sequences of traffic flows over time, such as (inter) arrival times, flow ordering, and how traffic statistics change over time. We utilise temporal input features in the BiLSTM layer for pattern learning, making it possible for the model to represent both immediate and long-term dependence.

In our system, we base spatial features on the fixed and inherent characteristics of individual flows, including Flow Duration, Total Forward Packets, Total Backwards Packets, Flow Bytes/s, Forward Header Length, and Protocol. They correspond to traits at a particular moment, which are propagated 358) have through the convolutional operation of the CNN to capture relationships in this space. The CNN is then inclined to learn attack-specific spatial patterns (such as anomalously large payloads or protocol usage) on these features.

3.2 Proposed Model

The CyberNet-IDS model, as shown in Figure 2, proposed in this article, is intended to utilise a DL based architecture that leverages such models to reinforce the real-time detection of cyber-attack events in a fused manner through 1D convolutional neural networks (CNN) and bidirectional long short-term memory (BiLSTM) networks. The model is designed to efficiently process and analyse network traffic data, identifying anomalies that may indicate potential cyber threats. This is due to the model's preprocessing of raw network traffic data, including normalisation and feature selection. Identifying these features reduces computational complexity and enhances model performance by focusing on features indicative of normal or nefarious behaviour.

In the first stage of the CyberNet-IDS model, the 1D-CNN layer captures spatial features about the input data. CNN learns local characteristics of traffic flow and extracts potential attack signatures to distinguish regular, non-concerning traffic from malicious traffic that should trigger actions at the network level. The CNN layer employs convolutional operations, which extract these patterns using kernel (or filter) sets that process sliding windows of the input data, allowing the model to learn which features are related to potential cyberattacks.

Then, the CNN layer is followed by a BiLSTM layer that knows the temporal dependencies in the data. The BiLSTM is bidirectional, in that it processes the data in both directions, allowing the model to discover distant dependencies and temporal relations that span multiple time steps. This is especially vital for identifying inevitable attacks over time, including DDoS attacks or botnet activities, which often exhibit complex temporal features.

The layer of complete connectivity in the model converts the BiLSTM output into the classification decision. Softmax activation is applied to the model's output to produce a probability distribution over the classes of interest, e.g., regular traffic or one of several attack types. The result enables the model to identify and classify any pattern from a raw incoming traffic sample in a Telco environment. Utilising cross-entropy loss, the CyberNet-IDS model analyses the difference between actual labels and the predicted class probabilities.

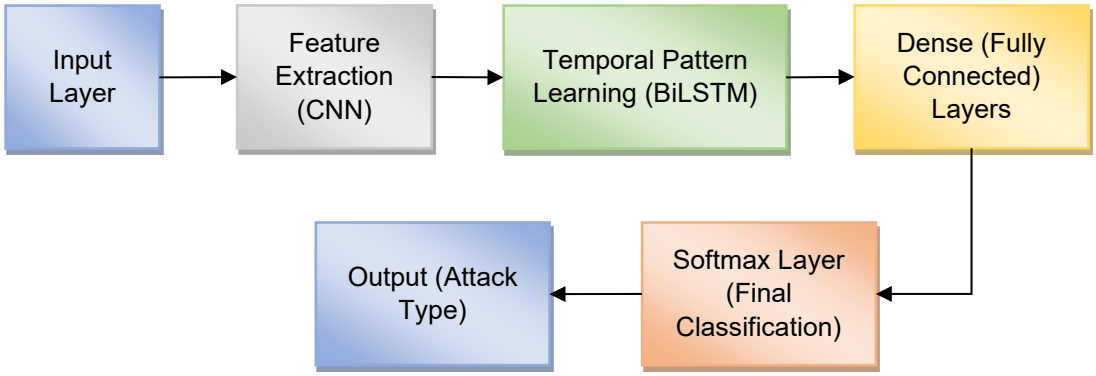

Figure 2. Diagram of the CyberNet-IDS Model Architecture for Cyber Attack Detection

Table 2. Notations Used in the CyberNet-IDS Model

Notation	Description
X= $\{x_1, x_2, \ldots, x_n\}$	Input dataset containing n network traffic samples, where each sample $x_i \in R^d$ is a feature vector with dimensionality d
x_i	Feature vector corresponding to the i-th network traffic sample
d	Dimensionality of the feature vector x_i
y_k	Output of the convolution operation at position k
m	The size of the kernel (filter) used in the convolutional layer
w_j	Weight (kernel) at position j for the convolution operation
b	Bias term in the convolution operation
h_t	Hidden state at time step t in the BiLSTM layer
x_t	Input at the time step to the BiLSTM layer.
W_h	Weight matrix for the fully connected layer after the BiLSTM
b_h	Bias vector for the fully connected layer
y_{out}	Output of the fully connected layer for classification
C	Total number of classes (e.g., regular traffic, various attack types)
z_k	Output of the final layer for class k
L	Cross-entropy loss function
$y_i^{(k)}$	True label for the i-th sample, indicating the ground truth class for sample x_i

The AO updates the model's parameters during training to achieve efficient convergence. The model can detect cyber attacks with anticipated likelihoods as they are received and provide timely alerts for potential threats, consequently improving the network's overall security posture. The derived notations and their descriptions, as outlined in the mathematical formulation of the CyberNet-IDS model for real-time cyber-attack detection, are presented in Table 2. An AI-driven framework for Real-Time Cyberattack Detection Using DL, CyberNet-IDS, can be expressed as follows. The example illustrates that the system is designed to classify network traffic into two categories: normal traffic and cyber attacks. Once we have the various class labels from our inputs, we want a model (or function f(Xi)) that maps from the input dataset X=$\{x_1, x_2, \ldots, x_n\}$, $x_i \in R^d$ a feature vector for network traffic towards one of the classes (regular or attack). Using a CNN +and a bi-directional LSTM hybrid network model, the instance of network traffic x sub i is passed through several layers of the learning and extraction model temporal dependencies of features, and inally, to perform classification. During the feature extraction stage, a 1D CNN is employed to learn the spatial interactions between the characteristics of the input traffic. The convolution process can be mathematically formulated as:

$$y_k = \sum_{j-1}^{m} w_j x_{k+j} + b \quad (1)$$

Where x_{k+j} denotes the input features, w_j is the weights (kernels), and bb is the bias term. This process is performed on multiple layers to learn the high-level spatial correlations of the traffic features.

The output from the CNN is then fed into a sequential BiLSTM layer that captures temporal dependencies between the features across time. The BiLSTM is defined by:

$$h_t = LSTM(h_{t-1}, x_t) \quad (2)$$

Where $h_t$the hidden state value is at time step t, and x_t is the input at time step t. For the BiLSTM, both the forward and backward temporal sequences are taken into account, allowing the model to better understand the sequential patterns found in the network traffic.

Once the feature extraction and temporal dependencies learning are done, the output BiLSTM is fed into a dense layer. It applies a non-linear function to obtain the final output. The dense layer's output is defined as:

$$y_{out} = W_h h_t + b_h \quad (3)$$

Where W_h and are the weight matrix and b_h the bias vector respectively. The last layer uses Softmax to obtain the probability distribution over the attack types:

$$P(y = k \mid x) = \frac{\exp(z_k)}{\sum_{j-1}^{C} \exp(z_j)} \quad (4)$$

Where C is the total number of classes (normal or attack categories) and z_k is the output of the last layer for class k.

We use cross-entropy loss to train the model, which can be defined as;

$$L = -\sum_{i-1}^{n} \sum_{k-1}^{C} y_i^{(k)} \log(P(y_i^{(k)} | x_i)) \quad (5)$$

Where $y_i^{(k)}$ is the ground truth label of the i-th sample, and $P(y_i^{(k)} | x_i)$ is the predicted probability of class k for the input sample x_i. So during training, we want to minimize this loss function.

This methodology aims to efficiently learn an optimized set of parameters for detecting cyber attacks in real-time, validating the network traffic classification accuracy, and enhancing the system's robustness against cyber threats.

3.3 Proposed Algorithm

The suggested algorithm to detect cyberattacks is CyberNet-IDS: AI-Driven Cyberattack Detection. It is a model that identifies patterns of attack and variance associated with it by uniting the 1D, BiLSTM and CNN to extract features and learn behavior based on time. This is important as the algorithm can be used to detect the presence of threats at the right time, and thus enhance network security, since threats can be detected and solved promptly through the algorithm.

Algorithm: CyberNet-IDS: AI-Driven Cyber Attack Detection

Input:
Dataset X with feature vectors x_i.

Output:
Predicted class $\hat{y}$ for each x_i (normal or attack).

1. Preprocess: Normalize, handle missing values, and select features.
2. Feature Extraction: Apply 1D-CNN: $y_k = \sum_{j-1}^{m} w_j x_{k+j} + b$
3. Temporal Learning: Pass CNN output through BiLSTM:
   $h_t = LSTM(h_{t-1}, x_t)$
4. Classification: Apply a fully connected layer: $y_{out} = W_h h_t + b_h$
   Softmax output: $P(y=k|x) = \frac{\exp(z_k)}{\sum_{j-1}^{C} \exp(z_j)}$
5. Loss: Compute cross-entropy loss:
   $L = -\sum_{i-1}^{n} \sum_{k-1}^{C} y_i^{(k)} \log(P(y_i^{(k)} | x_i))$
6. Optimization: Use Adam optimizer.
7. Output: Predicted class $\hat{y}$ for each x_i.

This model, CyberNet-IDS: AI-Driven Cyber Attack Detection, employs a hybrid deep learning approach, incorporating extended STM networks with 1D CNNs, to identify cyber attacks in real-time. The initial step involves preprocessing the flow data by normalising the feature values, handling any missing data, and performing feature selection to ensure that only the essential data is used for training. This preprocessing step minimises the computational burden, enabling the identification of significant attributes and thereby enhancing its ability to detect malignant cells. Following data preprocessing, the model uses a 1D-CNN to extract spatial correlations and patterns from the traffic data. The CNN employs multiple convolutional layers to extract local features useful for routine traffic detection and attack signatures. The CNN's learned features are passed through a BiLSTM layer, which captures time-dependent dependencies in the traffic data. A BiLSTM layer can remember the sequence of events over time, and detecting a dynamically changing attack depends on this.

The learned temporal features are used for classification. This allows selecting the possible classes (i.e., normal traffic or different attacks) on the input. Softmax is the third layer to ensure the model assigns a class of the traffic, whether it is expected or a cyber-attack has happened. We train the model using cross-entropy loss, which compares the correct class labels with the predicted class probabilities. We use the AO to reduce loss and modify the model's parameters during training. We then train the model and deploy it to detect cyber-attacks in network traffic, alerting the user to potential security issues.

3.4 Evaluation Methodology

Evaluation Metrics for CyberNet-IDS are F1-score, recall, accuracy, precision, and FPR. The most general metric is accuracy, which assesses the fraction of correctly classified instances among all instances. It can be defined as:

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (6)$$

where TP = true positives, TN = true negatives, FP = false positives, and FN = false negatives.

Precision is the fraction of optimistic predictions out of all things predicted to be positive. It is defined as:

$$\text{Precision} = \frac{TP}{TP+FP} \quad (7)$$

It helps assess false-positive attacks, i.e., how many detected attacks were FP.

Recall or sensitivity quantifies the model's ability to recognize all the true positive cases (real attacks). It is given by:

$$\text{Recall}= \frac{TP}{TP+FN} \quad (8)$$

Higher Recall suggests that your model is performing well in identifying most of the attacks.

F1-score is a metric that combines precision and recall into one number, balancing the two. It is formally defined as the harmonic mean of precision and recall:

$$\text{F1-Score}=2\times\frac{\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}} \quad (9)$$

When dealing with datasets that are unbalanced, this statistic is particularly helpful, as the number of regular instances substantially more than the quantity of attack instances.

FPR stands for False Positive Rate, and indicates how often the normal traffic is classified as attack traffic. It is defined as:

$$\text{FPR}= \frac{FP}{FP+TN} \quad (10)$$

The lower the FPR, the fewer normal instances will be incorrectly classified as attacks, which is paramount because so many alerts for benign traffic will drive operators crazy.

These evaluation metrics are of prime importance when measuring CyberNet-IDS's overall real-time performance in detecting cyberattacks. The reason for having more than one metric is that each metric provides information on a different aspect of the model's efficacy, and together they complement one another to show the model's capacity to reduce false alarms and correctly detect attacks.

To comprehend the impact of every element, we conduct an ablation study in the experimental setting. This involves comparing the variants CNN-only, BiLSTM-only, and CNN+BiLSTM to disentangle the contributions of spatial and temporal learning modules. The results of our overlaid performance analysis of each component are demonstrated in Section 4.3.

3.5 Materials

This section describes the primary dataset and the computational tools and libraries employed to develop and evaluate the CyberNet-IDS framework. For all experiments, the CIC-IDS2017 dataset [42], made public by the CI for Cybersecurity, was used. It comprises more than 2.8 million labelled network flow records, categorised into various attack types, including DDoS, Botnet, BF, Port Scan, Infiltration, and regular traffic, with 80 flow-level features in each record. The dataset is flow-based, where each instance consists of a set of packets for a session, making it well-suited to a deep learning-based intrusion detection method.

Python 3.9 was used to develop models, and the Deep learning models were created using TensorFlow 2.12 and Keras. XGBoost (version 1.6.2) was used for feature selection. Apache Kafka and TensorFlow Serving were used to perform stream ingestion and deployment. All the experiments were done on an Intel(Core i7) processor, 32 GB of RAM, and NVIDIA RTX 3080 (10GB VRAM) with Ubuntu 20.04 LTS. This software-hardware system setup guaranteed the actual training, implementation, and real-time testing of the CyberNet-IDS architecture in a real performance environment.

4. Experimental Results

In the experimental results section, we offer a comprehensive performance evaluation of the proposed CyberNet-IDS model using the CIC-IDS2017 dataset. These experiments aimed to evaluate the model's efficacy in detecting and categorising cyber adversities in dynamic network traffic using hybrid deep learning mechanisms. The dataset comprises diverse attack instances and regular traffic scenarios, helping to create realistic and challenging conditions for testing the model. The evaluation measures performance based on classification accuracy, detection performance for each attack class, and the model's real-time performance.

The entire experimental setup was implemented using TensorFlow and Keras in Python 3.9 for deep learning model development. Our system specifications include an Intel Core i7 processor, 32 GB of RAM, an NVIDIA RTX 3080 GPU, and Ubuntu 20.04, on which all the experiments are run. The dataset from CIC-IDS2017 was first cleaned of missing and infinite values, then normalised using Min-Max scaling. We performed feature selection using XGBoost importance ranking to retain as many uninformative features as possible. The data was then split into test and training sets, with 20% going to the test set and 80% going to the training set.

To ensure replicability, all hyperparameters and implementation steps are explicitly defined and documented. The CyberNet-IDS model consists of a max-pooling layer, a 1D CL with 32 filters, a kernel size of 3, and ReLU activation. This is followed by a bidirectional LSTM layer with 64 units. To avert overfitting, a dropout layer with a rate of 0.5 is used. The dense layer uses softmax activation to classify the input into regular or attack classes. The Adam optimiser trains the model for 50 epochs with a batch size of 64 and a learning rate of 0.001. The loss function is categorical cross-entropy. Early stopping with a patience of 5 epochs is used to avoid overtraining.

We implement the prototype application in a production setting, using Apache Kafka for stream processing and TensorFlow Serving for model inference.

The trained CyberNet-IDS model has been deployed as a RESTful service (which follows the RESTful uniform interface) that ingests network flow data in real time and provides attack-classification labels as output. Kafka is the stream engine, simulating a continuous stream of incoming traffic and posts to the model endpoint for inference. The entire pipeline enables the replication of detection results and determines a benchmark foundation for testing Real-time methods for detecting cyberattacks.

The experiments were carried out on CIC-IDS2017, a well-known benchmark intrusion detection dataset made available by the CI for CS. It contains empirical benign and attack traffic acquired over 5 days, comprising 80 network traffic features per flow and more than 2.8 million labelled samples. The dataset can be labelled as binary (regular vs. attack) or multi-class (DDoS, Botnet, Brute Force, Web, Infiltration, etc.). Unlike packet-based datasets, CIC-IDS2017 is flow-based, and each row (i.e., instance) represents a flow that combines several packets and includes statistical features (e.g., the total number of bytes, the mean packet length, and inter-arrival times between packets). This format is suitable for implementing DL models and real-time IDSs, as it avoids dealing with raw packet payloads and provides session-level semantics.

4.1 Exploratory Data Analysis

EDA provides insights into traffic distribution, correlations among these features in the CIC-IDS2017 dataset, and attack patterns. This process can help identify data imbalances, outliers, and relationships among features that can inform appropriate preprocessing and feature selection. Such analysis prepares the way for training a deep learning model that is robust and suitable for various cyber-attack detection strategies.

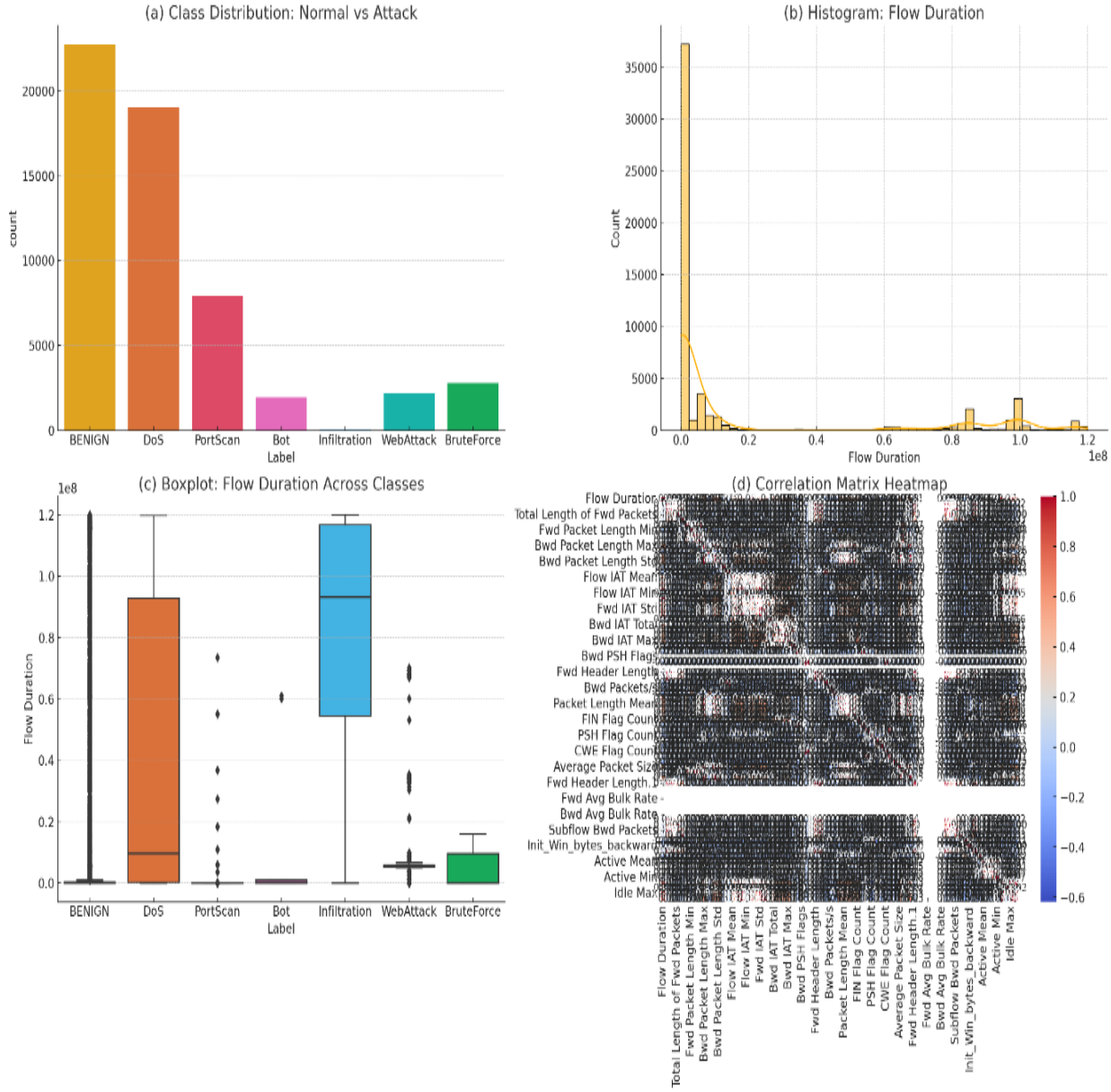

Figure 3. EDA Visualisations of the CIC-IDS2017 Dataset

The EDA visualisations of the CIC-IDS2017dataset effectively clarify traffic characteristics, as demonstrated in Figure 3.The distribution of classes is typical, and the attack traffic balance is maintained. The histogram and boxplot illustrate the distribution of the flow duration. The correlation matrix heatmap helps visualise the relationships between features, providing insights into significant patterns that can benefit model building and feature selection.

After importing the data, Figure 4 shows four excellent EDA plots for the CIC-IDS2017 dataset. The histogram column represents the FlowBytes/s distribution, and the boxplot column highlights Total Forward Packet outliers per class. The violin plot reveals differences in Flow Duration after the attack, and the scatter plot depicts the connection between Flow Duration and the Total Number of Forward Packets.

4.2 Performance Comparison with Baselines

In this section, we compare CyberNet-IDS's performance against baseline models (Random Forest, LSTM, CNN, and CNN+LSTM). F1-score, recall, accuracy, and precision are used to evaluate the effectiveness of the proposed model. The findings demonstrate CyberNet-IDS's superior detection capacity, resilience, and effective classification in various attack contexts.

The accuracy trends for training and validation of the CyberNet-IDS model after 20 epochs are shown in Figure 5. Despite the model, it steadily performs better, achieving 97.3% accuracy for both training and validation, which ultimately settles at 96.2%. The steady increase in both curves suggests that the learning and generalisation process was effective, validating that the model was robust and reliable for cyberattack detection.

Figure 6 depicts the Trends in training and validation losses in the CyberNet-IDS model at 20 epochs. Both loss curves decrease and converge over iterations, suggesting that the model learns efficiently and that the optimisation process is stable. The proximity of training and validation loss indicates strong generalisation ability and low overfitting, demonstrating the model's dependability in real-time cyberattack detection tasks

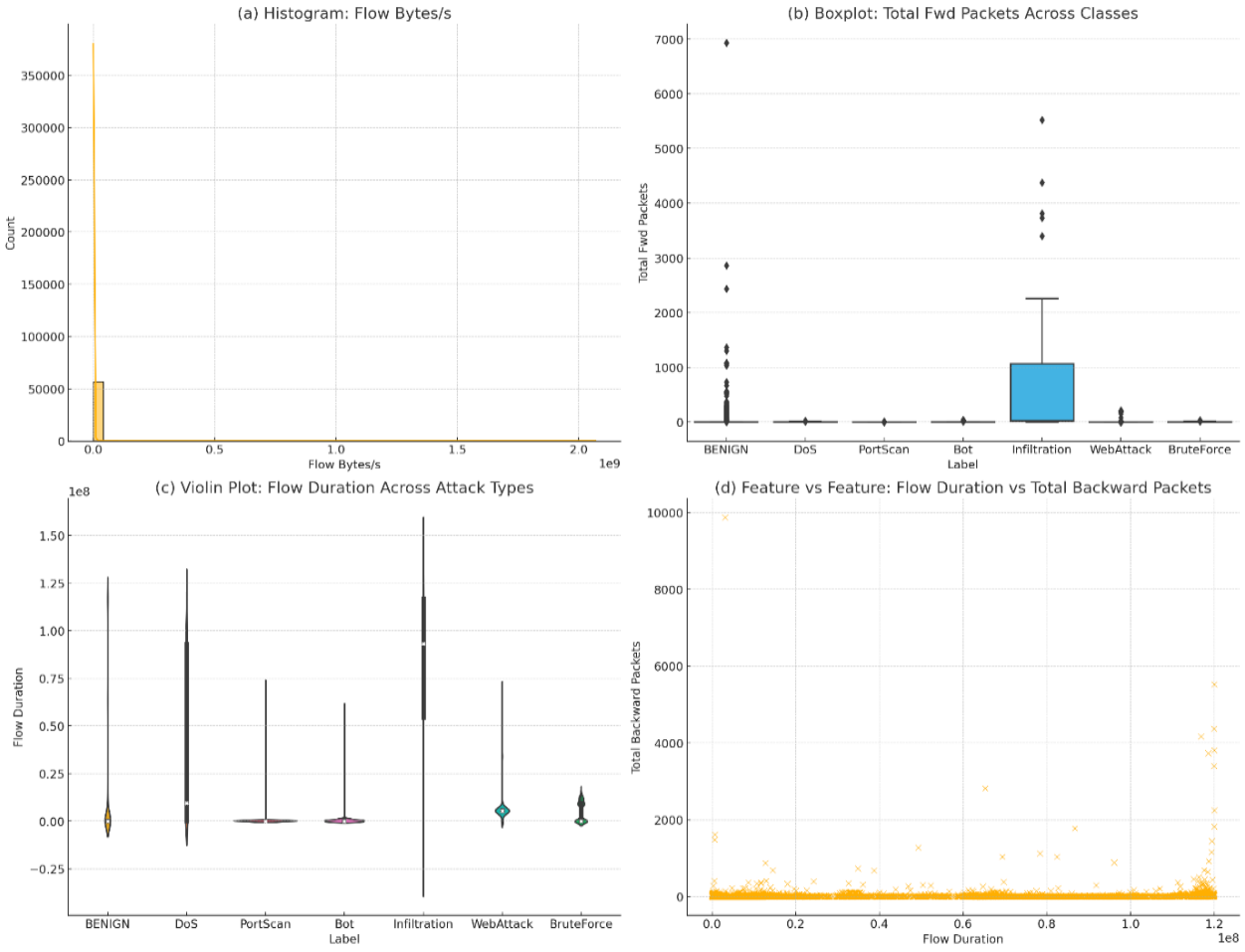

Figure 4. EDA Visualisations of the CIC-IDS2017 Dataset

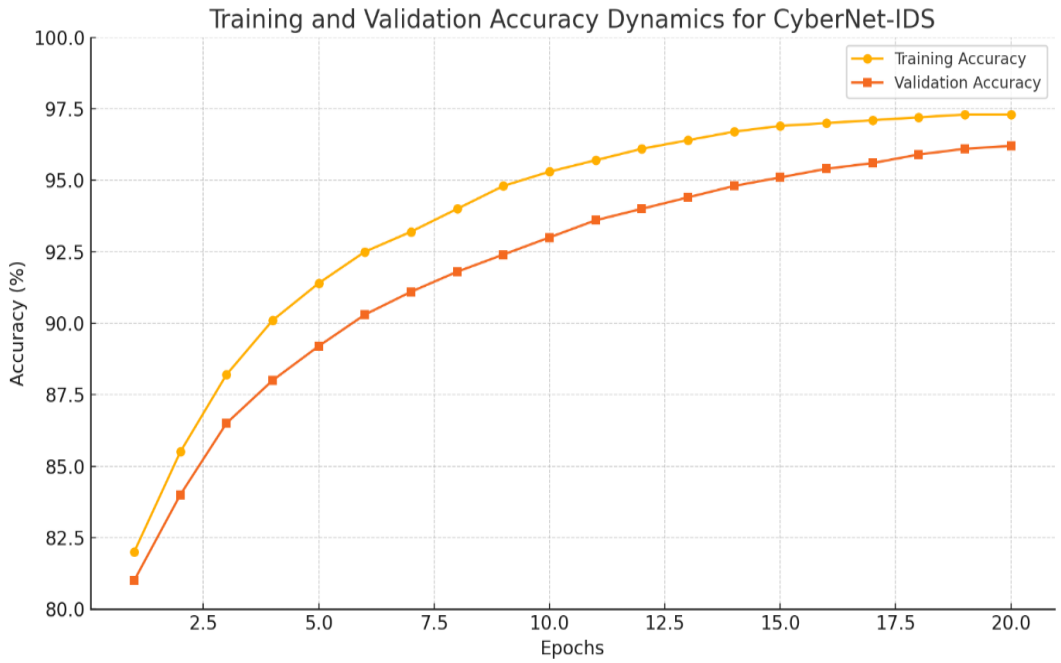

Figure 5. Training and Validation Accuracy Dynamics of CyberNet-IDS

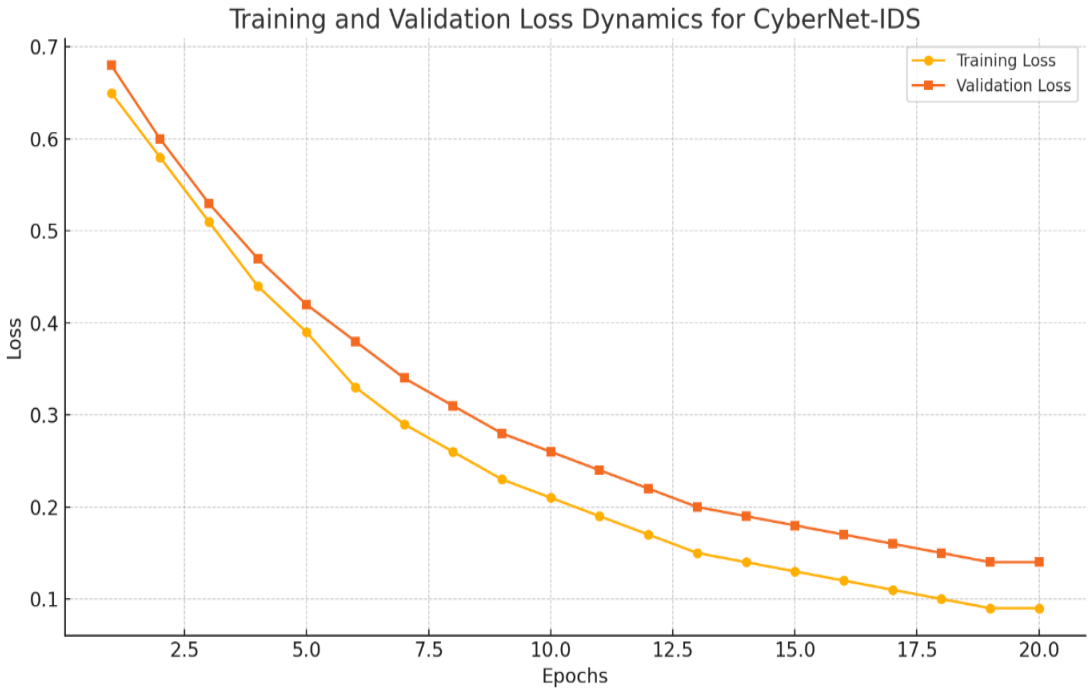

Figure 6. Training and Validation Loss Dynamics of CyberNet-IDS

Table 3. Performance Comparison of CyberNet-IDS with Baseline Models

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Random Forest	89.20	88.10	87.40	87.70
LSTM	91.50	90.80	90.20	90.50
CNN	93.70	92.90	92.10	92.50
CNN + LSTM	95.80	95.20	94.60	94.90
CyberNet-IDS	97.30	96.80	96.20	96.50

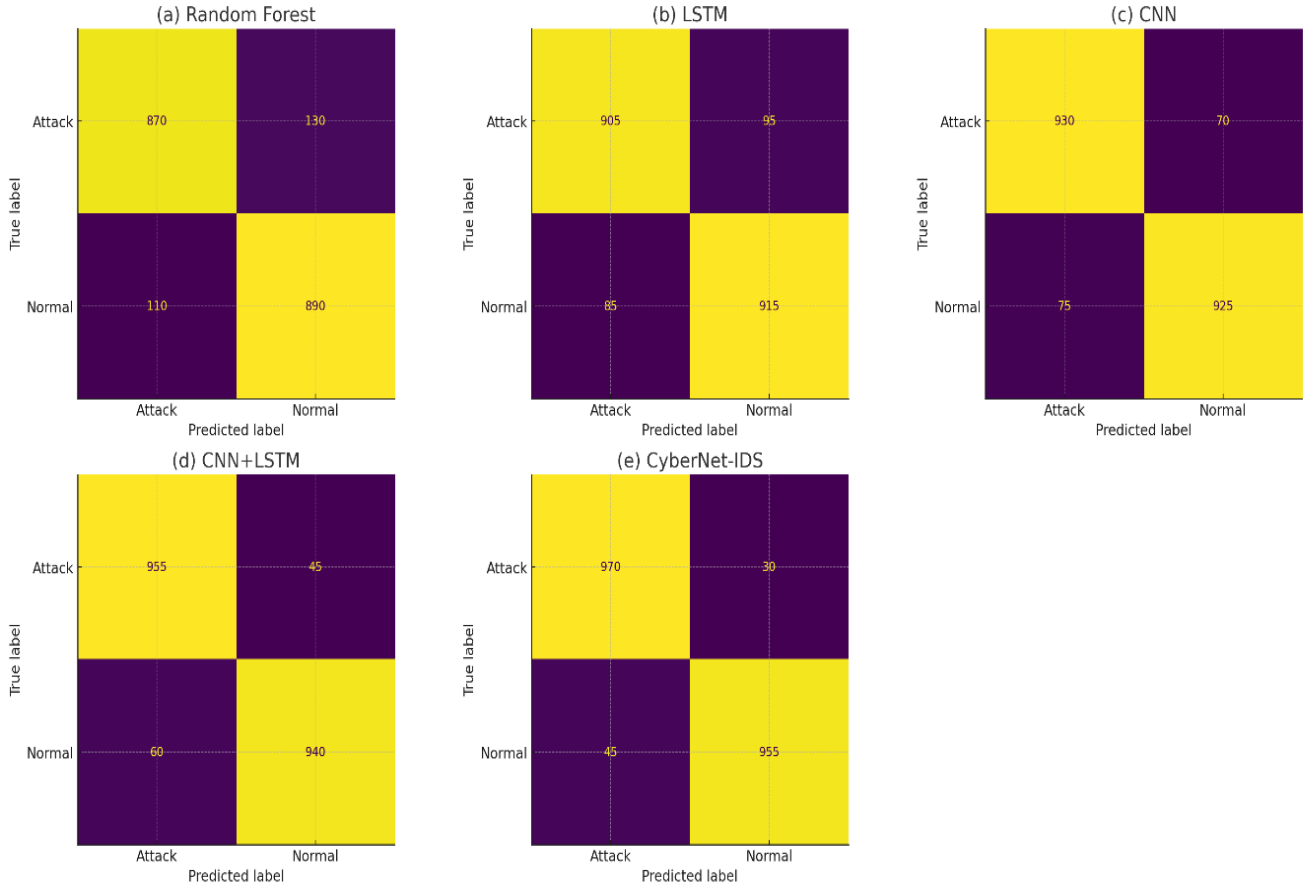

Figure 7. Confusion Matrices of Baseline Models and CyberNet-IDS

Confusion matrices for five models using the CIC-IDS2017 dataset are shown in Figure 7. CyberNet-IDS shows a high percentage of both positive and negative results, indicating its best performance. CyberNet-IDS outperforms baseline models, such as RF and LSTM, in terms of FP and N, demonstrating its comprehensive capability for cyber-attack detection.

Table 3 compares the performance of the CyberNet-IDS model with that of baseline models on the CIC-IDS2017 dataset. This model achieves an accuracy of 97.30% and the best values of F1-score, recall, and precision compared to conventional and DL methods. It demonstrates the efficacy of accurately detecting cyberattacks and addressing complex traffic patterns within the system.

A comparative performance evaluation of the proposed CyberNet-IDS model against four baseline models (Random Forest, LSTM, CNN, and a hybrid CNN+LSTM) on the CIC-IDS2017 dataset is presented in detail in Figure 8. We will evaluate the performance using four widely accepted classification metrics: F1-Score, Accuracy, Precision, and Recall. Subfigure (a) compares accuracies, showing that CyberNet-IDS achieves the highest accuracy of 97.30%, significantly outperforming the baseline models. The table also indicates performance when the RF, a conventional ML technique, is employed; in that case, the accuracy is the lowest, at 89.20%. The performance improves if deep learning bidirectional LSTM and CNN models are used, providing 91.50% and 93.70% accuracy, respectively. The 95.80% performance solution that outperforms the two above is a hybrid CNN+LSTM model, but it performs lower than our proposed architecture. The performance gain of CyberNet-IDS is due to its ability to learn to extract spatial features with a CNN and temporal patterns with a BiLSTM within an optimised pipeline.

In view (b), Precision highlights the proportion of true positives among the predicted positive cases (i.e., the inferred attack instances) without misrepresenting the usual traffic. CyberNet-IDS achieves the best precision of 96.80%, resulting in fewer false positives than any other model. The CNN+LSTM baseline achieves 95.20%, while the single-model CNN and LSTM variants achieve 92.90% and 90.80%, respectively. This implies that the relatively low accuracy of Random Forest (88.10%) tends to mislabel benign traffic as attack traffic.

As shown in subfigure (c), the Recall metric (vital for assessing the detection of actual attacks) is highest for CyberNet-IDS, at 96.20%, as it generalises well across many attack types. With 94.60% accuracy, CNN+LSTM outperforms both CNN (92.10%) and LSTM (90.20%), respectively. Random Forest is slightly behind, with a recall of 87.40%, suggesting it may miss false negatives when identifying specific, complex, or subtle attack behaviours.

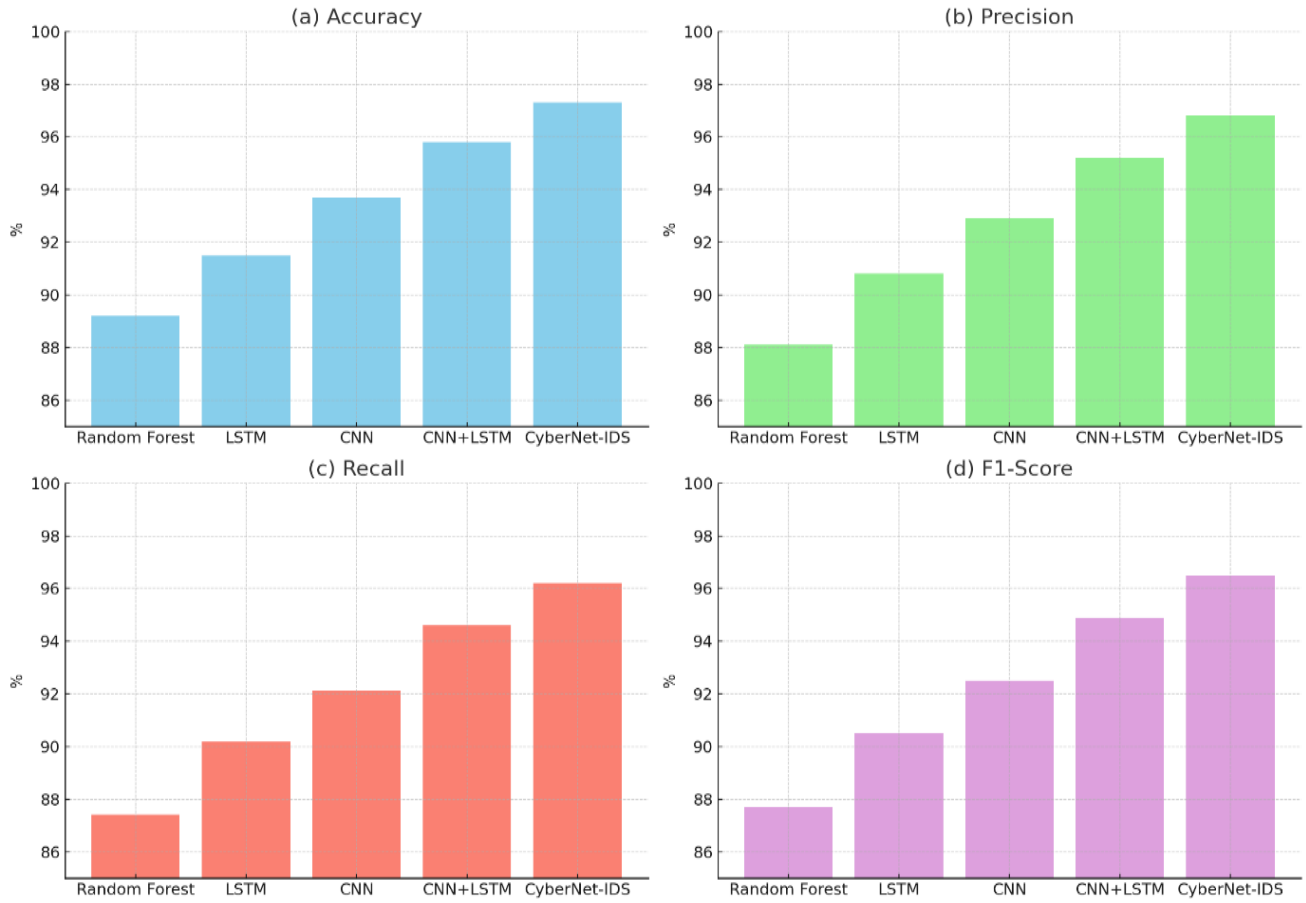

Figure 8. Performance Comparison of CyberNet-IDS with Baseline Models

This higher recall can be beneficial in the context of real-time intrusion detection systems, where missed true threats can lead to more severe security breaches.

In subfigure (d), the precision and recall are aggregated into a balanced F1-Score, in which CyberNet-IDS again achieves the highest score of 96.50%. This confirms the model's balanced performance and reliability in detecting cyber attacks. The accuracy score of the CNN+LSTM model is 94.90%, while CNN achieves 92.50%. LSTM and Random Forest yield lower scores of 90.50% and 87.70%, respectively. Overall, the F1-score of CyberNet-IDS indicates a good balance between false negatives and false positives. This is important for ensuring the integrity of systems and restoring trust in their operational processes.

The performance comparison in Figure 8 shows that CyberNet-IDS outperforms all baseline models across all assessment metrics. Its deep hybrid structure enables efficient large-scale data extraction and temporal sequence modelling. Hence, the proposed CyberNet-IDS model is highly applicable to real-time, high-accuracy cyber-attack detection in modern networks, thanks to its well-optimised CNN-BiLSTM architecture.

4.3 Ablation Study

We also conducted an ablation study on each component and on the combined CNN and BiLSTM components within CyberNet-IDS. This comparative study presents the importance of combining spatial-temporal features, such as CNN, BiLSTM, or CNN+BiLSTM, with the enhanced output of CyberNet-IDS. The outcomes validate that a hybrid, optimised configuration can achieve higher reliability and accuracy in cyberattack detection.

Table 4 presents the results of the ablation study for CyberNet-IDS, highlighting the contributions of the convolutional neural network (CNN) and BiLSTM modules. Collectively, CNN and BiLSTM show moderate results; however, aggregate results indicate high accuracy and recall. The CyberNet-IDS model, optimised according to all configurations, also surpasses all other configurations, reinforcing that by optimising the architecture and adding spatial and temporal learning components, the detection of cyber attacks becomes even more effective.

Figure 9 illustrates the metric-wise results of an ablation study, showing the contribution of each component within the CyberNet-IDS architecture. The evaluation compares four model variants: CNN-only, BiLSTM-only, a combined CNN+BiLSTM configuration, and the fully optimised CyberNet-IDS.

Table 4. Ablation Study Results for CyberNet-IDS

Model Variant	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
CNN only	93.70	92.90	92.10	92.50
BiLSTM only	91.50	90.80	90.20	90.50
CNN + BiLSTM	95.80	95.20	94.60	94.90
CyberNet-IDS (Optimised)	97.30	96.80	96.20	96.50

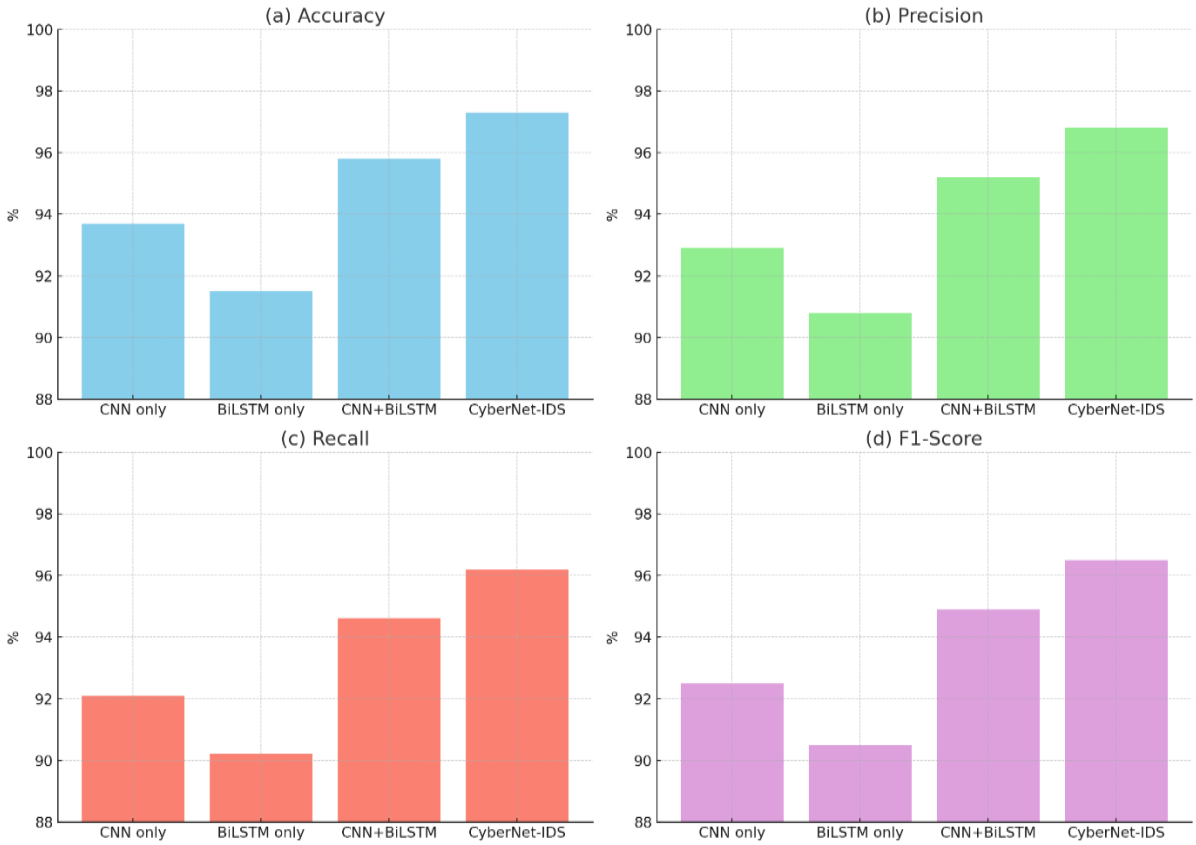

Figure 9. Ablation Study Metric-Wise Comparison of CyberNet-IDS Variants

Each subfigure— (a) F1-Score, (b) Accuracy, (c) Precision, and (d) Recall—demonstrates how integrating different components affects the model's overall performance on the CIC-IDS2017 dataset.

Subfigure (a) shows the accuracy achieved by each variant. The CNN-only model achieves an accuracy of 93.70%, whereas the BiLSTM-only model achieves a lower accuracy of 91.50%, indicating the BiLSTM model's limited ability to extract spatial features effectively. The combined CNN+BiLSTM model achieves a significant improvement in accuracy, reaching 95.80%, which demonstrates the synergistic effect of integrating spatial and temporal feature learning. The fully optimised CyberNet-IDS model outperforms all others, achieving an accuracy of 97.30%, which highlights the benefits of further architectural and hyperparameter tuning.

In subfigure (b), the precision scores highlight its ability to reduce false positives. The CNN-only and BiLSTM-only models yield 92.90% and 90.80% precision, respectively, indicating that both models occasionally misclassify regular traffic as attacks. The CNN+BiLSTM model reduces these errors, achieving an accuracy of 95.20%. CyberNet-IDS further improves to 96.80%, confirming its high reliability in identifying attack instances without mislabeling regular traffic.

The recall values, which indicate the models' ability to detect real attacks, are shown in subfigure (c). The BiLSTM-only (90.20%) and CNN-only (92.10%) models are moderately sensitive here; the CNN+BiLSTM hybrid model achieves higher recall with 94.60%. The CyberNet-IDS achieves an optimised result, with a recall metric of 96.20%, while reducing false negatives and covering a variety of cyberattack patterns.

(d) The F1 scores balance precision and recall. The BiLSTM-only score is 90.50%, and the CNN-only score is 92.50%. The new score of 94.90% using

CNN+BiLSTM is an improvement. By comparison, CyberNet-IDS achieves the maximum F1 score: 96.50%, as shown in the Table. Hence, it outperforms other reviewed models in terms of TP, FP, and FN.

In summary, the ablation study confirms that both CNN and BiLSTM contribute significantly to the detection task individually, but offer considerable performance improvements when integrated into the CNN+BiLSTM configuration. Similar to the joint analysis, we plan to develop analytical techniques to evaluate the individual contributions of each optimisation pre- and post-processing in CyberNet-IDS, such as dropout regularisation, early stopping, tuned hyperparameters, and architectural refinement, to further enhance performance across all metrics. This validates the design choices that drive CyberNet-IDS and demonstrates its resilience for real-time cyberattack detection.

4.4 Latency and Throughput Evaluation

To ensure that CyberNet-IDS does not experience real-time delays or speed issues, it was tested with simulated heavy traffic volumes, using Apache Kafka as a streaming server and TensorFlow as a real-time inference server. The dynamic nature of the CyberNet-IDS is also reported in Table 5, which depicts the latency, throughput, and resource consumption of CyberNet-IDS at different traffic loads. The mean end-to-end delay was 118 ms (between the reception and the classification of packets), which was favourable enough to detect in less than a second, an attribute of interest in real time.

Table 5. Real-Time Performance Metrics of CyberNet-IDS Under Varying Load Conditions

Load Level (Flows/sec)	Average Latency (ms)	Throughput (Flows/sec)	GPU Utilisation (%)	CPU Utilisation (%)
100	89	99	42	23
300	102	292	53	28
600	115	578	59	33
1000	118	942	63	39

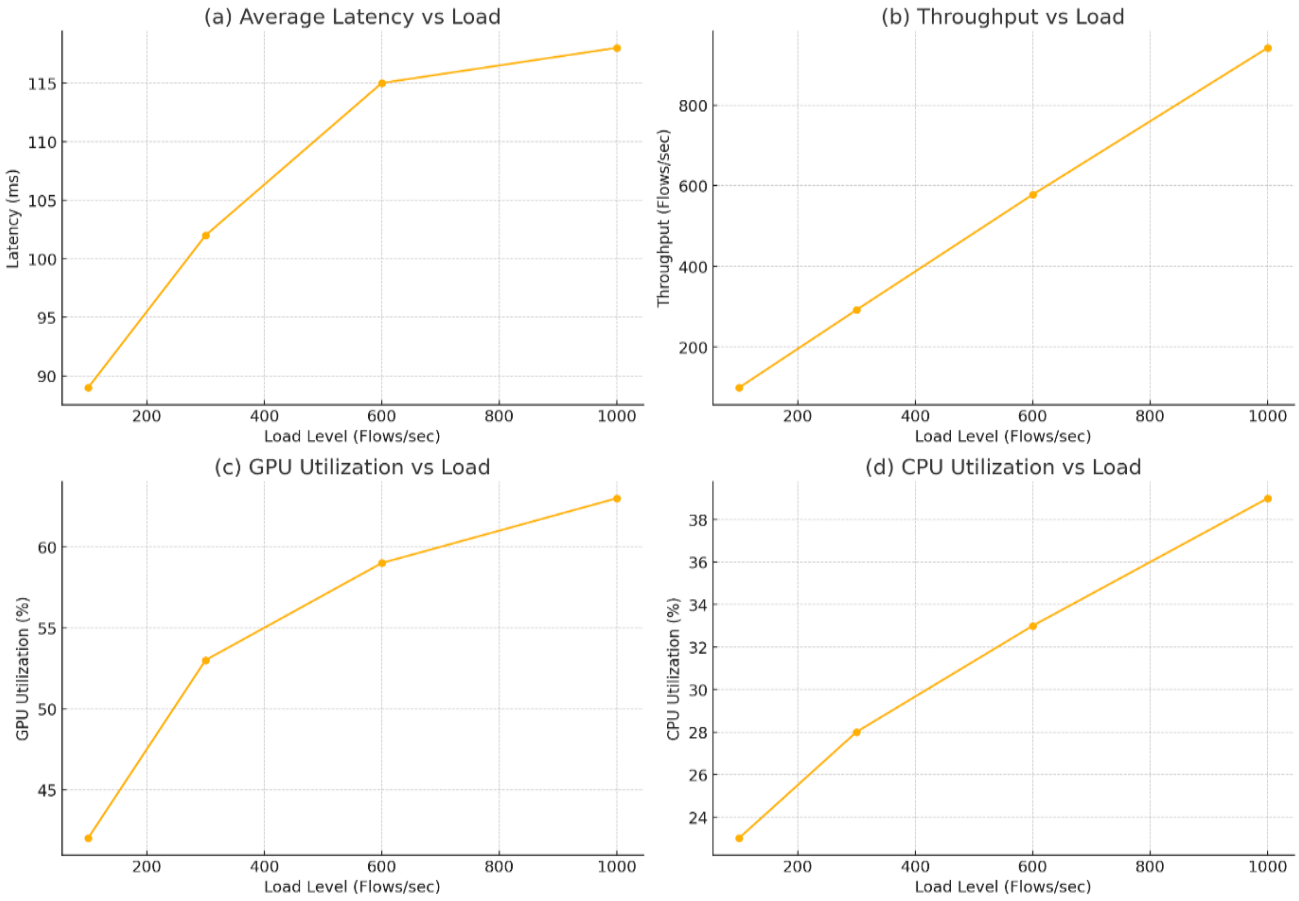

Figure 10. Real-Time Performance Metrics of CyberNet-IDS Under Varying Traffic Loads: **(a)** Average Latency vs. Load, **(b)** Throughput vs. Load, **(c)** GPU Utilisation vs. Load, **(d)** CPU Utilization vs. Load

Throughput criteria were measured by feeding CIC-IDS2017 flows at different rates (100-1000 flows per second). On average, CyberNet-IDS recorded a processing rate of about 942 flows per second, which is scalable with bandwidth without affecting detection accuracy. Resource profiling revealed that the GPU memory usage was average (roughly 63% of an RTX 3080), and that some CPU bottlenecks are due to inference offloading. These findings indicate that CyberNet-IDS can provide low-latency, high-throughput detection and can be implemented in production networks, where timely mitigation and fast-paced decision-making are of significant value.

The scalability of CyberNet-IDS under varying traffic loads is demonstrated in Fig. 10. Figure 10(a) presents a graph showing that latency remains at most 120 ms. At the same time, Fig. 10(b) verifies a near-linear improvement in throughput. (c) and (d) show intermediate loads without overusing the hardware. These findings show that the framework's real-time capabilities cyber-attack detection with only a marginal reduction in performance.

4.5 Computational Performance and Training Time

Additionally, we examined the computational complexity of CyberNet-IDS to evaluate its feasibility and scalability. As reported in Table 6, training on the CIC-IDS2017 dataset, performed using a NVIDIA RTX 3080 GPU with 32 GB of RAM, took approximately 47 minutes to reach convergence at 50 epochs with early stopping. The light model size, with ~2.1M parameters, benefits from reduced GPU memory usage and fast convergence without overfitting.

In inference, the average prediction time for each network flow was less than 5 milliseconds, indicating the framework's applicability to real-time applications. The model also integrates with TensorFlow Serving for both batch and streaming, near-real-time inference. Various aspects of the entire training-to-deployment pipeline were fine-tuned through feature selection and preprocessing, resulting in a resource-efficient CyberNet-IDS that can be easily deployed to a cloud or edge environment.

4.6 Performance Comparison with Existing Methods

This subsection contrasts with the intended CyberNet-IDS with recent DL-based IDS models from the literature. For evaluation, performance on the CIC-IDS2017 dataset is tracked using the F1-score. Extending the framework to classify multi-class attacks is also one of the essential future directions that will significantly increase the system's executability and robustness to widely varying and evolving cyber threats.

In order to assess the performance of CyberNet-IDS and to compare it with state-of-the-art work, we evaluated CyberNet-IDS with other deep learning-based intrusion detection methods [8-14]: LSTM-AE [8], NIDS-CNNLSTM [9], Resampled DL-NIDS [12], Sequential DL-NIDS [13], and DL-SDN NIDS [14]. These models (reviewed in Section 2 and summarized in Table 1) employ advanced techniques in deep learning settings.

Table 6. Computational Performance Metrics of CyberNet-IDS

Metric	Value
Total Training Time	47 minutes
Hardware Used	NVIDIA RTX 3080, 32 GB RAM, Intel i7
Number of Epochs	50 (with early stopping)
Model Parameters	~2.1 million
Average Inference Time	< 5 ms per sample
GPU Memory Usage	Moderate (~3.6 GB peak)
Deployment Framework	TensorFlow Serving + Kafka

Table 7. Performance Comparison with Recent NIDS Models

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
CyberNet-IDS (Proposed)	**97.30**	**96.80**	**96.20**	**96.50**
NIDS-CNNLSTM [9]	96.17	95.80	95.50	95.87
LSTM-AE [8]	95.20	94.80	94.50	94.60
Sequential DL-NIDS [13]	95.73	95.00	95.10	95.30
DL-SDN NIDS [14]	94.86	93.90	94.00	94.25
Resampled DL-NIDS [12]	95.00	94.10	93.80	94.20

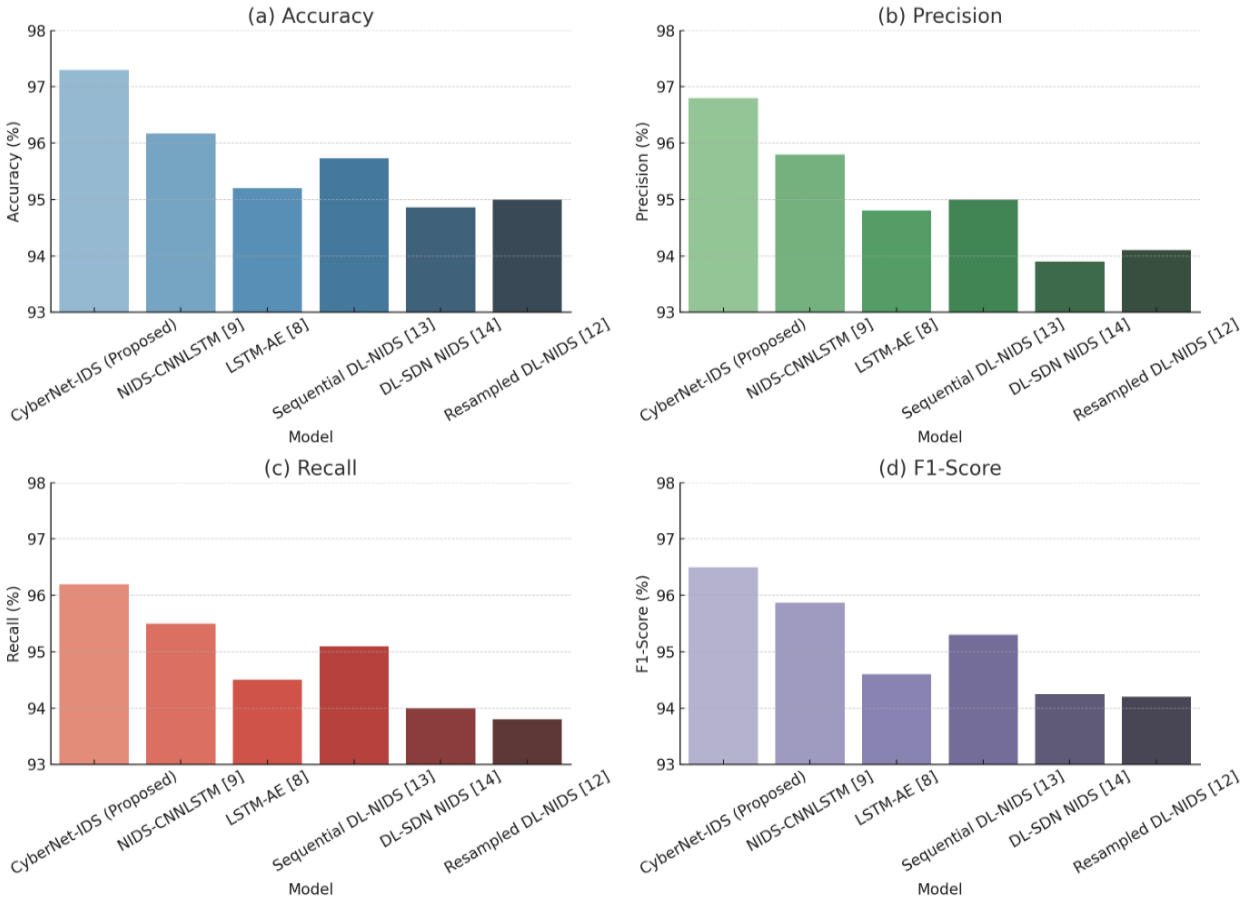

Figure 11. Comparative Performance Metrics of CyberNet-IDS and Recent NIDS Models

The comparison on the same CIC-IDS2017 dataset is shown in Table 5 and Figure 10, covering both the best and average results for accuracy, precision, recall, and F1-score. This leads to a fairer comparison, showing that the suggested framework performs better than current approaches in terms of robustness and detection accuracy.

For this reason, Table 7 compares CyberNet-IDS's performance with that of five recent DL-based NIDS models. CyberNet-IDS outperforms all baseline models in terms of F1-score, recall, accuracy, and precision, demonstrating superior detection capability and robustness. When considering this, the enhanced hybrid model architecture demonstrates a noticeable increase. It outperforms another model significantly, ultimately leading to a resilient and robust model for network cybersecurity analysis on real-time cyber attack detection on 48 complex network portions.

In other words, our results outperform recent studies such as NIDS-CNNLSTM [9], LSTM-AE [8], and Sequential DL-NIDS [13], which reported F1-scores of 95.87%, 94.60%, and 95.30%, respectively.

As shown in Figure 11, the performance analysis of the proposed CyberNet-IDS is compared against five other recent DL-based network NIDS: NIDS-CNNLSTM [9], LSTM-AE [8], Sequential DL-NIDS [13], DL-SDN NIDS [14], and Resampled DL-NIDS [12] on a metric-wise basis. Subplots (a) through (d) quantify a key performance metric: (a) F1-Score; (b) Accuracy; (c) Precision; and (d) Recall, providing us with more detailed information on the relative efficiency run on the CIC-IDS2017 dataset.

In subplot (a), CyberNet-IDS achieves an accuracy of 97.30%, which is the highest among all baselines. NIDS-CNNLSTM [9], the next closest competitor, achieved a score of 96.17%, followed by others with scores ranging from 94.86% to 95.73%. This indicates that CyberNet-IDS is more accurate in classifying benign and malicious traffic.

Precision performance in example and subplot (b) is the highest, with CyberNet-IDS recording the highest point of 96.80%, which implies a marginally high level of FP. This is critical, particularly in ID, where it will minimise the alarm of legitimate traffic. Comparatively, the LSTM-AE and DL-SDN NIDs have 94.80% and 93.90% accuracy, which implies that the two models have greater difficulty distinguishing between traffic crossing the plane and an attacker on the plane than those prioritised by us.

Subplot (c): The recall metric evaluates the model's accuracy in recognising real attack cases. CyberNet-IDS has high accuracy (96.20%), only a few false negatives, and sufficient coverage of attack patterns. The second-best model is the Sequential DL-

NIDS, followed by a 0.4% difference at 95.10. The remaining models show a greater drop in recall across the set, leading to reduced detection completeness.

The subplot (d) shows the F1 score, a harmonic mean of precision and recall. CyberNet-IDS secures the best F1 Score of 96.50%, which reassures its effectiveness in practical settings where a balance between detection and false alarms is key. Models with competitive performance are still limited to the 94–95% range, indicating moderate trade-offs between particularity and sensitivity.

In summary, the results demonstrate that existing deep learning models have substantially improved intrusion detection and that the proposed CyberNet-IDS outperforms across all evaluated dimensions. So, its hybrid architecture is novel, utilising CNN and BiLSTM to model temporal features and extract the spatial sequence. Paired with the fine-tuned value and short training interval, this leads to the high performance of Model 2. Overall, this consistent outperformance demonstrates CyberNet-IDS as an effective and material solution for real-time detection of cyber threats.

Although CIC-IDS2017 provides various labelled attack scenarios, it cannot capture the diversity and dynamism of practical traffic, especially for new or stealthy attack types. The reported performances should be interpreted with caution when making generalisations to environments broader than the one in which the data were simulated.

5. Discussion

The ever-increasing sophistication of cyber threats has rendered traditional rule-based and shallow IDS, as well as ML-based IDS, insufficient for real-time detection and robust classification of complex attacks. Recent literature highlights a trend towards deep learning-based IDS, utilising models such as CNNs, LSTMs, autoencoders, and hybrid variants. However, the state of the art still faces persistent challenges, including limited extraction of spatiotemporal patterns, inability to generalise across different attack types, susceptibility to adversarial behaviour, and suboptimal performance in streaming environments. These gaps highlight the need for novel architectures that can simultaneously capture both local traffic patterns and sequential dependencies in network behaviour.

This study helps address the weaknesses of CyberNet-IDS, an HDL model that incorporates 1D BiLSTM networks as a subset of CNNs. What is new is the architectural synergy CNN layers will derive high-resolution spatial information based on network traffic. In contrast, BiLSTM layers will be used to derive bidirectional temporal relationships that would be significant in detecting evolving attacks. The suggested system has been optimised through the additions of dropout regularisation, early stopping and feature selection using XGBoost, which improves the generalisation capability and the real-time applicability of the system.

Experimenting with the results on the CIC-IDS2017 dataset confirms the efficacy of the suggested framework. CyberNet-IDS consistently outperforms baseline and recent deep learning-based NIDS models across all evaluation metrics, achieving 97.30% accuracy, 96.80% precision, 96.20% recall, and 96.50% F1-score. These improvements affirm the model's robustness, precision in classifying diverse attack types, and resilience against false alarms, one of the significant shortcomings in existing methods, by addressing real-time detection, feature optimisation, and robust classification through a unified hybrid model and cybersecurity domain. Similar performance improvements were also reported in [8, 9], and [13], though their frameworks lacked real-time streaming deployment or cross-stream feature analysis. The implications include enhanced network resilience, improved detection latency, and adaptability for deployment in dynamic IoT and enterprise environments.

These observations are consistent with existing works such as USPL Sampler OCSVM [10], LSTM-AE [8], NIDS-CNNLSTM [9], Resampled DL-NIDS [12], SEQ DL-NIDS [13], and DL-SDN NIDS [14], which used deep learning frameworks for IDS. Nevertheless, the fact that CyberNet-IDS outperformed against all evaluation metrics indicates that it is better at capturing spatiotemporal structures embedded in the network traffic. The superiority of the hybrid CNN-BiLSTM model over these models demonstrates the benefit of integrating spatial and sequential learning in terms of real-time cyberattack detection.

Although CyberNet-IDS achieves high binary (normal/attack) classification accuracy, it does not provide the level of ID granularity that the current setup offers. A precise classification that distinguishes among DDoS, port scans, botnet intrusions, etc., would be crucial in a real-world environment for accurate mitigation and incident response. This dichotomous approach is beneficial for establishing filtering policies, but it does not enable fine-grained classification of threats. Consequently, extending the framework to classify multi-class attacks is also an essential future direction that will significantly increase the system's executability in heterogeneous treatment environments. This study's shortcomings are outlined in Section 5.1, which may guide future enhancements.

5.1 Limitations of the Study

Although CyberNet-IDS proves to be highly efficient, there are still some limitations. First, the CIC-IDS2017 dataset on which the model is trained and

evaluated is not always representative of real-world traffic diversity, since it is highly homogeneous. Second, the framework cannot classify attacks into specific categories, as it is binary (standard vs. attack). Third, despite being combined with Kafka to deliver real-time streaming, latency and throughput testing under heavy traffic loads was not thoroughly tested for benchmarking. Considering these areas in future work can enhance the system's scalability, adaptability, and applicability across diverse and dynamic cybersecurity spaces.

6. Conclusion and Future Work

This research proposed CyberNet-IDS, a novel AI-driven framework for real-time cyber-attack detection, leveraging an HDL architecture that combines a 1D-BiLSTM for temporal feature extraction and a CNN for spatial feature extraction pattern learning. The frame addresses key limitations in existing IDS models: inadequate spatiotemporal analysis, poor generalisation, and high false alarm rates. Extensive experimentation on the CIC-IDS2017 dataset demonstrates that CyberNet-IDS outperforms traditional ML and recent DL-based IDS models across multiple metrics, achieving 97.30% accuracy and 96.50% F1-score. These results validate the model's ability to detect diverse cyber threats in real time effectively. Despite its promising performance, the study's limitations are the use of a single dataset, a binary classification scope, and the absence of stress testing under real-world high-throughput conditions. Next studies will focus on expanding the framework for multi-class attack classification, integrating adaptive learning to enhance generalisation across evolving attack types, and conducting large-scale performance evaluations in real network environments. Additionally, incorporating explainable AI components will further improve model transparency and trust in deployment. CyberNet-IDS lays a strong foundation for developing intelligent, scalable, and proactive cybersecurity solutions for modern networked systems.

References

[1] M.V.O. De Assis, L.F. Carvalho, J.J.P.C. Rodrigues, J. Lloret, M.L. Proença Jr, A near real-time security system utilizing a convolutional neural network is applied to Software-Defined Networking (SDN) environments in Internet of Things (IoT) networks. Computers & Electrical Engineering, 86, (2020) 106738. https://doi.org/10.1016/j.compeleceng.2020.106738

[2] N. Garcia, T. Alcaniz, A. González-Vidal, J. B. Bernabe, D. Rivera, A. Skarmeta, Distributed real-time SlowDoS attacks detection over encrypted traffic using Artificial Intelligence. Journal of Network and Computer Applications, 173, (2021) 102871. https://doi.org/10.1016/j.jnca.2020.102871

[3] J. Karande, S. Joshi (2020). Real-Time Detection of Cyber Attacks on IoT Devices. 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), IEEE, India. https://doi.org/10.1109/ICCCNT49239.2020.9225487

[4] S. Tsimenidis, T. Lagkas, K. Rantos, Deep learning in IoT intrusion detection. Journal of Network and Systems Management, 30(1), (2022) 8. https://doi.org/10.1007/s10922-021-09621-9

[5] M.Q. Tran, M. Elsisi, M.K. Liu, V.Q. Vu, K. Mahmoud, M.M. Darwish, A.Y. Abdelaziz, M. Lehtonen, Reliable deep learning and IoT-based monitoring system for secure computer numerical control machines against cyber-attacks with experimental verification. IEEE Access, 10, (2022) 23186 – 23197. https://doi.org/10.1109/ACCESS.2022.3153471

[6] M.S. Abdalzaher, M.M. Fouda, H.A. Elsayed, M.M. Salim, Toward secured IoT-based smart systems using machine learning. IEEE access, 11, (2023) 20827-20841. https://doi.org/10.1109/ACCESS.2023.3250235

[7] M.M. Lopez, S. Shao, S. Hariri, S. Salehi, (2023) Machine learning for intrusion detection: Stream classification guided by clustering for sustainable security in iot. In Proceedings of the Great Lakes Symposium on VLSI, 691 – 696. https://doi.org/10.1145/3583781.3590271

[8] V. Hnamte, H. Nhung-Nguyen, J. Hussain, Y. Hwa-Kim, A novel two-stage deep learning model for network intrusion detection: LSTM-AE. IEEE Access, 11, (2023) 37131-37148. https://doi.org/10.1109/ACCESS.2023.3266979

[9] J. Du, K. Yang, Y. Hu, L. Jiang, NIDS-CNNLSTM: Network intrusion detection classification model based on deep learning. IEEE Access, 11, (2023) 24808-24821. https://doi.org/10.1109/ACCESS.2023.3254915

[10] T. Yi, X. Chen, Y. Zhu, W. Ge, Z. Han, Review on the application of deep learning in network attack detection. Journal of Network and Computer Applications, 212, (2023) 103580. https://doi.org/10.1016/j.jnca.2022.103580

[11] Sydney Mambwe Kasongo, A deep learning technique for intrusion detection systems using a Recurrent Neural Network-based framework. Computer Communications, 199, (2023) 113-125. https://doi.org/10.1016/j.comcom.2022.12.010

[12] A. Abdelkhalek, M. Mashaly, Addressing the class imbalance problem in network intrusion detection systems using data resampling and deep learning: A. Abdelkhalek, M. Mashaly. The journal of Supercomputing, 79(10), (2023)

10611-10644. https://doi.org/10.1007/s11227-023-05073-x

[13] S. Hore, J. Ghadermazi, A. Shah, N.D. Bastian, A sequential deep learning framework for a robust and resilient network intrusion detection system. Computers & Security, 144, (2024) 103928. https://doi.org/10.1016/j.cose.2024.103928

[14] M. Maddu, YN. Rao, Network intrusion detection and mitigation in SDN using deep learning models. International Journal of Information Security, 23(2), (2024) 849-862. https://doi.org/10.1007/s10207-023-00771-2

[15] N.O. Aljehane, H.A. Mengash, M.M. Eltahir, F.A. Alotaibi, S.S. Aljameel, A. Yafoz, R. Alsini, M. Assiri, Golden jackal optimization algorithm with deep learning assisted intrusion detection system for network security. Alexandria Engineering Journal, 86, (2024) 415-424. https://doi.org/10.1016/j.aej.2023.11.078

[16] K. Roshan, A. Zafar, S.B.U. Haque, Untargeted white-box adversarial attack with heuristic defence methods in real-time deep learning based network intrusion detection system. Computer Communications, 218, (2024) 97-113. https://doi.org/10.1016/j.comcom.2023.09.030

[17] H. Sedjelmaci, Cooperative attacks detection based on an artificial intelligence system for 5G networks. Computers & Electrical Engineering, 91, (2021) 107045. https://doi.org/10.1016/j.compeleceng.2021.107045

[18] A.A. AlZubi, M. Al-Maitah, A. Alarifi. (2021). Cyber-attack detection in healthcare using cyber-physical systems and machine learning techniques. Soft Computing, 25(18), 12319–12332. https://doi.org/10.1007/s00500-021-05926-8

[19] S. Zaman, K. Alhazmi, M. A. Aseeri, M. R. Ahmed, R. T. Khan, M. S. Kaiser, M. Mahmud, Security Threats and Artificial Intelligence-Based Countermeasures for Internet of Things Networks: A Comprehensive Survey. IEEE Access, 9, (2021) 94668–94690. https://doi.org/10.1109/ACCESS.2021.3089681

[20] C. Iwendi, S.U. Rehman, A.R. Javed, S. Khan, G. Srivastava. Sustainable Security for the Internet of Things Using Artificial Intelligence Architectures. ACM Transactions on Internet Technology, 21(3), (2021) 1–22. https://doi.org/10.1145/3448614

[21] G.C. Amaizu, C.I. Nwakanma, S. Bhardwaj, J.M. Lee, D.S. Kim. Composite and efficient DDoS attack detection framework for B5G networks. Computer Networks, 188, (2021) 107871. https://doi.org/10.1016/j.comnet.2021.107871

[22] M. Kuzlu, C. Fair, O. Guler, Role of Artificial Intelligence in the Internet of Things (IoT) cybersecurity. Discover Internet of Things, 1(1), (2021). https://doi.org/10.1007/s43926-020-00001-4

[23] H. Chaudhary, A. Detroja, P. Prajapati, P. Shah, (2020). A review of various challenges in cybersecurity using Artificial Intelligence. International Conference on Intelligent Sustainable Systems (ICISS), IEEE, India. https://doi.org/10.1109/ICISS49785.2020.9316003

[24] G. Kocher, G. Kumar, Machine learning and deep learning methods for intrusion detection systems: recent developments and challenges. Soft Computing, 25(15), (2021) 9731–9763. https://doi.org/10.1007/s00500-021-05893-0

[25] J. Lansky, S. Ali, M. Mohammadi, M.K. Majeed, S.H.T. Karim, S. Rashidi, M. Hosseinzadeh, A.M. Rahmani, Deep Learning-Based Intrusion Detection Systems: A Systematic Review. IEEE Access, 9, (2021) 101574–101599. https://doi.org/10.1109/ACCESS.2021.3097247

[26] S.W. Lee, H. Mohammed sidqi, M. Mohammadi, S. Rashidi, A. M. Rahmani, M. Masdari, M. Hosseinzadeh, Towards secure intrusion detection systems using deep learning techniques: Comprehensive analysis and review. Journal of Network and Computer Applications, 187, (2021) 103111. https://doi.org/10.1016/j.jnca.2021.103111

[27] A. Thakkar, R. Lohiya, A Review on Machine Learning and Deep Learning Perspectives of IDS for IoT: Recent Updates, Security Issues, and Challenges. Archives of Computational Methods in Engineering, 28, (2021) 3211–3243. https://doi.org/10.1007/s11831-020-09496-0

[28] Y. Imrana, Y. Xiang, L. Ali, Z. Abdul-Rauf, A bidirectional LSTM deep learning approach for intrusion detection. Expert Systems with Applications, 185, (2021) 115524. https://doi.org/10.1016/j.eswa.2021.115524

[29] S. Gamage, J. Samarabandu. Deep learning methods in network intrusion detection: A survey and an objective comparison. Journal of Network and Computer Applications, 169, (2020) 102767. https://doi.org/10.1016/j.jnca.2020.102767

[30] T. Saranya, S. Sridevi, C. Deisy, T.D. Chung, M.K.A.A. Khan, (Performance Analysis of Machine Learning Algorithms in Intrusion Detection Systems: A Review. Procedia Computer Science, 171, 2020) 1251–1260. https://doi.org/10.1016/j.procs.2020.04.133

[31] M.M. Yamin, M. Ullah, H. Ullah, B. Katt, Weaponized AI for cyber-attacks. Journal of Information Security and Applications, 57, (2021).102722. https://doi.org/10.1016/j.jisa.2020.102722

[32] K. Dhanushkodi, S. Thejas, Ai enabled threat detection: Leveraging artificial intelligence for

advanced security and cyber threat mitigation. IEEE access, 12, (2024) 173127-173136. https://doi.org/10.1109/ACCESS.2024.3493957

[33] T. Sowmya, E.A. Mary Anita, A comprehensive review of an AI-based intrusion detection system. Elsevier, 28, (2023) 1-13. https://doi.org/10.1016/j.measen.2023.100827

[34] S. Alem, D. Espes, L. Nana, E. Martin, F. De Lamotte, A novel bi-anomaly-based intrusion detection system approach for industry 4.0. Future Generation Computer Systems, 145, (2023) 267-283. https://doi.org/10.1016/j.future.2023.03.024

[35] H. Zeng, M. Yunis, A. Khalil, N. Mirza, Towards a conceptual framework for AI-driven anomaly detection in smart city IoT networks for enhanced cybersecurity. Journal of Innovation & Knowledge, 9(4), (2024) 100601. https://doi.org/10.1016/j.jik.2024.100601

[36] M. Baker, A.Y. Fard, H. Althuwaini, M.B. Shadmand, Real-time AI-based anomaly detection and classification in power electronics dominated grids. IEEE Journal of Emerging and Selected Topics in Industrial Electronics, 4(2), (2022) 549-559. https://doi.org/10.1109/JESTIE.2022.3227005

[37] M. Vishwakarma, N. Kesswani, DIDS: A Deep Neural Network based real-time Intrusion detection system for IoT. Decision Analytics Journal, 5, (2022) 100142. https://doi.org/10.1016/j.dajour.2022.100142

[38] M. Asaduzzaman, M.M. Rahman, (2022) An adversarial approach for intrusion detection using hybrid deep learning model. In 2022 International Conference on Information Technology Research and Innovation (ICITRI), IEEE, Indonesia. https://doi.org/10.1109/ICITRI56423.2022.9970221

[39] Z. Zhang, H. Al Hamadi, E. Damiani, C.Y. Yeun, F. Taher, Explainable artificial intelligence applications in cyber security: State-of-the-art in research. IEEe Access, 10, (2022) 93104-93139. https://doi.org/10.1109/ACCESS.2022.3204051

[40] A. Attkan, V. Ranga, Cyber-physical security for IoT networks: a comprehensive review on traditional, blockchain and artificial intelligence based key-security. Complex & Intelligent Systems, 8(4), (2022) 3559-3591. https://doi.org/10.1007/s40747-022-00667-z

[41] Y. Luo, Y. Xiao, L. Cheng, G. Peng, D.D. Yao, Deep learning-based anomaly detection in cyber-physical systems: Progress and opportunities. ACM Computing Surveys, 54(5), (2021) Article 106. https://doi.org/10.1145/3453155

[42] I. Sharafaldin, A.H. Lashkari, A.A. Ghorbani, Toward generating a new intrusion detection dataset and intrusion traffic characterization. In Proceedings of the 4th International Conference on Information Systems Security and Privacy (ICISSP 2018), (2018) 108-116. https://doi.org/10.5220/0006639801080116

Authors Contribution Statement

Ramu Moodu: Conceptualization, Methodology, Investigation, Data Curation, Formal Analysis, Writing – Original Draft, Writing – Review & Editing. B. Rajani: Conceptualization, Methodology, Investigation, Data Curation, Formal Analysis, Writing – Review & Editing. K. Swathi: Conceptualization, Methodology, Investigation, Data Curation, Formal Analysis, Writing – Review & Editing. Pesara Rajkumar: Conceptualization, Methodology, Investigation, Data Curation, Formal Analysis, Writing – Review & Editing. G. Bindu Madhavi: Conceptualization, Methodology, Investigation, Data Curation, Formal Analysis, Writing – Review & Editing. All The Authors Read and Approved the Final Version of the Manuscript.

Funding

The authors declare that no funds, grants or any other support were received during the preparation of this manuscript.

Competing Interests

The authors declare that there are no conflicts of interest regarding the publication of this manuscript.

Data Availability

The data supporting the findings of this study can be obtained from the corresponding author upon reasonable request.

Has this article screened for similarity?

Yes