Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2023.037386
Article

DDoS Attack Detection in Cloud Computing Based on Ensemble Feature Selection and Deep Learning

Yousef Sanjalawe[1,2,*] and Turke Althobaiti[3,4]

[1]Cybersecurity Department, Faculty of Information Technology, American University of Madaba (AUM), Amman, 11821, Jordan
[2]Computer Science Department, PY Collage, Northern Border University (NBU), 9280 NBU, Ar'ar, Saudi Arabia
[3]Computer Science Department, Faculty of Science, Northern Border University (NBU), 9280 NBU, Ar'ar, Saudi Arabia
[4]Remote Sensing Northern Border University (NBU), 9280 NBU, Ar'ar, Saudi Arabia
*Corresponding Author: Yousef Sanjalawe. Email: y.sanjalawe@aum.edu.jo
Received: 02 December 2022; Accepted: 15 January 2023

Abstract: Intrusion Detection System (IDS) in the cloud Computing (CC) environment has received paramount interest over the last few years. Among the latest approaches, Deep Learning (DL)-based IDS methods allow the discovery of attacks with the highest performance. In the CC environment, Distributed Denial of Service (DDoS) attacks are widespread. The cloud services will be rendered unavailable to legitimate end-users as a consequence of the overwhelming network traffic, resulting in financial losses. Although various researchers have proposed many detection techniques, there are possible obstacles in terms of detection performance due to the use of insignificant traffic features. Therefore, in this paper, a hybrid deep learning mode based on hybridizing Convolutional Neural Network (CNN) with Long-Short-Term Memory (LSTM) is used due to its robustness and efficiency in detecting normal and attack traffic. Besides, the ensemble feature selection, mutualization aggregation between Particle Swarm Optimizer (PSO), Grey Wolf Optimizer (PSO), Krill Hird (KH), and Whale Optimization Algorithm (WOA), is used to select the most important features that would influence the detection performance in detecting DDoS attack in CC. A benchmark dataset proposed by the Canadian Institute of Cybersecurity (CIC), called CICIDS 2017 is used to evaluate the proposed IDS. The results revealed that the proposed IDS outperforms the state-of-the-art IDSs, as it achieved 97.9%, 98.3%, 97.9%, 98.1%, respectively. As a result, the proposed IDS achieves the requirements of getting high security, automatic, efficient, and self-decision detection of DDoS attacks.

Keywords: CIC IDS 2017; cloud computing; distributed denial of service; ensemble feature selection; intrusion detection system

1 Introduction

Cloud Computing is a growing area that facilitates the Information Technology (IT) infrastructure, network facilities, and applications. CC may be officially known as "a model for enabling convenient, on-demand network access to a shared pool of adjustable computing materials, including the networks, storage, servers, services, and applications that could be supplied and produced at a fast rate with the minimum management attempt or service provider" [1]. Clouds could be separated into three notable categories including private, public, and hybrid clouds. Specifically, the CC framework could be separated into two categories, namely the front and back ends [2]. These ends are related to each other through a network, specifically the web. The front end is viewed by the customer (client), while the back end denotes the computing resources that include storage and services among others. In most cases, the back end characterizes the CC administrations that include information storage, various Personal Computers (PCs), and servers, while the front end is the user's PC and the application necessary to reach the cloud. A central server takes control of the clients' and users' requests and supervises the network performance. It gains specific directions, such as the conventions and implementations of special programming known as middleware [2], which also allows communication among organized PCs.

Similar to other conventional networks, CC is prone to several types of attacks, including the DDoS attack. This type of attack becomes the aim of multiple compromised computers that are known as bots or zombies that target a single system. The attack also aims towards the objective system or network resource weakening and the incidental disruption of service, which makes the service unavailable. The DDoS attack is classified into seven notable categories, namely amplification attack, flood attack, coremelt attack, land attack, authentication server attack, Common Gateway Interface (CGI) request attack, and Transmission Control Protocol-Synchronize (TCP SYN) attack, as illustrated in Fig. 1.

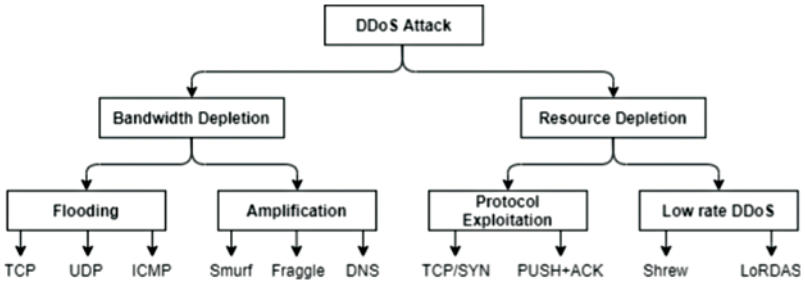

Figure 1: Classes of DDoS attack

The management of DDoS attacks in CC is challenging in the network security field, specifically when the SDN and CC perspectives are present. As a result of the significant evaluation of information kept over clouds, CC becomes prone to DDoS attacks. The availability of CC and the high volume of information leads to the rapid development of the measure of information [1]. However, the recent safety initiatives are unable to fulfill the security standards of distributed computing. Gartner made a prediction that the application layer DDoS attacks would experience thrice the increase annually in distributed computing. Based on the prediction, DDoS attacks would denote 25% of most application-layer attacks [3]. Moreover, customary defense elements are faced with various challenges in the identification of DDoS attacks in CC. A progressing Cloud Security Alliance (CSA) research indicated that DDoS attacks are the common attacks on cloud security [4]. To illustrate, distinguished scientific works presented in [5–9] focused on creating a more distinguished phase to combat DDoS attacks.

Despite the extended studies performed to identify and predict DDoS attacks, the expansion of security breaks has been taking place at an alarming degree in the undertakings and distributed computing environments. Fig. 2 depicts DDoS attacks in CC environment.

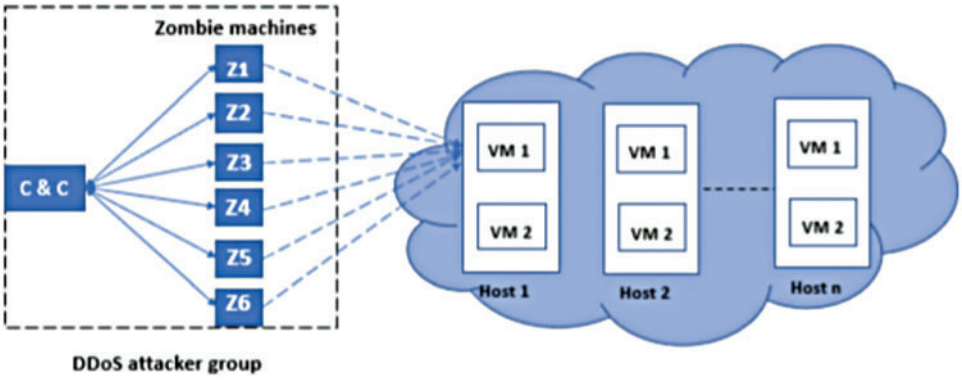

Figure 2: DDoS in cloud

Motivation and Gaps

In DDoS attacks [10], diverse attackers adjust the identification of fake data in a high range, which creates a challenge under the victim's ownership or the near network associates. This category of attack is initiated through the exploitation of the system susceptibility with a large number of unsuccessful traffic in the network to be present in the appointed attributes, such as memory, network processor time, and bandwidth [11,12], which lead to critical disruption to the victim. This attack could be multi-sourced, in which various hosts organize attack bundles against the victim or launch a single-source attack that begins with a single host. Currently, attack toolkits are created to be readily available online [13,14] to ensure that any internet clients are able to use these toolkits to launch attacks with minimum issues. Therefore, a higher quantity of examinations is prominent in recent times through the development of a system for preventing DDoS attacks [15]. At certain times, this attack occurs at a high speed and endures several minutes that do not reach an hour. Putting these causes into consideration, the security team fails to locate their system in the attack due to the incapability of the traditional techniques to predict this type of attack [16]. This condition leads to an issue in corporate networks and works from the memory, processing capacity of a destination system, and bandwidth [17,18]. To overcome the drawback of the DDoS attack, the attempt of IDS is important. The IDS is a network security approach for protecting the computer network system from attacks. To ensure that the network services are available and reliable, the research community and industry sectors make a significant effort for the development of IDS. Despite the implementation of popular IDS, the DDoS category of attack would be avoided in most cases. As a result of the increasing amount of information constantly distributed from the individual network to a supplementary network, IDS successfully identifies the disruption in large datasets [19].

The IDS operates under two detection models including the signature- or anomaly-based detection approaches. The anomaly detection method uses the signature, mark, or rule match to identify the intrusive behaviour. Frameworks that depend on misuse detection systems are able to achieve significant location degrees in the recognized attacks [20]. However, they are ineffective in the identification of any unknown harmful actions or still variations of the present attacks. The production of distinguished signatures for covered attacks is a work-concentrated task that majorly involves network security mastery [21]. Notably, an anomaly-based detection system uses an alternative detection

method that separates and labels any network operations that introduce significant differences from the individual legitimate traffic profiles as untrustworthy and attacked information. Provided that these outlines follow the information on common network practices, the anomaly-based detection system is able to identify obscure attacks. Overall, it is accepted in the research field [22]. A wide variety of anomaly-based detection frameworks have been implemented since the most current decade. The recent frameworks are faced with adverse impacts of a common problem in the achievement of strong accuracy in identifying the attack and ordinary traffic [23]. The approaches employed in these anomaly-based detection frameworks could be separated into two categories, namely statistical investigation, and Artificial Intelligence (AI) [23]. Machine Learning (ML)-based methods assist in the categorization of examined information by the identified attributes identified from the training data. Statistical analysis-based methods are conducted for the investigation into the features of network traffic packets and the development of a rationale threshold to differentiate the attacks from the legitimate traffic. However, some drawbacks are present. Therefore, the current study in DDoS attack identification mainly highlights anomaly-based approaches, while diverse detection procedures have been forecasted. The approaches include statistical detection approaches, clustering, Neural Network (NN) [24], Support Vector Machine (SVM) [25], and nearest neighbor [26]. The present study demonstrates that many investigations have been conducted to propose common means to solve the obstacle, particularly the emergence of DDoS attacks. Despite the launch of several DDoS attack detection systems, a range of issues should be addressed. Following are several current obstacles found in the current research on DDoS attack detection:

- Most of the published existing IDS methods prioritized the detection of DDoS attacks with low-false alarms. However, these approaches are not effective in positive detection rates in most cases.
- Although the awareness of the features of DDoS attacks is significant, the identification of the effectual attributes to detect an attack.
- Some of the existing approaches are often faced with a high false positive rate due to unsuccessful selection of the features.
- Past approaches were lacking in the acquirement of effective precision. For this reason, a properly arranged IDS for DDoS attacks remains an interest in the research field.

The identification of these issues encouraged the study in this field to be conducted. Therefore, an efficacious IDS is proposed to detect DDoS attacks by categorizing the attack and normal packet effectively to improve the DDoS attack detection rate. The main contributions of this paper are:

- Enriching the existing literature by proposing an efficient IDS to detect cyber-attacks in the cloud environment.
- Proposing an ensemble feature selection to select the most significant features subset that would enhance the performance of DDoS attacks detection.
- Hybridizing the CNN and LSTM to improve the classification process by taking advantage of both models in detecting DDoS attacks efficiently.

The organization of the manuscript begins with Section 2, which presents the related work. This is followed by Section 3, which illustrates the suggested methodology and Section 4, which demonstrates the experimental findings. The final section presents the conclusion.

2 Related Works

Many researchers conducted an analysis of the DoS attack detection systems. A small number of the studies are highlighted in this section. An analysis of the most current patterns of attacks in the cloud environment is conducted in [27]. In this procedure, Software Defined Networking (SDN) offered a chance to control DDoS attacks in the distributed computing states. The DDoS on SDN and the approaches against DDoS in SDN have been highlighted. The methods demonstrate numerous challenges to be identified for relieving the DDoS involved in SDN with distributed computing. Based on [28], an introduction to DDoS attack migration was made, which led to a detailed survey of the prevention, identification, and alleviation approach for these attacks. The authors also illustrated an extensive organization of scientific classification to identify the order of attack. Moreover, [29] demonstrated a supporting system to improve the DDoS services, which reduces the attack mitigation time and general downtime. The present approach leads to service downtime as the main reduction. In line with this, they [30] demonstrated Log-Based Intrusion Detection for Cloud Web Applications Using Machine Learning. In this case, two classifiers were involved, namely the neural networks and decision trees to detect intrusion. Specifically, the classifiers gain access logs, which require simple interpretation. A wide range of Evolutionary Algorithms (EAs) was employed to detect the attack, which included Random Forest Algorithm [31], Moth-flame Optimization Algorithm (MFOA) [32], Whale Optimization Algorithm [33]. However, a small number of these approaches have weaknesses. To illustrate, EA is trapped in some local optimums and unable to memorize any history. The RF has high complexity, with the training and decision-making for it requiring more time. Meanwhile, MFOA is involved in local optima.

Following the aforementioned weaknesses, a wide range of adjusted and hybrid algorithms have been suggested to detect the attack, which leads to higher general performance by shifting to the incorporations of the original versions. Despite the number of approaches suggested to detect malware, the hybridized methods show exceptional precision and identification rates, which allow new attack signatures to be detected, rule sets to go through dynamic updates, labelling time to be reduced, and false positives to be reduced. Several research works emphasized the hybridizations of EAs to detect attacks. To illustrate, a newly developed intrusion detection model is demonstrated in [34], through the combination of Modified Particle Swarm Optimization-Back Propagation (MPSOBP) and Laplacian Eigenmaps (LE). The Knowledge Discovery and Data (KDD) cup 99 dataset is employed for simulation. The introduction of LE is made to reduce the dimensionality, which is followed by the use of MPSO-BP for feature selection. As a result, the suggested work shows a comparatively better detection rate and higher speed of velocity.

In addition, Ho et al., in [35] proposed a CNN-based IDS to support the security of internet-based environments. It aims to detect network intrusions (attacks) by classifying all the network packets in the network as normal packets or malicious ones. The benchmark dataset CICIDS2017 has been utilized to evaluate and train the proposed IDS. It outperforms the nine other well-known state-of-the-art IDSs in most multi-class classification categories. In the same manner, Gao in [36] took advantage of CNN and Bidirectional LSTM (Bi-LSTM), by hybridizing both deep learning models to propose a hybrid-based IDS. CNN model is utilized to extract the parallel local features of network traffic, whilst BiLSTM is utilized to extract the features of long-distance-dependent network traffic. Then, Finally, the decision tree combined with CNN to surpass the design feature selection and directly utilize the DL model to learn the representational features of high-dimensional data. The proposed model was trained and tested using KDD CUP 99 dataset, and it achieved noticeable results in terms of accuracy, and false-positive rate.

According to [37], in the cloud environment, IDS is used to detect abnormal behavior in the connection and in the host. But it is hard to protect the cloud environment from DDoS attacks since they produce a huge volume of harmful information on the network. This type of attack forces the services provided by the cloud providers to become unavailable to the target users, which exhausts computational resources and leaves the cloud service provider risky to massive financial and reputational losses. Consequently, a machine learning-based IDS is proposed in this paper. This IDS utilized a filtering technique, called learning vector quantization, and a dimensionality-simplifying method called Principal Component Analysis. The selected attributes from each technique are used for categorization before being tested against a DoS attack. The proposed IDS was trained and tested using refined dataset propose by the Network Security Laboratory (NSL), called NSL-KDD dataset.

Following that, a new algorithm for anomaly identification is proposed in [38], specifically the hybridization of the Firefly Algorithm and K-Means and evaluated on the NSL-KDD dataset. A differentiation research work was conducted between the currently created algorithm with other clustering algorithms, which included K-Means + Canopy, K-Means + Cuckoo, K-Means + Bat, and K-Means. As a result, K-Means + Bat and K-Means + Firefly showed a higher performance by a large margin. Two novel algorithms were suggested in [39], which included (i) packet scrutinization algorithm that performed an examination on the packets from the users, and (ii) hybrid classification model known that denotes the incorporation of normalized K-means clustering algorithm with the recurring neural network. A one-time signature for cloud users was suggested by more authors to acquire the cloud data and prevent the user from attackers. A hybrid model was constructed in [40], which supports the unmanaged learning algorithm. It is also the incorporation with the Generative Local Metric Learning (GLML) and identifies the abnormal disruptions that include User to Root (U2R) and Remote to Local (R2L).

In [41], two deep generative models were used to automatically generate synthesized traffic malicious samples on the cloud environment. The first model, conditional denoising adversarial autoencoder, was utilized to generate specific types of malicious samples. The second model was a hybridized model among the first model with the K-nearest neighbor algorithm, which was used to generate malicious borderline samples that further improve the accuracy of a cloud IDS. The synthesized samples are merged with the original samples to form the augmented datasets. Three machine learning algorithms are trained on the augmented datasets and their effectiveness is analyzed. The experiments conducted on four popular IDS datasets show that the proposed techniques significantly improve the accuracy of the cloud IDSs compared with the baseline technique and the state-of-the-art approaches. Moreover, the proposed models also enhance the accuracy of machine learning algorithms in detecting some currently challenging attacks, including low-rate DDoS attacks and application layer DDoS attacks.

Notably, feature selection brings a substantial effect on the most effective removal of irrelevant and redundant features. It could also reduce the classifier performance, computational cost, and needed storage. Therefore, an intelligent search algorithm is required, with the use of search agents, evolutionary algorithms are able to seek the feature space to determine the ideal solution space. Although the recent literature work was insufficient in terms of the best feature subset selection, the importance of further studies has also been acknowledged. However, based on the current methods, it could be seen that the prediction of DDoS attacks is either inaccurate or it significantly consumes time, storage, and resources among others. In fact, DDoS attacks have increased in intensity throughout the years, which increases the possibility for the whole network or system to be destroyed. Therefore, a number of authors suggested a wide range of studies to solve these limitations. However, the studies were not adequate in offering an effective lightweight method in the CC environment. For this reason,

this paper suggests an effective method for DDoS attack detection by adapting ensemble feature selection and a hybrid CNN-LSTM model.

3 Proposed IDS

It can be seen from Fig. 3 that the proposed IDS approach comprises three main phases, including (i) the data preprocessing phase, (ii) ensemble feature selection, and (iii) the detection phase.

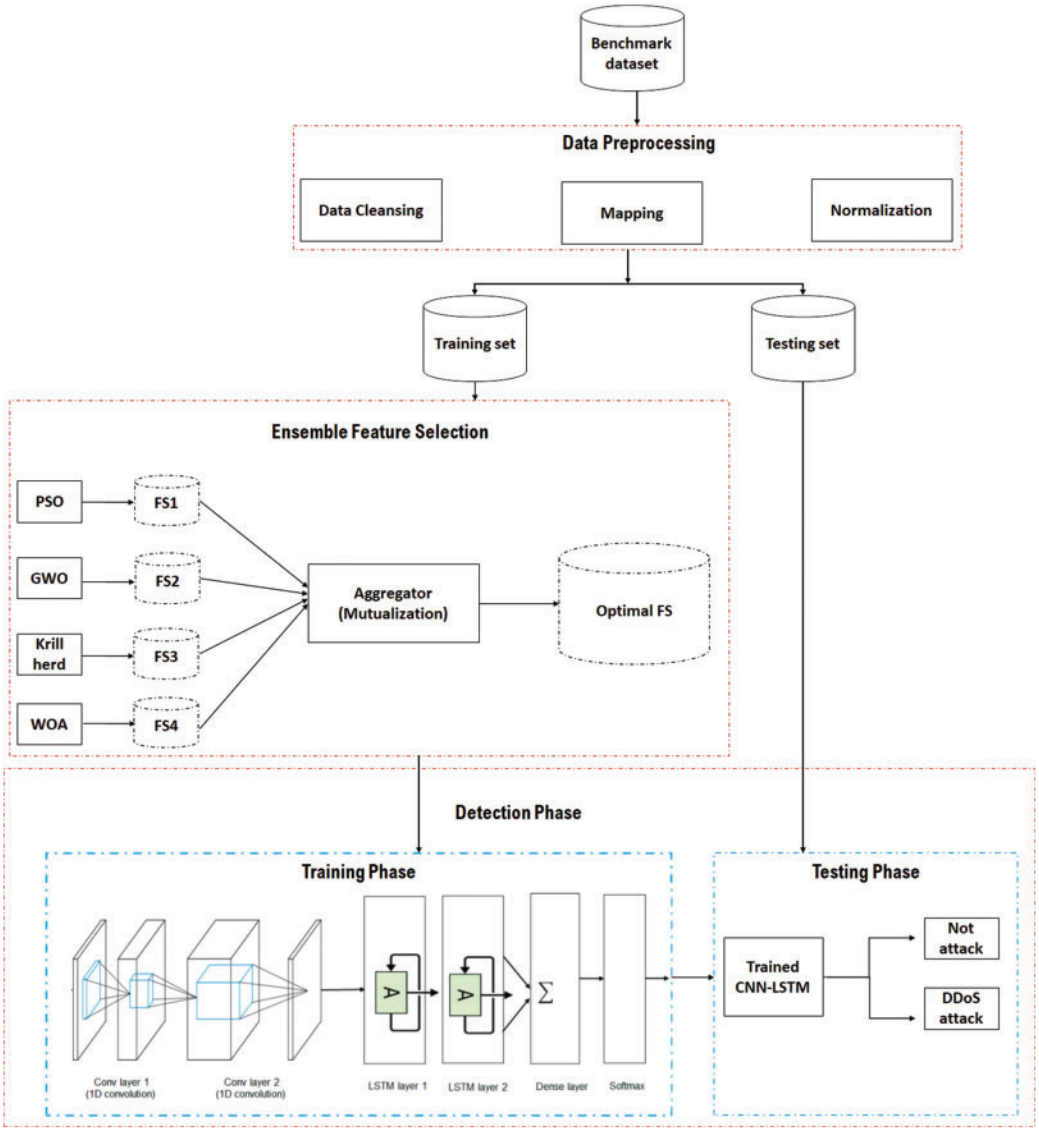

Figure 3: Proposed IDS approach

3.1 Data Preprocessing Phase

Data preprocessing is essential in IDS due to its capability of simplifying and improving the efficacy of data classification stages [42]. Furthermore, the data commonly originates from heterogeneous platforms and could produce noise and become excessive, lacking, and complicated. Therefore, adjustment on the unrefined data is meaningful to create important data for examination, categorization, and information reveal. The followings are the stages in pre-preprocessing in this research:

3.1.1 Data Cleansing

This article begins with an important process of eliminating or repairing false, redundant, or imperfect records. Following that, the missing values are filled in the provided datasets that are usually known as data scrubbing or cleaning. As a result, precise, efficacious, credible, and successful predictions could be achieved [43], while a small number of redundancies in the training and testing datasets are deleted. Nevertheless, redundant features are not present.

3.1.2 Mapping

As shown in Fig. 1, DDoS attacks comprise several categories, which include User Datagram Protocol (UDP), Transmission Control Protocol (TCP), The Internet Control Message Protocol (ICMP), Fraggle, Smurf, TCP/SYN, Domain Name Server (DNS), shrew, Low-Rate DoS Attack against Application Servers (LoRDAS), and push and acknowledgment (PUSH + ACK) attacks. All these attacks are delineated under the parent category (e.g., DDoS attack) in this substage. Fig. 4 presents the mapping process of the attack. Consequently, all abnormal traffic that suffers from any type of DDoS should belong to one attack category (e.g., DDoS).

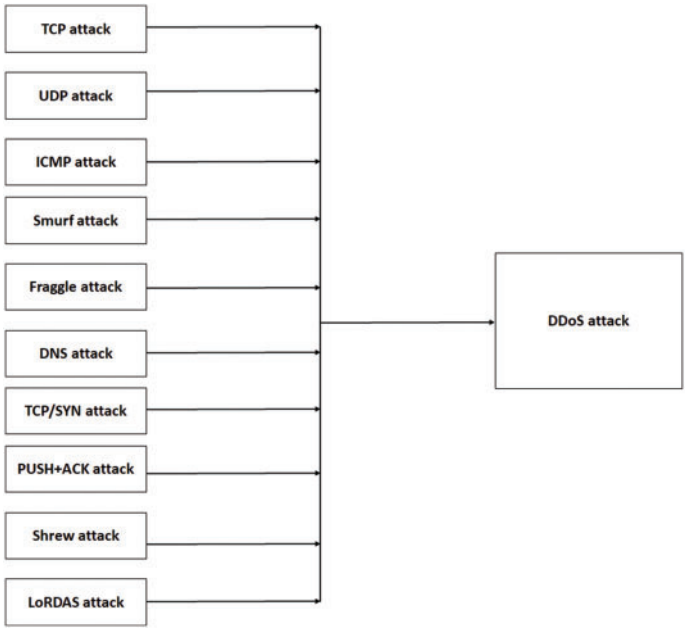

Figure 4: Attacks mapping

3.1.3 Normalization

A wide range of imbalanced scales of attributes could lead to lower performance of regression or classification models [44]. For this reason, the normalization of these differences in the dataset features is important to ensure that the overlooked and prominent values are present in the suitable range. To illustrate, the high values of "Duration", source and destination bytes ("Src Bytes", and "Dst Bytes") of the link of the CICIDS2017 dataset could outperform the lower values of "Num Failed Logins". Subsequently, this article systematically selects the minimum-maximum method [45] for the normalization of the dataset in the normalized range of [0, 1], which facilitates the comprehension of the data. Following is the standard equation of the minimum-maximum approach:

$$X_{norm} = (x - x_{min})/(x_{max} - x_{min}) \tag{1}$$

where, Xnormalized_value denotes the normalized outcome, with the Xmin_value and Xmax_value of the outcomes amounting to 0 and 1, respectively. The rest of the values would amount to the aforementioned ranges, allowing the attributes to show an identical base point and range. As a result, this method solves the bias, significantly shortens the duration for model testing and training, and allows a rapid rate of convergence, which improves the categorization and reliability performance [45].

3.2 Ensemble Feature Selection Phase

To solve the limitations caused by the use of a single feature selection algorithm, an ensemble feature selection approach employs mutual data for the selection of an optimal subset of features. Four primary feature selection algorithms are used, namely (i) Particle Swarm Intelligence (PSO) [46], (ii) Grey Wolf Optimizer (GWO) [47], (iii) Krill Herd (KH) [48], and (iv) Whale Optimization Algorithm (WOA) [49]. The prepossessed dataset is fed into these algorithms simultaneously, in which every algorithm creates the optimum feature subset. Therefore, this substage creates four feature subsets.

These subsets are fed into the aggregator to incorporate the chosen subset of features according to the feature-feature and feature-class shared information. The party in charge of combining acquires the first features on the rank from the entire chosen subsets. The similar features on the first rank obtain the normal (e.g., shared) feature as an optimum feature subset without the computation of feature mutual and feature-class information. However, the distinguishing features calculate the feature-class shared information for each feature and take into account the strongest feature-class shared information. This feature subset calculates the feature-feature mutual information with the chosen attributes as optimum. Given that the feature-feature shared information of the entire chosen features is lower than what is highlighted by the user-defined threshold, α, the feature would be chosen. The feature-feature shared information is employed for the measurement of the feature-relevance of an unselected attribute with the chosen attributes. Following the in-depth research, the threshold value amounts to ($\alpha = 0.75$). Based on the ensemble method, a 'combiner' holds a crucial function in the ensemble of diverse feature selection approaches. The party in charge of combining the suggested method of selecting ensemble feature highlights the reduction of the duplicate within the chosen subset of attributes by incorporating feature-class and feature-feature mutual information and preventing the bias in feature selection that is caused by a single feature selection algorithm.

3.3 Detection Phase

In this method, the hybrid CNN-LSTM model is employed to detect the DDoS attacks with the objective of classifying the network traffic data from phases into the respective categories: (i) normal traffic, or (ii) DDoS attack. In LSTM, the DL-based DDoS attack identification approach trains the

normal and harmful traffics to develop a model that highlights the baseline profile for these traffics. A schematic depiction of the CNN-LSTM network is presented in Fig. 3 (stage 3–detection stage). The end-to-end CNN comprises two elements, namely (i) a classifier and (ii) a feature extractor. Specifically, the feature extractor consists of two layers known as the pooling and convolution layers. The extracted output, which is also identified as the features map, is the input to the second element of the categorization. This condition allows CNN to successfully master the local attributes. Nevertheless, the drawback of it is that it overlooks the long-range interdependency of crucial attributes. For this reason, LSTM layers [50–52] are introduced after the CNN layers to substantially acquire the local and global features. This method allows the issues of exploding and vanishing gradient issues to be addressed effectively, which increases the capability to contribute to more significant reliance and efficient learning from variable extent orders [53,54].

In the Convolutional-LSTM (Conv-LSTM) network, the processing of the input is first performed by the CNN, while the output of CNN is distributed through the LSTM layers to create orders on every time step, which enables the modelling of long-term and short-term temporal attributes [54]. The sequence vector flows through a completely linked layer prior to being fed into the Softmax layer for the possibility allocation throughout the classes. In this phase, the test set is employed as an input to the model that has received training to determine whether the trained traffic is in a regular state or harmful. The attack traffic forecasted by the classic models is incorporated with the test set.

4 Results and Discussion

This section discusses the details of the benchmark dataset and evaluation metrics used to assess the performance of the proposed IDS, then, results and findings are presented in detail.

4.1 Benchmark Dataset and Evaluation Metrics

The CIC IDS 2017 dataset [55] is the data established by the Faculty of Computer Science, at the University of New Brunswick in 2017. Following a past study by Shiravi et al. [56], CICIDS2017 is a refined version of the ISCX 2012 dataset [57]. The CICIDS2017 dataset is developed from the real generalization of traffic. The study in [55] demonstrated the features of the IDS dataset and the method employed to develop the dataset. Faith, Arash, and Ali made a comparison between the dataset and other datasets. The assessment framework of the latest dataset produced in 2016 [58] comprises 11 requirements that are nearly fulfilled by the datasets, namely Labelled Dataset, Complete Traffic, Complete Network configuration, Available Protocols, Complete Interaction, Complete Capture, Heterogeneity, Meta Data, Feature Set, and Attack Diversity. These requirements are fulfilled by CICIDS 2017, which comprises five days of data gathering with 225,745 packages that comprise 80 attributes and have gained over seven-day network operations (e.g., intrusion and regular). The attack simulation in the CICIDS2017 dataset is divided into seven classes including the Heart Bleed Attack, Brute Force Attack, DoS Attack, Botnet, DDoS Attack, Infiltration Attack, and Web Attack. This article shows an analysis of DDoS attacks on IDS. In normal cases, DDoS attacks take place when the system is inundating the victim or bandwidth resources. These attacks occur due to several system compromises (e.g., botnets) that inundate the aimed systems by developing immense network traffic. Moreover, [55] demonstrated that the attribute relation affecting DDoS attacks is similar to the time span of Inter-Arrival Time (IAT) Flow associated with the mean, min, bandwidth packet, max, IAT bandwidth, total bandwidth, and flow duration. Essentially, a higher attribute value increases the possibility of its categorization as DDoS attack. The accuracy, recall, precision, and F-measure metrics (mentioned in Eqs. (2)–(5)) are used to evaluate the effectiveness of the proposed IDS. Table 1 presents

the confusion matrix used to calculate the evaluation metrics.

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

$$R = \frac{TP}{TP + FN} \tag{3}$$

$$P = \frac{TP}{TP + FP} \tag{4}$$

$$F = 2 * \frac{P * R}{P + R} \tag{5}$$

where Acc denotes detection accuracy, R denotes recall, P denotes precision, F denotes F-measure. And TP, TN, FP, and FN are true positives, true negatives, false positives, and false negatives, respectively.

Table 1: Confusion matrix

Symbol	Actual	Predicted	Description
TP	Attack	Attack	Malicious network traffics classified correctly
TN	Normal	Normal	Legitimate network traffics classified correctly
FP	Normal	Attack	Legitimate network traffics classified wrongly as malicious ones
FN	Attack	Normal	Malicious network traffics classified wrongly as legitimate ones

4.2 Experimental Setup

CNN-LSTM is applied in Python using Keras. Examinations are conducted on a PC with 32 Gigabyte (GB) of Random Access Memory (RAM) running 64-bit Ubuntu 14.04 Operating System (OS) and Core i7 processor. The software stack consists of Scala 2.11.8, Apache Spark v2.3.0, Keras, and Java Development Kit (JDK) 1.8. Furthermore, 80% of the data is employed for the training with 10-fold cross-validation, followed by evaluation of the trained model following the 20% held-over data. The CNN-LSTM is applied in Keras, followed by its training on an Nvidia TitanX Graphics Processing Unit (GPU) with (Compute Unified Device Architecture) CUDA and CUDA Deep Neural Network (cuDNN), which speeds up the entire experiment.

4.3 Results and Findings

The dataset was randomly split into training (80%) and test sets (20%) for testing. While 10% of the training set was employed for the verification. In the training stage, first-order gradient-based enhancement approaches including Root Mean Squared Propagation (RMSprop), Adam Maximum Optimizer (AdaMax), Adam, and Adam Gradient Optimizer (AdaGrad), and the learning rates were employed for the optimization of the binary cross-entropy loss of the forecasted network packet *vs.* the real network packet. This network packet goes through optimization with diverse incorporations of hyperparameters from 10-fold cross-validation and grid search for the training of every model on a batch size of 128. The performance was evaluated with the addition of Gaussian noise, convolutions, and LSTM layers to decrease overfitting and increase model generalization.

As per the suggestion, the CICIDS 2017 Monday Working Hours dataset was employed. Following the acquirement of the necessary attributes, the data are cleaned and normalized to allow the availability of the dataset for training. CIC2017 DDoS Monday-Working-Hours-DDoS-Attack is divided into two sections, namely 30% of testing data and 70% of training data. Specifically, the testing data comprises 67,723 packets (55,173 are classified as normal and 12,550 are classified as DDoS), while the training data comprises 158,022 packets (128,737 are classified as normal and 29,285 are classified as DDoS). The feature selection employs the ensemble approach, as highlighted in Section 3.2. On the other hand, 24 attributes are chosen in the last optimal subset, with the findings of the feature names of the ensemble selection presented in Table 2.

Table 2: Selected features

Total.length.of.Bwd.packets	Bwd.IAT.total	Packet.length.variane
Fwd.packet.length.min	Bwd.IAT.mean	PSH.flag.count
Bwd.packet.length.min	Bwd.IAT.std	ACK.flag.count
Bwd.packet.length.std	Bwd.IAT.min	Down.up.ratio
Flow.IAT.mean	Fwd.packets.s	Average.packet.size
Flow.IAT.mean	Bwd.packets.s	Average.Fwd.segment.size
Fwd.IAT.mean	Min. packet. length	Subflow.FWD.bytes
Init_Win_bytes_forward	Init_Win_bytes_backward	Active.mean

Following the experimental findings resulting from the evaluation of the proposed IDS, the ideal results are computed in respect of detection accuracy, precision, recall, and F-measure. It could be seen that the CNN-LSTM IDS proposed in this paper shows high performance in respect of all evaluation metrics primarily because of the use of effective ensemble feature selection and an appropriate CNN-LSTM detection model. In sum, CNN-LSTM IDS, as depicted in below Fig. 5, achieved 97.9%, 98.3%, 97.9%, and 98.1% in accuracy, precision, recall, and F-measure, respectively. The IDS was trained using significant subset of features will positively affect the detection model.

To demonstrate the influence of ensemble feature selection on the performance of IDS while detecting DDoS attacks, a comparison experiment is conducted, as illustrated in Fig. 6, among CNN-LSTM IDS with and without feature selection. Besides, it is easily noticeable that an exceptional level of security against DDoS attacks might be gained through the hybrid detection model, which is solidly built, straightforward, speedy, and proper for immediate use.

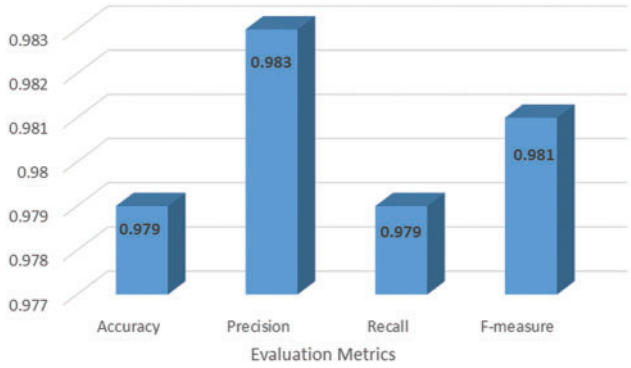

Figure 5: Evaluation results of CNN-LSTM IDS

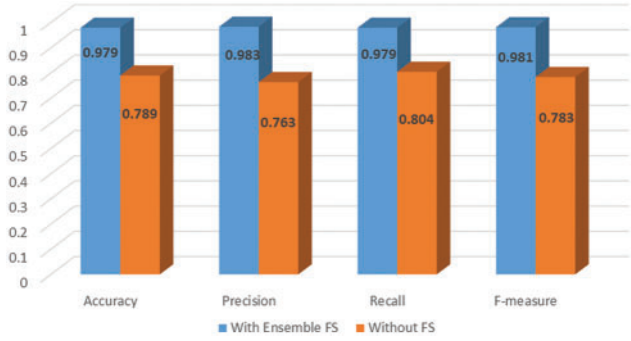

Figure 6: Comparison among CNN-LSTM IDS with and without feature selection

The performance of the CNN-LSTM IDS is compared with some of the state-of-the-art IDS based on the evaluation metrics that are mentioned in Section 4.1. The comparative evaluation test is used to estimate the accuracy, precision, recall, and F-measure of the CNN-LSTM IDS against related IDSs for detecting the DDoS attack in the Cloud environment. The CNN-LSTM IDS is compared with the CNN-IDS [59], the RNN-IDS [60], the LSTM-IDS [61], and the LigthGBM-IDS [62], as they are used as benchmark models due to their comparable performance. Table 3 shows the evaluation metrics of CNN-LSTM IDS and state-of-the-art IDSs. In specific, the results demonstrated in Table 3 reveal that the CNN-LSTM IDS outperforms all state-of-the-art IDSs that are compared with, in terms of accuracy, precision, recall, and F-measure. As aforementioned, this is due to the fact that the CNN-LSTM IDS ensures an efficient detection of DDoS attacks in the Cloud environment since it employs an efficient ensemble-feature selection model and hybrid-detection model.

To demonstrate the significance of the enhancement of the CNN-LSTM IDS to the state-of-the-art IDSs, a t-test is used, which is one of the common statistical tests used for the means [63]. Herein, it is used to test whether there is a difference between two independent means or not, by computing the probability of error (i.e., p-value). The difference between the two means is significant only if the p-value < 0.05; otherwise, the difference is not significant. As a result, the hypotheses formulated to determine the significance of the CNN-LSTM IDS compared with the state-of-the-art IDSs are formulated as follows:

I. H_0: CNN-LSTM IDS does not significantly enhance the state-of-the-art IDSs in terms of accuracy, precision, recall, and F-measure.

II. H_1: CNN-LSTM IDS significantly enhances the state-of-the-art IDSs in terms of accuracy, precision, recall, and F-measure.

Table 3: Comparison with state-of-the-art IDSs

Metric	CNN-LSTM IDS	CNN-IDS	RNN-IDS	LSTM-IDS	LigthGBM-IDS
Accuracy	0.979	0.76	0.794	0.832	0.838
Precision	0.983	0.683	0.807	0.841	0.828
Recall	0.979	0.827	0.811	0.853	0.867
F-measure	0.981	0.748	0.809	0.847	0.847

Table 4 illustrates the t-test results used to determine the significance of CNN-LSTM IDS enhancement on the state-of-the-art IDSs by comparing the means with CNN-IDS, the RNN-IDS, the LSTM-IDS, and the LigthGBM-IDS in terms of accuracy, precision, recall, and F-measure.

Table 4: Significant of enhancement

IDS	Accuracy	Recall	False Alarm	Precision	F-measure
CNN-IDS	9.32608E-29	9.849E-144	2.8121E-140	8.9667E-144	9.6938E-145
RNN-IDS	1.9719E-135	1.7446E-143	1.6342E-146	2.0471E-144	4.03E-144
LSTM-IDS	9.32608E-29	5.5167E-133	1.1482E-144	2.9552E-144	8.5209E-142
LigthGBM-IDS	1.517E-132	5.4897E-144	6.5876E-141	7.3049E-144	2.1863E-144

As depicted from the above table, all values of P-value < 0.05; thus, II. H1 is accepted, which means CNN-LSTM IDS significantly enhances the state-of-the-art IDSs in terms of accuracy, precision, recall, and F-measure. In sum, the CNN-LSTM IDS can be adopted to secure the Cloud environment in particular and could be adapted in other network-based environments in general, without being endangered by DDoS attacks. Once the CNN-LSTM IDS is implemented in real Cloud environment, the network traffic is efficiently classified into normal or DDoS attack ones; therefore, the ability to distinguish the traffic either it is a DDoS attack or not is figured out automatically with high performance. Consequently, the Cloud environment is secured from DDoS attacks and its negative impact on the whole network. The CNN-LSTM IDS then achieves the requirements of getting high security and automatic efficient self-decision.

5 Conclusion

IDS on the CC environment has received paramount interest over the last few years. Among the latest approaches, ML-based IDS provides the ability to discover and detect attacks efficiently. DDoS attacks are one of the most frequent that inflict serious damage and affect cloud performance. In a DDoS attack, the attacker usually uses innocent compromised computers (called zombies) by taking advantage of known or unknown bugs and vulnerabilities to send a large number of packets from these already-captured zombies to a server. This may occupy a major portion of the network bandwidth of the victim cloud infrastructures or consume much of the servers' time. Therefore, this paper presents an efficient IDS based on ensemble feature selection and hybrid deep learning to detect DDoS attacks in the CC environment. The proposed IDS was compared with recent state-of-the-art IDSs including

CNN-IDS, the RNN-IDS, the LSTM-IDS, and the LigthGBM-IDS, and it shows an outstanding performance in terms of accuracy, recall, precision, and F-0measure over these models due to the use of ensemble feature selection and the hybrid DL classifier.

However, a probable drawback of this method is that the IDS is tested only on a single dataset. Thus, testing the IDS on a more current dataset is important, considering the constant changes in the attack traffic signature. It is suggested that future studies (i) expand the current study to ensure the detection of network and anomaly misuses on streaming data in real-time, and (ii) emphasize the exploration of deep learning as an attribute extraction instrument to gain an understanding of the competent data illustrations when other anomaly recognition issues take place in the more recent dataset.

Acknowledgement: I express my gratitude to Northern Border University, Saudi Arabia, for administrative and technical support.

Funding Statement: The authors gratefully acknowledge the approval and the support of this research study by the Grant No. SCIA-2022-11-1545 from the Deanship of Scientific Research at Northern Border University, Arar, K.S.A.

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

References

[1] D. Bhamare, R. Jain, M. Samaka and A. Erbad, "A survey on service function chaining," *Journal of Network and Computer Applications*, vol. 75, no. 2, pp. 138–155, 2016.

[2] Y. Sanjalawe, M. Anbar, S. Al-E'mari, R. Abdullah, I. Hasbullah *et al.,* "Cloud data center selection using a modified differential evolution," *Computers, Materials & Continua*, vol. 69, no. 3, pp. 3179–3204, 2021.

[3] A. Shameli-Sendi, M. Pourzandi, M. Fekih-Ahmed and M. Cheriet, "Taxonomy of distributed denial of service mitigation approaches for cloud computing," *Journal of Network and Computer Applications*, vol. 58, no.23, pp. 165–179, 2015.

[4] D. Zeng, J. Zhang, L. Gu, S. Guo and J. Luo, "Energy-efficient coordinated multipoint scheduling in green cloud radio access network," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 10, pp. 9922–9930, 2018.

[5] A. Josep, R. Katz, A. KonWinSki, L. Gunho, D. Patterson *et al.,* "A view of cloud computing," *Communications of the ACM*, vol. 53, no. 4, pp. 50–58, 2010.

[6] A. Bakshia and Y. Dujodwala, "Securing cloud from DDoS attacks using intrusion detection system in virtual machine," in *2010 Second Int. Conf. on Communication Software and Networks*, Perth, WA, Australia, IEEE, pp. 260–264, 2010.

[7] S. Chapade, K. Pandey and D. Bhade, "Securing cloud servers against flooding-based DDoS attacks," in *2013 Int. Conf. on Communication Systems and Network Technologies*, Gwalior, India, IEEE, pp. 524–528, 2013.

[8] Y. Mehmood, M. Shibli, U. Habiba and R. Masood, "Intrusion detection system in cloud computing: Challenges and opportunities," in *2013 2nd National Conf. on Information Assurance (NCIA)*, Rawalpindi, Pakistan, IEEE, pp. 59–66, 2013.

[9] S. Tummalapalli and A. Chakravarthy, "Intrusion detection system for cloud forensics using Bayesian fuzzy clustering and optimization based SVNN," *Evolutionary Intelligence*, vol. 14, no. 2, pp. 699–709, 2021.

[10] J. Cheng, X. Tang, V. Sheng, Y. Liu and W. Guo, "Flow correlation degree optimization driven random forest for detecting DDoS attacks in cloud computing," *Security and Communication Networks*, vol. 68, no. 6, pp. 40–51, 2018.

[11] P. Lin, K. Ye and C. Xu, "Dynamic network anomaly detection system by using deep learning techniques," in *Int. Conf. on Cloud Computing*, Perth, WA, Australia, Cham, Springer, pp. 161–176, 2019.
[12] M. Aamir and S. Zaidi, "Clustering based semi-supervised machine learning for DDoS attack classification," *Journal of King Saud University-Computer and Information Sciences*, vol. 33, no. 4, pp. 436–446, 2021.
[13] N. Hoque, M. Bhuyan, R. Baishya, D. Bhattacharyya and J. Kalita, "Network attacks: Taxonomy, tools and systems," *Journal of Network and Computer Applications*, vol. 40, no. 12, pp. 307–324, 2014.
[14] S. Vimal, L. Kalaivani and M. Kaliappan, "Collaborative approach on mitigating spectrum sensing data hijack attack and dynamic spectrum allocation based on CASG modeling in wireless cognitive radio networks," *Cluster Computing*, vol. 22, no. 5, pp. 10491–10501, 2019.
[15] S. Annamalai, R. Udendhran and S. Vimal, "An intelligent grid network based on cloud computing infrastructures," in *Novel Practices and Trends in Grid and Cloud Computing*, Pennsylvania, USA: IGI Global, pp. 59–73, 2019.
[16] S. Zargar, J. Joshi and D. Tipper, "A survey of defense mechanisms against distributed denial of service (DDoS) flooding attacks," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 4, pp. 2046–2069, 2013.
[17] G. Kumar, "Denial of service attacks–An updated perspective," *Systems Science & Control Engineering*, vol. 4, no. 1, pp. 285–294, 2016.
[18] M. Masdari and M. Jalali, "A survey and taxonomy of DoS attacks in cloud computing," *Security and Communication Networks*, vol. 9, no. 16, pp. 3724–3751, 2016.
[19] F. Palmieri, S. Ricciardi, U. Fiore, M. Ficco and A. Castiglione, "Energy-oriented denial of service attacks: An emerging menace for large cloud infrastructures," *The Journal of Supercomputing*, vol. 71, no. 5, pp. 1620–1641, 2015.
[20] E. Kabir, J. Hu, H. Wang and G. Zhuo, "A novel statistical technique for intrusion detection systems," *Future Generation Computer Systems*, vol. 79, no. 11, pp. 303–318, 2018.
[21] S. Vimal and P. Subbulakshmi, "Secure data packet transmission in MANET using enhanced identity-based cryptography," *International Journal of New Technologies in Science and Engineering*, vol. 3, no. 12, pp. 35–42, 2016.
[22] S. Pasupathi, S. Vimal, Y. Harold-Robinson, M. Khari, E. Verdú *et al.,* "Energy efficiency maximization algorithm for underwater mobile sensor networks," *Earth Science Informatics*, vol. 14, no. 1, pp. 215–225, 2021.
[23] Z. Tan, A. Jamdagni, X. He, P. Nanda and R. Liu, "Denial-of-service attack detection based on multivariate correlation analysis," in *Int. Conf. on Neural Information Processing*, Berlin, Heidelberg, Springer, pp. 756–765, 2021.
[24] H. Om and A. Kundu, "A hybrid system for reducing the false alarm rate of anomaly intrusion detection system," in *2012 1st Int. Conf. on Recent Advances in Information Technology (RAIT)*, Dhanbad, India, IEEE, pp. 131–136, 2012.
[25] N. Kumar and M. Sharma, "Study of intrusion detection system for DDoS attacks in cloud computing," in *2013 Tenth Int. Conf. on Wireless and Optical Communications Networks (WOCN)*, Bhopal Madhya Pradesh, India, IEEE, pp. 1–5, 2013.
[26] T. Pandit and A. Dudy, "A feed forward artificial neural network-based system to minimize Dos attack in wireless network," *International Journal of Advances in Engineering & Technology*, vol. 7, no. 3, pp. 938–951, 2014.
[27] Q. Gaur, D. Sanghi, M. Conti and R. Buyya, "DDoS attacks in cloud computing: Issues, taxonomy, and future directions," *Computer Communications*, vol. 107, no. 22, pp. 30–48, 2017.
[28] K. Srinivasan, A. Mubarakali, A. Alqahtani and A. Kumar "A survey on the impact of DDoS attacks in cloud computing: Prevention, detection and mitigation techniques," in *Intelligent Communication Technologies and Virtual Mobile Networks: ICICV 2019*, Springer International, Tirunelveli, India, pp. 252–270, 2019.
[29] G. Somani, M. Gaur, D. Sanghi, M. Conti and R. Buyya, "Service resizing for quick DDoS mitigation in cloud computing environment," *Annals of Telecommunications*, vol. 72, no. 5, pp. 237–252, 2017.

[30] J. Fontaine, C. Kappler, A. Shahid and E. Poorter, "Log-based intrusion detection for cloud web applications using machine learning," in *Int. Conf. on P2P, Parallel, Grid, Cloud and Internet Computing*, Tirana, Albania, Cham, Springer, pp. 197–210, 2019.
[31] M. Hasan, M. Nasser, S. Ahmad and K. Molla, "Feature selection for intrusion detection using random forest," *Journal of Information Security*, vol. 7, no. 3, pp. 129–140, 2016.
[32] H. Zawbaa, E. Emary, B. Parv and M. Sharawi, "Feature selection approach based on moth-flame optimization algorithm," in *2016 IEEE Congress on Evolutionary Computation (CEC)*, Vancouver, Canada, IEEE, pp. 4612–4617, 2016.
[33] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Advances in Engineering Software*, vol. 95, no. 33, pp. 51–67, 2016.
[34] Y. Liu, D. Qiu and H. Li, "The intrusion detection modle utilizing LE and modified PSO-BP," in *2017 8th IEEE Int. Conf. on Software Engineering and Service Science (ICSESS)*, Beijing, China, IEEE, pp. 318–321, 2017.
[35] S. Ho, S. Al Jufout, K. Dajani and M. Mozumdar, "A novel intrusion detection model for detecting known and innovative cyberattacks using convolutional neural network," *IEEE Open Journal of the Computer Society*, vol. 2, no. 11, pp. 14–25, 2021.
[36] J. Gao, "Network intrusion detection method combining CNN and BiLSTM in cloud computing environment," *Computational Intelligence and Neuroscience*, vol. 21, no. 1, pp. 11–32, 2022.
[37] P. Kshirsagar, H. Manoharan, H. Alterazi, N. Alhebaishi, O. Rabie *et al.,* "Construal attacks on wireless data storage applications and unraveling using machine learning algorithm," *Journal of Sensors*, vol. 13, no. 2, pp. 13–26, 2022.
[38] A. Kaur, S. Pal and A. Singh, "Hybridization of K-means and firefly algorithm for intrusion detection system," *International Journal of System Assurance Engineering and Management*, vol. 9, no. 4, pp. 901–910, 2018.
[39] V. Balamurugan and R. Saravanan, "Enhanced intrusion detection and prevention system on cloud environment using hybrid classification and OTS generation," *Cluster Computing*, vol. 22, no. 6, pp. 13027–13039, 2019.
[40] M. Saharkhizan, A. Azmoodeh, H. HaddadPajouh, A. Dehghantanha, R. Parizi *et al.,* "A hybrid deep generative local metric learning method for intrusion detection," in *Handbook of Big Data Privacy*, Cham: Springer, pp. 343–357, 2020.
[41] L. Vu, Q. Nguyen, D. Nguyen, D. Hoang and E. Dutkiewicz, "Deep generative learning models for cloud intrusion detection systems," *IEEE Transactions on Cybernetics*, vol. 1, no. 11, pp. 23–35, 2022.
[42] N. Paulauskas and J. Auskalnis, "Analysis of data pre-processing influence on intrusion detection using NSL-KDD dataset," in *2017 Open Conf. of Electrical, Electronic and Information Sciences (eStream)*, Vilnius, Lithuania, IEEE, pp. 1–5, 2017.
[43] Y. Zhong, W. Chen, Z. Wang, Y. Chen, K. Wang *et al.,* "A novel network anomaly detection model based on heterogeneous ensemble learning," *Computer Network*, vol. 169, no. 21, pp. 107–121, 2020.
[44] A. Mahfouz, A. Abuhussein, D. Venugopal and S. Shiva, "Ensemble classifiers for network intrusion detection using a novel network attack dataset," *Future Internet*, vol. 12, no. 180, pp. 11–23, 2020.
[45] P. Devan and N. Khare, "An efficient XGBoost–DNN-based classification model for network intrusion detection system," *Neural Computing Applications*, vol. 32, no. 11, pp. 12499–12514, 2020.
[46] Y. Shi, "Particle swarm optimization," *IEEE Connections*, vol. 2, no. 1, pp. 121–139, 2004.
[47] S. Mirjalili, M. Mirjalili and A. Lewis, "Grey wolf optimizer," *Advances in Engineering Software*, vol. 69, no. 1, pp. 46–61, 2014.
[48] A. Gandomi and A. Alavi, "Krill herd: A new bio-inspired optimization algorithm," *Communications in Nonlinear Science and Numerical Simulation*, vol. 17, no. 12, pp. 4831–4845, 2012.
[49] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Advances in Engineering Software*, vol. 95, no. 11, pp. 51–67, 2016.
[50] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computing*, vol. 9, no. 21, pp. 1735–1780, 1997.

[51] F. Gers, N. Schraudolph and J. Schmidhuber, "Learning precise timing with LSTM recurrent networks," *Journal of Machine Learning Research*, vol. 3, no. 11, pp. 115–143, 2002.
[52] B. Pang, E. Nijkamp and Y. Wu, "Deep learning with Tensorflow: A review," *Journal of Educational and Behavioral Statistics*, vol. 45, no. 2, pp. 227–248, 2020.
[53] M. Khan, M. Karim and Y. Kim, "A two-stage big data analytics framework with real-world applications using spark machine learning and long short-term memory network," *Symmetry*, vol. 10, no. 21, pp. 21–37, 2018.
[54] J. Shook, T. Gangopadhyay, L. Wu, N. Ganapathysubramanian, S. Sarkar *et al.,* "Crop yield prediction integrating genotype and weather variables using deep learning," *Plos One*, vol. 16, no. 6, pp. 34–49, 2021.
[55] I. Sharafaldin, A. Lashkari and A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *4th Int. Conf. on Information Systems Security and Privacy (ICISSP)*, Madeira, Portugal, pp. 108–116, 2016.
[56] A. Shiravi, H. Shiravi, M. Tavallaee and A. Ghorbani, "Toward developing a systematic approach to generate benchmark datasets for intrusion detection," *Computers & Security*, vol. 31, no. 3, pp. 357–374, 2012.
[57] I. Sharafaldin, A. Gharib, A. Lashkari and A. Ghorbani, "Towards a reliable intrusion detection benchmark dataset," *Software Networking*, vol. 1, no. 31, pp. 177–200, 2018.
[58] A. Gharib, I. S. A. Lashkari and A. Ghorbani, "An evaluation framework for intrusion detection dataset," in *2016 Int. Conf. on Information Science and Security (ICISS)*, Pattaya, Thailand, IEEE, pp. 1–6, 2016.
[59] M. ElSayed, N. Le-Khac, M. Albahar and A. Jurcut, "A novel hybrid model for intrusion detection systems in SDNs based on CNN and a new regularization technique," *Journal of Network and Computer Applications*, vol. 191, no. 3, pp. 103–116, 2021.
[60] I. Ullah and Q. Mahmoud, "Design and development of RNN anomaly detection model for IoT networks," *IEEE Access*, vol. 10, no. 2, pp. 62722–62750, 2022.
[61] Y. Imrana, Y. Xiang, L. Ali and Z. Abdul-Rauf, "A bidirectional LSTM deep learning approach for [61] intrusion detection," *Expert Systems with Applications*, vol. 185, no. 11, pp. 115–129, 2021.
[62] J. Liu, Y. Gao and F. Hu, "A fast network intrusion detection system using adaptive synthetic oversampling and LightGBM," *Computers & Security*, vol. 106, no. 11, pp. 102–119, 2021.
[63] K. Kim, "T test as a parametric statistic," *Korean Journal of Anesthesiology*, vol. 68, no. 6, pp. 540–546, 2015.