applied sciences

Article

Detection of Unknown DDoS Attacks with Deep Learning and Gaussian Mixture Model

Chin-Shiuh Shieh [1], Wan-Wei Lin [1,*], Thanh-Tuan Nguyen [1], Chi-Hong Chen [1], Mong-Fong Horng [1] and Denis Miu [2]

[1] Department of Electronic Engineering, National Kaohsiung University of Science and Technology, Kaohsiung 807618, Taiwan; csshieh@nkust.edu.tw (C.-S.S.); tuannt@ntu.edu.vn (T.-T.N.); f107152131@nkust.edu.tw (C.-H.C.); mfhorng@nkust.edu.tw (M.-F.H.)

[2] Genie Networks Ltd., Taipei 11444, Taiwan; denis@genie-networks.com

* Correspondence: i109152103@nkust.edu.tw

Abstract: DDoS (Distributed Denial of Service) attacks have become a pressing threat to the security and integrity of computer networks and information systems, which are indispensable infrastructures of modern times. The detection of DDoS attacks is a challenging issue before any mitigation measures can be taken. ML/DL (Machine Learning/Deep Learning) has been applied to the detection of DDoS attacks with satisfactory achievement. However, full-scale success is still beyond reach due to an inherent problem with ML/DL-based systems—the so-called Open Set Recognition (OSR) problem. This is a problem where an ML/DL-based system fails to deal with new instances not drawn from the distribution model of the training data. This problem is particularly profound in detecting DDoS attacks since DDoS attacks' technology keeps evolving and has changing traffic characteristics. This study investigates the impact of the OSR problem on the detection of DDoS attacks. In response to this problem, we propose a new DDoS detection framework featuring Bi-Directional Long Short-Term Memory (BI-LSTM), a Gaussian Mixture Model (GMM), and incremental learning. Unknown traffic captured by the GMM are subject to discrimination and labeling by traffic engineers, and then fed back to the framework as additional training samples. Using the data sets CIC-IDS2017 and CIC-DDoS2019 for training, testing, and evaluation, experiment results show that the proposed BI-LSTM-GMM can achieve recall, precision, and accuracy up to 94%. Experiments reveal that the proposed framework can be a promising solution to the detection of unknown DDoS attacks.

Keywords: distributed denial of service (DDoS); machine learning; long short-term memory (LSTM); gaussian mixture model; incremental learning

check for updates

Citation: Shieh, C.-S.; Lin, W.-W.; Nguyen, T.-T.; Chen, C.-H.; Horng, M.-F.; Miu, D. Detection of Unknown DDoS Attacks with Deep Learning and Gaussian Mixture Model. *Appl. Sci.* **2021**, *11*, 5213. https://doi.org/10.3390/app11115213

Academic Editor: Eui-Nam Huh

Received: 6 May 2021
Accepted: 1 June 2021
Published: 4 June 2021

1. Introduction

By flooding malicious traffic, DoS (Denial of Service) attacks deplete the network bandwidth and computing resources of a targeted system, preventing the target system from offering regular services to legitimate users. DDoS (Distributed Denial of Service) [1] goes even further on a much larger scale. DDoS attacks take over the control of a large number of comprised systems, called a botnet, and launch coordinated attacks on the victim system, as illustrated in Figure 1. Along with the emergence and advancement of disruptive Internet technologies, DDoS attacks are evolving and proliferating in scale, frequency, and sophistication. Organizations face potential threats to their network environment that may cause severe impacts to their operations, such as business downtime, data breaches, or even ransom demands from hackers [2].

Upon the occurrence of DDoS attacks, actions for DDoS mitigation should be taken, as suggested in [3]. The detection of DDoS attacks is essential before any mitigation approaches can be taken. In the early era, the alarm of DDoS attacks was triggered by rules programmed by traffic engineers. This approach apparently failed to catch up with the dynamic and evolving natures of DDoS attacks. As ML/DL (Machine Learning/Deep

Learning) unleashes their great potential in different fields, academics and industries are exploring the possibility of applying ML/DL to DDoS detection. Traditional manual methods suffer from low accuracy and long latency problems in risk identification. With ML approaches, such as Naive Bayesian, KNN, and Random Forest, threats can be captured more quickly and more accurately [4]. In ML, features for classification must be selected by human experts or by certain feature selection schemes. On the other hand, feature selection is an integral part of DL. Deep Learning models such as CNN and RNN are built based on a series of nonlinear processing layers to learn many levels of data representation from a large volume of labeled samples. Therefore, DL can serve as a powerful tool for DDoS detection [5]. Some successful stories of ML/DL use for DDoS detection will be reviewed in Section 2. After exploring several alternatives, we employ the Bi-Directional Long Short-Term Memory (BI-LSTM) [6] DL architecture in this study. BI-LSTM is capable of capturing essential characteristics of DDoS traffic, in particular, the time domain correlation. Experiments show that it serves well for the intended purpose.

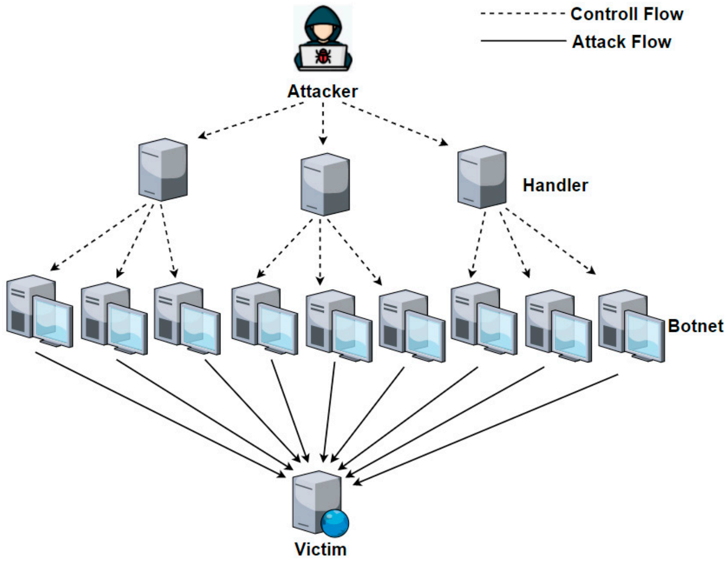


Figure 1. DDoS attack with a botnet.

ML and DL have proven themselves effective solutions to the detection of DDoS attacks. However, they are trained to recognize only instances drawn from the distribution models constructed from the training set. Thus, they could fail in cases they have never learned. To know what one doesn't know is the problem called the Open Set Recognition (OSR) problem [7]. This problem has a severe impact on the detection of DDoS attacks since DDoS attacks' technology keeps evolving and results in changing traffic characteristics. A Gaussian Mixture Model (GMM) [8] is incorporated into our framework to differentiate trained samples and novel instances. Unknown traffics captured by the GMM are then subject to discrimination and labeling by traffic engineers, and then fed back to the framework as additional training samples in the incremental learning phase. The main contributions of this paper are summarized as follows:

- We identify the detection of unknown DDoS attacks as an Open Set Recognition problem and demonstrate its impact on conventional detection approaches.
- We propose a new BI-LSTM-GMM model to detect the unknown network attack. The proposed framework can successfully differentiate novel instances from samples drawn from trained models.
- Using the data sets CIC-IDS2017 and CIC-DDoS2019 for training, testing, and evaluation, experiment results show that the proposed BI-LSTM-GMM can achieve recall, precision, and accuracy up to 94%.

The rest of this paper is organized as follows. Previous works and related technologies are briefly reviewed in Section 2. The framework of the proposed approach is presented in Section 3. Then, experiment results are reported in Section 4. Finally, some conclusions are drawn in Section 5.

2. Related Works

Various ML technologies have been employed, mainly as classifiers, in the detection of DDoS attacks. These include Support Vector Machines (SVM), k-Nearest Neighbors (KNN), the Naïve Bayes Classifier, Random Forest (RF), Density-Based Spatial Clustering of Applications with Noise (DBSCAN), and Artificial Neural Networks (ANNs), to name a few. With SVM, based on labeled training data, a hyperplane is constructed in the transform domain to classify unseen data. Cheng has built an IAI (IP Address Interaction Feature) model that can effectively discriminate normal from abnormal flows in traffic streams and helps to identify attack flows quickly and precisely [9]. In KNN, *k* nearest neighbors of incoming data are located. A majority of these *k* neighbors decide the classification of the incoming data. Vu has applied KNN [10] to classify the state of networks according to each stage of DDoS attack and obtained optimific results. A Naïve Bayes classifier is a classification technique based on Bayes' theorem assuming independence among predictors. A Naïve Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature. Fadil used the NB method [11] to predict the existence of DDoS attacks based on the mean and standard deviation of network packets and achieved precise results. An RF is a collection of decision trees. The majority of the outcomes of individual decision trees determine the classification. In [12], Wang et al. proved that with well-computed key features in DDoS data, experimental results show an RF algorithm can attain valid classification performance and an optimal feature subset. DBSCN finds core samples of high density and expands clusters from them. It particularly suits data that contains clusters of similar density. Dincalp used a DBSCAN clustering algorithm [13] to deal with diversity in attack vectors. The proposed system worked well with chosen attributes in their experiments. ANNs emulate biological neural networks. Given label data, ANNs learn the mapping function using the back-propagation algorithm. Ahanger proposed an ANN-based DDoS detection method that provided 99.8% detection accuracy in recognizing DDoS attacks [14].

There are also successful stories for DL in DDoS detection. Li and Lu [15] combined Long Short-Term Memory (LSTM) and Bayesian methods to detect DDoS attacks. LSTM is suitable for events with long intervals and delays in the time domain. In other words, LSTM can control the value of the indefinite length of time and decide whether the information should be retained or removed. The author used LSTM to identify the confidence index of DDoS attacks and further used the Bayesian method to make a second judgment to improve detection accuracy. Yang et al. [16] adopted the autoencoder for the detection of DDoS attacks. Autoencoder is a multi-layer neural network with an unsupervised training algorithm. It removes less relevant information and noise during the training process and retains essential information. Doriguzzi-Corin et al. [17] employed CNN in their detection system, named LUCID, featuring a dataset-agnostic preprocessing mechanism and an activation analysis. The proposed LUCID has 40 times of reduction in the processing time, rendering it well-suited in resource constrained environments. Yong et al. [18] applied machine learning models to detect webshell to build secure solutions for IoT networks. Ensemble methods, including random forest (RF), extremely randomized trees (ET), and Voting, are used to improve the performances of these machine learning models. Their findings show that RF and ET are suitable for lightweight IoT scenarios, and the Voting method is effective for heavyweight IoT scenarios. Hemalatha et al. [19] propose an efficient malware detection system based on deep learning. The system uses a reweighted class-balanced loss function in the final classification layer of the DenseNet model to achieve significant performance improvements in classifying malware by handling imbalanced data issues. Comprehensive experiments performed on four benchmark malware datasets showed that the proposed scheme has superior performance with a higher detection rate and lower computational cost.

Despite the success attained by the ML/DL-based schemes, a critical issue, the OSR problem, is overlooked. Bendale et al. [20] and Sabeel et al. [21] point out that, without proper measures, ML/DL could forcibly classify instances from new sample space into the wrong category. The correct action is to differentiate novel instances from training samples, which is the strategy adopted by [20,22]. They examined the hyper distance from incoming data to known classes and marked it as a new instance if the distance exceeded a certain threshold. In [20], a new model layer, OpenMax, is introduced, which estimates the probability of an input being from an unknown class. In [22], the Extreme Value Machine (EVM) is proposed, which uses the Weibull function for model construction from the training data set. Given a new sample, EVM will estimate the inclusion probability of each existing class for classification purposes. A new instance, distant from all classes, is considered belonging to a new class. This characteristic renders EVM a good candidate for OSR problems, including the detection of known DDoS attacks.

We follow the same philosophy but with different technology. To be of practical significance, this study deals with the OSR problem by incorporating the GMM into the proposed DDoS detection framework. The OSR problem can then be adequately addressed, as shown later.

3. Proposed Framework—BI-LSTM-GMM

As a solution to the OSR problem in DDoS attack detection, this study proposes a framework integrating Bi-Directional Long Short-Term Memory (BI-LSTM), a Gaussian Mixture Model (GMM), and incremental learning. The functional block diagram of the proposed BI-LSTM-GMM framework is given in Figure 2.

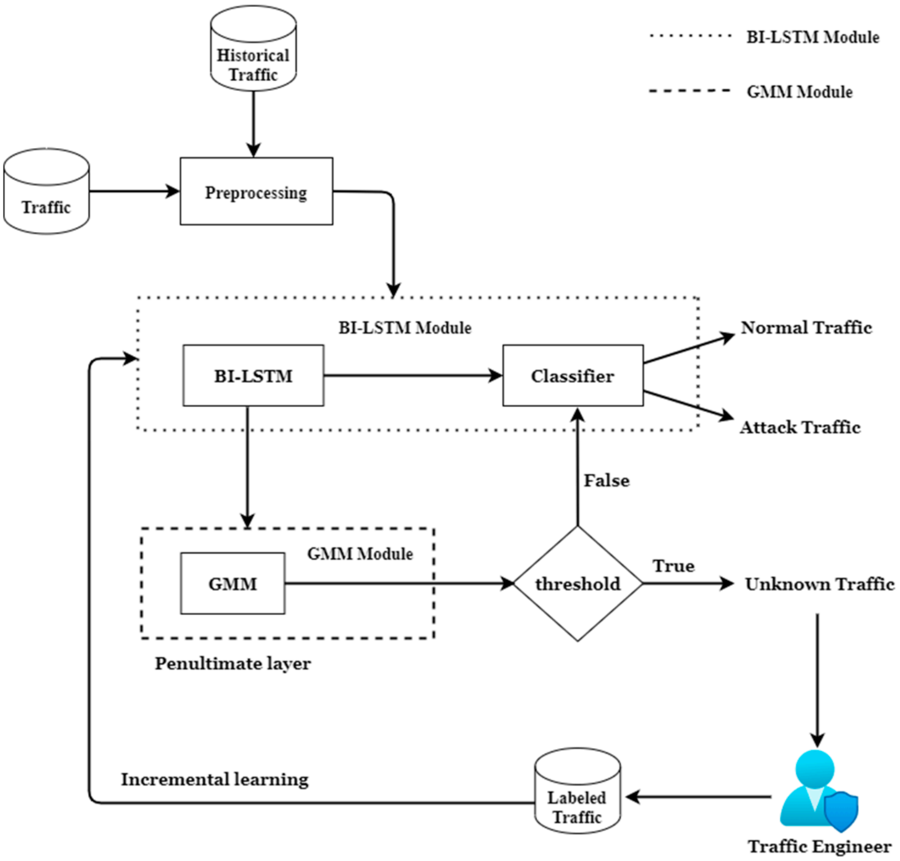


Figure 2. The functional block diagram of the proposed BI-LSTM-GMM.

With supervised learning, the BI-LSTM is employed for the discrimination of regular traffic and DDoS attacks. Based on a concept similar to unsupervised learning, the GMM is adopted to construct the training data's distribution model. With GMM, new attacks or new legitimate traffic can be captured if they fall beyond a specific range of the constructed distribution model. Captured new traffics are subject to the identification and labeling of the traffic engineer and then used to update the BI-LSTM and the GMM in the incremental learning phase. To a certain extent, the GMM can be regarded as a classifier of differentiating learned samples and novel instances and the BI-LSTM as a classifier for the discrimination of good and bad traffics.

Our framework uses historical traffic data for deep learning training to generate models that can be used to detect traffic. We also introduce the Gaussian mixture module to make the model capable of identifying unseen attacks. Finally, unknown attacks or traffic classified by the Gaussian mixture model will be identified and labeled by experts, and the deep learning model will be updated through incremental learning.

3.1. BI-LSTM Module

The LSTM is a Recurrent Neural Network (RNN) architecture with a memory cell and gates for the explicit control of input, output, and forget, as shown in Figure 3. With its unique design, LSTM is particularly suited for applications related to data sequences. The operation of an LSTM is governed by (1), as follows

$$\begin{cases} i_t = \sigma(W_t,\ [h_{t-1}, x_t] + b_i) \\ o_t = \sigma(W_o,\ [h_{t-1}, x_t] + b_o) \\ f_t = \sigma\left(W_f,\ [h_{t-1}, x_t] + b_f\right) \\ \widetilde{C}_t = tanh(W_c,\ [h_{t-1}, x_t] + b_c) \\ C_t = f_t\ \cdot\ C_{t-1} + i_t \cdot\ \widetilde{C}_t \\ h_t = o_t\ \cdot \tanh(C_t) \end{cases} \tag{1}$$

where t is the index of LSTM stage; x_t is the input; h_t is the output; W is the weight; b is the bias, C_t and $\widetilde{C}_t$ are the current and the last states, respectively; and $\sigma(\cdot)$ is the sigmoid function.

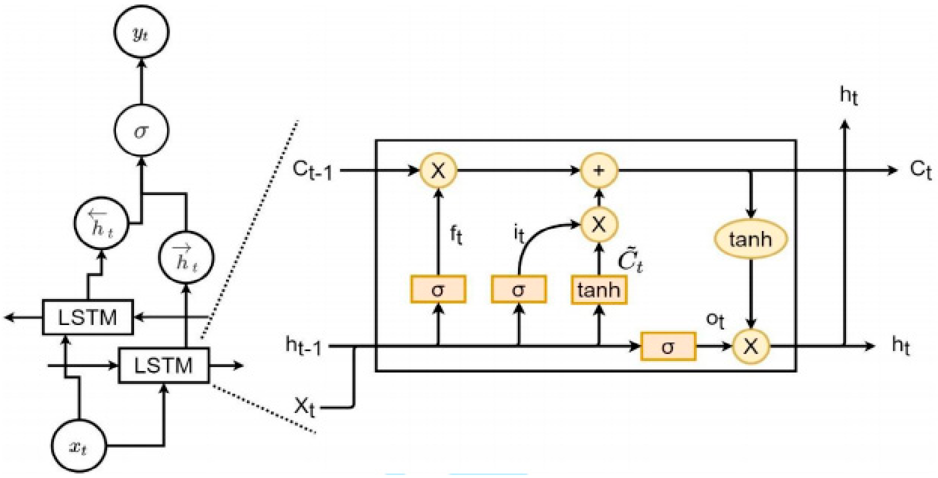


Figure 3. The internal structure of an LSTM stage.

A BI-LSTM is a DL architecture with dual LSTMs, one for the forward direction and the other for the reverse direction, as shown in Figure 4. A BI-LSTM becomes a non-causal system in the sense of data sequences. It is capable of gripping correlations on both sides along the time axis, which is a desired attribute for the detection of DDoS attacks.

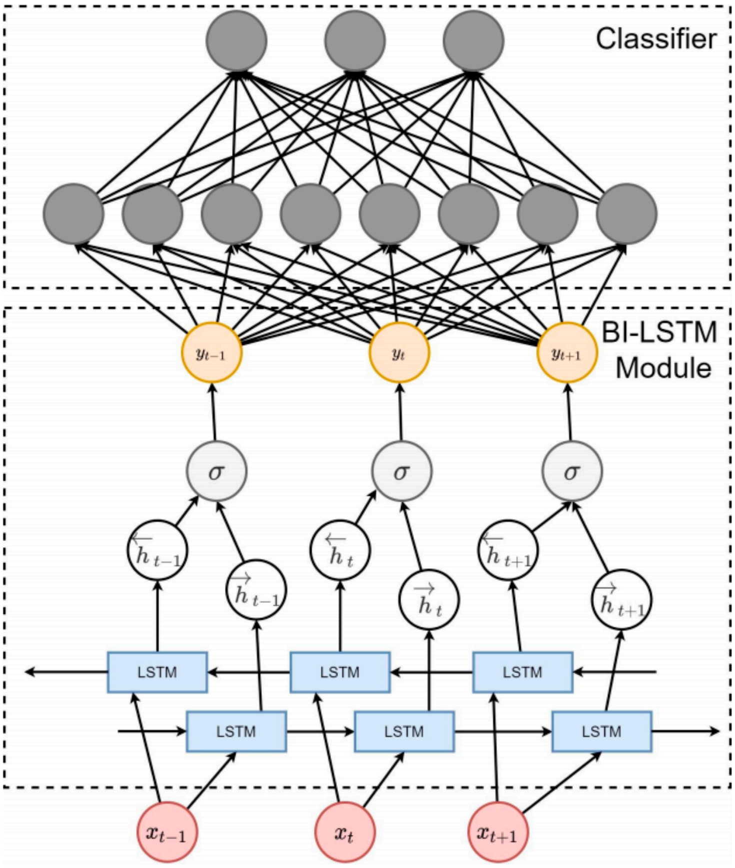


Figure 4. The BI-LSTM and the classifier.

The output of the BI-LSTM is directed to a fully connected layer serving as a classifier for the discernation of normal and malicious traffic. This classifier is gated by the outcome of the GMM. The classifier's output is counted only when the GMM reports the given instance falling within the training data distribution.

3.2. GMM Module

It is critical for an ML/DL-based system to know what it does not know. To endow the DDoS attack detection system with the ability to pick up instances not drawing from the distribution model of the training set, we adopt the GMM in our design, which is an extension of the single Gaussian model. With GMM, multiple Gaussian probability density functions are employed to model clustered data, which is a typical scenario in DDoS attack detection. Different types of legitimate traffic or DDoS attacks bear similar characteristics and form individual clusters in the feature space. With unsupervised learning, GMM can establish a model matching the distribution of the training data set. A GMM is a statistic model consisting of K Gaussian distributions. Each kernel can be specified by its weighting factor w_k, mean $\vec{\mu}_k$, and covariance matrix Σ_k, as follows

$$\Lambda = \left\{ \lambda_k = \left(w_k, \vec{\mu}_k, \Sigma_k \right) \right\},\ k = 1, \ldots, K \tag{2}$$

The likelihood of a given instance $\vec{x}$ belonging to a GMM can be evaluated according to (3), as follows

$$p\left(\vec{x} \middle| \Lambda \right) = \sum_{k=1}^{K} w_k \mathcal{N}\left(\vec{x} \middle| \vec{\mu}_k, \Sigma_k \right) \tag{3}$$

where

$$\begin{cases} N\left(\vec{x}\,\middle|\,\vec{\mu}_k, \Sigma_k\right) = \frac{1}{\sqrt{(2\pi)^K|\Sigma_k|}} \exp\left(-\frac{1}{2}\left(\vec{x}-\vec{\mu}_k\right)^{\mathrm{T}} {\Sigma_k}^{-1}\left(\vec{x}-\vec{\mu}_k\right)\right) \\ \sum\limits_{k=1}^{K} w_k = 1 \end{cases} \tag{4}$$

Given a training data set and K, the expectation–maximization algorithm is the technique most commonly used to estimate the mixture model's parameters. Once the model is constructed, the GMM can be used to reject unseen instances. According to common practice in statistics, an instance falling within three times of the standard deviation, i.e., $\vec{\mu} - 3\vec{\sigma} \leq \vec{x} \leq \vec{\mu} + 3\vec{\sigma}$, is considered an instance drawing from the training data distribution. In this case, the instance will be accepted, and the output of the BI-LSTM will be enabled. Otherwise, the instance will be rejected, and the BI-LSTM will be disabled. A rejected instance will be logged and subject to identification and labeling by data engineers. The control flow of the GMM in the proposed framework is presented in Figure 5.

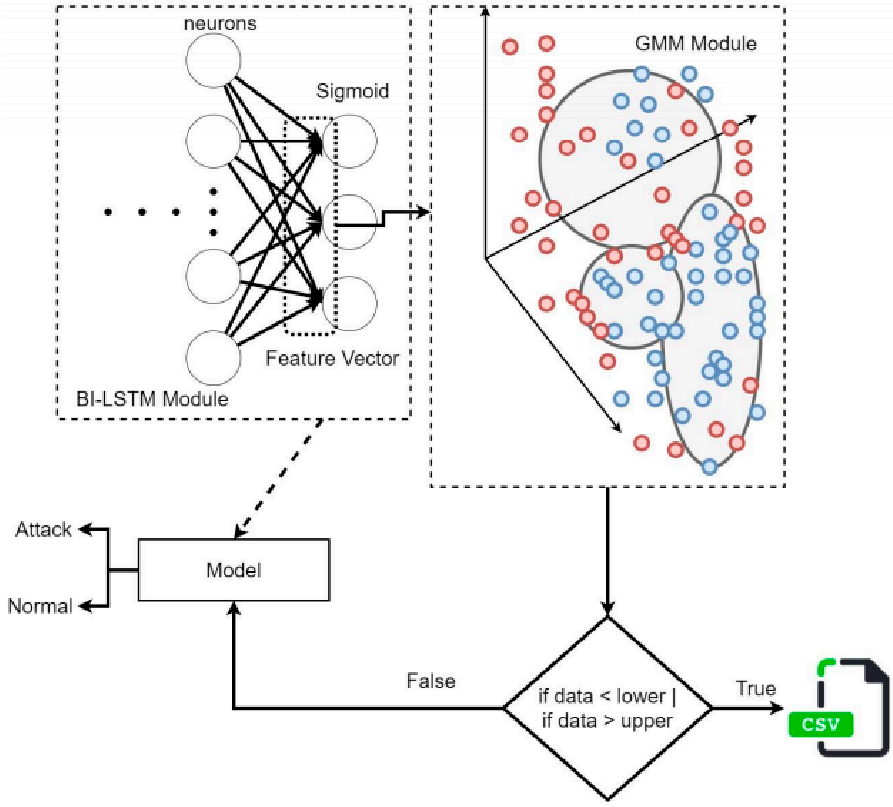


Figure 5. The control flow of the GMM module.

3.3. *Incremental Learning*

An entity can keep improving its ability if it can keep learning. Given the existing training set and newly obtained training data, retraining the system from scratch is a possible yet highly time-consuming process. On the contrary, incremental learning is a more plausible and more efficient alternative. Incremental learning is the procedure that keeps refining an existing model with newly obtained data. This capability is essential for DDoS attack detection since the attacking technology and resulting traffic keeps evolving.

In the proposed framework, traffic rejected by the GMM module will be logged and labeled by traffic engineers. The labeled traffic will be fed to the BI-LSTM and the GMM, serving as the training data in the incremental learning phase. The control flow for the incremental learning procedure is given in Figure 6.

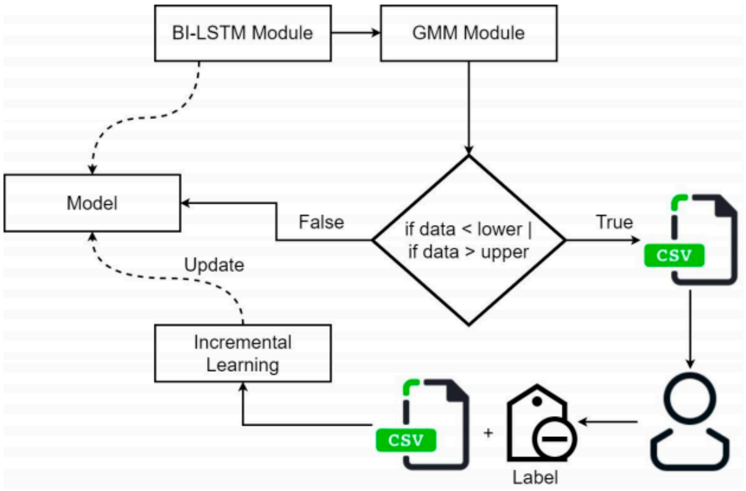


Figure 6. The control flow of the incremental learning procedure.

4. Experiments

A series of experiments were conducted to validate the feasibility and effectiveness of the proposed framework. The experiments were implemented using Python language on TensorFlow 2.0 and Keras platforms and executed on Intel i5-9500 CPU with 32 GB RAM.

4.1. Data Set

As shown in Table 1, two well-recognized datasets, CIC-IDS2017 [23] and CIC-DDoS2019 [24], are adopted in our experiments. The datasets were collected by the Canadian Institute for Cybersecurity with Wireshark in emulated environments. They are generated using two types of usage profiles and multistage attacks, such as Heartbleed, and a variety of DoS and DDoS attacks. Collected traffic is then pre-processed with the CICFlowMeter [25]. It has 80 network traffic features, aiming to generate the varied DoS and DDoS traffic data. The resulting data set contains records of traffic features in CSV format. To fit in the numeric nature of the proposed framework, the non-numeric fields are transformed using One-Hot encoding. All fields are then normalized for the rescaling of their dynamic ranges.

Table 1. CIC-IDS2017 and CIC-DDoS2019.

Data Set	Traffic Type	# of Instances	Ratio	Total # of Instances
CIC-IDS2017/Wednesday	BENIGN	440,031	0.63	692,703
	DoS GoldenEye	10,293	0.014	
	DoS Hulk	231,073	0.333	
	DoS Slowhttptest	5499	0.008	
	DoS Slowloris	5796	0.008	
	Heartbleed	11	1.5×10^{-5}	
CIC-IDS2017/Friday	BENIGN	97,718	0.432	225,745
	DDoS	128,027	0.567	
CIC-DDoS2019/NTP	BENIGN	14,365	0.0118	1,217,007
	DDoS/NTP	1,202,642	0.9881	
CIC-DDoS2019/LDAP	BENIGN	1612	0.0007	2,181,542
	DDoS/LDAP	2,179,930	0.9992	
CIC-DDoS2019/SSDP	BENIGN	763	0.0002	2,611,374
	DDoS/SSDP	2,610,611	0.9997	
CIC-DDoS2019/ Syn	BENIGN	392	0.0002	1,582,681
	Syn	1,582,289	0.9997	
CIC-DDoS2019/ NetBIOS	BENIGN	1707	0.0004	4,094,986
	DDoS/ NetBIOS	4,093,279	0.9995	

Performance indices include the confusion matrix, as shown in Table 2, and the Accuracy, Precision, and Recall, as defined in (5). Precision attempts to answer the question as to what proportion of positive identifications are actually correct. Recall concerns what proportion of actual positives are identified correctly. Precision measures the percentage of identified instances that are correctly classified.

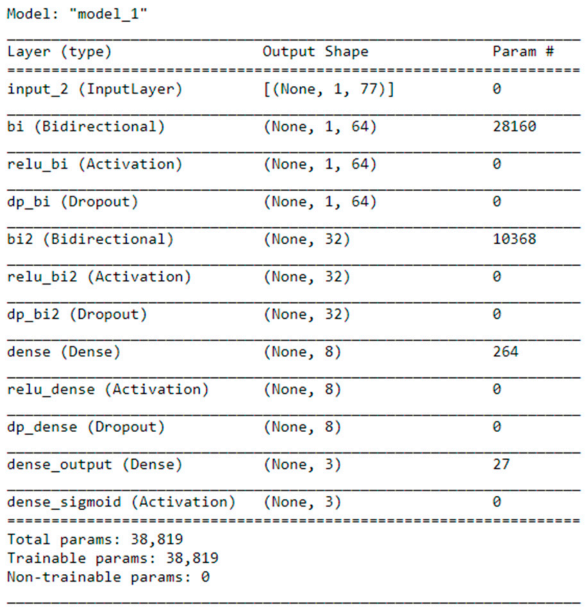
$$\begin{cases} \text{Accuracy} \triangleq \frac{TP+TN}{TP+FP+FN+TN} \\ \text{Precision} \triangleq \frac{TP}{TP+FP} \\ \text{Recall} \triangleq \frac{TP}{TP+FN} \\ \text{F1 Score} \triangleq \frac{2*TP}{2*TP+FP+FN} \end{cases} \tag{5}$$

Table 2. Confusion matrix.

Actual / Predicted	Attack	Normal
Attack	TP (True Positive)	FP (False Positive)
Normal	FN (False Negative)	TN (True Negative)

4.2. BI-LSTM Module

With certain efforts of investigation, we arrived at a BI-LSTM architecture with configuration in Figure 7 and parameter settings in Table 3. The kernel, bias, and activation are constrained by L2 regularization. We adopted a 10-fold cross-validation procedure in the training and testing. The dropout mechanism was invoked to avoid the problem of over-fitting.



```
Model: "model_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_2 (InputLayer)         [(None, 1, 77)]           0
_________________________________________________________________
bi (Bidirectional)           (None, 1, 64)             28160
_________________________________________________________________
relu_bi (Activation)         (None, 1, 64)             0
_________________________________________________________________
dp_bi (Dropout)              (None, 1, 64)             0
_________________________________________________________________
bi2 (Bidirectional)          (None, 32)                10368
_________________________________________________________________
relu_bi2 (Activation)        (None, 32)                0
_________________________________________________________________
dp_bi2 (Dropout)             (None, 32)                0
_________________________________________________________________
dense (Dense)                (None, 8)                 264
_________________________________________________________________
relu_dense (Activation)      (None, 8)                 0
_________________________________________________________________
dp_dense (Dropout)           (None, 8)                 0
_________________________________________________________________
dense_output (Dense)         (None, 3)                 27
_________________________________________________________________
dense_sigmoid (Activation)   (None, 3)                 0
=================================================================
Total params: 38,819
Trainable params: 38,819
Non-trainable params: 0
_________________________________________________________________
```

Figure 7. Configuration of the BI-LSTM.

Table 3. Parameter Settings of the BI-LSTM.

Parameter	Setting
Epoch/Batch Size	500/1024
Clipnorm	0.9
Learning rate	0.00859
Momentum	0.89
Decay	1.0×10^{-3}
Bidirectional Layer	2

We first examined the effectiveness of the BI-LSTM in serving as a DDoS detection module. “CIC-IDS2017/Wednesday” was used as the training data set to train the BI-LSTM. After training, the same data set, “CIC-IDS2017/Wednesday” was used to test the BI-LSTM. As indicated in the first row of Table 4, the BI-LSTM was well-trained, well-behaved, as intended. The BI-LSTM is highly capable of discrimination between malicious traffics and legitimate ones from the training data set.

Table 4. Performance of the BI-LSTM trained with “CIC-IDS2017/Wednesday”.

Test Data Set	Recall	Precision	Accuracy	AUC	F1
CIC-IDS2017/Wednesday	0.998	0.972	0.989	0.986	0.985
CIC-IDS2017/Friday	0.412	0.984	0.662	0.703	0.581

We now turn our attention to the impact of the OSR problem. That is, we investigated how a well-trained BI-LSTM reacts to novel instances never learned beforehand. “CIC-IDS2017/Friday” was used to test the BI-LSTM trained with “CIC-IDS2017/Wednesday”. As shown in the second row of Table 4, the performance, recall and accuracy dropped dramatically to approximately half, which is essentially a random guess. The reason behind this phenomenon is that the BI-LSTM has no built-in mechanism for knowing what it does not know. For a novel instance not drawn from the distribution model of the training set, the BI-LSTM compulsorily determines the category based on a mismatched model, and in general, this results in incorrect classification.

Notably, the precision does not significantly degrade in Table 4. Because of that, the legitimate traffic in “CIC-IDS2017/Friday” and “CIC-IDS2017/Wednesday” share a similar distribution pattern in the feature space.

4.3. GMM Module

GMM is our choice of solution to the OSR problem. With GMM, K Gaussian probability density functions are weighted to model the distribution of a given training data set. The number of Gaussian distributions, K, is a critical system parameter. There is an inevitable tradeoff between the model accuracy and the computational cost for the K value. Based on information entropy, the AIC (Akaike Information Criterion), as defined in (6), is commonly used for the evaluation of the fitting accuracy. The lower the AIC is, the better the accuracy is.

$$AIC \triangleq 2K - 2\ln(L) \tag{6}$$

As we constructed the GMM model with “CIC-IDS2017/Wednesday”, the change in AIC for different K values is presented in Figure 8. It is apparent that a larger K value leads to better accuracy. However, on the other hand, a larger K value implies a higher computational cost. Figure 8 suggests a compromised K value of 5 and, consequently, its associated $\mu = 12.23$ and $\sigma = 2.20$ in subsequent experiments.

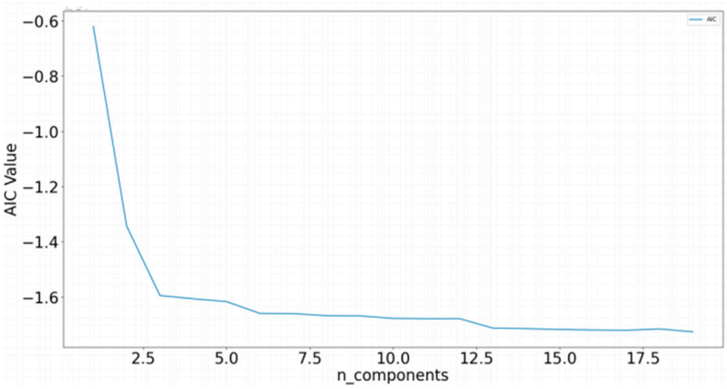


Figure 8. Change in AIC for different K values.

4.4. BI-LSTM-GMM with Incremental Learning

With GMM, traffics never learned can be picked up and logged. In the proposed framework, an instance three times of deviation away from the mean, i.e., $x \leq \mu - 3\rho$ or $x \geq \mu + 3\rho$, is considered a novel instance. It will be logged and labeled by traffic engineers. The intervention of data engineers is required since a novel instance can be malicious traffic, or equally likely, a new type of legitimate traffic. Collected, labeled new instances are fed back to the BI-LSTM and the GMM in the incremental learning phase. Table 5 reports the results after the completion of the loop. Compared with Table 4, the detection rate of unknown traffic, the "CIC-IDS2017/Friday" is greatly improved, both in its recall and accuracy.

Table 5. Performance of the BI-LSTM-GMM with incremental learning.

Test Data Set	Recall	Precision	Accuracy	AUC	F1
CIC-IDS2017/Wednesday	0.953	0.895	0.942	0.930	0.923
CIC-IDS2017/Friday	0.998	0.979	0.982	0.966	0.988

Notice that the detection of the old attack, CIC-IDS2017/Friday, slightly degrades. This is a side effect of incremental learning. An existing model will be biased by the incremental training data set and move in the feature space.

We also let data sets in "CIC-DDoS2019" play the role of unknown attacks to examine the general behavior of the proposed framework. The results are summarized in Table 6. Unknown traffic is the primary source of mis-classification. As shown in Table 5, the pure BI-LSTM is unable to address unknown attacks correctly. The recall indicator declines dramatically due to unknown traffic. There is a consistent tendency that the BI-LSTM alone failed to handle unknown traffics adequately. However, the GMM is capable of filtering new instances that were not learn beforehand. As indicated in Table 6, the integration of BI-LSTM and GMM can be an effective solution to the Open Set Recognition (OSR) problem in detecting unknown attacks. With the help of traffic engineers, labeled new instances are fed back to the BI-LSTM and the GMM for incremental learning. The updated model can then deal with both the old traffic and the new traffic correctly and gracefully.

Table 6. Performance of the BI-LSTM and BI-LSTM-GMM with "CIC-IDS2017/Wednesday" as the old traffic and "CIC-DDoS2019" as the new traffic.

Model	Test Data Set	Recall	Precision	Accuracy	AUC	F1
BI-LSTM	CIC-DDoS2019/NetBIOS	0.898	0.999	0.898	0.853	0.946
BI-LSTM-GMM	CIC-IDS2017/Wednesday	0.995	0.736	0.868	0.836	0.846
BI-LSTM-GMM	CIC-DDoS2019/NetBIOS	0.982	0.999	0.980	0.967	0.990
BI-LSTM	CIC-DDoS2019/NTP	0.362	0.995	0.368	0.606	0.531
BI-LSTM-GMM	CIC-IDS2017/Wednesday	0.985	0.750	0.875	0.850	0.852
BI-LSTM-GMM	CIC-DDoS2019/NTP	0.932	0.987	0.923	0.927	0.959
BI-LSTM	CIC-DDoS2019/LDAP	0.392	0.999	0.392	0.568	0.563
BI-LSTM-GMM	CIC-IDS2017/Wednesday	0.999	0.872	0.946	0.909	0.931
BI-LSTM-GMM	CIC-DDoS2019/LDAP	0.956	0.996	0.953	0.948	0.976

Referring to Table 6, the performance degradation of CIC-DDoS2019/NetBIOS is not so profound. It could be that its traffic pattern resembles that of CIC-IDS2017/Wednes-day. The performance of CIC-DDoS2019/NTP and CIC-DDoS2019/LDAP is much more evident. However, with the help of the proposed BI-LSTM-GMM framework together with the incremental learning scheme, all performance indices return to satisfactory levels.

The proposed framework has demonstrated optimistic results in the detection of unknown DDoS attacks. However, several issues demand further investigation for its universal applicability. The first issue relates to the test datasets. Validation on more datasets is required before a solid conclusion can be reached. The second issue is that, for the current design, the BI-LSTM configuration and the number of kernels in the Gaussian Mixture model are subject to a certain amount of trial-and-error. Automatic setting of these essential system parameters will be a fruitful research direction. The third problem is that the intervention of traffic engineers is still demanded. A possible solution to this issue is a global center for the collection of ever evolving attack traffic.

5. Conclusions

This study is a proof of concept for the detection of DDoS attacks with Deep Learning (DL) and a Gaussian Mixture Model (GMM). As a solution to the Open Set Recognition (OSR) problem in DDoS detection, the proposed framework consists of Bi-Directional Long Short-Term Memory (BI-LSTM), GMM, and incremental learning. The Bi-LSTM has demonstrated itself as a practical approach for the discrimination of malicious and legitimate traffic sampled from the distribution of the training data set. The GMM has shown to be an effective measure for the differentiation of novel instances and trained samples. Unknown traffic can be captured by the GMM and labeled by data engineers, and then fed back to the BI-LSTM and the GMM for incremental learning. Both the new traffic and the old traffic can be handled by the updated model correctly and gracefully. The feasibility and effectiveness of the proposed framework has been validated by a series of experiments on data sets CIC-IDS2017 and CIC-DDoS2019.

BI-LSTM is fully capable of performing what it has been trained to do, such as detecting known DDoS attacks. However, when confronted with novel attacks, the system performance degrades severely. The recall drops from 99.8% to 41.2% for Dataset CIC-IDS2017/Wednesday and Friday. Unknown traffic can be captured by GMM and then labeled by traffic engineers. After that, incremental learning regains detection rates up to 95.3% and 99.8% for Dataset CIC-IDS2017/Wednesday and Friday.

To explore the full potential of the proposed BI-LSTM-GMM framework there are some research directions that deserve further investigation: validation on more datasets, auto-configuration of BI-LSTM and GMM, and the elimination of intervention traffic engineers.

Author Contributions: Conceptualization, C.-S.S.; methodology, C.-H.C.; software, C.-H.C.; validation, W.-W.L. and T.-T.N.; writing—original draft preparation, C.-S.S.; writing—review and editing, W.-W.L. and T.-T.N.; supervision, M.-F.H.; funding acquisition, D.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was partly funded by the Minister of Science and Technology, Taiwan, under the grant number 109-2637-E-992-006, MOST 109-2622-E-992-033, and 109-2221-E-992-073-MY3, and the Ministry of Education, Taiwan, under the grant code of University-Industry Innovation R&D (RSC).

Acknowledgments: The authors would also like to thank the Editor and Reviewers for their constructive comments which led to improvements to the presentation of the manuscript.

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

References

1. Mahjabin, T.; Xiao, Y.; Sun, G.; Jiang, W. A survey of distributed denial-of-service attack, prevention, and mitigation techniques. *Int. J. Distrib. Sens. Netw.* **2017**, *13*. [CrossRef]
2. Genie-Networks. DDoS Attack Statistics and Trends Report for 2020. 2021. Available online: https://www.genie-networks.com/gnnews/ddos-attack-statistics-and-trends-report-for-h1-2020/ (accessed on 6 May 2021).
3. Jonker, M.; Sperotto, A.; Pras, A. DDoS Mitigation: A measurement-based approach. In *NOMS 2020–2020 IEEE/IFIP Network Operations and Management Symposium*; IEEE: Piscataway Township, NJ, USA, 2020; pp. 1–6.
4. Priya, S.S.; Sivaram, M.; Yuvaraj, D.; Jayanthiladevi, A. Machine learning based DDoS detection. In Proceedings of the 2020 International Conference on Emerging Smart Computing and Informatics, Pune, India, 12–14 March 2020; IEEE: Piscataway Township, NJ, USA, 2020; pp. 234–237.
5. Pouyanfar, S.; Sadiq, S.; Yan, Y.; Tian, H.; Tao, Y.; Reyes, M.P.; Shyu, M.; Chen, S.; Iyengar, S.S. A survey on deep learning: Algorithms, techniques, and applications. *ACM Comput. Surv.* **2018**, *51*, 1–36. [CrossRef]

6. Yulita, I.N.; Fanany, M.I.; Arymuthy, A.M. Bi-directional Long Short-Term Memory using Quantized data of Deep Belief Networks for Sleep Stage Classification. *Procedia Comput. Sci.* **2017**, *116*, 530–538. [CrossRef]
7. Geng, C.; Huang, S.J.; Chen, S. Recent advances in open set recognition: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *14*, 1–19. [CrossRef] [PubMed]
8. Cao, A.; Luo, Y.; Klabjan, D. Open-set recognition with Gaussian mixture variational autoencoders. *arXiv* **2020**. Available online: https://arxiv.org/abs/2006.02003 (accessed on 6 May 2021).
9. Cheng, J.; Yin, J.; Liu, Y.; Cai, Z.; Wu, C. DDoS attack detection using IP address feature interaction. In Proceedings of the IEEE International Conference on Intelligent Networking and Collaborative Systems, Thessalonika, Greece, 24–26 November 2010; IEEE: Piscataway Township, NJ, USA, 2009; pp. 113–118.
10. Vu, N.H. DDoS attack detection using K-Nearest Neighbor classifier method. In Proceedings of the International Conference on Telehealth/Assistive Technologies, Baltimore, Maryland, USA, 16–18 April 2008; IEEE: Piscataway Township, NJ, USA, 2008; pp. 248–253.
11. Fadlil, A.; Riadi, I.; Aji, S. Review of detection DDoS attack detection using Naïve Bayes classifier for network forensics. *Bull. Electr. Eng. Inform.* **2017**, *6*, 140–148. [CrossRef]
12. Wang, C.; Zheng, J.; Li, X. Research on DDoS attacks detection based on RDF-SVM. In Proceedings of the 10th International Conference on Intelligent Computation Technology and Automation, Changsha, China, 9–12 October 2017.
13. Dincalp, U. Anomaly based distributed denial of service attack detection and prevention with machine learning. In Proceedings of the 2nd International Symposium on Multidisciplinary Studies and Innovative Technologies, Ankara, Turkey, 19–21 October 2018.
14. Ahanger, T.A. An effective approach of detecting DDoS using artificial neural networks. In Proceedings of the 2017 International Conference on Wireless Communications, Signal Processing and Networking, Chennai, India, 22–24 March 2017; IEEE: Piscataway Township, NJ, USA, 2017; pp. 707–711.
15. Li, Y.; Lu, Y. LSTM-BA: DDoS detection approach combining LSTM and Bayes. In Proceedings of the 7th International Conference on Advanced Cloud and Big Data, Suzhou, China, 21–22 September 2019; IEEE: Piscataway Township, NJ, USA, 2009; pp. 180–185.
16. Yang, K.; Zhang, J.; Xu, Y.; Chao, J. DDoS attack detection with AutoEncoder. In *IEEE/IFIP Operations and Management Symposium*; IEEE: Piscataway Township, NJ, USA, 2020; pp. 1–9.
17. Doriguzzi-Corin, R.; Millar, S.; Scott-Hayward, S.; Martinez-del-Rincon, J.; Siracusa, D. LUCID: A practical, lightweight deep learning solution for DDoS attack detection. *IEEE Trans. Netw. Serv. Manag.* **2020**, *17*, 876–889. [CrossRef]
18. Yong, B.; Wei, W.; Li, K.-C.; Shen, J.; Zhou, Q.; Wozniak, M.; Połap, D.; Damaševičius, R. Ensemble machine learning approaches for webshell detection in Internet of things environments. *Trans. Emerg. Telecommun. Technol.* **2020**, *30*. [CrossRef]
19. Hemalatha, J.; Roseline, S.A.; Geetha, S.; Kadry, S.; Damaševi˘cius, R. An efficient DenseNet-based deep learning model for malware detection. *Entropy* **2021**, *23*, 344. [CrossRef] [PubMed]
20. Bendale, A.; Boult, T.E. Towards open set deep networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; IEEE: Piscataway Township, NJ, USA, 2016; pp. 1563–1572.
21. Sabeel, U.; Heydari, S.S.; Mohanka, H.; Bendhaou, Y.; Elgazzar, K.; El-Khatib, K. Evaluation of deep learning in detecting unknown network attacks. In Proceedings of the 2019 International Conference on Smart Applications, Communications and Networking, Sharm El Sheik, Egypt, 17–19 December 2019; pp. 1–6.
22. Rudd, E.M.; Jain, L.P.; Scheirer, W.J.; Boult, T.E. The extreme value machine. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 762–768. [CrossRef] [PubMed]
23. University of New Brunswick. Intrusion Detection Evaluation Dataset (CIC-IDS2017). 2017. Available online: https://www.unb.ca/cic/datasets/ids-2017.html (accessed on 6 May 2021).
24. University of New Brunswick. DDoS Evaluation Dataset (CIC-DDoS2019). 2019. Available online: https://www.unb.ca/cic/datasets/ddos-2019.html (accessed on 6 May 2021).
25. Canadian Institute for Cybersecurity. CICFlowMeter (4.0) [Source Code]. 2016. Available online: https://github.com/CanadianInstituteForCybersecurity/CICFlowMeter (accessed on 6 May 2021).