Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2023.046101

ARTICLE

Cybernet Model: A New Deep Learning Model for Cyber DDoS Attacks Detection and Recognition

Azar Abid Salih[1,*] and Maiwan Bahjat Abdulrazaq[2]

[1]Department of Information Technology, Technical College of Duhok, Duhok Polytechnic University, Duhok, Iraq

[2]Department of Computer Science, Faculty of Science, University of Zakho, Duhok, Iraq

*Corresponding Author: Azar Abid Salih. Email: azar.abid@dpu.edu.krd

Received: 19 September 2023 Accepted: 30 November 2023 Published: 30 January 2024

ABSTRACT

Cyberspace is extremely dynamic, with new attacks arising daily. Protecting cybersecurity controls is vital for network security. Deep Learning (DL) models find widespread use across various fields, with cybersecurity being one of the most crucial due to their rapid cyberattack detection capabilities on networks and hosts. The capabilities of DL in feature learning and analyzing extensive data volumes lead to the recognition of network traffic patterns. This study presents novel lightweight DL models, known as Cybernet models, for the detection and recognition of various cyber Distributed Denial of Service (DDoS) attacks. These models were constructed to have a reasonable number of learnable parameters, i.e., less than 225,000, hence the name "lightweight." This not only helps reduce the number of computations required but also results in faster training and inference times. Additionally, these models were designed to extract features in parallel from 1D Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM), which makes them unique compared to earlier existing architectures and results in better performance measures. To validate their robustness and effectiveness, they were tested on the CIC-DDoS2019 dataset, which is an imbalanced and large dataset that contains different types of DDoS attacks. Experimental results revealed that both models yielded promising results, with 99.99% for the detection model and 99.76% for the recognition model in terms of accuracy, precision, recall, and F1 score. Furthermore, they outperformed the existing state-of-the-art models proposed for the same task. Thus, the proposed models can be used in cyber security research domains to successfully identify different types of attacks with a high detection and recognition rate.

KEYWORDS

Deep learning; CNN; LSTM; Cybernet model; DDoS recognition

1 Introduction

In recent times, the threat of cyberattacks has grown due to vulnerabilities in some internet-connected devices, making them attractive targets for malicious activities. Cyberattacks can lead to the exposure of sensitive user information and significant damage to critical infrastructure [1]. DDoS attacks, one of the most common types of cyberattacks, present significant challenges compared to other harmful cyberattacks. Detecting these attacks has proven difficult due to their rapid expansion and the complexity involved in their detection, making them a challenging area for researchers [2]. Moreover, DDoS attacks involve multiple distributed threats that aim to disrupt or disable internet-connected hosts' services. Typically, these attacks target specific applications, such as the web servers of banks, organizations, online shopping, and credit card payment networks [3]. To protect against various DDoS attacks, it is crucial to build robust systems. An effective cyberattack detection system requires a DL model that minimizes false alarms and achieves high detection accuracy. The method of learning, which can be either supervised or unsupervised, has been applied to improve the performance of models in cyber security [4].

DDoS attacks can be detected and recognized through the application of various DL architectures or the development of innovative network models, contributing to enhanced cybersecurity measures. Within the era of data science, the identification of cyber DDoS attacks seamlessly aligns with the realm of big data analytics [5]. Given the vast datasets prevalent in cybersecurity, accurately characterizing intricate network traffic patterns replete with complex features and values presents a difficult challenge. Previously, traditional machine learning algorithms carried out the task of attack detection, applying the rules to a small amount of data [6]. These algorithms caused the system to make a lot of false-positives and misclassifications, making security administration harder and causing more damage to systems than DL models with high-performance detection and satisfactory results [7]. DL is a crucial technique for addressing difficult challenges with voluminous datasets and demanding computational requirements because it enables the solution of complex problems. It can be used to meet the needs of the cybersecurity domain for cyber-attack detection [8]. A main contribution within the realm of DL resides in its capacity for feature learning. Also, extract information from incomplete data and extract features in addition to classifying the data. This proficiency is deeply embedded in the training phase, utilizing multiple hidden layers and parameters for complex mathematical computations [9]. This work introduces novel DL models that incorporate both 1D CNNs and LSTM networks for the purpose of cyber security for binary detection and multi-attack recognition.

The main contribution of this study is delineated as follows:

- Two new DL-based cyberattacks models were proposed, both utilize a 1D CNN–LSTM. Notably, our new approach, rather than utilizing the output of an LSTM layer as input solely for a subsequent 1D CNN layer, employs a novel architectural configuration wherein both the LSTM and 1D CNN layers operate on the same input data in parallel. Subsequently, the output representations generated by these parallel processing pathways are combined through an element-wise addition operation. This same procedure is applied to the dense layers, further enhancing the model's performance.
- The models proposed were for binary and multiclass classification tasks, in which the binary ones detected the instances as either malicious or benign. Whereas the second one classify the instances into 12 categories, 11 of them are attacks and one is benign.
- These new models were tested on the large dataset CIC-DDoS2019. Notably, these models exhibited superior detection and recognition capabilities, particularly when applied to unseen data, surpassing the performance of most existing models.
- The developed models are lightweight, whereas most of the existing DL-based models were constructed using millions of parameters and are inappropriate for devices with limited resources.
- To demonstrate the effectiveness of the suggested models, the performance evaluation of the model has been compared with results from baseline models and previous research.

The rest of this paper is structured as follows: Section 2–Related Works, Section 3–Background. Section 4–Proposed Methodology, Section 5–Experimental Results and Discussion, Section 6–Comparison Analysis, Section 7–Limitation of the Study, Section 8–Validity Threats. Finally, Section 9 addressed the conclusion.

2 Related Works

The DL is obtaining more widespread attention and applications across diverse research domains due to its exceptional performance. Cyberattacks on computer networks remain a relevant and challenging area of study, evident in daily incident reports that highlight the evolving strategies of cyber attackers. This section explores relevant literature on DL-based cyberattack detection.

Elsayed et al. [10] introduced the "DDoSNet" model, a novel attack detection system designed for SDN environments. Built on DL techniques, DDoSNet combines Recurrent Neural Networks (RNN) autoencoders with a softmax regression model at the output layer to binary classify the network traffic as malicious or normal. In the evaluation, the comprehensive CIC-DDoS2019 dataset is utilized, employing balanced data techniques. The total number of instances used is 230,673 samples. The results showed that the overall accuracy of the model was 99%.

In Chartuni et al. [11], the aim of the study was to select a representative dataset of DDoS attack events, preprocess it, and develop a sequential neural network model for multi-class classification. The model employs 78 features as input. The SMOTE was applied as a class-balancing technique. For the built model, three dense layers are connected. The hidden layer structure consists of neurons distributed as (128, 256, 512), respectively, and the model is trained across 10 classes, with three similar classes combined as one group based on the similarity of features in the dataset. The CIC-DDoS2019 dataset is utilized, and the proposed model achieves approximately 94% precision, accuracy, recall, and F1 score.

Wei et al. [12] proposed an AE-MLP model that combines an AE for automated feature extraction and a MLP network for classification. The latent space of the AE extracts 24 features, while the optimized MLP architecture consists of five layers: one input, three hidden, and one output layer. The testing dataset encompasses 5% of the original CIC-DDoS2019. The proposed model effectively classifies five types of attacks and has been rigorously tested on six subsets of DDoS attack samples. Notably, it achieves remarkable performance, with subset 6 exhibiting an accuracy of 98.34% based on 977,239 instances.

Alghazzawi et al. [13] presented a hybrid model that combines CNN with BiLSTM for the prediction and classification of DDoS attacks. This model is trained and evaluated using the recently introduced CIC-DDoS2019 dataset. The proposed model objective is to classify instances as either normal (T1) or attack (T2). In the feature selection process, the top 10 features were ranked based on their high scores. In the testing phase randomly selected 100 records from the dataset to predict the labels. The accuracy achieved for this prediction was 94.52%.

Nie et al. [14] developed a robust attack detection approach for CEC-based SIoT structured around three main phases. Initially, employed a feature selection module to process collaborative edge network traffic and selected 10 highly ranked features using Information Gain (IG) analysis. Subsequently, a DL architecture utilizing Generative Adversarial Networks (GAN) is designed to detect individual attacks. The effectiveness of our approach is evaluated using the CSE-CIC-IDS2018 and CIC-DDoS2019 datasets. In the case of the CIC-DDoS2019 dataset, which involves 13 classes, the proposed methodology was evaluated across 531,819 instances. The proposed model showed that it achieved 98.53% accuracy.

Boonchai et al. [15] utilized DNN models for robust multiclass (13-class) classification capabilities in DDoS scenarios. In this study, two models were implemented: one with a simple DNN structure and the other using a CNN-AE architecture. The DNN structure comprises six sequentially dense layers. The CNN-AE model structure, including convolutional, max pooling, and upsampling layers, is guided by an AE technique. Both models were applied to the CIC-DDoS2019 dataset. The dataset consists of 83 attributes related to DDoS attacks. After the data was preprocessed, it was also chosen at 50,000 records per class. The accuracy achieved by these models is notable, reaching up to 87% and 91.9%, respectively. In terms of CNN-AE's performance, it demonstrated lower prediction rates for specific attack types: LDAP (0.76), MSSQL (0.77), and SSDP (0.80).

Haq et al. [16] developed two advanced DNN models, DNNBoT1 and DNNBoT2, for detecting IoT botnet attacks like Mirai and BASHLITE. The models were constructed with a sophisticated architecture, incorporating a fifth layer that utilized a kernel initializer, followed by a dense layer with a softmax function for classification in the sixth layers. By employing PCA for feature extraction, they achieved high accuracy; DNNBoT1 and DNNBoT2 achieved accuracy rates of 90.71% and 91.44%, respectively, on the N-BaIoT data set collected from nine compromised industrial IoT devices. This dataset consisted of 1,486,418 instances of normal and attack occurrences, each characterized by nearly 58 features.

Srinivas et al. [17] presented a hybrid technique named DFNN-SAE-DCGAN, combining three deep learning models. In this framework, the Deep Convolutional Generative Adversarial Networks (DCGAN) component efficiently classifies various DDoS attack types using characteristic sets generated by the DFNN-SAE. Their experiments resulted in an impressive accuracy rate of 98.5% when applied to the CIC-DDoS2019 dataset. The optimal DFNN-SAE architecture consisted of 21 features in the final hidden layer. The study employed only 10% of the original dataset for training and testing.

Zainudin et al. [18] applied feature selection technique Extreme Gradient Boosting (XGBoost), to identify the top 10 relevant features. These selected features are then used in conjunction with a hybrid (CNN-LSTM) model for DDoS attack classification. Different subsets of data were used to evaluate the model. The CIC-DDoS2019 dataset, which includes three types of DDoS attacks (DNS, UDP, and SYN) as well as benign, was utilized to evaluate the model's performance. The dataset included 138,839 samples, and the results reveal that the proposed model achieves an impressive accuracy of 99.5%.

Kumar et al. [19] employed the LSTM DL architecture to detect DDoS attacks using the CIC-DDoS2019 dataset. This binary classification task distinguishes between "Benign" and "Attacks" with the utilization of 70 features. Our experimental trials with the suggested LSTM model, utilizing an epoch value of 25, resulted in the highest accuracy of 98.6%.

Even though the existing DL models have shown promising outcomes in previous studies, there are still a number of limitations that need to be resolved. First, most of these previous models require millions of parameters that need to be tweaked during the training stage. This makes them computationally expensive for both training and inference stages while being impractical for devices with constrained resources. Furthermore, many of these existing models were primarily designed for binary class classification, which fails to include all categories of DDoS attacks. Moreover, even those models attempting multiclass classification often overlook certain DDoS attack types. For instance, classes such as DNS, LDAP, and NetBIOS, as well as UDP and SSDP [11] classes, tend to be marginalized due to their similar behavioral characteristics. Finally, existing models frequently employ a combination of different architectures, such as LSTM and 1D-CNN, sequentially to enhance accuracy. However, to the best of our knowledge, no studies have explored the simultaneous utilization of these architectures–that is, feeding the same input into both LSTM and 1D-CNN–and subsequently adding their outputs to extract crucial features that significantly contribute to distinguishing between the various classes of DDoS attacks. Motivated by the above facts, we aim in this research to develop a new DL model that addresses the above-mentioned issues in the area of cybersecurity.

3 Background

3.1 Cyber DDoS Attack

Recent cyberattacks have become a critical problem in the technology era. According to a CSO study, by 2021 the yearly cost of cybercrime harm will be beat trillions of dollars [20]. One of the most dangerous cyberattacks on the internet is DDoS. This type of attack is able to cause serious damage to network infrastructure and can affect any server, causing connectivity issues or even complete loss of service. It was accomplished by using vulnerabilities in systems, applications, or protocols [21].

DDOS can flood the network, systems, or servers with traffic to drain out the bandwidth and affect the resources. Multiple attackers carry out a DDoS attack on a system, which floods it with more incoming requests. The destination system cannot immediately respond to the huge requests due to this server's crashes or breaks [22]. DDoS attacks encompass various strategies. Firstly, volume-based attacks saturate bandwidth; secondly, protocol attacks exploit protocol stack weaknesses in the network layer; and finally, application layer attacks flood networks with requests, rendering them unresponsive to legitimate traffic. The fact that DDoS employs a variety of attack strategies together with a number of potential combinations makes it difficult to detect such security mechanism scenarios in order to protect a network from this threat [23]. The solution to prevent DDOS attacks is real-time packet monitoring of network traffic by a security administration to store traffic data. As packets reach the system, they are analyzed based on several rules, and the possibly harmful ones are discarded [24]. Building a DDoS protection system that employs DL to recognize content with malicious intent that appears to be legitimate without the need for any human involvement defends against both protocol and volumetric attacks [25].

3.2 Convolution 1D CNN

CNNs represent a distinct class of feed-forward deep neural networks specifically designed for data processing. Within the CNN framework, multiple convolutional layers are systematically stacked, each endowed with the ability to distinguish increasingly complicated patterns. CNNs fall under the expansive umbrella of DL, a domain that has reaped significant scholarly attention in recent times. The CNN architecture encompasses three distinct types of layers: convolutional layers, pooling layers, and fully connected layers. CNN1D is an architecture designed specifically for processing one-dimensional data, such as time series or sequential data. It utilizes convolutional layers to automatically learn relevant features from the input sequence, making it highly effective in tasks that involve pattern recognition. The 1D CNN is constructed using the following hyperparameters: the number of CNN layers, the number of neurons in each layer, the number and size of the filters, and the subsampling factor of each layer. The kernel will slide through a list of objects in sequence 1D, and the filter in the convolution slides along one dimension only [26]. The primary function of 1D CNN is to extract hierarchical representations of sequential data through convolutional filters. By combining these features with pooling layers and fully connected layers, 1D CNN models can perform tasks such as classification, regression, and sequence generation with remarkable accuracy. The most important point is that CNN used to reduce the number of features and automatically detect and extract important features [27]. 1D CNN has been recently used for attack detection. Its utilization in cybersecurity involves automatically learning and identifying complex patterns within sequential data, such as network traffic or system logs, to recognize potential threats [28].

The 1D CNN technique uses the kernel weights for each convolution as the trainable weights for the convolutional neural network. The fundamental concept of performing a single convolution operation on a one-dimensional array is shown in Fig. 1. The given example multiplies the input array by the trainable weights for each output element. Based on the value of the loss function, the kernel weights are adjusted using backpropagation [29].

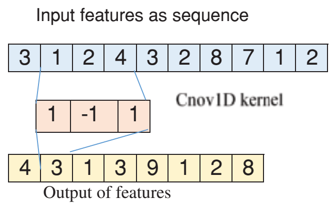

Figure 1: The operation of a single convolution unit

3.3 Long Short Term (LSTM)

LSTMs are special types of Recurrent Neural Network (RNN) architecture that can handle long-term dependencies resolved without being impacted by an unstable gradient. The gradient vanishing/exploding problem can affect the training of traditional RNNs. It is worth noting that in comparison to an LSTM, which is computationally intensive, an RNN features approximately four times as many trainable weights. A distinguishing feature of LSTM is the presence of a horizontal line, known as the cell state. The LSTM eliminates or adds information to the cell state by using three gates [30]. In the context of DDoS attack detection, LSTMs can be employed to analyze network traffic data over time and identify patterns that may indicate an ongoing attack. The ability of LSTMs to capture and learn from long-term dependencies in network traffic data makes them a valuable technique for enhancing the resilience of systems against cyber threats. LSTM has the capability to extract features and can also find application in the detection, recognition, and prediction of DDoS attacks through the design of a neural network architecture [31].

The LSTM architecture encompasses three crucial gates: input gates represented by $I(t)$, with its primary function being the discernment of permissible data for network entry. This gate computes the relevant information to be incorporated into the LSTM's cell state, leveraging both the antecedent hidden state and new input data. The gate's utility facilitated through the sigmoid function (σ), which assigns a value within the range of 0 to 1, followed by the tanh function that imparts weighting to values spanning from -1 to 1. The forget gate, denoted as $F(t)$, plays a pivotal role in determining which values to exclude. It employs a sigmoid function that considers both the previous state ($h_{(t-1)}$) and the input (xt) to assign a value ranging from 0 to 1 for each element within $C_{(t-1)}$, thus effectively managing information retention or exclusion. The final gate, is the output gate, denoted as $O(t)$, and assumes responsibility for shaping the new hidden state. This block's input governs the output by employing a sigmoid function to allocate a value within the range from 0 to 1. Subsequently, this value is multiplied by a tanh function, determining its degree of significance, and thereby assigning a value spanning from -1 to 1. If the sigmoid function's value approximates 1, it signifies data retention;

conversely, proximity to 0 signifies data exclusion. The Eqs. (1)–(8) of LSTM architecture as shown below:

$$F\left(t\right)=\sigma\left(w_{f}\left[h_{\left(t-1\right)},\ X_{t}\right]+b_{f}\right. \tag{1}$$

$$I\left(t\right)=\sigma\left(w_{i}\left[h_{\left(t-1\right)},\ X_{t}\right]+b_{i}\right. \tag{2}$$

$$O\left(t\right)=\sigma\left(wo\left[h_{\left(t-1\right)}\right],\ X_{t}\right]+b_{o} \tag{3}$$

$$f\left(x\right)=1/1+e^{axt} \tag{4}$$

$$\tanh\left(x\right)=\left(2/1+e^{\left(-2x\right)}\right)-1 \tag{5}$$

$$\hat{C}_{t}=\tanh\left(w_{c}\left[h_{\left(t-1\right)}\right],+b_{c}\right) \tag{6}$$

$$C_{t}=F_{t}.C_{\left(t-1\right)}+I_{t}.\hat{C}_{t} \tag{7}$$

$$h_{t}=O_{t}.\tanh\left(C_{t}\right) \tag{8}$$

These components are symbolically denoted as the cell state (C), the weight coefficient matrices (W) and bias terms (b). The activation functions in play are sigmoid and hyperbolic tangent (tanh). In this context, $h_{(t-1)}$ signifies the previous hidden state output, α represents the learning rate, X_t denotes the current input, and the biases for the forget (b_f), input (b_i), and output (b_o) gates are integral [19,32].

3.4 Dataset Description

The CIC-DDoS2019 dataset, the most recent version published in Cybersecurity, was developed by researchers from the Canadian Institute for Cybersecurity (CIC) and the "University of New Brunswick", in order to provide the research community with high-quality datasets that have been useful for the past few years in assisting analysts and cybersecurity experts in understanding and mitigating DDoS threats [33]. The dataset consists of 87 features, encompassing label a total of 88 columns, and it has been extracted employing CICFlowMeter tools for the acquisition of traffic data. The dataset includes more than a hundred thousand benign traffic samples and over fifty million malicious flows, distributed over 13 classes of recent DDoS attacks with one benign [34]. The DDoS attacks can be carried out in the application layer via transport layer protocols such as Transmission Control Protocol (TCP) and User Datagram Protocol (UDP). The dataset was collected on two separate days for training and testing evaluation, both of which provide a compilation of information exceeding 20 GB [10,35].

Fig. 2 illustrates the distribution of various attacks within the dataset, categorized according to exploitation-based and reflection-based attack types. In reflection-based DDoS, attacks are categorized into TCP,UDP or TCP/UDP. The exploitation-based DDoS in this category, attacks are subcategorized into TCP or UDP. Notably, LDAP, MSSQL, NetBIOS, SSDP, and SYN DDoS attacks showed high-volume traffic and noisy data. In contrast, UDP-Lag and Web DDoS attacks displayed low-volume traffic. It is worth mentioning that the authors excluded the WebDDoS attack due to its limited recording [36].

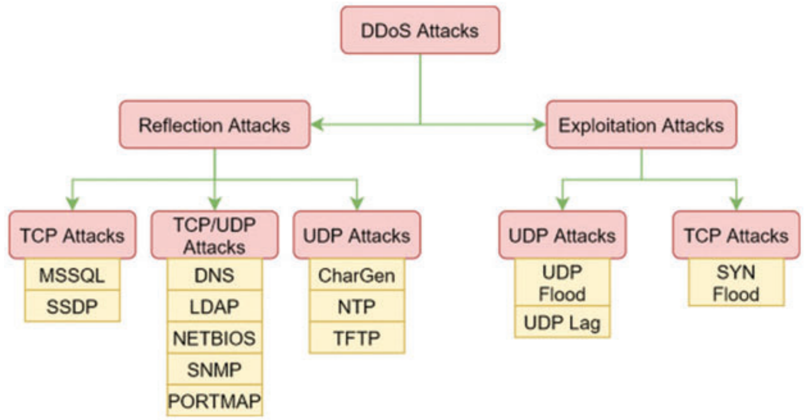

Figure 2: The distribution of DDoS attack throughout the CIC-DDoS2019 dataset

4 Proposed Methodology

In this section, the proposed model, starting with dataset preparation and the preprocessing procedures applied to the dataset, is described. Then, the building Cybernet model was presented. In Fig. 3, the general framework of the proposed DL model is deployed on the datasets to evaluate their performance in detecting and recognizing DDoS attacks.

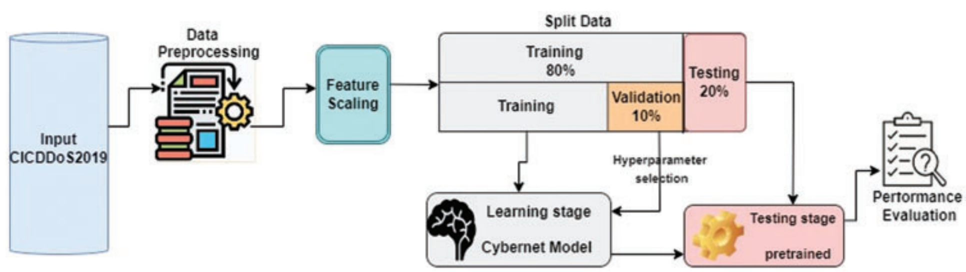

Figure 3: Outline of the proposed methodology

4.1 Data Preparation

In the context of this research, a newly released, extensive dataset containing the latest DDoS attack information is considered. Specifically, the CICDDoS2019 dataset, a recently published resource, has been chosen. This dataset encompasses both benign and the most current realistic DDoS traffic profiles, closely emulating real-world data. In the first step, all recorded files are concatenated into one dataset. To import data and read it to feed the model, the data must be in a standard format in the form of a CSV file. It is essential to note the presence of various data types, including categorical and numerical data. Gathering network traffic data from diverse sources inevitably introduces challenges such as poor or noisy data, incomplete data, and instance inconsistencies, necessitating data transformation to ensure uniform formatting, especially when dealing with extensive datasets. The data collection carried out by Python software is needed to import and analyze the dataset in the next stage of data preprocessing.

4.2 Data Preprocessing

The data preprocessing operations performed on the data set are presented below. The following stage is generalizing the data set and making it standardized to prepare the data before the model training to achieve accurate results.

4.2.1 Remove Irrelevant or Socket Features

The initial step involves removing irrelevant socket information features, such as Unnamed:0, Source Port, Destination Port, Flow ID, Source IP, Destination IP, Timestamp, and SimilarHTTP, as they do not significantly contribute to DDoS attack detection. These features exhibit variations across different networks, necessitating the model's training on packet characteristics themselves. Subsequently, after excluding these eight features, we are left with 79 relevant features for the model's input out of 87. Furthermore, both the attacker and normal users may have the same IP address. Therefore, training the DL model with socket feature values can cause an overfitting problem [12,37,38].

4.2.2 Remove/Replace Missing and Infinite Values

Due to the massive amount of data in CIC-DDoS2019, the original dataset contains either an infinite number of missing values or a large number of missing values. It is important to remove all these values from the dataset. These values of features represent attributes that have an effect on the results of the training phase.

4.2.3 Remove Duplicates

Removing duplicates enhances data quality, reduces redundancy, and helps maintain the integrity of the dataset, thus enabling more accurate and reliable analysis. The original data set contains huge duplicates, especially for new types of DDoS attacks such as LDAP, DNS, NetBIOS, UDP-lag and SSDP.

4.2.4 Random Selection Subset of Data

Due to the large volume of data in the CIC-DDoS2019 dataset, there are limitations in applying it to computers, especially in analysis data and visualization issues. Researchers often use random subsets of datasets for analysis. After cleaning the dataset, we select random subsets for each attack class with less than 150,000 instances. This subset CIC-DDoS2019 used because it offers a wide range of attack types and a large sample of data. Moreover, the dataset was created by authors from a reputable institution, and many related works performed experimentation; thus, our results are more comparable to the related work. However, the WebDDoS attack type contains only 363 samples in the range of 0.0003 overall samples in the dataset. It is fewer than other types and occurs in severely imbalanced data, so we exclude it from our experiment due to its minimal impact on the dataset's composition [39]. Also, there is an imbalance between the different classes. The TFTP represent the largest number of instances, while the LDAP has a smaller number of instances than the other types of attacks. In Figs. 4 and 5, both models illustrate data samples after the cleaning stage, obtaining a dataset including 79 attributes and yielding a total of 1,060,572 records. This dataset is categorized into 12 distinct classes. In the context of binary classification, the dataset is divided into Malicious and Benign.

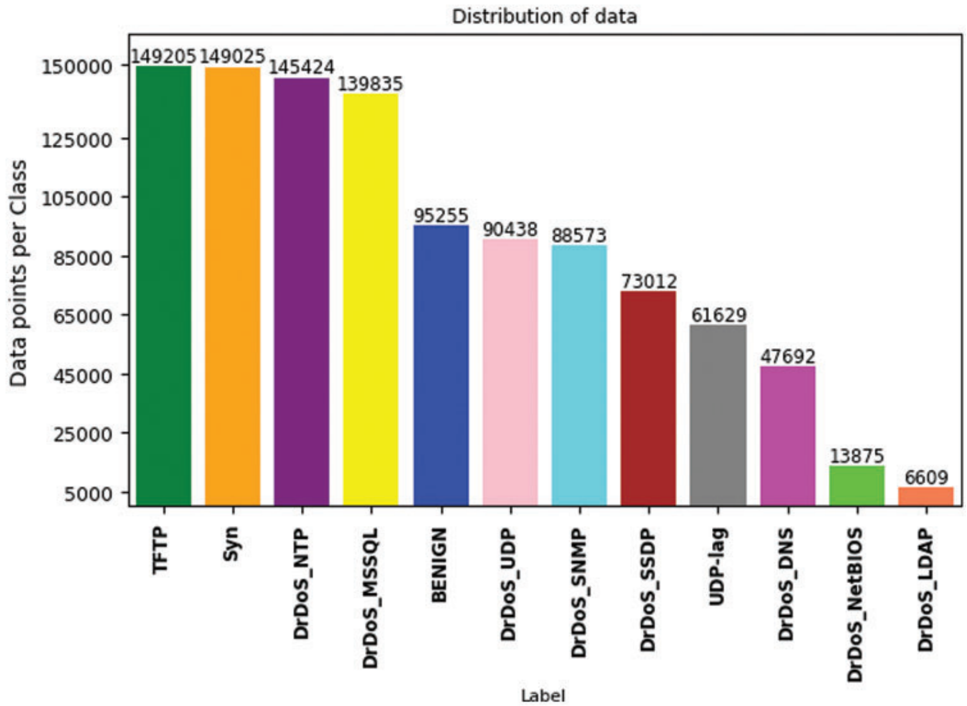

Figure 4: Distribution of data 12 classes

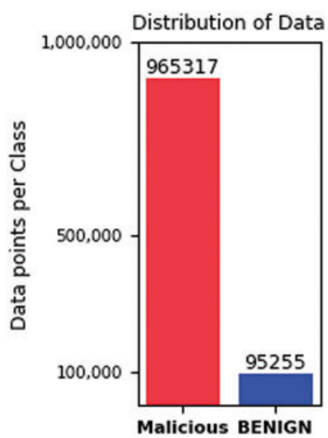

Figure 5: Distribution of data 2 classes

4.2.5 Data Split

The dataset, which represents the captured network traffic, is further split into three sections: the training set at 80% and the testing set at 20% for evaluating the performance of the model. In the training phase, validation sets take 10%. After trying different ways of splitting the data, the best result was achieved at 80:20. The split data for both multiclass and binary classification models is shown in Tables 1 and 2. In data science, the optimal data-splitting ratio is dependent on the size of the data, and there is no agreement based on theoretical or numerical research. In this study, we used a massive amount of data, so it is sufficient to use 20% of the dataset for the testing phase to prove a true prediction and the rest for the training and validation stages.

Table 1: Data split on a binary classification model

Class	Training	Validation	Testing
Benign	68584	7620	19051
Malicious	695027	77226	193064

Table 2: Data split on a multiclass classification model

Class	Training	Validation	Testing
Multiclass12	763611	84846	212115

4.2.6 Feature Scaling

A diverse range of imbalanced attribute scales may result in reduced performance for classification models. The feature scaling process is applied to the input training data phase. It is a process that handles overfitting and reduces training data time. After splitting the dataset into training and testing sets, apply the data normalization technique. The dataset contains feature values with a huge difference between their maximum and minimum values. For stable convergence of weights and biases and to avoid gradient vanishing and exploding, the dataset features have been normalized using the MinMaxScaler transform features by scaling each feature within the range 0, 1. After generalizing features, the model enhances precision and accuracy. The formula for MinMax scalar is according to Eq. (9).

$$X' = X - X_{\min}/X_{\max} - X_{\min} \tag{9}$$

where X_{max} and X_{min} are the maximum and minimum feature values (X), the output within the 0–1 range.

4.3 Cybernet Model Building

In this section, we provide detailed explanations of our methodology regarding the proposed models. The main objective is to develop a new DL architecture named Cybernet model for evaluating cybersecurity's reliability and effectiveness in the context of cyber DDoS attacks. In this study, we have chosen to incorporate LSTM, 1D CNN, and the fully connected layer (commonly referred to as the dense layer) within our network models. The methodology for designing these network models begins with the initial layer, known as the input layer, which consists of 79 neurons corresponding to the attributes present in the dataset. A general structure of the proposed model is presented in Fig. 6. The model consists of an input layer, two blocks of LSTM and 1D CNN with add output, a flatten layer, two blocks of dense layer with added output, and an output dense layer. In this model, HeNormal initialization is applied to the kernel weights. This method randomly initializes the weights for deep neural networks.

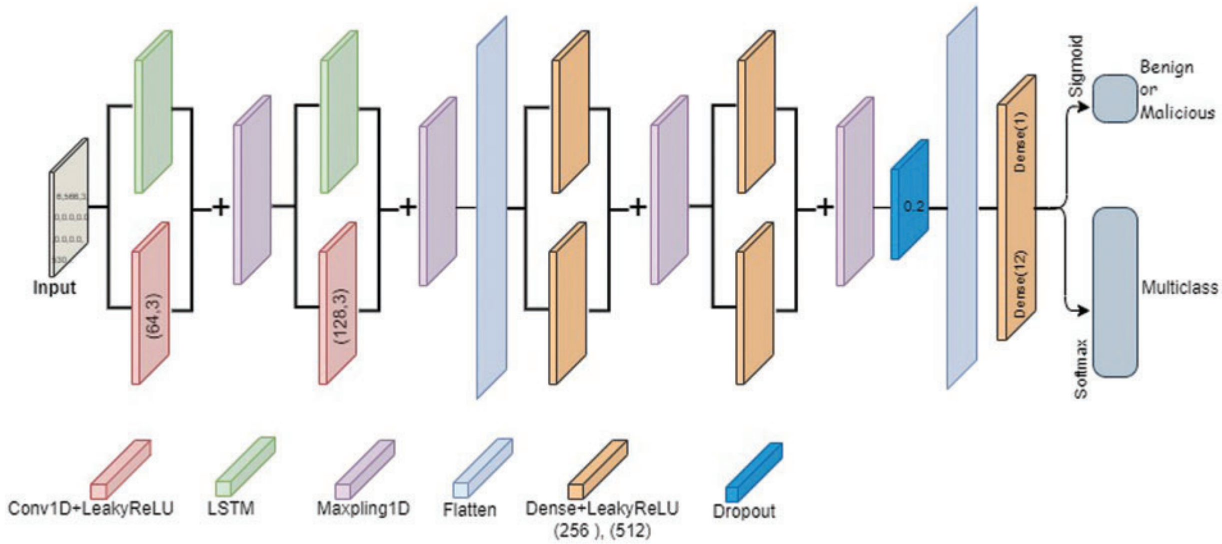

Figure 6: Structure of the proposed Cybernet models

The structure of the newly developed models is as follows: First, in parallel, a 1D-CNN layer with 1 filter and a kernel size of 3 was applied to the input layer. Second, an LSTM layer with units equal to the number of features is also applied to the input layer. Each of the aforementioned layers were then subjected to the LeakyReLU activation function. The output of the LSTM and convolutional layers was then combined using element-wise addition and passed through a max-pooling layer. The processes were repeated again, except instead of using the input layer as input, we used the output of the max-pooling layer as input to the 1D-CNN and LSTM layers. Furthermore, used 2 filters instead of 1 for 1D-CNN and the down-sampled feature number as the unit's number for the LSTM layer. The output of the max-pooling layers was then flattened. The flattened layers were then used as an input to two dense layers separately, each with 256 units, followed by element-wise addition and max-pooling layers. This step is repeated with another two dense layers of 512 units. Further, to prevent overfitting, a dropout rate of 0.2 is applied to the max-pooling layer. Next, the max-pooling layer was flattened and passed to a dense layer with a softmax activation function representing the output layer, which consists of 12 units in the multiclass model. In binary model, the output layer consists of 1 unit with a sigmoid activation function, which indicates 2 types of attacks. Table 3 provides a detailed description of the cybernet model layers.

Table 3: The architecture of proposed Cybernet model layers and parameters settings

Layer type	Filter	Activation function	Output shape	Parameters
Input			(None, 79)	0
LSTM		Leaky ReLU	(None, 1, 79)	50244
Conv1D	64 (size = 3)	Leaky ReLU	(None, 79, 1)	4
Add			(None, 79, 1)	
MaxPooling1D			(None, 39, 1)	0
LSTM		Leaky ReLU	(None, 1, 39)	12324

(Continued)

Table 3 (continued)

Layer type	Filter	Activation function	Output shape	Parameters
Conv1D	128 (size = 3)	Leaky ReLU	(None, 39, 2)	8
Add			(None, 39, 2)	
MaxPooling1D			(None, 19, 2)	
Flatten			(None, 38)	
Dense		Leaky ReLU	(None, 256)	9984
Dense		Leaky ReLU	(None, 256)	9984
Add			(None, 256, 1)	
MaxPooling1D			(None, 128, 1)	
Flatten			(None, 128)	
Dense		Leaky ReLU	(None, 512)	66048
Dense		Leaky ReLU	(None, 512)	66048
Add			(None, 512, 1)	
MaxPooling1D			(None, 256, 1)	
Dropout (0.2)			(None, 256, 1)	
Flatten			(None, 256)	
Dense		Binary class Sigmoid	(None, 1)	257 Total params: 214,901
Dense		Multiclass Softmax	(None, 12)	3084 Total params: 217,728

Table 4 describes the hyperpameters used in the Cybernet model. The model's performance during training is checked using the validation set for hyperparameter selection. The performance of the model depends on various hyperparameters used to tune the multiclass and binary class models.

Table 4: The hyperparameters employed in Cybernet model during training

Hyperparameter	Value
Activation function	LeakyReLU
Classification function	SoftMax, Sigmoid
Batch size	32
Epoch	30
Learning rate	0.001
Optimizer	Adam

4.4 Model Training and Testing

First, data sets were divided into three sets: training, validation, and testing. After preprocessing the dataset, the data was split into training and testing sets, with 80% allocated to training and 20% to testing. Furthermore, 10% was extracted from the initial 80% of training data to form the validation set. The model's training set and determined the model's best hyperparameters using the validation set. Finally, we evaluated both models on the testing (unseen) set using various performance metrics. The models are trained using the training set, with a batch size of 32 and 30 epochs. Model training sets using the Adam optimizer and the categorical cross-entropy loss function. After trying different hyperparameters on the model, according to Table 4, the network is stable, and the best values were selected for the Cybernet models. Consider using techniques such as early stopping and dropout to prevent overfitting and improve the generalization performance of the model. Experiment with different hyperparameters of the model, such as the learning rate, filter number and size, and LSTM units, to optimize the model's performance.

4.5 Evaluation Metrics

The evaluation of the model's performance relies on metrics and optimizing the model's fine-tuning hyperparameter. The accuracy measure how well the model correctly predicts the target class. Recall measures how many of the actual positive cases the model correctly predicted, while precision measures how many of the predicted positive cases are actually positive. The F1 score is the harmonic mean of precision, recall and provides a balanced measure of the model's performance. The Eqs. (10)–(13) of metrics are shown below [40]:

$$\mathrm{Accuracy} = \mathrm{TP} + \mathrm{TN}/\mathrm{TP} + \mathrm{FP} + \mathrm{TN} + \mathrm{FN} \tag{10}$$

$$\mathrm{Precision} = \mathrm{TP}/\mathrm{TP} + \mathrm{FP} \tag{11}$$

$$\mathrm{Recall} = \mathrm{TP}/\mathrm{TP} + \mathrm{FN} \tag{12}$$

$$\mathrm{F1\_score} = 2 * \frac{\mathrm{precision} \times \mathrm{recall}}{\mathrm{precision} + \mathrm{recall}} \tag{13}$$

where TP (true positive) is a correctly classified positive instance; TN (true negative) is a correctly classified negative instance; FP (false positive) is an incorrectly classified positive instance; and FN (false negative) is an incorrectly classified negative instance.

4.6 Computational Complexity of the Proposed Models

The most difficult component of designing the proposed models was combining all the aforementioned layers and activation functions to get sufficient accuracy. In particular, we concentrated on building a cost-effective architecture to reduce the computational complexity of the network while maintaining a high level of accuracy. This was achieved by using fewer layers, smaller filters, and lower-dimensional extract features in parallel, as shown in Table 3. The binary and multiclass models have 214,901 and 217,728 trainable parameters, respectively. After improving the Cybernet models, the binary model required an average 58 s for each of the 30 epochs, totaling 29 min for complete training. In contrast, the multiclass model took an average of 106 s for each of the 30 epochs, summing up to 53 min for the entire training process.

5 Experimental Results and Discussion

The proposed model for DDoS attack detection employs widely adopted performance measures. The implementation of these models using the Python programming language installed on a computer equipped with a 12th genH Intel Core i7-12650H 2.30 GHz, 16 GB of RAM, 2X512GB SSD, RTX3060 GPU and Windows 10 as an operating system Install tensorflow-gpu on the Python environment as well to use models. Subsequently, provide a report, and then discuss the results of the proposed models. After the dataset pre-processing, normal and the model applied the dataset into three sections:

training, validation, and testing subsets. We obtained results using the proposed 2-class and 12-class classification models using widely recognized performance metrics. The results are detailed in Table 5. The detection binary model exhibits an impressive overall accuracy of 99.99%. In contrast, the 12-class model achieves an overall accuracy of 99.76%. In the case of the binary classification model in terms of additional metrics such as precision, recall, and F1 score achieved 99.99%. Whereas the 12-class multiclassification model in terms of precision, recall, and F1 score achieved 99.76%. Furthermore, upon comparing both models, it becomes evident that the 2-class classification model outperforms the 12-class classification model. However, it is crucial to emphasize that the dataset is imbalanced. Therefore, relying solely on accuracy as the primary performance metric can be misleading. The class with the most samples significantly impacts the overall model performance, potentially overshadowing other classes.

Table 5: Performance of the Cybernet model under different evaluation metrics

Classification (%)	Accuracy	Precision	Recall	F1 score
Binary	99.99	99.99	99.99	99.99
Multiclass	99.76	99.76	99.76	99.76

Figs. 7 and 8 show the confusion matrix for the proposed models for prediction results, which is obtained using a test dataset. The multiclass model misclassified 502 instances, whereas the binary class model misclassified 3 instances. The performance of accuracy and loss of models as illustrated in Figs. 9 and 10.

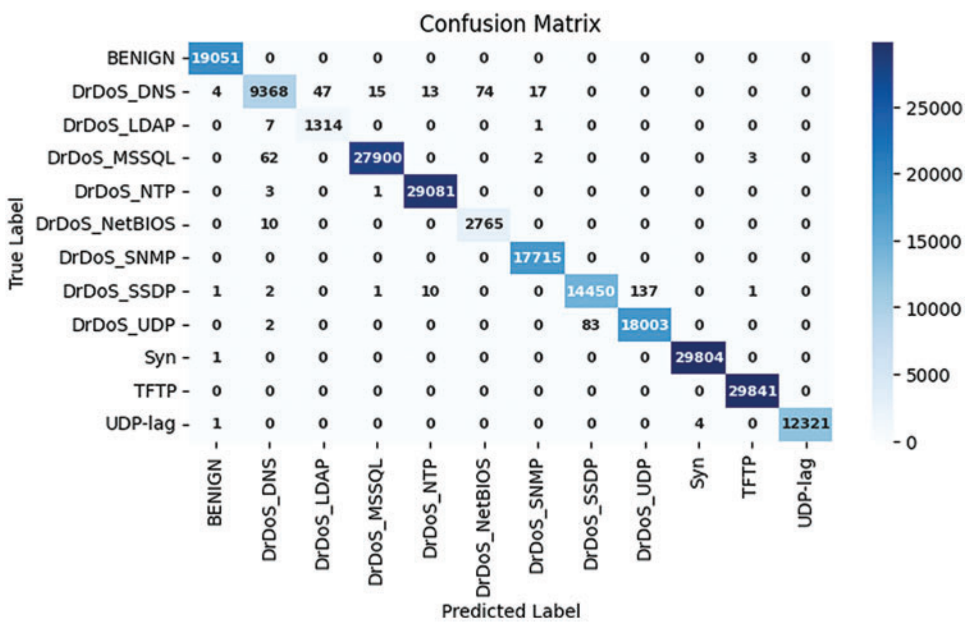

Figure 7: Confusion matrix multiclass model

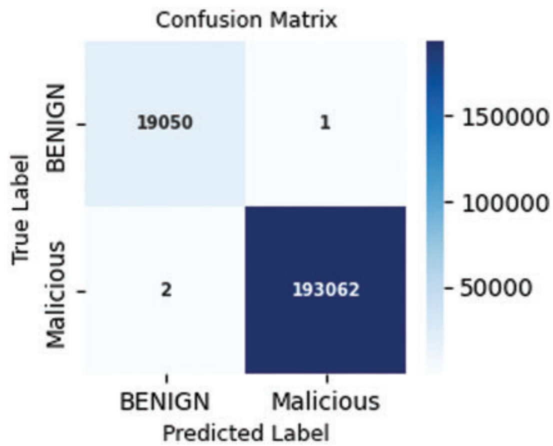

Figure 8: Confusion matrix binary class model

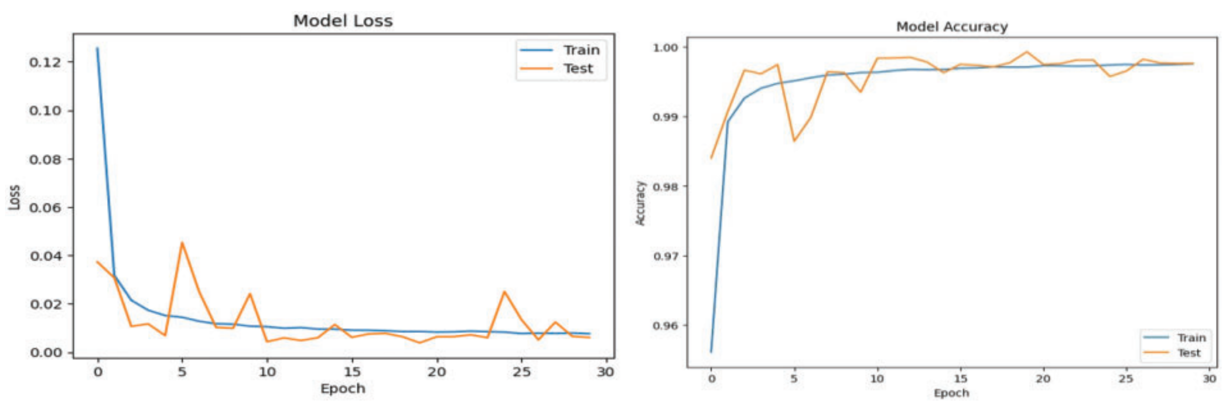

Figure 9: Performance of accuracy and loss of multiclass (recognition) model

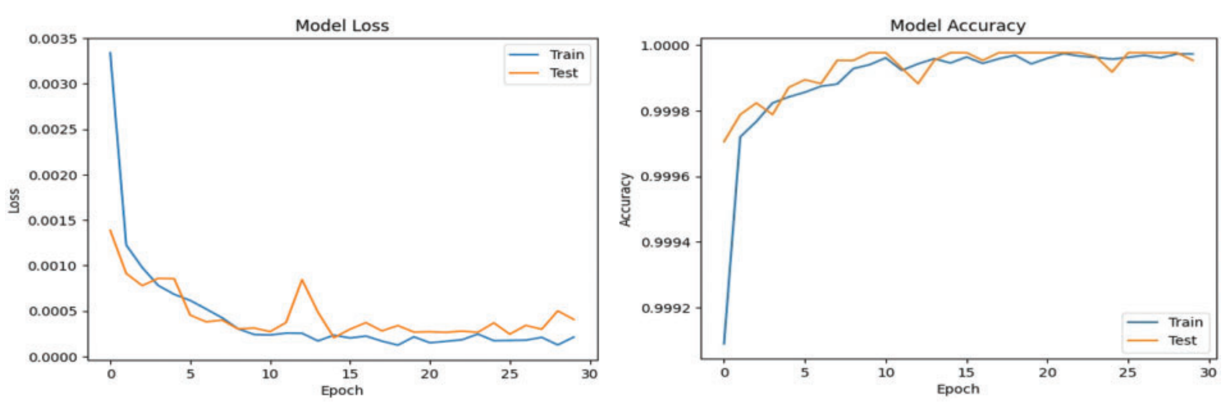

Figure 10: Performance of accuracy and loss of binary (detection) model

6 Comparative Analysis

Table 6 compares the performance of the DL network to the state-of-the-art results. To compare the results obtained from the new Cybernet model with those obtained from most existing hybrid 1D CNN with LSTM and using traditional ML algorithms. These models were applied to the same dataset. The results showed that the Cybernet model outperformed in terms of accuracy, precision, recall, and F1 score. The omission of the DNS class, given its similarities with the LDAP class, aligns with prevalent practices in previous research. Most of the previous works using the CIC-DDoS2019 dataset were based on binary classification to detect benign and malicious behavior with a smaller number of instances. Most of the work does not mention the total trainable parameter or the number of instances of the data sample used. Furthermore, in the absence of reporting the employment of imbalanced data classes, most works use balanced data techniques, making these investigations susceptible to various drawbacks, including an elevated risk of overfitting. Additional to that, using the feature selection procedure in the case of a big dataset and employing it in DL is not an effective strategy it filters the feature before applying it to the training model. The main idea of DL in the feature learning process is that a network learns which features to extract on its own by using multiple layers of neural networks. The Table 6 summarizes relevant research on Cyber DDoS using DL algorithms based on models.

Table 6: Model results comparison performance analysis of the proposed model

Ref.	Dataset	Data	Model	Features	Classification	Accuracy
[10]	CIC-DDoS2019	230,673	RNN+AE	77	Binary	99%
[11]	CIC-DDoS2019		3 Dense layers	78	10 classes	94%
[12]	CIC-DDoS2019	977,239	AE-MLP	24	5 classes	98.34%
[13]	CIC-DDoS2019	100–500	CNN+BiLSTM	10	Binary	94.52%
[14]	CIC-DDoS2019	531,819	GAN	10	13-classes	98.53%
[15]	CICDDoS2019	650,000,	DNN, CNN-AE	83	13-classes	87% 91.9%
[16]	N-BaIoT	1,486,418	DNNBoT1 and DNNBoT2	58	11 classes	90.71% 91.44%
[17]	CICDDoS2019	10%	DFNN-SAE-DCGAN	21	Binary	98.5%
[18]	CICDDoS2019	138,839	CNN+LSTM	10	Binary, 3 classes	99.5%
[19]	CICDDoS2019		LSTM	70	Binary	98.6%
Cybernet	CICDDoS2019	1060572	Cybernet model	79	Binary, 12 classes	99.99% 99.76%

7 Limitations of the Study

In this research paper, our primary focus was on the new architecture of DL for Cyber DDoS attack detection and recognition within the field of cybersecurity. The models we proposed showed satisfactory performance; however, there are several limitations that need to be addressed. A significant challenge we encountered was the absence of a dedicated lab for network security administration, which hindered our ability to collect real-time data, thus affecting our capacity to evaluate the models' performance on them. Additionally, our approach to data scaling was limited; we employed the Min-Max scaler to bring our data within the range of 0 to 1, neglecting the exploration of other existing normalization techniques. Testing various normalization methods is crucial as it could significantly influence the model's performance and generalizability. Another notable gap in our research was the lack of in-depth exploration into the interpretability of the proposed models. Understanding the underlying rationale behind the models' decisions is paramount for establishing trust and acceptance in practical cybersecurity applications. To address this, future research endeavors should emphasize enhancing the models' explainability. This could potentially be achieved by incorporating techniques such as feature importance analysis or model-agnostic interpretability methods, shedding light on the intricate workings of the models, and making their decisions more transparent and interpretable for stakeholders. By addressing these limitations and delving deeper into the interpretability aspects, future studies in this domain can significantly advance the field of cybersecurity.

8 Validity Threats

Our newly proposed DL models were trained and tested on real data obtained from Canadian Institute for Cybersecurity at https://www.unb.ca/cic/datasets/ddos-2019.html. Therefore, we believe that the results obtained from the proposed models reflect real-world scenarios, meaning they can detect the attacks they were trained on. However, when new attacks emerge in the future, our models, along with the existing ones, cannot detect them. Addressing this challenge requires ongoing vigilance and a proactive approach. Retraining the models in response to new attack strategies is essential to maintain their relevance and effectiveness. Furthermore, considering the evolving nature of cybersecurity threats, continuous monitoring and adaptation are vital to ensure the robustness of our models against novel attacks. Hence, while our models show promise in the current landscape, their long-term efficacy necessitates vigilant consideration of validity threats and a strategy for adaptability to unknown threats.

9 Conclusion

In recent years, increased attention has been given to the use of DL algorithms for cyberattack detection and recognition. In this study, we developed a new DL architecture named the Cybernet model with a specific focus on LSTM and 1D CNN architectures. The objective was to learn the necessary behaviors of cyber attacks in the cybersecurity field, recognizing and detecting DDoS attack types with high accuracy. The main architectural point is that the outputs of LSTM and 1D-CNN are added before moving to the subsequent layer. During the process of feature learning, when LSTM layers are applied to non-time series data, they enable the model to automatically learn relevant features from the data, especially when combined with other layers like convolutional layers. The evaluation was conducted on the comprehensive CIC-DDoS2019 dataset. The Cybernet models achieved remarkable results in various evaluation metrics. For instance, we achieved accuracy rates of 99.76% for multi-class classification and an outstanding 99.99% for binary-class classification.

In the future, our aim is to include the attention layer into the proposed models to determine its impact on the overall performance. Additionally, conducting further experiments with diverse datasets could offer valuable insights into the models' generalizability and robustness.

Acknowledgement: We are grateful to the anonymous reviewers who helped us enhance the quality of the paper.

Funding Statement: The authors received no specific funding for this study.

Author Contributions: Study conception and design: Azar Abid Salih; data collection: Azar Abid Salih; analysis and interpretation of results: Maiwan Bahjat Abdulrazaq; draft manuscript preparation: Azar Abid Salih. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The data used in this study is publicly available in Canadian Institute for Cybersecurity at https://www.unb.ca/cic/datasets/ddos-2019.html.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] K. Li, H. Zhou, Z. Tu, W. Wang and H. Zhang, "Distributed network intrusion detection system in satellite-terrestrial integrated networks using federated learning," *IEEE Access*, vol. 8, pp. 214852–214865, 2020.

[2] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, A. Al-Nemrat *et al.,* "Deep learning approach for intelligent intrusion detection system," *IEEE Access*, vol. 7, pp. 41525–41550, 2019.

[3] Y. Fu, F. Lou, F. Meng, Z. Tian, H. Zhang *et al.,* "An intelligent network attack detection method based on RNN," in *IEEE Third Int. Conf. on Data Science in Cyberspace*, Guangzhou, China, pp. 483–489, 2018.

[4] S. Zhang and C. Du, "Semi-supervised deep learning based network intrusion detection," in *Int. Conf. on Cyber-Enabled Distributed Computing and Knowledge Discovery*, Chongqing, China, pp. 35–40, 2020.

[5] K. P. Reddy, S. Kodati, M. Swetha, M. Parimala and S. Velliangiri, "A hybrid neural network architecture for early detection of DDOS attacks using deep learning models," in *2021 2nd Int. Conf. on Smart Electronics and Communication (ICOSEC)*, vol. 2, pp. 323–327, 2021.

[6] S. Dhir and Y. Kumar, "Study of machine and deep learning classifications in cyber physical system," in *Third Int. Conf. on Smart Systems and Inventive Technology*, Tirunelveli, India.

[7] S. Nayyar, S. Arora and M. Singh, "Recurrent neural network based intrusion detection system," in *2020 Int. Conf. on Communication and Signal Processing (ICCSP)*, Chennai, India, 2020.

[8] M. Mittal, K. Kumar and S. Behal, "Deep learning approaches for detecting DDoS attacks: A systematic review," *Soft Computing*, vol. 27, pp. 13039–13075, 2023. https://doi.org/10.1007/s00500-021-06608-1

[9] M. Xin and Y. Wang, "Research on feature selection of intrusion detection based on deep learning," in *IWCMC Conf., 2020*, Limassol, Cyprus, IEEE, pp. 1431–1434, 2020. https://doi.org/10.1109/IWCMC48107.2020.9148217

[10] M. S. Elsayed, N. A. Le-Khac, S. Dev and A. D. Jurcut, "DDoSNet: A deep-learning model for detecting network attacks," in *IEEE 21st Int. Symp. on "A World of Wireless, Mobile and Multimedia Networks"*, India, vol. 1, pp. 391–396, 2020.

[11] A. Chartuni and J. Márquez, "Multi-classifier of DDoS attacks in computer networks built on neural networks," *Applied Sciences*, vol. 1, pp. 413–426, 2021.

[12] Y. Wei, J. Jang-Jaccard, F. Sabrina, A. Singh, W. Xu *et al.,* "AE-MLP: A hybrid deep learning approach for DDoS detection and classification," *IEEE Access*, vol. 9, pp. 146810–146821, 2021.

[13] D. Alghazzawi, O. Bamasag, H. Ullah and M. Z. Asghar, "Efficient detection of DDoS attacks using a hybrid deep learning model with improved feature selection," *Applied Sciences*, vol. 11, pp. 11634, 2021.

[14] L. Nie, Y. Wu, X. Wang, L. Guo, G. Wang *et al.,* "Intrusion detection for secure social Internet of Things based on collaborative edge computing: A generative adversarial network-based approach," *IEEE Transactions on Computational Social Systems*, vol. 9, pp. 134–145, 2022.

[15] J. Boonchai, K. Kitchat and S. Nonsiri, "The classification of DDoS attacks using deep learning techniques," in *IEEE, 7th Int. Conf. on Business and Industrial Research (ICBIR)*, Bangkok, Thailand, pp. 544–550, 2022.

[16] Haq and Khan, "DNNBoT: Deep neural network-based botnet detection and classification," *Computers, Materials & Continua*, vol. 71, no. 1, pp. 1729–1750, 2022.

[17] C. Srinivas, T. Kumar, E. Yunus and Singh, "An optimized DDoS attack detection using deep convolutional generative adversarial networks," in *IEEE, 2023 5th Int. Conf. on Inventive Research in Computing Applications (ICIRCA)*, Coimbatore, India, pp. 668–673.

[18] A. Zainudin, L. A. C. Ahakonye, R. Akter, D. S. Kim and J. M. Lee, "An efficient hybrid-DNN for DDoS detection and classification in software-defined IIoT networks," *IEEE Internet of Things Journal*, vol. 10, pp. 8491–8504, 2023.

[19] D. Kumar, R. K. Pateriya, R. K. Gupta, V. Dehalwar and A. Sharma, "DDoS detection using deep learning," *Procedia Computer Science*, vol. 218, 2023.

[20] Y. Zhou, M. Han, L. Liu, J. S. He and Y. Wang, "Deep learning approach for cyberattack detection," in *IEEE INFOCOM 2018-IEEE Conf. on Computer Communications Workshops*, Honolulu, HI, USA, pp. 262–267, 2018.

[21] M. Cirillo, M. D. Mauro, V. Matta and M. Tambasco, "Application-layer DDOS attacks with multiple emulation dictionaries," in *ICASSP 2021-2021 IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, Toronto, ON, Canada, pp. 2610–2614, 2021.

[22] R. R. A. G. R., R. Sunitha and H. B. Prasad, "Mitigating DDoS flooding attacks with dynamic path identifiers in wireless network," in *2020 Second Int. Conf. on Inventive Research in Computing Applications (ICIRCA)*, Coimbatore, India, pp. 869–874, 2020.

[23] V. Borgiani, P. Moratori, J. F. Kazienko, E. R. R. Tubino and S. E. Quincozes, "Toward a distributed approach for detection and mitigation of denial-of-service attacks within Industrial Internet-of-Things," *IEEE Internet of Things Journal*, vol. 8, pp. 4569–4578, 2021.

[24] A. D. S. Ilha, A. C. Lapolli, J. A. Marques and L. P. Gaspary, "Euclid: A fully in-network, P4-based approach for real-time DDoS attack detection and mitigation," *IEEE Transactions on Network and Service Management*, vol. 18, pp. 3121–3139, 2021.

[25] Y. W. Chen, J. P. Sheu, Y. C. Kuo and N. V. Cuong, "Design and implementation of IoT DDoS attacks detection system based on machine learning," in *2020 European Conf. on Networks and Communications (EuCNC)*, pp. 122–127, 2020.

[26] S. Sumathi, R. Rajesh, S. Lim and J. Lloret, "Recurrent and deep learning neural network models for DDoS attack detection," *Journal of Sensors*, vol. 2022, pp. 1–21, 2022.

[27] G. Andresini, A. Appice, N. D. Mauro, C. Loglisci and D. Malerba, "Multi-channel deep feature learning for intrusion detection," *IEEE Access*, vol. 8, pp. 53346–53359, 2020.

[28] M. D. Hossain, H. Inoue, H. Ochiai, D. Fall and Y. Kadobayashi, "LSTM-based intrusion detection system for in-vehicle can bus communications," *IEEE Access*, vol. 8, pp. 185489–185502, 2020.

[29] D. Akgun, S. Hizal and U. Cavusoglu, "A new DDoS attacks intrusion detection model based on deep learning for cybersecurity," *Computers & Security*, vol. 118, pp. 102748, 2022.

[30] T. A. Tang, L. Mhamdi, D. McLernon, S. A. R. Zaidi and M. Ghogho, "Deep recurrent neural network for intrusion detection in SDN-based networks," in *IEEE Int. Conf. on Network Softwarization*, Montreal, QC, Canada, pp. 202–206, 2018.

[31] X. Fang, M. Xu, S. Xu and P. Zhao, "A deep learning framework for predicting cyber attacks rates," *EURASIP Journal on Information Security*, vol. 2019, pp. 1–11, 2019.

[32] Y. Li and Y. Lu, "LSTM-BA: DDoS detection approach combining LSTM and Bayes," in *2019 Seventh Int. Conf. on Advanced Cloud and Big Data (CBD)*, pp. 180–185, 2019.

[33] O. R. Sanchez, M. Repetto, A. Carrega, R. Bolla and J. F. Pajo, "Feature selection evaluation towards a lightweight deep learning ddos detector," in *ICC 2021-IEEE Int. Conf. on Communications*, Montreal, QC, Canada, pp. 1–6, 2021.

[34] U. Saxena, U. Pradesh and Y. Singh, "An analysis of DDoS attacks in a smart home networks," in *10th Int. Conf. on Cloud Computing, Data Science & Engineering (Confluence)*, Noida, India, 2020.

[35] M. Najafimehr, S. Zarifzadeh and S. Mostafavi, "A hybrid machine learning approach for detecting unprecedented DDoS attacks," *The Journal of Supercomputing*, vol. 78, pp. 8106–8136, 2022.

[36] S. U. Rehman, M. Khaliq, S. I. Imtiaz, A. Rasool, M. Shafiq *et al.,* “DIDDOS: An approach for detection and identification of distributed denial of service (DDoS) cyberattacks using gated recurrent units (GRU),” *Future Generation Computer Systems*, vol. 118, pp. 453–466, 2021.
[37] R. Chauhan and S. S. Heydari, “Polymorphic adversarial DDoS attack on IDS using GAN,” in *Int. Symp. on Networks, Computers and Communications (ISNCC) Conf.*, Montreal, QC, Canada, pp. 1–6, 2020.
[38] B. M. Kanber, N. F. Noaman, A. M. H. Saeed and M. Malas, “DDoS attacks detection in the application layer using three level machine learning classification architecture,” *International Journal of Computer Network and Information Security*, vol. 14, pp. 33–46, 2022.
[39] I. Sharafaldin, A. H. Lashkari, S. Hakak and A. A. Ghorbani, “Developing realistic distributed denial of service (DDoS) attack dataset and taxonomy,” in *Int. Carnahan Conf. on Security Technology*, Chennai, India, pp. 1–8, 2019.
[40] O. Ussatova, A. Zhumabekova, Y. Begimbayeva, E. T. Matson and N. Ussatov, “Comprehensive DDoS attack classification using machine learning algorithms,” *Computers, Materials & Continua*, vol. 73, no. 1, pp. 577–594, 2022.