future internet

MDPI

Article

Explainable Lightweight Block Attention Module Framework for Network-Based IoT Attack Detection

Furkat Safarov [1], Mainak Basak [2], Rashid Nasimov [3], Akmalbek Abdusalomov [1,3,*] and Young Im Cho [1,*]

[1] Department of Computer Engineering, Gachon University, Sujeong-gu, Seongnam-si 461-701, Gyeonggi-do, Republic of Korea
[2] Department of AI Software, Gachon University, Sujeong-gu, Seongnam-si 461-701, Gyeonggi-do, Republic of Korea
[3] Department of Artificial Intelligence, Tashkent State University of Economics, Tashkent 100066, Uzbekistan
* Correspondence: bobomirzaevich@gmail.com (A.A.); yicho@gachon.ac.kr (Y.I.C.)

Abstract: In the rapidly evolving landscape of internet usage, ensuring robust cybersecurity measures has become a paramount concern across diverse fields. Among the numerous cyber threats, denial of service (DoS) and distributed denial of service (DDoS) attacks pose significant risks, as they can render websites and servers inaccessible to their intended users. Conventional intrusion detection methods encounter substantial challenges in effectively identifying and mitigating these attacks due to their widespread nature, intricate patterns, and computational complexities. However, by harnessing the power of deep learning-based techniques, our proposed dense channel-spatial attention model exhibits exceptional accuracy in detecting and classifying DoS and DDoS attacks. The successful implementation of our proposed framework addresses the challenges posed by imbalanced data and exhibits its potential for real-world applications. By leveraging the dense channel-spatial attention mechanism, our model can precisely identify and classify DoS and DDoS attacks, bolstering the cybersecurity defenses of websites and servers. The high accuracy rates achieved across different datasets reinforce the robustness of our approach, underscoring its efficacy in enhancing intrusion detection capabilities. As a result, our framework holds promise in bolstering cybersecurity measures in real-world scenarios, contributing to the ongoing efforts to safeguard against cyber threats in an increasingly interconnected digital landscape. Comparative analysis with current intrusion detection methods reveals the superior performance of our model. We achieved accuracy rates of 99.38%, 99.26%, and 99.43% for Bot-IoT, CICIDS2017, and UNSW_NB15 datasets, respectively. These remarkable results demonstrate the capability of our approach to accurately detect and classify various types of DoS and DDoS assaults. By leveraging the inherent strengths of deep learning, such as pattern recognition and feature extraction, our model effectively overcomes the limitations of traditional methods, enhancing the accuracy and efficiency of intrusion detection systems.

Keywords: network; cybersecurity; DDoS; attention; IoT; Densenet

Citation: Safarov, F.; Basak, M.; Nasimov, R.; Abdusalomov, A.; Cho, Y.I. Explainable Lightweight Block Attention Module Framework for Network-Based IoT Attack Detection. *Future Internet* **2023**, *15*, 297. https://doi.org/10.3390/fi15090297

Academic Editors: Georgios Kavallieratos and Georgios Spathoulas

Received: 14 July 2023
Revised: 23 August 2023
Accepted: 28 August 2023
Published: 1 September 2023

1. Introduction

Computer network systems have been implemented to enable device communication and perform crucial business functions. However, this creates a higher reliance on an entity's connection systems' primary functions. Due to their extensive and critical reliance on computer networks, key sectors including banking, healthcare organizations, and service providers are subject to instability threats [1–4]. Due to this dependence, maintaining ideal networks is necessary to maintain accessibility, efficiency, and safety. A security breach can significantly impact network performance, leading to instability and eventual network incompatibility.

Moreover, cyberattacks may result in blackouts, issues in weapon systems, and confidential information releases. They might cause the loss of priceless sensitive data, such as hospital files, military records, etc. Furthermore, they can disable phone and computer

networks, making data unavailable or rendering systems unusable [5–7]. Banking and government networks are particularly vulnerable because of the tremendous value of the data they contain. The hackers steal the information (especially other people's banking details) and profit from that information.

Over the past decade, there have been instances of different kinds of hybrid network attacks, causing severe system anomalies. Such attacks have been more prevalent over the past ten years, posing a severe threat to the stability of networks due to the modification of numerous services [8–10]. Denial of service (DoS) attacks fall into mainly two categories: service outages and service flooding.

The internet-of-things network has experienced severe losses due to DDoS attacks. Therefore, IoT users consequently have paid great attention to the vulnerabilities. Numerous devices or systems work together to attack a single target, making it challenging to locate and disable the attacking devices [11–15]. Cyberattackers frequently use a botnet to interfere with internet infrastructure. DDoS attacks are difficult to identify and prevent in real time, yet this approach has enormous utility because attacks can have significant effects.

Many intrusion detection systems (IDSs) have been developed in the past to identify these assaults, utilizing a variety of techniques involving mathematical modeling, and data mining techniques such as machine learning techniques, etc. Due to their difficulties in processing high-dimensional network information, these analytical and conventional machine learning models perform poorly [16–20]. Therefore, deep learning-based techniques are essential to handle these issues.

Recently, deep learning has attracted much interest in attack detection due to its efficient feature extraction and learning abilities, specifically in settings with massive datasets. Without contextual information, deep learning techniques eventually capture significant characteristics from the input data using numerous layers [21–24]. Therefore, in this paper, Densenet-based deep learning was implemented to perform multi-class classification on DoS and DDoS attacks. To solve the imbalanced data issue, a self-organized generative adversarial network (SOMGAN) was implemented to perform data augmentation. Afterwards, the feature extraction and selection are performed using a pyramid atrous attention network and artificial bee colony optimization algorithm (ABC). Finally, the attacks were detected and classified using a convolution block attention classifier.

The primary contributions of this paper include the following:

- A self-organizing map generative adversarial network (SOMGAN)-based data augmentation technique was utilized to address the challenge of the imbalanced dataset and enhance the effectiveness of the proposed network;
- Departing from traditional feature extraction methods, the paper adopts a deep learning approach incorporating the pyramid atrous attention module to extract crucial attributes from raw network traffic data;
- The development of a feature selection and classification system based on the artificial bee colony optimization algorithm (ABC) and convolutional dense-attention module to identify various types of attacks;
- The investigation results demonstrate that the proposed approach outperforms previous techniques in terms of attack detection on ot-IoT, CIC-IDS2017, and UNSW_NB15 datasets.

The study is subdivided into the following sections. The study's concept introduction is presented in Section 1, and a literature review is briefly described in Section 2. The methodology, experimental results, and discussion follow in Sections 2–4, respectively. Finally, the conclusions are delivered.

2. Literature Review

DoS and DDoS assaults are a severe threat to many organizations because of their tremendous ability to bring down unprotected servers in a short period. Therefore, the prevention of DoS and DDoS attacks has been the subject of many research proposals.

Current studies suggest some robust defensive frameworks from network breach attacks, which are briefly explained below.

Detection and classification of DDoS attacks was studied by Wei et al. [25], who incorporated two deep learning-based techniques using an auto encoder (AE) multi-layer perceptron (MLP). To perform feature extraction without human assistance, AE was implemented by the authors. Using the extracted features, various kinds of DDoS attacks were classified by MLP network. To assess the effectiveness of the suggested approach, large DDoS attack samples from CICDDoS2017 were extracted to access the accuracy metrics.

Shroff et al. [26] studied a generative adversarial network (GAN)-based reliable detector for identifying cyberattacks. In this system, two distinct GAN-based models were implemented. The first generator produced benign instances that closely resembled benign samples from the dataset and the second generator was capable of producing DDoS cases that closely resembled those from the dataset. Moreover, the creation of a DNN classifier-based framework facilitated distinguishing between huge samples of DDoS and benign classes over structural similarity metrics. GANs are a powerful deep generative model trained with an adversarial procedure. GANs have undergone several modifications since they were first proposed to solve several different problems in different domains [27].

Azzaoui et al. [28] implemented a deep neural network (DNN)-based intrusion detection model to effectively classify dynamic network traffic outside the sandbox. The kernel consisted of a four-layer network, and each layer contained 136 neurons. To analyze the effectiveness of the suggested approach, numerous experiments were carried out with various hyperparameter combinations, and the results were compared with those of other shallow and deep ANN models. They used CICIDS2017 and NSL-KDD datasets with standard performance metrics for this assessment. They then created and tested 36 alternative DNN model combinations, each producing different outcomes.

To identify unknown DDoS attacks, Shieh et al. [29] created a method that employed reconstruction error and distributed hidden layer features. The deep hierarchical reconstruction nets (DHRNet) structure was used in this research to recompile it with a 1D interconnected neural network using a spatial location constraint prototype loss function. A random gradient descent approximation-based one-class SVM (support vector machine) was implemented to identify the unidentified patterns in the following stage. The performance of this approach was assessed using the CICIDS2017 Friday Open Dataset.

Alduailij et al. [30] developed a system for detecting DDoS attacks by employing various machine learning and feature selection algorithms. The initial step involved selecting the most relevant attributes from the network-IoT datasets using dual machine learning (ML) approaches, namely correlative and mutual information random forest. Subsequently, the attack detection was carried out using an ensemble-weighted voting method and then scored with random forest (RF) algorithms. The performance of the system was assessed using evaluation metrics and coefficient metrics confirmed the higher true positives of the target class.

Smith et al. [31] utilized the UNSW-NB15 dataset to evaluate the effectiveness of machine learning algorithms for network intrusion detection. They compared the performance of various classifiers, including random forest, support vector machines, and neural networks, using a range of features extracted from the dataset. The results demonstrated that the random forest classifier outperformed other algorithms, achieving an accuracy of 95% and a low false-positive rate. The study highlighted the significance of leveraging the UNSW-NB15 dataset as a benchmark for assessing the efficacy of intrusion detection systems and emphasized the potential of machine learning techniques in enhancing network security.

DenseNet [32] is a deep learning architecture that achieves efficient information flow across levels by directly connecting all of its layers. Each layer passes its feature maps to all succeeding layers and receives extra input from all earlier layers. Concatenation is used to merge the output feature maps from the previous layer with those from the current layer. Each layer of the network is connected to all of the successive levels, and together they

are known as DenseNets. Comparatively speaking, this model needs fewer parameters than conventional CNNs. It also reduces the overfitting problem that occurs with smaller malware training sets [33–35].

3. Proposed Methodology

This section introduces a novel intrusion detection method based on deep learning, which aims to classify various forms of DoS and DDoS attacks. The proposed method's high-level architecture is depicted in Figure 1, consisting of four key phases. These phases include preprocessing and data augmentation, feature extraction, feature selection, and classification using a Densenet convolutional block attention module (DCBAM). It mainly uses the improved convolutional block attention (CBA) DenseNet algorithm to enhance beneficial features to better integrate the attention module into DenseNet without increasing too many parameters and wasting computing resources. Initially, the raw data undergoes a series of preprocessing steps to remove unwanted information. To address the issue of imbalanced data, a data augmentation technique based on a conditional generative adversarial network (SOMGAN) is applied, resulting in improved performance of the classifier. Subsequently, the augmented data are subjected to feature extraction using the pyramid atrous attention-based deep learning technique. The artificial bee colony optimization algorithm (ABC) is then employed to identify significant features from the extracted set. Lastly, a classifier based on Densenet architecture analyzes these features and accurately classifies the detected cyberattacks. DenseNet [36] uses a dense connection layer, in which each layer can obtain the connected feature map of the previous layer. Model redundancy is reduced by feature reuse at each level of the network [37].

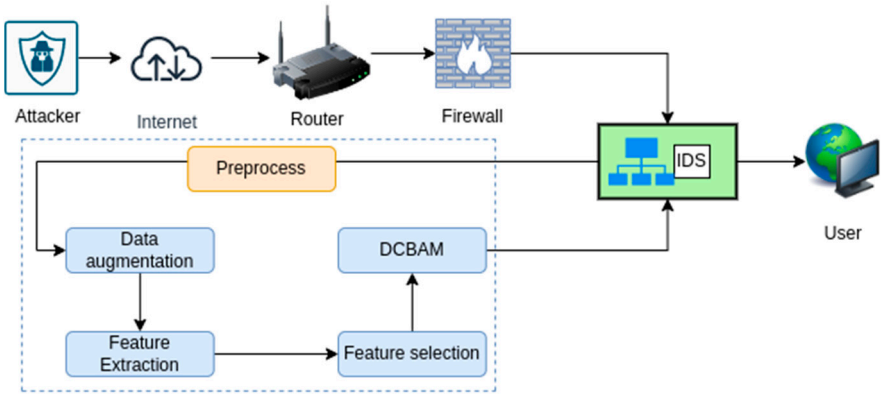

Figure 1. System pipeline.

3.1. Preprocessing

The data preprocessing stage is crucial to the overall kernel learning phase and training process, as it becomes more robust and yields a more precise model. Consequently, undesired characteristics such as "infinity" or "NaN" values in "flow packets/s" are eliminated during this phase. Additionally, redundant rows, including Fwd Avg Bytes, Bwd Avg Bulk, Fwd Avg Bulk, are removed.

The main objective of the approach is to perform multi-class categorization of DDoS attacks, which necessitates encoding. To achieve this, the study employed a one-hot encoder (OHE). This involved adding a new column for each label and assigning a value of 1 or 0 which denoted an attack or benign class.

After the labels were encoded, the next step involved data normalization using L2 normalization, and consequent columns were processed to its standard. The attributes of the label datasets are categorized by the equation below, where x represents each instance of a record.

$$\|x\|_2 = \sqrt{\sum_{i=1}^{n} |x_i|^2} \quad (1)$$

The process of normalizing the dataset records generally leads to significantly faster training. By ensuring that the dataset attributes are within a consistent range, this normalization step contributes to the generation of a more accurate model.

3.2. Data Augmentation Using a Self-Organizing Map Generative Adversarial Network

While using a SOMGAN, self-organizing maps (SOM) can be used for feature extraction. A SOM is a type of unsupervised neural network that is used for dimensionality reduction and clustering of input data. It works by mapping high-dimensional data onto a low-dimensional grid and preserving the topological properties of the input space. The resulting map can be used to identify clusters in the data and reduce the dimensionality of the input space. The generator is then used to generate new samples that are similar to the input data but also diverse. Using a self-organizing map generative adversarial network (SOMGAN) for data augmentation in the context of identifying cyberattacks can offer several benefits, primarily related to improving the robustness and generalization of the cyberattack detection model. Here is why SOMGAN techniques might be used for data augmentation:

1. Limited real data: In many cybersecurity applications, obtaining a diverse and extensive dataset of real-world cyberattacks can be challenging due to their infrequent occurrence or limited availability. Data augmentation techniques, like SOMGAN, can artificially expand the dataset, making the model more robust by exposing it to a wider range of possible attack scenarios.
2. Class imbalance: Cyberattack datasets often suffer from class imbalance, where certain attack types are rare compared to normal instances. This can lead to biased models that perform well on the majority class but poorly on the minority class (attacks). By generating synthetic attack instances, a SOMGAN can balance the class distribution and help the model better understand the characteristics of various attack types.
3. Generalization: Data augmentation helps the model generalize better. By exposing the model to a more diverse set of attack patterns, it learns to differentiate between normal and attack instances more effectively, even when faced with previously unseen or slightly different attack variations.
4. Anomaly detection: Many cyberattacks are "anomalies" compared to normal network behavior. Data augmentation techniques like SOMGAN can help the model learn to identify subtle anomalies that may not be well-represented in the original dataset.
5. Zero-day attacks: Data augmentation can aid in preparing the model for detecting zero-day attacks, which are previously unseen attack types. The model's exposure to a wider range of attack patterns through synthetic data can enhance its ability to identify novel attacks.
6. Improved feature learning: A SOMGAN can help the model learn more robust and relevant features from the data. This is particularly useful for complex and high-dimensional data like network traffic or system logs, where manual feature engineering can be challenging.
7. Reducing overfitting: By augmenting the dataset with synthetic data, the model is less likely to overfit to the limited real data. This is especially important when building deep learning models for cybersecurity, as overfitting can lead to poor generalization and a high false-positive/negative rate.

In summary, SOM can be used as an alternative to other dense generative algorithms for data augmentation in a SOMGAN [38–43]. A SOMGAN is sufficiently employed in a number of applications for data augmentation with unbalanced distributions in fault diagnosis, anomaly detection, and DoS attack detection.

3.3. Feature Extraction by Pyramid Atrous Attention Module

The attention network plays a crucial role in extracting effective features from the preprocessed data. This sort of network is built upon encoder and decoder structures, consisting of five stages. The initial three stages employ 1×1 convolution layers for the

convolution process, while the subsequent two stages utilize atrous convolution with 3 × 3 convolution layers. A ReLU layer is introduced between the two convolution layers to generate nonlinear representations, capturing low-level specific features.

To upscale the high-level feature maps within all residual blocks, the deconvolution technique is employed. Ensuring uniform feature map sizes is necessary for conducting feature fusion operations. Subsequently, the convolutional block attention module (CBAM) [5] is incorporated into the lateral connections to fine-tune the feature maps layer by layer. This integration aids in reducing false detections and enhancing feature extraction accuracy.

The CBAM module holds the capability to enhance network feature learning and can be seamlessly integrated into any network architecture. In our approach, we incorporated a CBAM module after each dense block to refine the features and bolster the network's ability to represent features effectively. Please refer to Figures 2 and 3 for a visual representation of this process.

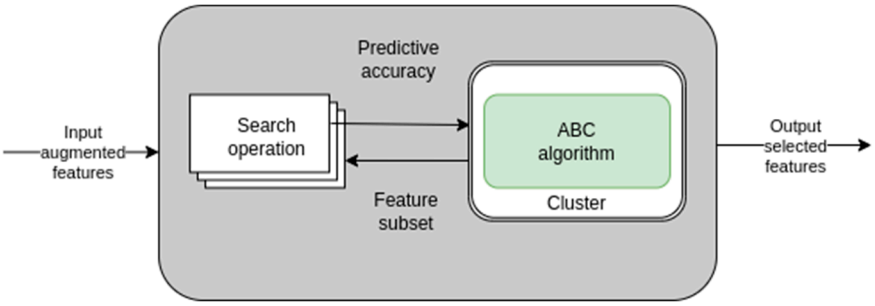

Figure 2. Demonstrate ABC algorithm mechanism.

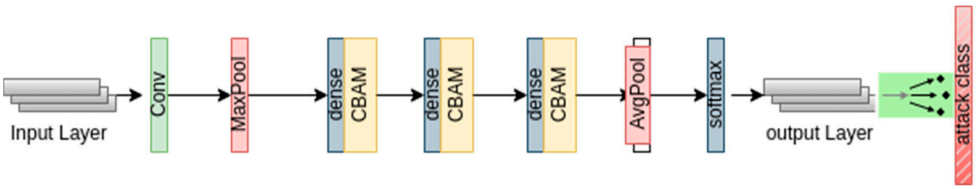

Figure 3. Proposed model architecture.

Overall, the attention network leverages encoder and decoder structures, employing convolution and atrous convolution layers to extract features. The deconvolution technique is used to upscale feature maps, and the CBAM module is added to adjust feature maps and improve feature representation accuracy. This comprehensive approach enhances the network's capability to learn and represent features accurately.

3.4. Feature Selection Using Artificial Bee Colony (ABC) Algorithm

An artificial bee colony (ABC) algorithm is a population-based stochastic optimization technique, which replicates the intelligent foraging behavior of honeybee swarms. It can be used for classification, clustering, and optimization studies. The food supply position—which represents the solution to the optimization problem—and the amount of nectar in the food source depends on the quality of the associated solution. This value is calculated in formula below:

$$\mathrm{fit_i} = 1/(1 + \mathrm{f_i}) \tag{2}$$

SN in the algorithm indicates the size of the population. Each z_i solution is a D-dimensional vector for $i = 1, 2, 3, \ldots,$ SN. Here, D is the numbers of cluster products and input size for each dataset. The probability value (p_i) is calculated in (3):

$$p_i = \mathrm{fit}_i/(\Sigma^{SN}{}_{n=1}\ \mathrm{fit_n}) \tag{3}$$

where SN is the number of food sources, which is equal to the number of employed bees and the goodness of the fit$_i$ solution given in (1). z^i_j represents comparison of two food sources to a bee.

$$z^j{}_i = z^j{}_{min} + \text{rand } (0,1) \ (z^j{}_{max} - z^j{}_{min}) \tag{4}$$

So a greedy selection mechanism is used to make selections among the old source and one of the candidates.

3.5. Intrusion Detection Using DCBAM Architecture

This section provides a comprehensive overview of the implementation process for an attack detection and classification system utilizing DenseNet201, a deep learning model with 201 layers. The framework leverages the unique characteristics of DenseNet201, which establishes direct connections between layers possessing the same feature map size. This design enables the reuse of extracted features across layers, resulting in a more precise and compact model. The DenseNet201 model is structured with four dense blocks interconnected by three transition layers responsible for downsampling. These blocks and layers are crucial for feature extraction. The final deep layers in DenseNet201 incorporate the information from all preceding layers, as denoted by Equation (4). This equation represents the composite function involving batch normalization, ReLu activation, and a 3×3 convolution layer. Furthermore, specific convolution layers with varying filter sizes and kernel shapes are appended after the fourth dense block to capture more detailed information. The features extracted from previous aggregated layers are subsequently passed to the classification head of the model. The classification layer comprises two dense layers, a batch normalization layer, and a convolution layer, with a Softmax. These components play a vital role in the final classification of attacks based on the extracted features. In summary, the implementation of the attack detection and classification system utilizes DenseNet201 as the underlying model, incorporating direct connections, dense blocks, transition layers, and specific convolution layers. The system effectively captures and reuses features, leading to a more precise and compact model for accurate classification of attacks.

$$X^l = H_l\Big(\Big[X^0, X^1, \ldots\ldots X^{l-1}\Big]\Big) \tag{5}$$

To extract important features while removing redundant information, the attention mechanism is employed. The convolutional block attention module (CBAM) proposed by Woo et al. [4] effectively extracts meaningful features in the channel and spatial dimensions, respectively, allowing for adaptive feature refinement. The module is shown in Figure 3. Using deep learning-based techniques, such as DenseNet, for cyberattack detection has become increasingly popular due to several advantages over other available techniques. Here are some reasons why DenseNet and similar deep learning approaches are favored [44–52]:

- Feature learning: Deep learning models like DenseNet automatically learn relevant features from the data, making them highly effective at capturing intricate patterns in complex data like network traffic or system logs. This adaptability is crucial in detecting new and evolving cyberattacks.
- End-to-end learning: Deep learning models are designed to learn from raw input data to make predictions directly. This end-to-end learning can help simplify the detection pipeline, reducing the need for manual feature engineering and potentially improving accuracy.
- Complex relationships: Cyberattacks can exhibit intricate relationships across multiple dimensions of data. Traditional techniques may struggle to capture these relationships effectively, whereas deep learning models can handle complex, nonlinear interactions in the data.
- Scalability: Deep learning models can handle large-scale datasets, making them suitable for real-time or near-real-time detection in high-speed network environments, which is essential for modern cybersecurity needs.

- Adaptability: Deep learning models can adapt to new attack patterns with minimal human intervention. This adaptability is crucial as cyberattacks constantly evolve, making it challenging to maintain rule-based or signature-based detection systems.
- Representation learning: Deep learning models can learn useful representations of the data, which can aid in identifying both known and novel attack types. This feature is particularly valuable in zero-day attack detection.
- Performance: In many cases, deep learning techniques like DenseNet can achieve state-of-the-art performance on benchmark datasets, demonstrating their effectiveness in cyberattack detection compared to other techniques.

4. Experimental Results

In this section, exploratory analysis is performed to expunge redundancies in the data and state the potency of the intrusion detection (ID) model. The results of the analysis are discussed in the following sections. The experiments were conducted on the system mentioned in the Table 1. These parameters were carefully selected to optimize the training process and achieve the best possible performance of the intrusion detection model. The Adam optimizer is known for its effectiveness in training deep learning models, while the ReLU activation function helps introduce nonlinearity, enhancing the model's representational power. The batch size and momentum values contribute to efficient gradient updates during training, and the dropout regularization technique aids in preventing overfitting. By conducting experiments with these specified settings, obtained evaluation metrics clearly denote the robustness of our proposed architecture over SOTA models.

Table 1. Software and hardware configurations.

System	Details	
Operating system	Linux	64-bit
Processor	I7	Intel
RAM	16	MB
Graphic memory	1080Ti	Nvidia
Backend	Pytorch	Python
Hyperparameter Optimization		
Optimizer	Adam	
Learning rate	0.001	
Activation function	ReLu	
Batch size	64	
Momentum	0.9	
Epoch	50	
Dropout rate	0.9	

4.1. Dataset Description

4.1.1. Bot-IoT Dataset

This dataset is the most recent in the industry. The dataset was released by Koroniotis et al. in 2018 [53]. It has a variety of synthetic and real-world scenarios and includes more than 72 million recordings. There are four different assault types, while DoS and DDoS-type packets make up the majority of the dataset. Similar to the UNSW-NB15 data collection, this set is imbalanced.

4.1.2. CICIDS2017 Dataset

The Canadian Institute of Cybersecurity has just produced an open-source dataset for intrusion detection called CICIDS-2017 [54]. Labeling the CICIDS-2017 dataset is based

on the date, destination, source IP addresses, attacks, protocols, destination, and source ports. It contains the characteristics of actual, realistic internet traffic. With 80 network traffic features and 2,830,743 records, this dataset was collected over five days. The dataset is a compilation of eight traffic surveillance periods and a CSV file with both regular and intruder traffic. DDoS, DoS, SSH, brute force, FTP, botnet, infiltration, heartbleed, and web attacks are the different types in this dataset.

4.1.3. UNSW-NB15 Dataset

UNSW-NB15 dataset has a hybrid of the real modern normal and the contemporary synthesized attack activities of the network traffic [55]. Smith et al. (2021) utilized the UNSW-NB15 dataset to evaluate the effectiveness of machine learning algorithms for network intrusion detection. They compared the performance of various classifiers, including random forest, support vector machines, and neural networks, using a range of features extracted from the dataset. The results demonstrated that the random forest classifier outperformed other algorithms, achieving an accuracy of 95% and a low false-positive rate. The study highlighted the significance of leveraging the UNSW-NB15 dataset as a benchmark for assessing the efficacy of intrusion detection systems and emphasized the potential of machine learning techniques in enhancing network security.

4.2. Discussion

In this section, the proposed framework is tested on the test datasets (Bot-IoT, CICIDS2017, and UNSW_NB15) over numerous evaluation metrics, and the detailed ablation is conducted to compare with other SOTA methods. To evaluate and analyze the effectiveness of attack detection cases, we compared the proposed approach with recently published attack detection methods. To perform this task, we employed widely used estimation metrics (precision, recall, and F1), as detailed in these publications [56–60]. To classify the results obtained, the following cases were distinguished:

True positives (TP): attack present and correct classification;
True negatives (TN): attack not present and correct classification;
False positives (FP): attack not present and incorrect classification;
False negatives (FN): attack present and incorrect classification.

A confusion matrix is a table used to estimate a classifier's goodness. There are the events considered in the rows, while in the columns, their classification is present. The data on the main diagonal represent correct classifications. From this table are also derived three merit factors that contribute to the analysis of a classifier's performance: the precision (P) (6) merit factor takes into account the number of correct attack identifications concerning the total number of detections. It is obtained with the following formula:

$$P = TP/TP + FP \tag{6}$$

The recall (R) (7) factor of merit takes into account the number of correct attack identifications compared to the total number of attacks made:

$$R = TP/TP + FN \tag{7}$$

Finally, the F1-score factor (F1) (8) is given by the harmonic average of precision and recall and measures the accuracy of the classification of events:

$$F1 = 2 \cdots R \cdots P/P + R \tag{8}$$

4.2.1. Performance Evaluation on Bot-IoT Dataset

Our proposed strategy's multi-class classification is shown in Table 2. The table illustrates that the performance of our proposed method across all classes was superior. The Bot-IoT dataset achieved 99.87% and 99.68% accuracy in the DDoS and DoS classifications, respectively. The performance on theft and reconnaissance classes was marginally worse

than that on other classes. Only 99% (theft) and 98.67% (reconnaissance) accuracy were achieved in those classes using our suggested approach. The behavior on reconnaissance assaults was similar to the DDoS/DoS attack behavior mirrored in the current feature set. This activity makes it more challenging using the model to identify the difference between those two attacks.

Table 2. Multi-class classification on Bot-IoT dataset.

Techniques	Precision	Recall	F1	Accuracy
Normal	99.78	99.89	99.74	99.69
DDoS	99.93	99.96	99.89	99.87
Dos	99.79	99.86	99.81	99.68
Theft	99.11	99.32	99.14	99
Reconnaissance	98.69	99.05	98.91	98.67

Figure 4 displays a visual depiction of Table 2. From the figure, it is observed that the F1 (99.89%), recall (99.96%), precision (99.93%), and accuracy (99.87%) value of the DDoS class was higher than that for all other classes. Overall, the results were positive for all types of attacks. However, the reconnaissance class received the lowest grade because it resembled regular data. Moreover, theft-exfiltration also attained the least values because a few instances in the dataset were mistakenly classified as belonging to a different class.

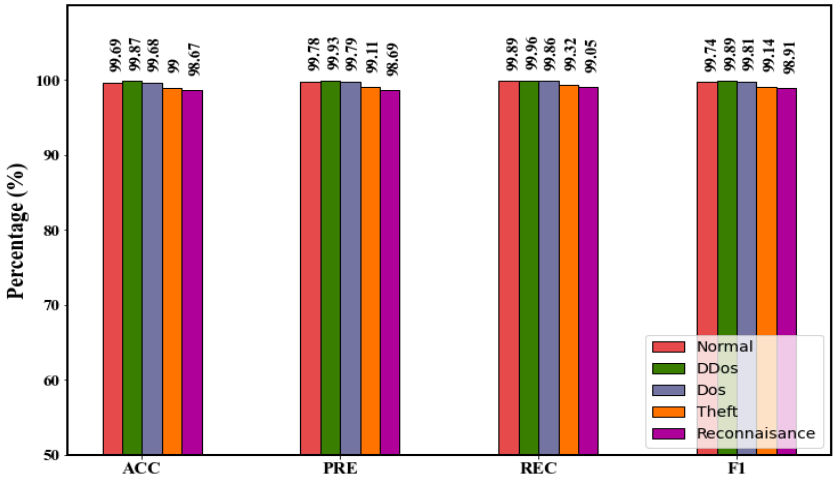

Figure 4. Multi-class classification on Bot-IoT dataset.

After the multi-class classification, the proposed approach's results were compared with the existing intrusion detection techniques evaluated on the Bot-IoT dataset, as shown in Table 3. The table shows that the proposed approach's precision, accuracy, F1, and recall were more significant than those of other existing techniques, which means that the proposed framework dramatically reduced the false positives in most classes. Compared to all other techniques, the support vector machine (SVM)'s performance could have been better and more accurate because it incorrectly divided all theft assaults into various classes. Additionally, many attacks were misidentified as regular packets, demonstrating the inability of SVM in intrusion detection.

Compared to all other approaches, the overall performance of XGBoost was superior. However, the accuracy of k-nearest neighbor (KNN) (99.03%) was higher than that of XGBoost (98.96%) because it easily handled multi-class cases and achieved better accuracy than SVM. The performance of C4.5 was also better than that of SVM. Nevertheless, merely one metric (accuracy) (Figure 5) does not adequately capture how effective the technique was at classifying intrusions. Last but not least, the results of the suggested strategy

acquired using the Bot-IoT dataset show that our technique produced more practical outcomes when compared to other techniques.

Table 3. Comparison of the proposed approach on Bot-IoT dataset.

Techniques	Precision	Recall	F1	Accuracy
SVM [31]	89.60	89.35	89.34	89.35
XGBOOST [61]	99.38	99.57	99.47	98.96
KNN [62]	99.04	99.03	99.04	99.03
C4.5 [63]	-	-	-	92
Proposed	**99.46**	**99.61**	**99.49**	**99.38**

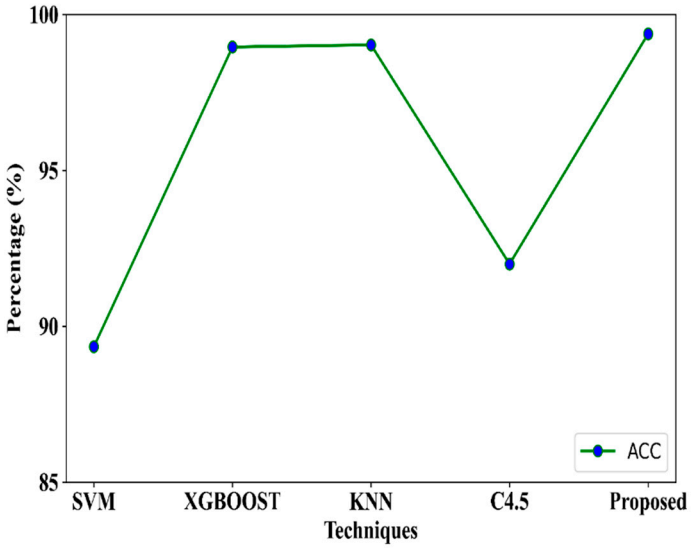

Figure 5. Accuracy comparison of the proposed approach on Bot-IoT dataset.

4.2.2. Performance Evaluation on the CIC-IDS2017 Dataset

Table 4 displays the suggested model's performance in multi-class classification on the CICIDS2017 dataset according to ACC, PRE, REC, and F1. The proposed method performed the best at detecting "benign" traffic (detection ACC of 99.97%) and the worst at detecting "SQL injection" traffic (detection ACC of 97.98%). The "SQL injection" data were scarce in the whole dataset, which caused the performance of the classifier to be poor. Moreover, the behavior pattern of a "bot" attack is similar to that of regular network traffic, making it harder for the proposed approach to recognize the attacks accurately, thus leading to average performance. Compared with heartbleed and SQL injection attacks, the brute force attack was more accurately predicted. Figure 6 displays a visual depiction of Table 4.

Regarding other performance scores, brute force and DDoS Hulk achieved similar PRE (99.67%), REC (99.65%), and F1 (99.67%) results. This demonstrated that the proposed classifier still displayed asymmetric behavior with regard to traffic classifications in this arrangement. Furthermore, ACC and REC rates are critical for assessing the classifier's performance for every attack. According to the statistics, a class with low accuracy has a lot of false positives, which implies that 'benign' classes are unnecessarily marked as assaults.

Additionally, a model with low recall may ignore actual intrusion. Therefore, to ensure that the model performs optimally, ACC and REC values must be high enough. As stated in Figure 6, the proposed model achieved superior values for all the parameters that describe the method's efficiency for multi-class categorization.

Table 4. Multi-class classification on the CIC-IDS2017 dataset.

Techniques	Precision	Recall	F1	Accuracy
Benign	99.95	99.93	99.95	99.97
DDoS	99.89	99.85	99.87	99.92
Infiltration	99.59	99.57	99.58	99.63
Portscan	99.87	99.85	99.86	99.92
Bot attack	98.85	98.81	98.83	98.89
Pataror-FTP	99.89	99.86	99.87	99.95
Parator SSH	99.23	99.21	99.22	99.27
Brute force	99.67	99.65	99.66	99.71
XSS	99.17	99.14	99.16	99.33
SQL injection	97.56	97.52	97.54	97.98
DDOs GoldenEye	98.92	98.89	98.91	98.96
DDOS Hulk	99.67	99.65	99.66	99.73
DDOS slowhttptest	98.68	98.66	98.67	98.71
DDOS-slowloris	98.83	98.8	98.82	98.86
Heartbleed	98.08	97.99	98.04	98.12

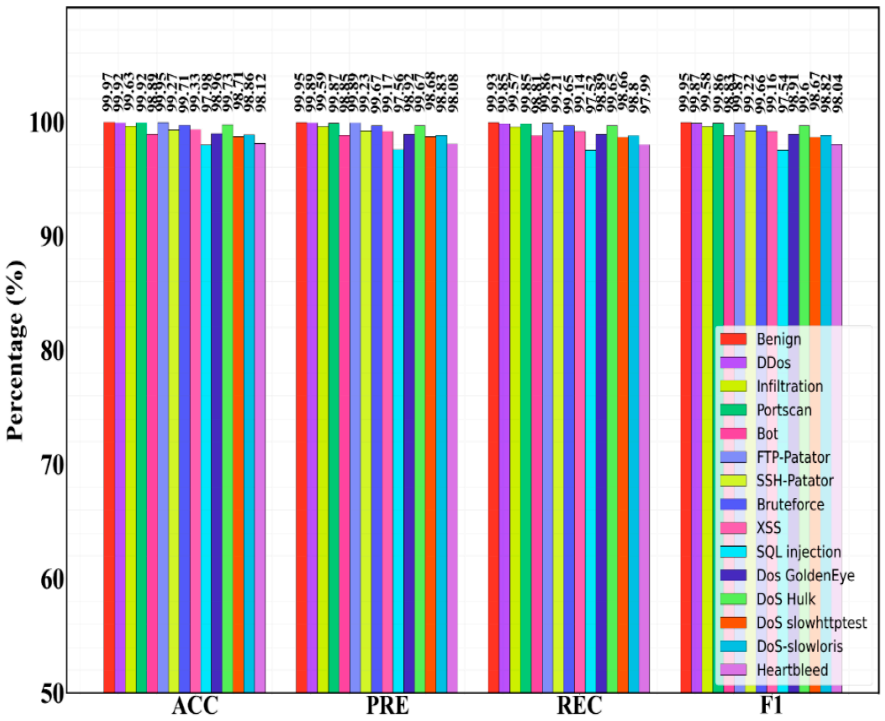

Figure 6. Multi-class classification on the CICIDS2017 dataset.

To show the effectiveness of the proposed approach, Table 5 compares the proposed approach's results with those of existing techniques on the CICIDS2017 dataset. Our suggested methodology produced better performance when compared to existing techniques. In terms of F1, the deep neural network (DNN) and recurrent neural network (RNN) achieved similar values. Compared to all other techniques, Adaboost's performance in terms of DDoS attack classification was substandard. However, the REC (100%) value of this technique was higher than that of all other approaches, which means this technique provided only a few false negatives.

Table 5. Comparison of the proposed model on CICIDS2017 dataset.

Techniques	Pre	Recall	F1	Acc
Decision tree [64]	97.5	85	90	96.67
DNN [65]	-	-	96	-
1D-CNN [66]	-	-	-	98.96
Adaboost [67]	81.83	100	90.01	81.83
RNN [68]	96	97	96	98
Proposed	**99.19**	**99.15**	**99.17**	**99.26**

On the other hand, the recurrent neural network (RNN) (98% ACC) and 1D-CNN (98.96% ACC) approaches outperformed comparable techniques and produced minimal misclassification errors. However, they did not match the performance of the proposed approach (99.26% ACC). This indicates that the proposed approach is more appropriate for DDoS attack categorization and detection.

4.2.3. Performance Evaluation on the UNSW-NB15 Dataset

Table 6 displays the outcomes of the proposed method's multi-class categorization using the UNSW_NB15 dataset. The multi-class categorization performance of the suggested method was outstanding and produced the best results for each attack class, as seen from the table. For every class, ACC rates were greater than 99%. The benign, DDoS, and DSproto classes achieved exceptional performance, with ACC rates of 99.94%, 99.83%, and 99.81%, respectively. The categorization performance on other assault categories also delivered the most significant result. Figure 7 shows a graphic depiction of Table 6.

Table 6. Evaluation of proposed model on UNSW-NB15 dataset.

Techniques	Precision	Recall	F1	Accuracy
Benign	99.94	99.86	99.85	99.89
DSproto	99.81	99.64	99.63	99.62
sbytes	99.47	99.5	99.48	99.55
sLoss	99.00	99.04	99.02	99.02
Service	99.53	99.57	99.55	99.62
Sload	99.59	99.56	99.54	99.45
sWin	99.28	99.31	99.28	99.32
stcpb	99.31	99.34	99.17	99.38
Fuzzers	99.38	99.42	99.34	99.42
Backdoor	99.35	99.38	99.36	99.45
DoS	99.58	99.21	99.2	99.25
DDoS	99.83	99.15	99.11	99.00

According to the figure, the proposed classification model performed less optimally on the "sLoss" and "sWin" classes compared to other label types. Through testing, we discovered that the properties of "sLoss" and "sWin" shared many of the same features. Therefore, to correctly classify the traffic data of these classes, the classifiers need more significant features, because our model combines the efficient ABC algorithm to select significant features. However, the overall model accuracy in distinguishing target classes was quite high for "sLoss" and "sWin" target labels, at 99.00% and 99.28%, respectively. This will lead to improving the overall accuracy of the classifier.

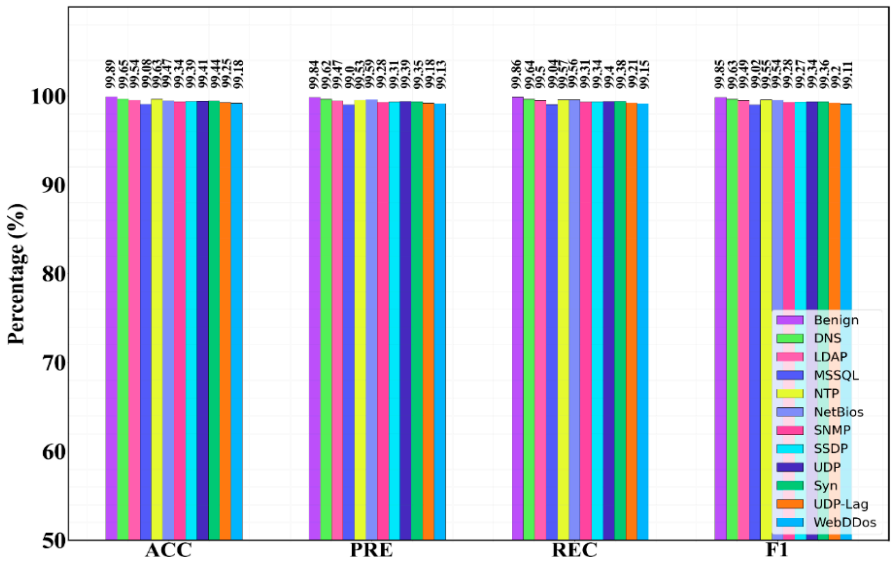

Figure 7. Multi-class classification on the UNSW_NB15 dataset.

The suggested technique's comparison to various existing intrusion detection algorithms is shown in Table 7 and Figure 8. The existing techniques, such as radial basis function neural network (RBFNN), Bayes point machine (BPM), explainable neural network (XNN), and convolutional neural network (CNN) were used. From the table, it is observed that the proposed strategy attained 99.58% accuracy compared to all the other state-of-the-art methods on the UNSW_NB15 dataset; also, it is observed that the overall accuracy was improved with CNN more than other methods. This might be due to the fact that the correlation between target features is better learned by a convolution kernel than other shallow networks. However, in terms of precision, all the techniques achieved better results.

Table 7. Ablation of proposed architecture on UNSW_NB15 dataset.

Techniques	Precision	Recall	F1	Accuracy
BPM [69]	98.00	97.00	97.00	97.00
RBFNN [70]	98.10	97.40	97.10	97.10
XNN [71]	99.00	99.20	99.00	98.10
CNN [72]	99.33	99.19	99.13	99.13
Proposed	**99.69**	**99.45**	**99.42**	**99.58**

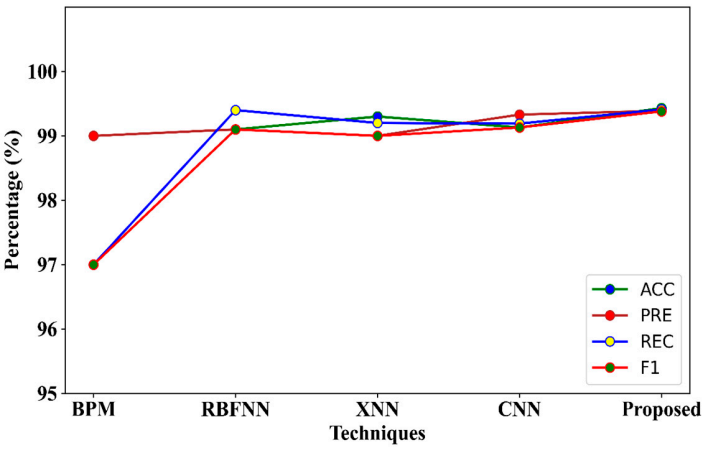

Figure 8. Ablation of proposed model with existing techniques on the UNSW_NB15 dataset.

4.2.4. Impact of Feature Selection Approach

By selecting the most pertinent aspects from the collected features, a binary POA method is applied to improve the suggested intrusion detection methodology. On the Bot-IoT, CICIDS2017, and UNSW_NB15 datasets, the ABC-based technique minimized the number of features and presented more information. Table 8 provides the proposed model performance feature selection criterion.

Table 8. Comparison with and without feature selection.

With Feature Selection				
Dataset	Precision	Recall	F1	Accuracy
Bot-IoT	99.46	99.61	99.49	99.38
CICIDS2017	99.19	99.15	99.17	99.26
UNSW-NB15	99.39	99.41	99.38	99.43
Without Feature Selection				
Bot-IoT	99.37	99.53	99.31	99.29
CICIDS2017	99.06	99.03	99.04	99.17
UNSW-NB15	99.28	99.30	99.26	99.32

Table 8 shows that using an efficient ABC-based feature selection approach enhanced the performance of the suggested strategy. Without the feature selection process, it achieved only 99.29%, 99.17%, and 99.32% accuracy for Bot-IoT, CICIDS2017, and UNSW_NB15 datasets, respectively. After the feature selection process, the classifier's performance was increased with the optimal set of features and achieved the best results without the feature selection technique. Finally, we hope that the proposed system will be effective in various real-world applications, particularly in bolstering cybersecurity measures against DoS and DDoS attacks. The integration of advanced deep learning techniques, such as the pyramid atrous attention module and the convolutional block attention module, enhances the accuracy and efficiency of intrusion detection systems. We plan to develop a small model with reliable cyberattack detection performance using YOLOv, dilated CNNs and weighted non-negative matrix factorization (WNMF) in IoT environments [73–77].

5. Conclusions

The demand for using more precise and effective IDS has grown more critical due to the quick increase in network traffic and the development of intrusions. Therefore, a deep learning-based network intrusion detection was implemented in this research. The outcomes demonstrated the performance of the proposed approach in terms of recognizing and categorizing cyber-security threats. Different performance metrics, including accuracy, F-score, recall (sensitivity), and precision (detection rate) were used in the evaluation process to analyze the usefulness of the suggested models on the three benchmark datasets. In contrast to previous attack detection techniques, the proposed framework achieved superior results with 99.38%, 99.26%, and 99.43% accuracy for Bot-IoT, CICIDS2017, and UNSW_NB15 datasets, respectively. This outcome was attained by the ABC-based feature selection method, which improved the data quality. Based on the findings of this study, it is determined that the recommended model will help create a successful intrusion detection system with a high detection rate. In the forthcoming endeavors, there will be a focus on refining the suggested IDS to detect different categories of attacks. Furthermore, the suggested approach has the potential to be adapted and utilized in a robust security application.

Author Contributions: Conceptualization, M.B. and R.N.; formal analysis, A.A., M.B. and F.S.; funding acquisition, Y.I.C.; investigation, A.A. and R.N.; methodology, M.B.; project administration, Y.I.C.; software, M.B.; supervision, A.A.; validation, R.N. and F.S.; writing—original draft, M.B., A.A. and F.S.; writing—review and editing, A.A., Y.I.C. and M.B. All authors have read and agreed to the published version of the manuscript.

Funding: This study was funded by Korea Agency for Technology and Standards in 2022, project numbers are K_G012002236201, K_G012002234001 and by the Gachon University research fund of 2022 (GCU- 202300770001).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to thank the editor and anonymous referees for their constructive comments toward improving the contents and presentation of this paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Aydın, H.; Orman, Z.; Aydın, M.A. A long short-term memory (LSTM)-based distributed denial of service (DDoS) detection and defense system design in public cloud network environment. *Comput. Secur.* **2022**, *118*, 102725. [CrossRef]
2. Ayvaz, U.; Gürüler, H.; Khan, F.; Ahmed, N.; Whangbo, T.; Bobomirzaevich, A.A. Automatic speaker recognition using mel-frequency cepstral coefficients through machine learning. *Comput. Mater. Contin.* **2022**, *71*, 5511–5521. [CrossRef]
3. Wu, Z.; Zhang, H.; Wang, P.; Sun, Z. RTIDS: A robust transformer-based approach for intrusion detection system. *IEEE Access* **2022**, *10*, 64375–64387. [CrossRef]
4. Okey, O.D.; Maidin, S.S.; Adasme, P.; Rosa, R.L.; Saadi, M.; Melgarejo, D.C.; Zegarra Rodriguez, D. BoostedEnML: Efficient technique for detecting cyberattacks in IoT systems using boosted ensemble machine learning. *Sensors* **2022**, *22*, 7409. [CrossRef] [PubMed]
5. Batchu, R.K.; Seetha, H. A hybrid detection system for DDoS attacks based on deep sparse autoencoder and light gradient boost machine. *J. Inf. Knowl. Manag.* **2022**, *22*, 2250071. [CrossRef]
6. Henry, A.; Gautam, S.; Khanna, S.; Rabie, K.; Shongwe, T.; Bhattacharya, P.; Sharma, B.; Chowdhury, S. Composition of Hybrid Deep Learning Model and Feature Optimization for Intrusion Detection System. *Sensors* **2023**, *23*, 890. [CrossRef]
7. Li, J.; Zhang, H.; Liu, Z.; Liu, Y. Network intrusion detection via tri-broad learning system based on spatial-temporal granularity. *J. Supercomput.* **2023**, *79*, 9180–9205. [CrossRef]
8. Teixeira, D.; Malta, S.; Pinto, P. A Vote-Based Architecture to Generate Classified Datasets and Improve Performance of Intrusion Detection Systems Based on Supervised Learning. *Future Internet* **2022**, *14*, 72. [CrossRef]
9. Mendonca, R.V.; Silva, J.C.; Rosa, R.L.; Saadi, M.; Rodriguez, D.Z.; Farouk, A. A lightweight intelligent intrusion detection system for industrial internet of things using deep learning algorithms. *Expert Syst.* **2022**, *39*, 12917. [CrossRef]
10. Abdusalomov, A.; Whangbo, T.K.; Djuraev, O. A Review on various widely used shadow detection methods to identify a shadow from images. *Int. J. Sci. Res. Publ.* **2016**, *6*, 2250–3153.
11. Kareem, M.I.; Jasim, M.N. Fast and accurate classifying model for denial-of-service attacks by using machine learning. *Bull. Electr. Eng. Inform.* **2022**, *11*, 1742–1751. [CrossRef]
12. Alqarni, A.A. Majority Vote-Based Ensemble Approach for Distributed Denial of Service Attack Detection in Cloud Computing. *J. Cyber Secur. Mobil.* **2022**, *10*, 265–278. [CrossRef]
13. Kuldoshbay, A.; Abdusalomov, A.; Mukhiddinov, M.; Baratov, N.; Makhmudov, F.; Cho, Y.I. An improvement for the automatic classification method for ultrasound images used on CNN. *Int. J. Wavelets Multiresolut. Inf. Process.* **2022**, *20*, 2150054.
14. Fatani, A.; Dahou, A.; Al-Qaness, M.A.; Lu, S.; Elaziz, M.A. Advanced feature extraction and selection approach using deep learning and Aquila optimizer for IoT intrusion detection system. *Sensors* **2022**, *22*, 140. [CrossRef] [PubMed]
15. Avazov, K.; Hyun, A.E.; Sami, S.; Khaitov, A.A.; Abdusalomov, A.B.; Cho, Y.I. Forest Fire Detection and Notification Method Based on AI and IoT Approaches. *Future Internet* **2023**, *15*, 61. [CrossRef]
16. Alzubi, Q.M.; Anbar, M.; Sanjalawe, Y.; Al-Betar, M.A.; Abdullah, R. Intrusion detection system based on hybridizing a modified binary grey wolf optimization and particle swarm optimization. *Expert Syst. Appl.* **2022**, *204*, 117597. [CrossRef]
17. Barut, O.; Luo, Y.; Li, P.; Zhang, T. R1dit: Privacy-preserving malware traffic classification with attention-based neural networks. *IEEE Trans. Netw. Serv. Manag.* **2022**, *1*, 1–15. [CrossRef]
18. Mamieva, D.; Abdusalomov, A.B.; Mukhiddinov, M.; Whangbo, T.K. Improved Face Detection Method via Learning Small Faces on Hard Images Based on a Deep Learning Approach. *Sensors* **2023**, *23*, 502. [CrossRef]
19. Safarov, F.; Temurbek, K.; Jamoljon, D.; Temur, O.; Chedjou, J.C.; Abdusalomov, A.B.; Cho, Y.-I. Improved Agricultural Field Segmentation in Satellite Imagery Using TL-ResUNet Architecture. *Sensors* **2022**, *22*, 9784. [CrossRef]

20. Kanber, B.M.; Noaman, N.F.; Saeed, A.M.; Malas, M. DDoS Attacks Detection in the Application Layer Using Three Level Machine Learning Classification Architecture. *Int. J. Comput. Netw. Inf. Secur.* **2022**, *14*, 1–16. [CrossRef]
21. Gaur, M.V.; Kumar, R. M-LSTM: Multi-class Long Short-Term Memory based approach for Detection of DDoS Attacks. *Math. Stat. Eng. Appl.* **2022**, *71*, 1375–1394.
22. Halladay, J.; Cullen, D.; Briner, N.; Warren, J.; Fye, K.; Basnet, R.; Bergen, J.; Dolek, T. Detection and Characterization of DDoS Attacks Using Time-Based Features. *IEEE Access* **2022**, *10*, 49794–49807. [CrossRef]
23. Mhawi, D.N.; Aldallal, A.; Hassan, S. Advanced feature-selection-based hybrid ensemble learning algorithms for network intrusion detection systems. *Symmetry* **2022**, *14*, 1461. [CrossRef]
24. Rao, Y.N.; Babu, K.S. An Imbalanced Generative Adversarial Network-Based Approach for Network Intrusion Detection in an Imbalanced Dataset. *Sensors* **2023**, *23*, 550. [CrossRef] [PubMed]
25. Wei, Y.; Jang-Jaccard, J.; Sabrina, F.; Singh, A.; Xu, W.; Camtepe, S. Ae-mlp: A hybrid deep learning approach for ddos detection and classification. *IEEE Access* **2021**, *9*, 146810–146821. [CrossRef]
26. Shroff, J.; Walambe, R.; Singh, S.K.; Kotecha, K. Enhanced Security against Volumetric DDoS Attacks Using Adversarial Machine Learning. *Wirel. Commun. Mob. Comput.* **2022**, *2022*, 5757164. [CrossRef]
27. Abdusalomov, A.B.; Nasimov, R.; Nasimova, N.; Muminov, B.; Whangbo, T.K. Evaluating Synthetic Medical Images Using Artificial Intelligence with the GAN Algorithm. *Sensors* **2023**, *23*, 3440. [CrossRef]
28. Azzaoui, H.; Boukhamla, A.Z.E.; Arroyo, D.; Bensayah, A. Developing new deep-learning model to enhance network intrusion classification. *Evol. Syst.* **2022**, *13*, 17–25. [CrossRef]
29. Shieh, C.S.; Nguyen, T.T.; Chen, C.Y.; Horng, M.F. Detection of Unknown DDoS Attack Using Reconstruct Error and One-Class SVM Featuring Stochastic Gradient Descent. *Mathematics* **2022**, *11*, 108. [CrossRef]
30. Alduailij, M.; Khan, Q.W.; Tahir, M.; Sardaraz, M.; Alduailij, M.; Malik, F. Machine-Learning-Based DDoS Attack Detection Using Mutual Information and Random Forest Feature Importance Method. *Symmetry* **2022**, *14*, 1095. [CrossRef]
31. Almaraz-Rivera, J.G.; Perez-Diaz, J.A.; Cantoral-Ceballos, J.A. Transport and Application Layer DDoS Attacks Detection to IoT Devices by Using Machine Learning and Deep Learning Models. *Sensors* **2022**, *22*, 3367. [CrossRef]
32. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708.
33. Hemalatha, J.; Roseline, S.A.; Geetha, S.; Kadry, S.; Damaševičius, R. An Efficient DenseNet-Based Deep Learning Model for Malware Detection. *Entropy* **2021**, *23*, 344. [CrossRef]
34. Saranya, P.; Devi, S.K.; Bharanidharan, B. Detection of Diabetic Retinopathy in Retinal Fundus Images using DenseNet based Deep Learning Model. In Proceedings of the 2022 International Mobile and Embedded Technology Conference (MECON), Noida, India, 10–11 March 2022; pp. 268–272. [CrossRef]
35. Nodirov, J.; Abdusalomov, A.B.; Whangbo, T.K. Attention 3D U-Net with Multiple Skip Connections for Segmentation of Brain Tumor Images. *Sensors* **2022**, *22*, 6501. [CrossRef]
36. Wan, S.; Zhang, Y. DenseNet-201-based deep neural network with composite learning factor and precomputation for multiple sclerosis classification. *ACM Trans. Multimed. Comput. Commun. Appl.* **2020**, *16*, 1–19.
37. Sha, M.; Zhang, N.; Ren, Y. X-DenseNet: Deep Learning for Garbage Classification Based on Visual Images. *J. Phys. Conf. Ser.* **2020**, *1575*, 012139.
38. Biondi, F.; Buonocore, G.; Matthews, R. Generative Adversarial Networks from a Cyber Intelligence Perspective. 2021. Available online: https://api.semanticscholar.org/CorpusID:237501625 (accessed on 23 May 2023).
39. Shorten, C.; Khoshgoftaar, T.M. A survey on Image Data Augmentation for Deep Learning. *J. Big Data* **2019**, *6*, 60. [CrossRef]
40. Abdusalomov, A.B.; Mukhiddinov, M.; Whangbo, T.K. Brain Tumor Detection Based on Deep Learning Approaches and Magnetic Resonance Imaging. *Cancers* **2023**, *15*, 4172. [CrossRef] [PubMed]
41. Avazov, K.; Jamil, M.K.; Muminov, B.; Abdusalomov, A.B.; Cho, Y.-I. Fire Detection and Notification Method in Ship Areas Using Deep Learning and Computer Vision Approaches. *Sensors* **2023**, *23*, 7078. [CrossRef] [PubMed]
42. Khan, F.; Tarimer, I.; Alwageed, H.S.; Karadağ, B.C.; Fayaz, M.; Abdusalomov, A.B.; Cho, Y.-I. Effect of Feature Selection on the Accuracy of Music Popularity Classification Using Machine Learning Algorithms. *Electronics* **2022**, *11*, 3518. [CrossRef]
43. Jiang, C.; Zhao, J.; Ding, Y.; Li, G. Vis–NIR Spectroscopy Combined with GAN Data Augmentation for Predicting Soil Nutrients in Degraded Alpine Meadows on the Qinghai–Tibet Plateau. *Sensors* **2023**, *23*, 3686. [CrossRef] [PubMed]
44. Ayas, S.; Ayas, M.S. A modified densenet approach with nearmiss for anomaly detection in industrial control systems. *Multimedia Tools Appl.* **2022**, *81*, 22573–22586. [CrossRef]
45. Wafa, R.; Khan, M.Q.; Malik, F.; Abdusalomov, A.B.; Cho, Y.I.; Odarchenko, R. The Impact of Agile Methodology on Project Success, with a Moderating Role of Person's Job Fit in the IT Industry of Pakistan. *Appl. Sci.* **2022**, *12*, 10698. [CrossRef]
46. Abdusalomov, A.B.; Islam, B.M.S.; Nasimov, R.; Mukhiddinov, M.; Whangbo, T.K. An Improved Forest Fire Detection Method Based on the Detectron2 Model and a Deep Learning Approach. *Sensors* **2023**, *23*, 1512. [CrossRef] [PubMed]
47. Liu, R. Multivariate Network Intrusion Detection Methods Based on Machine Learning. In Proceedings of the 2023 IEEE 2nd International Conference on Electrical Engineering, Big Data and Algorithms (EEBDA), Changchun, China, 24–26 February 2023; pp. 148–155. [CrossRef]

48. Ahsan, M.; Rifat, N.; Chowdhury, M.; Gomes, R. Intrusion Detection for IoT Network Security with Deep Neural Network. In Proceedings of the 2022 IEEE International Conference on Electro Information Technology (eIT), Mankato, MN, USA, 19–21 May 2022; pp. 467–472. [CrossRef]
49. Abdusalomov, A.; Whangbo, T.K. Detection and Removal of Moving Object Shadows Using Geometry and Color Information for Indoor Video Streams. *Appl. Sci.* **2019**, *9*, 5165. [CrossRef]
50. Apruzzese, G.; Colajanni, M.; Ferretti, L.; Guido, A.; Marchetti, M. On the effectiveness of machine and deep learning for cyber security. In Proceedings of the 2018 10th International Conference on Cyber Conflict (CyCon), Tallinn, Estonia, 29 May–1 June 2018; pp. 371–390.
51. Safarov, F.; Kutlimuratov, A.; Abdusalomov, A.B.; Nasimov, R.; Cho, Y.-I. Deep Learning Recommendations of E-Education Based on Clustering and Sequence. *Electronics* **2023**, *12*, 809. [CrossRef]
52. Berman, D.; Buczak, A.; Chavis, J.; Corbett, C. A survey of deep learning methods for cyber security. *Information* **2019**, *10*, 122. [CrossRef]
53. Koroniotis, N.; Moustafa, N.; Sitnikova, E. Towards the development of realistic botnet dataset in the Internet of things for network forensic analytics: Bot-IoT dataset. *Future Gener. Comput. Syst.* **2019**, *100*, 779–796. [CrossRef]
54. Sharafaldin, I.; Lashkari, A.H.; Ghorbani, A.A. Toward generating a new intrusion detection dataset and intrusion traffic characterization. In Proceedings of the 4th International Conference on Information System and Security Privacy, Madeira, Portugal, 22–24 January 2018; pp. 108–116.
55. Moustafa, N.; Slay, J. The evaluation of Network Anomaly Detection Systems: Statistical analysis of the UNSW-NB15 dataset and the comparison with the KDD99 dataset. *Inf. Secur. J. A Glob. Perspect.* **2016**, *25*, 1–14. [CrossRef]
56. Safarov, F.; Akhmedov, F.; Abdusalomov, A.B.; Nasimov, R.; Cho, Y.I. Real-Time Deep Learning-Based Drowsiness Detection: Leveraging Computer-Vision and Eye-Blink Analyses for Enhanced Road Safety. *Sensors* **2023**, *23*, 6459. [CrossRef]
57. Pascale, F.; Adinolfi, E.A.; Coppola, S.; Santonicola, E. Cybersecurity in Automotive: An Intrusion Detection System in Connected Vehicles. *Electronics* **2021**, *10*, 1765. [CrossRef]
58. Norkobil Saydirasulovich, S.; Abdusalomov, A.; Jamil, M.K.; Nasimov, R.; Kozhamzharova, D.; Cho, Y.-I. A YOLOv6-Based Improved Fire Detection Approach for Smart City Environments. *Sensors* **2023**, *23*, 3161. [CrossRef] [PubMed]
59. Mamieva, D.; Abdusalomov, A.B.; Kutlimuratov, A.; Muminov, B.; Whangbo, T.K. Multimodal Emotion Detection via Attention-Based Fusion of Extracted Facial and Speech Features. *Sensors* **2023**, *23*, 5475. [CrossRef] [PubMed]
60. Abdusalomov, A.B.; Safarov, F.; Rakhimov, M.; Turaev, B.; Whangbo, T.K. Improved Feature Parameter Extraction from Speech Signals Using Machine Learning Algorithm. *Sensors* **2022**, *22*, 8122. [CrossRef] [PubMed]
61. Ashraf, E.; Areed, N.F.; Salem, H.; Abdelhay, E.H.; Farouk, A. Fidchain: Federated intrusion detection system for blockchain-enabled iot healthcare applications. *Healthcare* **2022**, *10*, 1110. [CrossRef] [PubMed]
62. Ahmed, I.; Dahou, A.; Chelloug, S.A.; Al-qaness, M.A.; Elaziz, M.A. Feature Selection Model Based on Gorilla Troops Optimizer for Intrusion Detection Systems. *J. Sens.* **2022**, *2022*, 6131463. [CrossRef]
63. Khraisat, A.; Gondal, I.; Vamplew, P.; Kamruzzaman, J.; Alazab, A. A novel ensemble of hybrid intrusion detection system for detecting internet of things attacks. *Electronics* **2019**, *8*, 1210. [CrossRef]
64. Fuhr, J.; Wang, F.; Tang, Y. MOCA: A Network Intrusion Monitoring and Classification System. *J. Cybersecur. Priv.* **2022**, *2*, 629–639. [CrossRef]
65. Han, H.; Kim, H.; Kim, Y. An Efficient Hyperparameter Control Method for a Network Intrusion Detection System Based on Proximal Policy Optimization. *Symmetry* **2022**, *14*, 161. [CrossRef]
66. Qazi, E.U.H.; Almorjan, A.; Zia, T. A One-Dimensional Convolutional Neural Network (1D-CNN) Based Deep Learning System for Network Intrusion Detection. *Appl. Sci.* **2022**, *12*, 7986. [CrossRef]
67. Yulianto, A.; Sukarno, P.; Suwastika, N.A. Improving adaboost-based intrusion detection system (IDS) performance on CIC IDS 2017 dataset. *J. Phys. Conf. Ser.* **2019**, *1192*, 012018. [CrossRef]
68. Ravi, V.; Chaganti, R.; Alazab, M. Recurrent deep learning-based feature fusion ensemble meta-classifier approach for intelligent network intrusion detection system. *Comput. Electr. Eng.* **2022**, *102*, 108156. [CrossRef]
69. Rajagopal, S.; Kundapur, P.P.; Hareesha, K.S. Towards effective network intrusion detection: From concept to creation on Azure cloud. *IEEE Access* **2021**, *9*, 19723–19742. [CrossRef]
70. Lopez-Martin, M.; Sanchez-Esguevillas, A.; Arribas, J.I.; Carro, B. Network Intrusion Detection Based on Extended RBF Neural Network with Offline Reinforcement Learning. *IEEE Access* **2021**, *9*, 153153–153170. [CrossRef]
71. Aziz, S.; Faiz, M.T.; Adeniyi, A.M.; Loo, K.H.; Hasan, K.N.; Xu, L.; Irshad, M. Anomaly Detection in the Internet of Vehicular Networks Using Explainable Neural Networks (xNN). *Mathematics* **2022**, *10*, 1267. [CrossRef]
72. Perez-Diaz, J.A.; Vargas-Rosales, C. SDN-based architecture for transport and application layer DDoS attack detection by using machine and deep learning. *IEEE Access* **2021**, *9*, 108495–108512.
73. Abdusalomov, A.; Baratov, N.; Kutlimuratov, A.; Whangbo, T.K. An Improvement of the Fire Detection and Classification Method Using YOLOv3 for Surveillance Systems. *Sensors* **2021**, *21*, 6519. [CrossRef]
74. Valikhujaev, Y.; Abdusalomov, A.; Cho, Y.I. Automatic Fire and Smoke Detection Method for Surveillance Systems Based on Dilated CNNs. *Atmosphere* **2020**, *11*, 1241. [CrossRef]
75. Abdusalomov, A.B.; Mukhiddinov, M.; Kutlimuratov, A.; Whangbo, T.K. Improved Real-Time Fire Warning System Based on Advanced Technologies for Visually Impaired People. *Sensors* **2022**, *22*, 7305. [CrossRef]

76. Mukhiddinov, M.; Abdusalomov, A.B.; Cho, J. Automatic Fire Detection and Notification System Based on Improved YOLOv4 for the Blind and Visually Impaired. *Sensors* **2022**, *22*, 3307. [CrossRef]
77. Kutlimuratov, A.; Abdusalomov, A.; Whangbo, T.K. Evolving Hierarchical and Tag Information via the Deeply Enhanced Weighted Non-Negative Matrix Factorization of Rating Predictions. *Symmetry* **2020**, *12*, 1930. [CrossRef]