processes

MDPI

Article

HCRNNIDS: Hybrid Convolutional Recurrent Neural Network-Based Network Intrusion Detection System

Muhammad Ashfaq Khan

IoT and Big-Data Research Center, Department of Electronics Engineering, Incheon National University, Incheon 2012, Korea; ashfaq_jiskani@dongguk.edu; Tel.: +82-32-835-8784

Abstract: Nowadays, network attacks are the most crucial problem of modern society. All networks, from small to large, are vulnerable to network threats. An intrusion detection (ID) system is critical for mitigating and identifying malicious threats in networks. Currently, deep learning (DL) and machine learning (ML) are being applied in different domains, especially information security, for developing effective ID systems. These ID systems are capable of detecting malicious threats automatically and on time. However, malicious threats are occurring and changing continuously, so the network requires a very advanced security solution. Thus, creating an effective and smart ID system is a massive research problem. Various ID datasets are publicly available for ID research. Due to the complex nature of malicious attacks with a constantly changing attack detection mechanism, publicly existing ID datasets must be modified systematically on a regular basis. So, in this paper, a convolutional recurrent neural network (CRNN) is used to create a DL-based hybrid ID framework that predicts and classifies malicious cyberattacks in the network. In the HCRNNIDS, the convolutional neural network (CNN) performs convolution to capture local features, and the recurrent neural network (RNN) captures temporal features to improve the ID system's performance and prediction. To assess the efficacy of the hybrid convolutional recurrent neural network intrusion detection system (HCRNNIDS), experiments were done on publicly available ID data, specifically the modern and realistic CSE-CIC-DS2018 data. The simulation outcomes prove that the proposed HCRNNIDS substantially outperforms current ID methodologies, attaining a high malicious attack detection rate accuracy of up to 97.75% for CSE-CIC-IDS2018 data with 10-fold cross-validation.

Keywords: intrusion detection system; machine learning; recurrent neural network; deep learning; convolutional neural network; big data

Citation: Khan, M.A. HCRNNIDS: Hybrid Convolutional Recurrent Neural Network-Based Network Intrusion Detection System. *Processes* **2021**, *9*, 834. https://doi.org/10.3390/pr9050834

Academic Editor: Chien-Chih Wang

Received: 16 April 2021
Accepted: 6 May 2021
Published: 10 May 2021

1. Introduction

Nowadays, information and communication technology (ICT) systems play a crucial role in every area of business and people's lives. At the same time, cyber-attacks on ICT systems are becoming more complex and are steadily increasing. Therefore, ICT systems need a very efficient network security solution. An intrusion detection (ID) system is one of the widely used tools for detecting various types of malicious attacks in the network. Initially, a substantial amount of work in the ID domain was done by John Anderson in 1980 [1]. In most cases, an ID framework traces all interior and exterior packets in a network to find out if any of them have signs of interruption. A good-quality ID system can identify the characteristics of various cyberattacks' actions and react automatically by sending out warnings. An ID is classified into three categories according to the network architecture: (1) network-based ID systems, which evaluate the contents of distinct packets to detect malicious network traffic behavior; (2) host-based ID systems, which evaluate the event log files of each host separately to determine malicious attacks; and (3) hybrid ID systems, which combine both host and network-based ID systems with better-quality security mechanisms [2–4]. The classification and evaluation of collected network traffic packets are generally classified into anomaly detection, signature detection, and state protocol analysis.

The signature detection technique utilizes the predefined patterns and filters that efficiently detect malicious attacks. The signature detection technique uses existing knowledge to identify malicious threats, which is why it is referred to as a knowledge-based approach. The signature recognition method attains a low false alarm rate (FAR) and high accuracy; however, it cannot detect a new attack in the network [5]. The anomaly detection technique depends on heuristic methods to detect unknown attacks. However, the performance of the anomaly detection technique is effective and has a high false-positive rate. To overcome this issue, various organizations have used state protocol analysis, which combines the benefits of both signature and anomaly-based systems [6]. Two main types of ID systems can be identified according to their deployment structure: distributed and non-distributed. The first type contains numerous ID subsystems, and these subsystems connect over a large network, known as distributed implementation, while a non-distributed structure, on the other hand, may be placed in a unique location, such as an open-source snort [7].

Nowadays, statistical tests and threshold computing techniques are used in the current approaches to network intrusions in commercial markets. This statistical-test-based ID system depends on several traffic constraints, including packet arrival time, packet length, and network traffic flow size, on the model's network traffic in a predefined time. These kinds of approaches may not be efficient due to the complex nature of malicious attacks that are occurring nowadays. So, the most efficient intelligent solution is required instead of these statistically based approaches. ML-based approaches have been extensively used in detecting several types of malicious attacks and ML techniques can assist the network administrator with taking the appropriate actions to prevent these malicious attacks in the network [8]. Ensemble learning (EL) also helps us to boost the machine learning (ML) results by combining several models. Yong et al. [9] used ensemble ML-based approaches for web shell attack detection in Internet of things (IoT) environments. To develop a secure IoT system, authors apply machine learning models to detect web shells to create safe solutions for IoT networks. Future ensemble ML algorithms, such as extremely randomized trees (ET), random forest (RF), and Voting, are used to increase the performances of these machine learning models. Folino et al. [10] developed a novel ensemble-based deep learning framework for the analysis of non-stationary data, such as those that typically occur in IDS logs. The ability to design a better detection system is desired to achieve a higher detection rate, particularly when using ensemble learners. The choice of available base classifiers and the choice of available combiners are two major challenges when designing an ensemble. Tama et al. [11] provide a comprehensive review of ensemble learning for intrusion detection systems. However, most conventional ML methods belong to the shallow learning category and are less focused on feature engineering and selection; they are unable to effectively solve the massive attack data classification problem that occurs in a real-world network application context. As dataset sizes increase continuously, multi-classification attack detection tasks will lead to reduced accuracy. So, ML is incompatible with intelligent evaluation and the forecasting prerequisites of high-dimensional learning with an enormous amount of data [12]. Deep learning (DL) algorithms have recently gained popularity as powerful algorithms due to promising results in image processing, computer vision (CV), natural language processing (NLP), and other fields [13]. DL is popular among researchers due to its two primary characteristics: hierarchical feature representations and learning long-term temporal pattern dependencies. Therefore, DL methods have recently been considered to increase the intelligence of ID techniques, although there is a shortage of research to benchmark such ML techniques with publicly existing datasets. In a nutshell, DL has a nonlinear structural design that enables high-quality learning for complex data analysis. The fast development of parallel computing technology has delivered an extensive hardware foundation for DL methods. The most popular problems with current ML-based models are: (1) these models have a high false-positive rate (FPR) with a larger range of malicious intrusions [14]; (2) these models are not generalizable, as most of the existing ID systems miss novel attacks due to outdated ID datasets; and (3) state-of-the-art solutions are needed to maintain today's quickly growing high-speed

network traffic in a heterogeneous environment. These challenges are the motivation to develop a hybrid convolutional recurrent neural network-based ID system using a real-world dataset with a focus on evaluating the efficacy of ML and DL classifiers in the ID domain. As mentioned above, ID methods have their limitations; so, in our proposed ID system, we merge the two approaches to overwhelm their disadvantages and propose a new classical method combining the advantages of two approaches that have enhanced performance over traditional methods. To improve the learning capacity and performance of the ID system, we propose an improved IDS that consists of up-to-date DL methods, such as CNN, and classical ML, such as RNN. The important contributions of our research can be summarized as follows.

- We developed the HCRNNIDS, which combines both deep and shallow models to reduce analytical overheads and maximize benefits. The proposed HCRNNIDS focuses on identifying whether network traffic behavior is normal or malicious because attacks can be classified into the corresponding intrusion class.
- We address the problem of class imbalance that is common in ID data.
- We equate the proposed method with popular ML approaches. The empirical outcomes express that the HCRNNIDS very appropriate for attack detection and can accurately identify the misuses in 97.75% of incidents with 10-fold cross-validation.
- The output of the hybrid convolutional recurrent neural network-based network intrusion detection system is higher than that of traditional classification techniques when conducting experiments on the well-known and contemporary real-life CSE-CIC-IDS2018 dataset; it improves the accuracy of ID, thus providing a novel research method for ID.

To address the abovementioned challenge, we developed a hybrid convolutional recurrent neural network-based intrusion detection system (HCRNNIDS). The remainder of the article is arranged as follows. We introduce the background of the NID in Section 2. In Section 3, we provide an overview of the proposed HCRNNIDS structure as well as a comprehensive explanation of the ID data. The HCRNNIDS simulation is described in Section 4. Lastly, the conclusion of our research is illustrated in Section 5.

2. Related Work

During the last two decades, machine learning (ML) techniques have been used extensively in the network security domain because of their capability to extract the concealed information on the distinctions among malicious and normal behaviors [15–18]. So, the earlier researchers used various approaches based on conventional ML for intrusion detection (ID). Xu et al. [19] applied K-nearest neighbors (K-NN) for anomaly ID, and evaluated the efficacy of the proposed ID system using the KDDCup ID dataset. Bhati et al. [20] applied variants of support vector machine (SVM), such as quadratic, linear, fine, and medium Gaussian, to analyze the performance of SVM techniques using the NSL-KDD dataset. An integrated ID system was developed by Sumaiya et al. [21] using correlation-based feature selection and an artificial neural network (ANN). The authors performed an experimental analysis on the UNSW-NB and NSL-KDD ID datasets. Similarly, a Random Forest (RF)-based ID system was presented by Waskle et al. [22], and an ID system based on several classical ML classification methods was presented by Alqahtanet al. [23]. However, prior techniques used in the domain of ID have poor classification efficiency, with a high FAR and a low DR in the ID system. Deep learning (DL) is a subset of ML that consists of several hidden layers used to obtain the deep network's characteristics. Due to their deep structure and ability to learn the important features from the dataset on their own and produce an output, these techniques are more effective than ML [24].

DL has grown in popularity in recent years and has been applied for intrusion detection (ID); studies have shown that DL outperforms conventional methods. The authors of [25] use a DL method for flow-based anomaly ID based on a deep neural net (DNN), and the experimental results show that DL can be used for anomaly ID in a software-defined network. Nowadays, auto-encoders (AE), convolutional neural networks (CNNs), and

deep neural networks (DNNs), as well as variants of these methods, are used for ID [26]. Long-short-term-memory (LSTM) can be effective in the field of network security. Various LSTM deep-learning-based security policies have been investigated for ID [27], classification and detection of malicious apps [28], phishing exposure [29], and time-dependent botnet ID [30]. The ability to model the sequence is the primary benefit of a recurrent neural network (RNN) over a conventional network. Oliveira et al. [31] developed an intelligent ID and classification framework using LSTM deep learning and evaluated the proposed framework by using the CIDDS-001 dataset to achieve a higher ID accuracy as compared with traditional ML approaches. The convolutional neural network (CNN) is another popular DL approach that learns directly from the dataset without requiring manual feature extraction algorithms. A typical CNN consists of convolutional, pooling fully connected, input, and output layers. Even though CNNs are widely used to analyze visual images, they can also be utilized in the field of security. For example, in IoT networks [32], CNN-based models are used for ID, such as denial-of-service (DoS) ID [33], and android malware [34]. An auto-encoder (AE) [35] is a kind of ANN that is applied to economically learn data codes in an unsupervised fashion. An AE aims to learn a representation for a dataset by training the network to disdain "noise" signals to reduce the number of dimensions. An auto-encoder has three components: encoder, message, and decoder. A deep AE can be used to construct a useful security model in cybersecurity. As a result, the AE-based feature learning (FL) model in cybersecurity outperforms other advanced algorithms. The AE-based FL model uses the fewest security features when compared with other sophisticated algorithms in cybersecurity. The model is more effective and functional, even in small spaces like the IoT, because of the rich and tiny latent representation of security features. [36]. The authors in [37] provide an AE-based FL prototype for security purposes, proving its effectiveness in malware classification and detection. The authors in [38] propose a deep AE-based anomaly detection model. To create an effective ID model, the Restricted Boltzmann Machine (RBM) can be used. Yadigar et al. [39] present a denial-of-service attack detection model and achieve a higher attack detection accuracy with a RBM. The summary of various approaches in the ID domain is given in Table 1.

Table 1. Summary of different approaches in the ID domain.

Reference	Dataset	ID Technique	Performance
Tanet et al. [40]	KDD'99	MCA + EMD	99.95%
Tanet et al. [40]	ISCX 2012	MCA + EMD	90.12%
B. Inger et al. [41]	NSL-KDD	ANN	99.67%
Casas et al. [42]	KDD'99	Clustering-based IDS	92.0%
Ludwig et al. [43]	NSL-KDD	Deep learning ensemble	92.49%
Shone et al. [44]	KDD'99	Non-symmetric deep AE	97.90%
Kakavand et al. [45]	ISCX 2012	Ada boost+ DT	97.2%
Yu et al. [46]	CTU-UNB	Stacking dilated CAE	87.14%
Kumar et al. [47]	ISCX 2012	PCA	94.05%
Akyol et al. [48]	ISCX 2012	MHCVF	68.2%
Omar et al. [49]	ISCX 2012	HADM-IDS	87.2%
Monshizadeh et al. [50]	UNSW-N15	SVM, J48	89.01%
Wang et al. [51]	ISCX 2012	HAST-IDS	96.9

Recently, applications of the hybrid model have made various researchers attempt to develop an efficient and robust ID system in the cybersecurity domain. These ID systems have been found to be competent against malicious attacks as compared with separate conventional ML and DL-based ID systems [52]. Wang et al. [51] transformed two packets into an image and used the 2D-CNN to learn the characteristics of packet bytes while using the LSTM to learn the characteristics of the packet sequence, resulting in the simultaneous learning of two spacing and timing characteristics. Chencheng et al. [53] developed a hybrid ID system for the evaluation of multiple types of flow features using a hybrid NN and tested the efficacy of the proposed hybrid ID system in real-time using the ISCX2012

ID dataset. Zeng et al. [54] used a hybrid NN with a stack autoencoder (SAE) to evaluate the traffic features and chose the best feature vectors from the network traffic as label results. Hosseini et al. [55] used a fusion-based method that combined an ANN and a SVM and tested it on the benchmark NSL-KDD dataset, where the SVM was used for feature selection and the ANN was used for attack classification.

Most of the researchers in the ID domain use simulated datasets, such as NSL-KDD or KDDcup99, but these kinds of ID databases cannot accurately represent the scenarios of realistic network traffic. Erhan et al. [56] note that DDOS attacks are one of the most annoying types of malicious attacks for online activities on the internet. DDoS attacks are generally classified into two types: bandwidth depletion and resource depletion attacks. The authors created resource-depletion-type DDoS attacks and recorded the traffic from the backbone router's mirror port of the Bogaziçi University network. This dataset contains both attack-free and attack traffic, making it appropriate for testing network-based DDoS detection methods. Attacks are directed at a single victim server that is linked to the campus's backbone router. Damasevicius et al. [57] developed an annotated real-world network flow dataset for network ID called LITNET-2020. It consists of modern network traffic and various types of malicious attacks that occurred over 10 months. This gives an advantage over synthetically created ID datasets because an artificial synthesis of traffic might lead to inaccurate network attack models and behaviors. It is essential to utilize realistic flow-based ID datasets to guarantee accurate valuation of techniques [58]. As a result, in this study, the practical CSE CIC dataset was used to demonstrate a change from static datasets to dynamically created datasets that not only represent network traffic and log files at the time of the study, but can also be updated, reproduced, and extended.

3. Proposed Approach

Figure 1 shows the ID system's architecture. It consists of two learning steps, sketched as follows. In this approach, we proposed to construct an IDS using a HCRNN-based deep learning approach. Our proposed HCRNNIDS is economical in terms of computational complexity while using datasets with full features and offers improved accuracy with a minimal chance of a FAR.

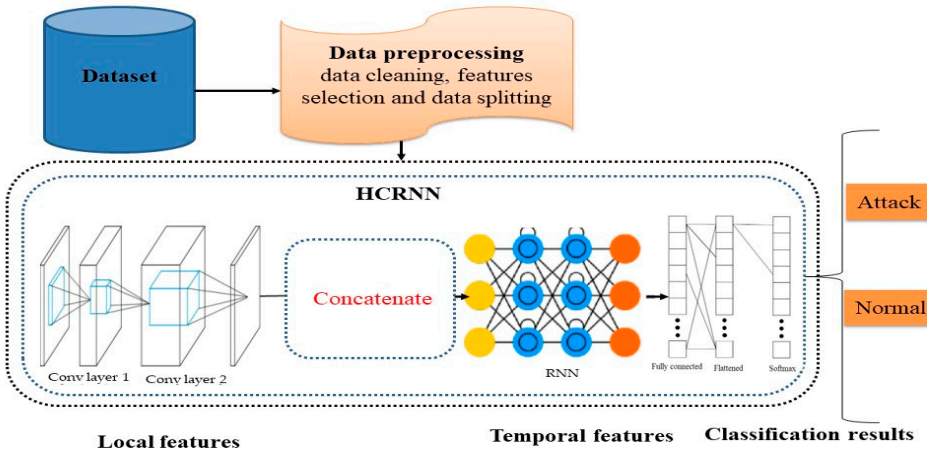

Figure 1. Overview of the HCRNNIDS.

3.1. Overview of the HCRNNIDS

HCRNN-based deep learning focuses on solving realistic ID problems with a big data processing framework. Due to a lack of time and space, resolving such a problem is not an easy job. Big data presently has huge, and increasing, volumes but needs an enormous amount of power, specialized resources, and a computational device to assist with the learning process that can handle the data competently.

HCRNN-based deep learning decreases these challenges by using a RNN with a CNN DL model. The source of the experiment is the key structure of the HCRNNIDS that exists here. The detail of the HCRNNIDS is depicted in Figure 1.

The overview of the HCRNNIDS shows that a CNN has two fundamental components: (i) a feature extractor; and (ii) a classifier. The feature extractor consists of two layers called convolution and pooling layers. The extracted output, which is known as the feature map, becomes the input to the second component for the classification. This way, the CNN learns the local features very well. However, the weakness is that it misses the temporal dependency among important features. Therefore, to capture both spatial as well as temporal features more robustly, we introduced recurrent layers after the CNN layers. This way, we addressed the vanishing and exploding gradient problems effectively, which improves the capability to capture spatial and temporal dependencies and learn efficiently from variable extent sequences. In the HCRNN network, the input is initially processed by the CNN, and then the output of the CNN is passed through the recurrent layers to generate sequences at each timestep, which helps us model both spatial and temporal features. Then, the sequence vector is passed through a fully connected layer before being fed into a Softmax layer for the probability distribution over the classes. The network traffic was first organized and preprocessed in the data pre-processing part. During pre-processing, all necessary conversions were made that used or were helpful for the HCRNNIDS and IDS-supported data formats. In the original CSE CIC IDS2018, a few features, such as IP addresses and timestamps, have slight importance on whether the network traffic is benign or malicious. The timestamp features are used to record the time when malicious traffic happened and provide only slight support when training the process, so during the pre-processing phase we removed these kinds of features. As in an anomaly intrusion detection system (AIDS), most of the traffic is categorized based on the traffic behavior, and should not be biased, conflicting with the IP address, so we also removed the IP address characteristics. Data pre-processing operations were implemented using Pandas NumPy and Scikit-learn libraries developed for the Python programming language. The research community has drawn substantial attention to the class imbalance issue. The problem of a class imbalance is created by an insufficient data distribution; one class contains most of the samples, while others contain comparatively few. The classification problem becomes more complicated as the data dimensionality increases due to unbounded data values and unbalanced classes. Bedi et al. [59] utilized several ML approaches to deal with the class imbalance issue. Thabtah et al. [60] also evaluated various approaches to the class imbalance problem. Most data samples are targeted by most of the algorithms, which miss the minority data samples. As a result, minority samples appear irregularly but constantly. The main algorithms for solving an unbalanced data problem are data pre-processing and feature selection techniques, and every approach has both benefits and shortcomings. The ID dataset has a high-dimensional imbalance problem, including missing features of interest, missing feature values, and the sole existence of cumulative data. The data appear to be noisy, containing errors and outliers, and unpredictable, comprising discrepancies in codes or names. We used over-sampling to resolve the imbalance problem; this involved enlarging the number of instances in the minority class by arbitrarily replicating them to increase the presence of the minority class in the sample. Although this procedure carries some risk of overfitting, no information was lost, and the over-sampling approach was found to outperform the under-sampling alternative. After finishing the data pre-processing, we split the dataset into testing, training, and validation sets, which were 9, 90, and 1% of the initial network traffic, respectively. The training set was utilized for training, the validation set was applied for fast evaluation of the prototype during training, and the testing set was utilized for the final evaluation of the model. Furthermore, we discovered that the dataset included far too many samples of normal network traffic, which could easily distort the model's classification preference.

3.2. Datasets

Since choosing proper ID data to assess the ID system plays a critical role, we selected the data before we performed the simulation of the proposed approach.

Explanation of the ID Data

Although numerous ID datasets are freely available, some of them contain old-fashioned, inflexible, under-verified, and irreproducible intrusions. To overcome these deficiencies and produce modern traffic patterns, the well-known CSE-CIC-DS2018 [61] dataset was produced by the Amazon Web Services (AWS) platform. It contains various types of datasets used to evaluate anomaly-based techniques. The CSE-CIC-DS2018 intrusion dataset presents real-time network behavior and comprises several intrusion states. Moreover, it is distributed as a whole network encapsulates all of the inner network traces to calculate payloads for data packets. These characteristics of the CSE-CIC-DS2018 dataset bring us to utilize it for the proposed intrusion detection system in our research. It is expected that the proposed HCRNN-based ID system will result in a more rational and valuable direction in the network security domain.

This dataset contains several intrusion profiles that can be utilized in the security domain and apply to a wide range of network protocols and topologies. This dataset was enhanced by the IDS2017 criteria. IDS2018 is a publicly available dataset that currently has two profiles and seven intrusion methods. Several data states were gathered, and the unprocessed data were edited regularly. So, IDS2018 has 80 statistical properties, including packet length, volume, and number of bytes, that were calculated in forward and reverse mode. Finally, the dataset was made available to all researchers via the internet, with approximately 5 million records. The CSE-CIC IDS2018 dataset is accessible in two formats: PCAP and CSV. The CSV format is primarily used in AI, while the PCAP format is utilized to extract new features [62,63].

This CSE-CIC IDS2018 dataset contains seven different types of attacks:

- Brute-force DOS attacks;
- DDOS attacks;
- Brute-force SSH;
- Infiltration;
- Heartbleed;
- Web attacks; and
- Botnet.

The dataset-attacking infrastructure consists of 50 computers, while the attacking companies consist of 30 servers and 420 terminals. CSE-CIC IDS2018 data signify the captured network traffic of AWS and a system log with 80 extracted attributes utilizing CICFlowMeter-V3. The size of the CSE-CIC IDS2018 dataset is around 400 GB, which is larger than that of CIC-IDS 2017. Table 2 presents a few extracted features of the CSE-CIC2018 dataset.

We compared the sample size of the CSE-CICIDS2018 dataset with that of CICIDS2017. The results are displayed in Table 3. The sample size of CSE-CICIDS2018 was significantly increased compared with the CICIDS 2017 ID dataset, particularly in the Botnet and Infiltration attacks, where it has risen by 143 and 4497, respectively. However, the number of Web Attacks available is extremely small (928) in CSE-CICIDS2018.

Table 2. Overview of the extracted features of the CSE-CIC IDS2018 ID dataset.

Features	Explanation
Fl-dur	Flow interval
Fl-iat-max	Maximum time between two flows
Tot-fw-pk	Aggregate data packets in a forward way
Tot-l-fw-pkt	Overall size of the packet in an up way
Tot-bw-pk	Overall data packets in a back way
Fw-pkt-l-min	The lowest volume of the packet in a further way
Fw-pkt-l-avg	The average amount of data in the packet in an up way
Fw-iat-min	Smallest time between two packets delivered in an onward way
Bw-iat-tot	Overall time between two packets delivered in a back way
Bw-iat-avg	Mean period between two packets delivered in a back way
Bw-iat-std	Average period between two packets forwarded in a back way
Bw-iat-max	Highest period between two packets forwarded in a back way
Bw-iat-min	Least time between two packets delivered in a forward way
Bw-iat-min	Lowest time between two packets forwarded in a reverse way

Table 3. Comparison of the CSE-CIC 2018 ID dataset with CICIDS-2017.

Dataset	Normal	DDoS	Dos	Botnet	Brute Force	Infiltration	Web Attacks	Port Scan
CICIDS-2017	1,743,179	128,027	252,661	1966	13,835	36	2180	158,930
CSE-CICIDS2018	6,112,151	687,742	654,301	286,191	380,949	161,934	928	-

3.3. Experimental Details

We implemented the HCRNN method in Java with Deeplearning4j to validate the efficacy of the proposed ID scheme. The experiment was done on a cluster computer (64-bit, 32 GB RAM, 32-core processor, desktop computer Core I7). The software stack contained Java (JDK) 12, Deeplearning4j 1.0.0. alpha, and Spark v2.3.0. The deep learning (DL) algorithm was trained on an NVIDIA GTX 1080 Ti GPU with cuDNN support to increase the pipeline speed. To evaluate the output of the HCRNNIDS, we first had to divide the data into training and testing sets. To develop an effective HCRNNIDS, we used a training set and analyzed our ID approach with the testing set. To show the dominance of the proposed solution, we used the CSE-CIC-IDS2018 dataset with all its original features. The network traffic was mixed with malicious and non-malicious data, which were classified into malicious and non-malicious groups by the HCRNNIDS. The proposed method reduces the computational complexity by using extensive features from the CSE-CIC-IDS2018 dataset to achieve high ID accuracy and a low FAR value. Though 90% of the CSE-CIC-IDS2018 data were used for training purposes with 10-fold cross-validation, the model was evaluated on a 10% held-out dataset. During the training phase, first-order gradient-based optimization techniques, such as Adam, Ada Grad, RMSprop, and Ada Max, with varying learning rates were used to optimize the binary cross-entropy loss of the predicted network packet, and the actual network packet was optimized with different combinations of hyperparameters from a grid search and 10-fold cross-validation to train each model on the batch size. We observed the performance by adding Gaussian noise layers followed by convolutional and recurrent layers to improve the model generalization and reduce overfitting.

4. Experimental Results

The HCRNNIDS's superiority was assessed by the false positive (FP), true positive (TP), false negative (FN), true negative (TN), ID accuracy, and error rate using the CSE-CIC-DS2018 ID dataset.

4.1. Evaluation Metrics

The confusion matrix (CM) supports the identification of the actual and predicted classification. The result of a categorization is based on two classes: Normal and Anomaly. In the confusion matrix, there are four critical states that we must measure.

- True Positive (TP): this indicates that the model is accurate and normal and predicts positive outcomes.
- False negative (FN) is characterized by incorrect prediction. It recognizes instances that are malicious with certainty as natural, and the model predicts negative outcomes incorrectly.
- False positive (FP): the model predicts a positive outcome when, in fact, the number of observed attacks is normal.
- True negative (TN): denotes instances that are properly monitored as an attack and predicts negative results. The overview of the overview of the confusion matrix is given in Table 4.

Table 4. Overview of the confusion matrix.

		Predicated Value	
Actual value	Normal	TP	FN
	Anomaly	FP	TN

We can calculate the system's output using the above specifications of the confusion matrix (CM). DR and FAR are two crucial and common parameters for the analysis of an IDS. The sum of misclassified regular instances is identified as FAR, while the amount of intrusion incidents identified by the model is recognized as DR.

$$FAR = FP/(TN + FP) \quad (1)$$

$$DR = TP/(TP + FN) \quad (2)$$

We say that the HCRNNIDS approach is better as compared with traditional approaches as DR rises and FAR drops.

4.2. Evaluation of the Proposed HCRNNIDS

Table 5 shows the classifier's performance with CSE-CIC-DS2018. The results were created through the random search hyperparameter optimization technique. The ensemble classifier XGB manages to boost the attack classification performance significantly, with an accuracy of 83%. The tree-based classifier gives better accuracy as compared with the ensemble-based classifiers.

Table 5. Classifier performance with CSE-CIC-DS2018.

Classifier	Precision	Recall	F1-Score	DR	FAR
LR	0.781	0.801	0.791	0.80	11.50
XGB	0.845	0.834	0.839	0.83	9.13
DT	0.8733	0.885	0.879	0.88	7.8
HCRNN	0.9633	0.9712	0.976	0.97	2.5

We had also tried different combinations of the algorithm. We used several traditional machine-learning-based classifiers, but as our main aim was to use the HCRNN to capture both spatial as well as temporal features more robustly, we introduced recurrent layers after the CNN layers. This way, we attempted to address the vanishing and exploding gradient problems effectively, which improves the capability to capture spatial and temporal dependencies and learn efficiently from variable extent sequences. While traditional machine-learning-based classifiers are not suitable and computationally efficient

for high-dimensional data (imbalances), typically the number of variables largely exceeds the sample size. Its scale invariance makes it a good candidate for such a high-dimensional dataset. However, the most important improvement was obtained with advanced the DL approaches like CNNRNN, which accurately detected misuse in up to 97.6% of instances. The performance enhancement was due to the long-term dependency among nonlinear features and implementation details are given in Supplementary Materials.

4.3. Overall Evaluation

Table 6 summarizes the outcomes with the existing methods for the CSE-CIC-IDS2018 dataset. Since these datasets were created after the DARPA and KDD data, there are a few preliminary results available. Because of the current simulation performance, the best outcomes for each phase in terms of FAR and accuracy were determined. The proposed HCRNNIDS outperforms the state-of-the-art techniques in terms of accuracy and FAR. This is due to the deep learning approach's execution. It is important to note that the similarities are only meant to act as a reference because various researchers have used distinct quantities of data distributions, pre-processing procedures, and sampling methods. As a result, a simple measurement of metrics such as the testing and training time is rarely appropriate. Though the proposed HCRNNIDS outperformed the evaluated metrics, it is difficult to believe that the proposed approach fully outperformed other approaches. With the proposed solution, we claim that one can achieve an exceptional amount of network protection and easily identify malicious threats.

Table 6. Comparison of existing approaches with the CSE-CIC-IDS2018 data.

Reference	Methods	Accuracy	False Alarm Rate
Peng et al. [64]	DBN	95%	0.98
Farhan et al. [65]	DNN	90.25%	-
Lin et al. [66]	Deep learning	95.0%	05.99
Rawaa et al. [67]	LSTM	96.2%	8.6
Zhou et al. [68]	IDS using DL	96.0%	
Kim et al. [69]	CNN IDS	96.0%	
Our approach	HCRNN	97.75%	1.4

5. Conclusions and Future Work

In this article, the NIDS was formed using a HCRNN, which is effective in cybersecurity. We trained the ID system framework utilizing a CSE-CIC-DS2018 dataset. We executed the IDS using a few traditional classification techniques (LR, DT, XGB, etc.) and the HCRNN technique for the proposed ID system. To capture both spatial as well as temporal features more robustly, we introduced recurrent layers after the CNN layers. This way, we attempted to address the vanishing and exploding gradient problems effectively, which improves the capability to capture spatial and temporal dependencies and learn efficiently from variable extent sequences. The key reason for the proposed ID system based on DL classification is to combine the benefits of both anomaly-based (AB) and signature-based (SB) methods. The proposed ID system helps to reduce the computational complexity and results in an enhancement in accuracy and DR for intrusion detection.

Both traditional ML and deep learning methods were assessed using renowned classification metrics (DR, Accuracy, Precision, Recall, and F1 score). The simulation results show that the proposed HCRNNIDS can successfully realize the calcification of malicious attack events. The overall accuracy of the normal and other types of attacks reaches around 97.75% in the CSE-CIC-IDS2018 data. Based on the simulation results, we can conclude that one can achieve an effective security solution against malicious attacks using the HCNRNN deep learning model.

However, one possible limitation of the proposed approach is that we have tested the HCRNNIDS only on a single ID dataset. It will also be important to test it on a more recent dataset since the signature of the attached traffic often changes. We think that the

proposed scheme can be extended, in the future, into many domains, such as anomalies, and misuses could be identified in various real image datasets in the IoT. We will focus on exploring various other deep learning methods with a feature extraction technique to learn knowledgeable data illustrations in the case of other ID issues in contemporary, realistic datasets.

Supplementary Materials: The source codes of the implementation are available on GitHub at https://github.com/Ashfaqjiskani/Hybrid-Convolutional-Recurrent-Neural-Network-Based-Network-IDS.

Author Contributions: Muhammad Ashfaq Khan conceived the research concept, built the prototype, and performed the experiment. The author has read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Acknowledgments: This research was supported by the IoT and Big-Data Research Center, Department of Electronics Engineering, Incheon National, University, Incheon 2012, South Korea.

Conflicts of Interest: The author declares no conflict of interest.

Abbreviations

ML	Machine Learning
DL	Deep learning
HCRNNIDS	Hybrid Convolutional Recurrent Neural Network Based Network Intrusion Detection System
CNN	Convolutional neural network
RNN	Recurrent neural network
CSE-CIC	A collaborative project between the Communications Security Establishment (CSE) & the Canadian Institute for Cybersecurity (CIC)
ICT	Information and communication technology
ID	Intrusion Detection
IDS	Intrusion Detection System
DR	Detection Rate
FAR	False Alarm Rate
CV	Computer vision
NLP	Natural Language Processing
FPR	False-positive rate
NID	Network Intrusion Detection
K-NN	K nearest neighbors
SVM	Support vector machine
ANN	Artificial neural network
RF	Random Forest
DN	Deep network's
LSTM	Long Short-Term-Memory
AE	Auto-encoder
FL	Feature learning
RBMs	Restricted Boltzmann Machines
LR	Logistic Regression
XGB	Extreme Gradient Boosting
DT	Decision Tree
AIDS	Anomaly intrusion detection system
SMOTE	Synthetic Minority Oversampling Technique
AWS platform	Amazon Web Services
DoS	Denial of Service

DDoS	Distributed Denial of Service
FN	False Negative
FP	False Positive
TP	True Positive
TN	True negative
AB	Anomaly-based
SB	Signature-based

References

1. Anderson, J.P. Technical Report. In *Computer Security Threat Monitoring and Surveillance*; James P. Anderson Company: Washington, DC, USA, 1980.
2. Asharf, J.; Moustafa, N.; Khurshid, H.; Debie, E.; Haider, W.; Wahab, A. A Review of Intrusion Detection Systems Using Machine and Deep Learning in Internet of Things: Challenges, Solutions and Future Directions. *Electronics* **2020**, *9*, 1177. [CrossRef]
3. Xu, C.; Shen, J.; Du, X.; Zhang, F. An Intrusion Detection System Using a Deep Neural Network with Gated Recurrent Units. *IEEE Access* **2018**, *6*, 48697–48707. [CrossRef]
4. Vinayakumar, R.; Soman, K.P.; Poornachandran, P. Applying convolutional neural network for network intrusion detection. In *Proceedings of the 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Udupi, India, 13–16 September 2017*; Institute of Electrical and Electronics Engineers (IEEE): Piscataway, NJ, USA, 2017; pp. 1222–1228.
5. Khan, M.A.; Kim, Y. Deep Learning-Based Hybrid Intelligent Intrusion Detection System. *Comput. Mater. Contin.* **2021**, *68*, 671–687. [CrossRef]
6. Devi, B.T.; Thirumaleshwari, S.S.; Jabbar, M.A. An Appraisal over Intrusion Detection Systems in Cloud Computing Security Attacks. In *Proceedings of the 2020 2nd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA), Bangalore, India, 5–7 March 2020*; Institute of Electrical and Electronics Engineers (IEEE): Piscataway, NJ, USA, 2020; pp. 722–727.
7. Thaseen, I.S.; Poorva, B.; Ushasree, P.S. Network Intrusion Detection using Machine Learning Techniques. In *Proceedings of the 2020 International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE), Tamil Nadu, India, 24–25 February 2020*; Institute of Electrical and Electronics Engineers (IEEE): Piscataway, NJ, USA, 2020; pp. 1–7.
8. Yin, C.; Zhu, Y.; Fei, J.; He, X. A Deep Learning Approach for Intrusion Detection Using Recurrent Neural Networks. *IEEE Access* **2017**, *5*, 21954–21961. [CrossRef]
9. Yong, B.; Wei, W.; Li, K.C.; Shen, J.; Zhou, Q.; Wozniak, M.; Połap, D.; Damaševičius, R. Ensemble machine learning approaches for web shell detection in Internet of things environments. In *Transactions on Emerging Telecommunications Technologies*; John Wiley & Sons, Ltd.: New Jersey, USA, 2020.
10. Folino, F.; Folino, G.; Guarascio, M.; Pisani, F.; Pontieri, L. On learning effective ensembles of deep neural networks for intrusion detection. *Inf. Fusion* **2021**, *72*, 48–69. [CrossRef]
11. Tama, B.A.; Lim, S. Ensemble learning for intrusion detection systems: A systematic mapping study and cross-benchmark evaluation. *Comput. Sci. Rev.* **2021**, *39*, 100357. [CrossRef]
12. Kim, K.; Aminanto, M.E.; Tanuwidjaja, H.C. *Network Intrusion Detection Using Deep Learning: A Feature Learning Approach*; Springer: Berlin/Heidelberg, Germany, 2018.
13. Avci, O.; Abdeljaber, O.; Kiranyaz, S.; Hussein, M.; Gabbouj, M.; Inman, D.J. A review of vibration-based damage detection in civil structures: From traditional methods to Machine Learning and Deep Learning applications. *Mech. Syst. Signal Process.* **2021**, *147*, 107077. [CrossRef]
14. Kumar, K.P.M.; Saravanan, M.; Thenmozhi, M.; Vijayakumar, K. Intrusion detection system based on GA-fuzzy classifier for detecting malicious attacks. *Concurr. Comput. Pr. Exp.* **2021**, *33*, 5242. [CrossRef]
15. Zhang, H.; Huang, L.; Wu, C.Q.; Li, Z. An effective convolutional neural network based on SMOTE and Gaussian mixture model for intrusion detection in imbalanced dataset. *Comput. Netw.* **2020**, *177*, 107315. [CrossRef]
16. Binbusayyis, A.; Vaiyapuri, T. Identifying and Benchmarking Key Features for Cyber Intrusion Detection: An Ensemble Approach. *IEEE Access* **2019**, *7*, 106495–106513. [CrossRef]
17. Bhavani, T.T.; Rao, M.K.; Reddy, A.M. Network Intrusion Detection System Using Random Forest and Decision Tree Machine Learning Techniques. In Proceedings of the Distributed Computing and Artificial Intelligence, 13th International Conference, Sevilla, Spain, 1–3 June 2016; Springer: Berlin/Heidelberg, Germany, 2019; pp. 637–643.
18. Karatas, G.; Demir, O.; Sahingoz, O.K. Increasing the Performance of Machine Learning-Based IDSs on an Imbalanced and Up-to-Date Dataset. *IEEE Access* **2020**, *8*, 32150–32162. [CrossRef]
19. Xu, H.; Przystupa, K.; Fang, C.; Marciniak, A.; Kochan, O.; Beshley, M. A Combination Strategy of Feature Selection Based on an Integrated Optimization Algorithm and Weighted K-Nearest Neighbor to Improve the Performance of Network Intrusion Detection. *Electronics* **2020**, *9*, 1206. [CrossRef]
20. Bhati, B.S.; Rai, C.S. Analysis of Support Vector Machine-based Intrusion Detection Techniques. *Arab. J. Sci. Eng.* **2019**, *45*, 2371–2383. [CrossRef]
21. Thaseen, I.S.; Banu, J.S.; Lavanya, K.; Ghalib, M.R.; Abhishek, K. An integrated intrusion detection system using correlation-based attribute selection and artificial neural network. *Trans. Emerg. Telecommun. Technol.* **2021**, *32*, 4014. [CrossRef]

22. Waskle, S.; Parashar, L.; Singh, U. Intrusion Detection System Using PCA with Random Forest Approach. In *Proceedings of the 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC), Coimbatore, India, 2–4 July 2020*; Institute of Electrical and Electronics Engineers (IEEE): Piscataway, NJ, USA, 2020; pp. 803–808.
23. Alqahtani, H.; Sarker, I.H.; Kalim, A.; Hossain, S.M.M.; Ikhlaq, S.; Hossain, S. Cyber Intrusion Detection Using Machine Learning Classification Techniques. In *Communications in Computer and Information Science*; Springer Science and Business Media LLC: Berlin/Heidelberg, Germany, 2020; Volume 1235, pp. 121–131.
24. Ahmad, Z.; Khan, A.S.; Shiang, C.W.; Abdullah, J.; Ahmad, F. Network intrusion detection system: A systematic study of machine learning and deep learning approaches. *Trans. Emerg. Telecommun. Technol.* **2021**, *32*, 4150. [CrossRef]
25. Girdler, T.; Vassilakis, V.G. Implementing an intrusion detection and prevention system using Software-Defined Networking: Defending against ARP spoofing attacks and Blacklisted MAC Addresses. *Comput. Electr. Eng.* **2021**, *90*, 106990. [CrossRef]
26. Aldweesh, A.; Derhab, A.; Emam, A.Z. Deep learning approaches for anomaly-based intrusion detection systems: A survey, taxonomy, and open issues. *Knowl. Based Syst.* **2020**, *189*, 105124. [CrossRef]
27. Jihyun, K.; Jaehyun, K.; Huong, L.T.T.; Howon, K. Long short-term memory recurrent neural network classifier for intrusion detection. In *Proceedings of the 2016 International Conference on Platform Technology and Service (PlatCon), Jeju, Korea, 15–17 February 2016*; IEEE: Piscataway, NJ, USA, 2016; pp. 1–5.
28. Vinayakumar, R.; Soman, K.P.; Poornachandran, P. Deep android malware detection and classification. In *Proceedings of the 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Udupi, India, 13–16 September 2017*; Institute of Electrical and Electronics Engineers (IEEE): Piscataway, NJ, USA, 2017; pp. 1677–1683.
29. Adebowale, M.A.; Lwin, K.T.; Hossain, M.A. Intelligent phishing detection scheme using deep learning algorithms. *J. Enterp. Inf. Manag.* **2020**, 1–20. [CrossRef]
30. Tran, D.; Mac, H.; Tong, V.; Tran, H.A.; Nguyen, L.G. A LSTM based framework for handling multiclass imbalance in DGA botnet detection. *Neurocomputing* **2018**, *275*, 2401–2413. [CrossRef]
31. Oliveira, N.; Praça, I.; Maia, E.; Sousa, O. Intelligent Cyber Attack Detection and Classification for Network-Based Intrusion Detection Systems. *Appl. Sci.* **2021**, *11*, 1674. [CrossRef]
32. Ahmad, R.; Alsmadi, I. Machine learning approaches to IoT security: A systematic literature review. *Internet Things* **2021**, *14*, 100365. [CrossRef]
33. Makuvaza, A.; Jat, D.S.; Gamundani, A.M. Deep Neural Network (DNN) Solution for Real-time Detection of Distributed Denial of Service (DDoS) Attacks in Software Defined Networks (SDNs). *SN Comput. Sci.* **2021**, *2*, 1–10. [CrossRef]
34. Millar, S.; McLaughlin, N.; del Rincon, J.M.; Miller, P. Multi-view deep learning for zero-day Android malware detection. *J. Inf. Secur. Appl.* **2021**, *58*, 102718. [CrossRef]
35. Guijuan, Z.; Yang, L.; Xiaoning, J. A survey of autoencoder-based recommender systems. *Front. Comput Sci.* **2020**, *14*, 430–450.
36. Liu, J.; Song, K.; Feng, M.; Yan, Y.; Tu, Z.; Zhu, L. Semi-supervised anomaly detection with dual prototypes autoencoder for industrial surface inspection. *Opt. Lasers Eng.* **2021**, *136*, 106324. [CrossRef]
37. Yousefi-Azar, M.; Varadharajan, V.; Hamey, L.; Tupakula, U. Autoencoder-based feature learning for cybersecurity applications. In *Proceedings of the 2017 International Joint Conference on Neural Networks (IJCNN), Anchorage, AK, USA, 14–19 May 2017*; IEEE: Piscataway, NJ, USA, 2017; pp. 3854–3861.
38. Khan, M.A.; Kim, J. Toward Developing Efficient Conv-AE-Based Intrusion Detection System Using Heterogeneous Dataset. *Electronics* **2020**, *9*, 1771. [CrossRef]
39. Yadigar, I.; Fargana, A. Deep learning method for denial-of-service attack detection based on restricted Boltzmann machine. *Big Data* **2018**, *6*, 159–169.
40. Tan, Z.; Jamdagni, A.; He, X.; Nanda, P.; Liu, R.P.; Hu, J. Detection of Denial-of-Service Attacks Based on Computer Vision Techniques. *IEEE Trans. Comput.* **2014**, *64*, 2519–2533. [CrossRef]
41. Ingre, B.; Yadav, A. Performance analysis of NSL-KDD dataset using ANN. In Proceedings of the 2015 International Conference on Signal Processing and Communication Engineering Systems, Guntur, India, 2–3 January 2015; pp. 92–96.
42. Casas, P.; Mazel, J.; Owezarski, P. Unsupervised Network Intrusion Detection Systems: Detecting the Unknown without Knowledge. *Comput. Commun.* **2012**, *35*, 772–783. [CrossRef]
43. Ludwig, S.A. Intrusion detection of multiple attack classes using a deep neural net ensemble. In *Proceedings of the 2017 IEEE Symposium Series on Computational Intelligence (SSCI), Honolulu, HI, USA, 27 November–1 December 2017*; Institute of Electrical and Electronics Engineers (IEEE): Piscataway, NJ, USA, 2017; pp. 1–7.
44. Shone, N.; Ngoc, T.N.; Phai, V.D.; Shi, Q. A Deep Learning Approach to Network Intrusion Detection. *IEEE Trans. Emerg. Top. Comput. Intell.* **2018**, *2*, 41–50. [CrossRef]
45. Kakavand, M.; Mustapha, N.; Mustapha, A.; Abdullah, M.T. Effective Dimensionality Reduction of Payload-Based Anomaly Detection in TMAD Model for HTTP Payload. *KSII Trans. Internet Inf. Syst.* **2016**, *10*, 3884–3910. [CrossRef]
46. Yu, Y.; Long, J.; Cai, Z. Network Intrusion Detection through Stacking Dilated Convolutional Autoencoders. *Secur. Commun. Netw.* **2017**, *2017*, 1–10. [CrossRef]
47. Kumar, G.; Kumar, K. Design of an Evolutionary Approach for Intrusion Detection. *Sci. World J.* **2013**, *2013*, 1–14. [CrossRef] [PubMed]
48. Akyol, A.; Hacibeyoğlu, M.; Karlik, B. Design of Multilevel Hybrid Classifier with Variant Feature Sets for Intrusion Detection System. *IEICE Trans. Inf. Syst.* **2016**, *E99*, 1810–1821. [CrossRef]

49. Almomani, O. A Feature Selection Model for Network Intrusion Detection System Based on PSO, GWO, FFA and GA Algorithms. *Symmetry* **2020**, *12*, 1046. [CrossRef]
50. Monshizadeh, M.; Khatri, V.; Atli, B.G.; Kantola, R.; Yan, Z. Performance Evaluation of a Combined Anomaly Detection Platform. *IEEE Access* **2019**, *7*, 100964–100978. [CrossRef]
51. Wang, W.; Sheng, Y.; Wang, J.; Zeng, X.; Ye, X.; Huang, Y.; Zhu, M. HAST-IDS: Learning Hierarchical Spatial-Temporal Features Using Deep Neural Networks to Improve Intrusion Detection. *IEEE Access* **2017**, *6*, 1792–1806. [CrossRef]
52. Bhati, N.S.; Khari, M. A Survey on Hybrid Intrusion Detection Techniques. In *Advances in Human Factors, Business Management, Training and Education*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 815–825.
53. Ma, C.; Du, X.; Cao, L. Analysis of Multi-Types of Flow Features Based on Hybrid Neural Network for Improving Network Anomaly Detection. *IEEE Access* **2019**, *7*, 148363–148380. [CrossRef]
54. Zeng, Y.; Gu, H.; Wei, W.; Guo, Y. Deep-Full-Range: A Deep Learning Based Network Encrypted Traffic Classification and Intrusion Detection Framework. *IEEE Access* **2019**, *7*, 45182–45190. [CrossRef]
55. Hosseini, S.; Zade, B.M.H. New hybrid method for attack detection using combination of evolutionary algorithms, SVM, and ANN. *Comput. Netw.* **2020**, *173*, 107168. [CrossRef]
56. Erhan, D.; Anarım, E. Boğaziçi University distributed denial of service dataset. *Data Brief* **2020**, *32*, 106187. [CrossRef] [PubMed]
57. Damasevicius, R.; Venckauskas, A.; Grigaliunas, S.; Toldinas, J.; Morkevicius, N.; Aleliunas, T.; Smuikys, P. LITNET-2020: An Annotated Real-World Network Flow Dataset for Network Intrusion Detection. *Electronics* **2020**, *9*, 800. [CrossRef]
58. Gogoi, P.; Bhuyan, M.H.; Bhattacharyya, D.K.; Kalita, J.K. Packet and flow-based network intrusion dataset. In *International Conference on Contemporary Computing*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 322–334.
59. Bedi, P.; Gupta, N.; Jindal, V. I-SiamIDS: An improved Siam-IDS for handling class imbalance in network-based intrusion detection systems. *Appl. Intell.* **2021**, *51*, 1133–1151. [CrossRef]
60. Thabtah, F.; Hammoud, S.; Kamalov, F.; Gonsalves, A. Data imbalance in classification: Experimental evaluation. *Inf. Sci.* **2020**, *513*, 429–441. [CrossRef]
61. A Collaborative Project between the Communications Security Establishment (CSE) & the Canadian Institute for Cybersecurity (CIC). Available online: https://www.unb.ca/cic/datasets/ids-2018.html (accessed on 31 March 2021).
62. Drhooge, L.; Wauters, T.; Volckaert, B.; de Turck, F. Classification Hardness for Supervised Learners on 20 Years of Intrusion Detection Data. *IEEE Access* **2019**, *7*, 167455–167469. [CrossRef]
63. Jaganathan, V.; Cherurveettil, P.; Sivashanmugam, P.M. Using a Prediction Model to Manage Cyber Security Threats. *Sci. World J.* **2015**, *2015*, 1–5. [CrossRef] [PubMed]
64. Wei, P.; Li, Y.; Zhang, Z.; Hu, T.; Li, Z.; Liu, D. An Optimization Method for Intrusion Detection Classification Model Based on Deep Belief Network. *IEEE Access* **2019**, *7*, 87593–87605. [CrossRef]
65. Farhan, R.I.; Abeer, T.M.; Nidaa, F.H. Optimized Deep Learning with Binary PSO for Intrusion Detection on CSE-CIC-IDS2018 Dataset. *J. Al Qadisiyah Comput. Sci. Math.* **2020**, *12*, 16.
66. Farhan, R.I.; Abeer, T.M.; Nidaa, F.H. Performance Analysis of Flow-Based Attacks Detection on CSE-CIC-IDS2018 Dataset Using Deep Learning. *Indones. J. Electr. Eng. Comput. Sci.* **2020**, *20*, 16–27. [CrossRef]
67. Lin, P.; Ye, K.; Xu, C.-Z. Dynamic Network Anomaly Detection System by Using Deep Learning Techniques. In *Lecture Notes in Computer Science*; Springer Science and Business Media LLC: Cham, Switzerland, 2019; pp. 161–176.
68. Zhou, Q.; Pezaros, D. Evaluation of Machine Learning Classifiers for Zero-Day Intrusion Detection—An Analysis on CIC-AWS-2018 dataset. *arXiv* **2019**, arXiv:1905.03685*v1*, 1–18.
69. Kim, J.; Shin, Y.; Choi, E. An Intrusion Detection Model based on a Convolutional Neural Network. *J. Multimed. Inf. Syst.* **2019**, *6*, 165–172. [CrossRef]