Journal of Computer Science

Original Research Paper

Deep Regularization Mechanism for Combating Class Imbalance Problem in Intrusion Detection System for Defending DDoS Attack in SDN

Narender M and Yuvaraju B N

Department of Computer Science Engineering, The National Institute of Engineering, Mysuru, Visvesvaraya Technological University, Belagavi, India

Article history
Received: 22-11-2022
Revised: 03-02-2023
Accepted: 09-02-2023

Corresponding Author:
Narender M
Department of Computer Science Engineering, The National Institute of Engineering, Mysuru, Visvesvaraya Technological University, Belagavi, India
Email: narender@nie.ac.in

Abstract: Integration of deep learning into Intrusion Detection Systems (IDS) for Software Defined Networking (SDN) is an emerging field of research. Most of the datasets used to build IDS are highly imbalanced, especially in the case of DDoS attacks, which account for a larger percentage of malicious samples than normal traffic. As a result of a class imbalance, the classification result is distorted since deep learning is limited in its ability to generalize and is misled into favoring the majority class. This study aims to confront the problem of class imbalance by introducing a new deep regularization mechanism that allows the learning model to unlearn biased information. Unlike the existing system, the proposed mechanism integrates two multi-layer neural networks to extract shared information and biased distribution. The first learning model generalizes the data distribution to classify network traffic as normal or attacks class. On the other hand, the second model is integrated with the embedding layers feature of the first model, which learns the bias distribution and then it regressively instructs the first network not to learn this biased information. The proposed regularization scheme is evaluated on the most recent and highly imbalanced network dataset CIC-DDoS2019. The proposed scheme is compared with different supervised learning classifiers executed on the same dataset in the experiment, balanced with the smote technique. The proposed model outperforms other learning techniques and reached an overall precision, recall, and F1-score of 96.71, 97.14, and 96.92%, respectively.

Keywords: Software Defined Networking, Distributed Denial of Service, SYN Flood, Deep Learning, Class Imbalance, Intrusion Detection System

Introduction

With the increased adoption of applications and services over the internet, there is a considerable increase in global usage, which leads to opportunities with increasing concern of threat (Wagner *et al*., 2019). The conventional adversaries, e.g., Trojans, Worms, and Denial of Service (DoS), have already created news of reported attacks in past decades (Hussain and Singha, 2020; El Mrabet *et al*., 2018). Various research has been conducted to develop a solution to resist DoS attacks (Benarous *et al*., 2017). However, a typical variant of a DoS attack is proven to have a more lethal effect on the network called a Distributed Denial of Service (DDoS) attack (Cihan and Akleylek, 2019). In this form of attack, the adversary explores the weak point in the network and injects a malicious code that keeps on replicating over multiple devices and spreads over a large network where all the compromised computer systems, called bots, are aggregated to initiate this form of attack (Stellios *et al*., 2018). It is nearly impossible to find the authenticity or source of the attacker in this process, as the attacks are initiated from multiple machines in highly uncertain networks. As a part of the solution, it has also been observed from the existing trends that the inclusion of a Software Defined Network (SDN) acts as a beneficial feature in securing this threat (Bannour *et al*., 2017). Characterized by a centralized controller system, SDN offers the construction of protocols for data communication on demand, thereby offering more transparency towards the networking system (Benzekki *et al*., 2016).

However, even SDN is not free from loopholes associated with supportability, scalability, and security issues. In the majority of the cases of DDoS attacks, the target of the attack is mainly either a switch on the data plane or a controller on the control plane (Han *et al*., 2020). There is no doubt that SDN is a potential approach to the networking concept compared to traditional networking system, which uses IP addresses (Chica *et al*., 2020). Therefore, researchers inevitably must emphasize developing an efficient IDS to deal with DDoS attacks with high detection capability.

Challenging Issues and Motivation

In the current literature, several network Intrusion Detection Systems (IDS) are available based on predictive modeling using deep learning methods to predict attack classes. A deep learning powered solution provides accurate results when executed on large amounts of uniformly distributed data samples. However, most deep learning-based IDS perform poorly when the training samples are few, or the dataset contains more training samples subject to some classes than others. The second case is known as the class imbalance problem and datasets referred to as imbalanced datasets prevent the learning model from generalizing patterns and data distribution from the minority class. As a result, the network IDS are prone to high false positive rates and lower detection rates. The class imbalance problem is closely related to the real-time system, where network traffic is dynamic and uncertain. There is no guarantee that the number of malicious traffic will be less, more, or equal to normal traffic. Hence, there is a higher probability that the intrusion detection system may be prone to a biased generalization about the majority class of packets. The problem of imbalanced data has received insufficient attention in recent research. In this regard, there is a need to develop an efficient intrusion detection system that can detect and classify attack traffic un-biasedly.

Contribution of the Proposed Work

Unlike the existing solutions, the proposed research work aimed to combat the issue of biases despite addressing the class-imbalanced problem for enhancing the performance of deep learning driven IDS for mitigating DDoS in the SDN enabled network. This study offers a synchronized bias regularizer scheme by integrating the feature embedding layer of the deep learning model for improving the detection rate of the minority attack class. In summary, the core contributions of this study are as follows:

- Firstly, a mathematical model has been devised to formulate both the problem and objective function as a solution to mitigate biased learning of the model
- Second, a multi-layer neural network is used in the study to develop a synchronized bias regularizer mechanism. This regularizer runs iteratively to capture the bias distribution from the feature map (i.e., the output of the hidden layer) of the main DL network
- The core logic behind the proposed methodology is to inform a DL-driven IDS what to learn and what not to learn from the dataset to better generalize the data distribution in the training phase. By eliminating information associated with bias, the network becomes capable of generalizing precise distribution along the output classes for attack detection and classification

This section discusses the existing technical approaches to safeguard the DDoS attack in the SDN system. The authors in the work of (Yan *et al*., 2015) have presented a theoretical discussion about various recent security approaches toward mitigating the security challenges associated with SDN. According to the author, existing approaches have not been sufficiently carried out to construct a bridge between DDoS attacks and SDN. The study has also facilitated an insight that it is possible to resist DDoS attacks by smartly harnessing SDN's potential. In a similar research direction, the authors (Dong *et al*., 2019; Aladaileh *et al*., 2020) have conducted a theoretical assessment of network security and revealed that channel capacity is another prominent factor that gets adversely affected by DDoS attacks. This problem has been addressed by Alamri and Thayananthan (2020), where a mechanism is presented to resist the DDoS attack over the SDN controller emphasizing proper utilization of resources and detection of the attack. The implementation is carried out by extreme gradient boosting and controlling bandwidth approaches. The system makes use of threshold-based profiling of adaptive bandwidth. Considering CAIDA and NSL-KDD datasets, the presented scheme identifies violated traffic flow with approximately 99% accuracy. However, this strategy is completely based on a threshold-based approach and there could be a scenario where the attack launches different variants of DDoS with less correlation with the threshold. In such a condition, an attacker can bypass this approach. In addition, the above-discussed approaches have not focused on handling the class imbalance problem since they are evaluated on the NSL_KDD dataset, which is quite imbalanced. Therefore, the performance claimed by all these approaches is either limited to specific scenarios or not designed considering the network's real-time constraint. This problem of determination of the flexible and untraceable identification of attack strategy must be subjected to deeper investigation. The authors in the study of Mbow *et al*. (2022) presented a scheme for handling the class imbalance problem by combining multiple data sampling schemes, smote and Tomek link, to improve the performance of learning-driven IDS. An application of the

Siamese learning model has been employed in the work of (Bedi *et al*., 2020) to combat the class imbalance problem. The presented scheme adopted a distance function to measure similarity among the features obtained from the Siamese learning model. Alqarni and El-Alfy (2022) utilize a generative deep learning technique in the design of IDS, executed on an imbalanced dataset NSL-KDD. Deep generative learning acts as an oversampling technique to increase the sample of minority classes.

Bagui and Li (2021), the authors have evaluated several conventional data sampling methods to balance the distribution of data samples subjected to different classes. The adoption of a Convolution Neural Network (CNN) is reported in the study of (Haider *et al*., 2020) to detect DDoS attacks for the system with SDN. Dong and Sarem (2019) have adopted enhanced K-nearest neighboring to understand the vulnerability of the threat associated with DDoS attacks in SDN. The adoption of an enhanced KNN algorithm is further reported by Xu *et al*. (2019) to offer better security options for SDN-based controller systems for resisting DDoS attacks. Phan *et al*. (2020) have developed a mitigation strategy using a deep reinforcement learning approach to offer more accuracy toward detection. Another unique work by Novaes *et al*. (2020) used a combination of fuzzy logic and long short-term memory to detect DDoS attacks. All these approaches have tried to improve the performance of IDS by introducing customization in the predictive model. However, they do not emphasize data modeling, which is important in improving attack classification performance. A concept of ensemble learning is presented by Alao *et al*. (2022), where a Support Vector Machine (SVM) classifier is ensembled with a gaussian mixture model to detect intrusions in the network. The validation of the model is done against the NSL KDD dataset and the simulation outcome claimed a 0.18% false acceptance rate achieved by the system.

Ravi and Shalinie (2020) have presented an attack identification scheme using a semi-supervised learning technique. The adoption of a self-exploration scheme is seen in the work of (Ouiazzane *et al*., 2021). The authors developed a hybrid detection scheme considering signature-based and anomaly IDS. This study proposes a multi-agent solution to simulate network attacks and explore optimal intrusion detection and prevention solutions. In the work of (Alfrhan *et al*., 2020), the authors handled the class imbalance problem associated with CIC-IDS2017 by oversampling the minority class using smote technique. Attack detection uses random forest, probabilistic model, and KNN classifiers. The researchers (Setiawan *et al*., 2020) adopted a feature learning scheme with a weighted-SVM classifier to IDS performance on the imbalanced dataset. Khudhu and Samsudin (2022) explored the potential of an autoencoder in feature modeling and implemented a random forest classifier for intrusion detection. A joint operation on multiple schemes is done in the work of (Gaffer *et al*., 2012), where smote was integrated with a genetic algorithm and fuzzy logic to determine optimal parameters to balance the dataset. Hence, considerable works are on controlling the impact of dataset imbalance or class imbalance on the network IDS. But all the approaches have limitations and most have not focused on controlling biased learning of deep learning models. The next section outlines the significant research problem explored based on above discussed related work.

Research Problem Being Explored

Lack of novelty in the existing research works: A closer analysis reveals that the existing methods in the context of DDoS detection have no significant novelty. The success rate of most of the existing solutions against DDoS highly depends on the pre-defined information given to the SDN controller.

Limitation to certain attacks: The existing works based on the thresholding approach are only effective for certain attacking scenarios. However, there could be a possible scenario where the attack launches different variants of DDoS and in such a condition, an adversary may bypass this security feature.

Lack of focus on class imbalance problem: Most DDoS or DoS datasets are associated with imbalance factors, leading to bias in the prediction and classification. It has been analyzed that most previous works using learning mechanisms lack effective data modeling and are subjected to impreciseness in classifying sophisticated attacks in large-scale networks.

No cost effectiveness: Most existing approaches to solve class imbalance problems are based on conventional oversampling and under sampling methods, which may result in overfitting. The smote technique for oversampling is widely adopted in the literature. However, it is limited to the binary classification problem. Also, most of the oversampling techniques lead to longer training times.

Problem Statement

Hence, there is a requirement to develop "an upgraded attack detection system that can provide a reliable security feature to the controller for timely detection of DDoS attacks in SDN oriented networking ecosystem".

Proposed System

The proposed system focuses on developing an efficient and upgraded attack detection system to address security issues in the SDN network. Therefore, the proposed work uses a deep neural network with a synchronized bias regularizer to develop an Intelligent Intrusion Detection Model (IIDM) that addresses DDoS attacks and protects the communication between the SDN controller and the device layer. The modeling of the proposed upgraded and intelligent attack detection model is shown in Fig. 1.

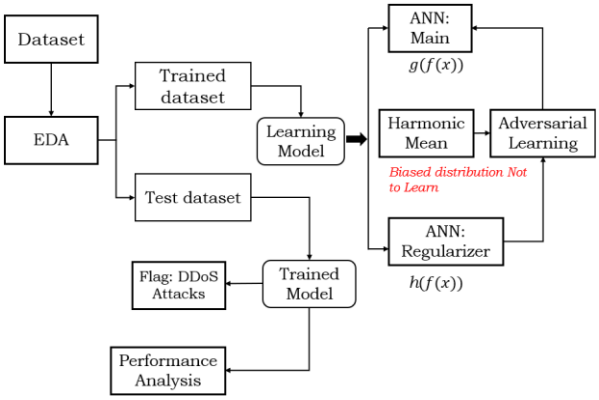

Fig. 1: Architecture of the proposed IIDM

The real time scenario of the DDOS attacks in the SDN enabled networking system inspires the design of the proposed system IIDM. In this case, the attack on the server gets exposed to a very high number of packets that obscures normal or benign traffic packets. For instance, a user sends between 15-20 packets during an actual transaction to a server; however, during a DDoS attack, at least 10000 packets are sent during the same time frame. Under this interpretation, any DDoS dataset is more likely to be biased. Therefore, the proposed study introduces a novel mechanism of regularization to control the bias phenomena in the training phase of the learning model. The ideology behind this is that if we train the model with the highly biased dataset, then the model becomes biased towards detecting only the class of majority samples. For example, if training of the learning model is carried out on the majority class in the dataset, i.e., is subjected to DDoS packets, the model becomes biased towards detecting packets as DDoS even if there is normal traffic flow in the network. Therefore, the feature representation bias during model training must be handled efficiently to achieve a reliable detection mechanism. Unlike the existing research work, the proposed study introduces a different approach to learning model design, where two multi layered, neural networks are implemented to perform better feature learning. The first type of neural network is implemented as the main network to predict network traffic as an attack or normal traffic. On the other hand, the second neural network acts as a regularizer that targets to learn the bias value in the data distribution intermediate output obtained from the main network. In this way, the model makes better generalizations of feature representation and can efficiently address security issues in large scale network traffic.

Dataset Description

The proposed system considers the case of reported DDoS attacks on the modern networking system. Therefore, the recent network dataset CIC-DDoS2019 is adopted, where each traffic flow is labeled along with time stamp information, destination port, and IP address of destination in the form of CSV files. This information within this dataset is designed considering a test environment of 25 subjects based on their normal user communication over all possible internet-based protocols (Email, SSH, FTP, HTTP, HTTPS). The dataset used includes two types of DDoS attacks, such as (i) Reflection attacks and (ii) Exploitation attacks. The reflection attack is designed to exploit the challenge-response authentication system. The exploitation attack is aimed at the system function located in the syn.csv file in the dataset. There are different variants of the DDoS attack available with this dataset: SNMP, DNS, NTP, SYN, UDP-Lag, UDP, MSSQL, LDAP, and NetBIOS. Another exclusive part of the database design is that a total of 12 variants of DDoS attacks were considered for training and seven different variants of DDoS attacks were considered for testing. However, the proposed study is interested in detecting TCP SYN flood attacks only because this dataset comes with massive traffic samples and is quite imbalanced, which is most suitable to evaluate the proposed methodology. The dataset has been reported to be used in many existing studies investigating networks and security. Hence, they are highly reliable to be used in the proposed study.

SYN flood attack is a special type of DDoS attack. The attacker or adversary forwards repeated SYN packets to each port on TCP, aiming to prevent legitimate users from using services and resources offered by the TCP server. The server sends SYN_ACK and waits for the identical user's ACK after sending SYN packets. However, the connection will never be completed without receiving the ACK. SYN backlog queues and other resources are exhausted whenever the server receives large spurious or deceptive SYN packets. Hence, there will be no scope for data packets from the legitimate user. The significant challenge is to differentiate between valid SYN and malicious SYN. Similar to the SYN flood attack just discussed, an SDN flood attack involves incoming SYN packets being dealt with by the SDN controller in additional or supplementary ways.

Rationale Behind Choosing CIC-DDoS2019 Dataset

Intrusion detection for network protection has been a topic of extensive research for a long time. It continues to be a relevant study area with technological advancements and networking. The literature shows a variety of datasets available for IDS (Manjula *et al*., 2023). However, in the current study, the CIC-DDoS2019 dataset is well-known for detecting DDoS attacks in computer networks. This dataset consists of real-world traffic samples from a modern production network. The dataset reflects the real-world distribution of DDoS attacks, which often exhibit class imbalance. The CIC-DDoS2019 dataset provides a better scenario for deep learning models to handle class imbalance problems than other datasets, including old and new datasets.

Exploratory Analysis

The study initially performs Exploratory Data Analysis (EDA) to get a comprehensive insight into the dataset and understand the suitable data treatment operation requirement. It has been analyzed that the dataset consists of 1582681 rows and 88 columns, each belonging to integer, float, and categorial datatypes. Based on the observation, it has been found that there is an unwanted column with extra wide spaces. Therefore, before proceeding to model implementation, the dataset is cleaned. The study uses the strip () function to clear the wide spaces and the extra columns are dropped out. Then, the proposed study analyses the class of the data packets and it is observed that there are two types of packets, namely SYN and Benign. The SYN class represents attack packets, whereas the benign class represents normal traffic packets. Based on further exploration, it has been analyzed that most of the time, SYN flood attacks are launched from two source IPs, i.e., 172.16.0.5 and 192.168.50.1, whereas the rest are genuine or legitimate users. An important factor is noted that the number of packets belonging to the SYN flood class is very high (i.e., 1582289) compared to the normal packets (i.e., 392). In other words, only 0.02% of packets are subjected to normal packets and the rest are attack packets. As a subsequent step, the next observation is done towards analyzing SYN and ACK flags pattern Table 1.

Table 1 shows that the adopted dataset is heavily imbalanced, leading to bias prediction toward the detection of attacks packet only. It can also be seen in the density plot shown in Fig. 2. Therefore, the present dataset cannot be used directly for training the model. However, the dataset can be down-sampled into a smaller size to make it balanced. In a real-time scenario, during a DDoS attack, the server gets bombarded with a very high number of packets that obscures normal packets. This is the fact that makes DDoS dataset imbalanced. Considering this fact, the proposed work does not down-sample; rather, it proposes a better model that uses novel regularization techniques that minimize training bias. In this regard, an upgraded deep neural network-driven attack detection system is built to meet the requirement of real-time implementation where the network traffic is uncertain and dynamic.

Mathematical Modelling

It has been observed from the data exploration and analysis that the dataset is biased since only 0.02% of all the packets are normal and the rest are attack packets. If the model is trained with the biased data, then the model generalizes and learns the biased data distribution as a useful feature to categorize traffic classes. It can lead to poor performance in the deployment or testing phase because the biased data set cannot accurately represent the model's use cases, leading to deviations in results and low levels of accuracy.

Let us consider the given dataset D consists of observations $x \in X$ and class label $y \in Y$. Let us consider another variable B representing a set of biases that input X can possess in the training phase. So, when a set of input data samples x is fed to the deep learning model in its training phase, then the learning of the model is more prone to the biased outcome, numerically expressed in Eq. 1:

$$M\left(b\left(X_{train}\right);Y''M\left(b\left(X_{test}\right),Y\right)\right)\approx 0 \quad (1)$$

where, X_{train} input data sample in the training dataset and X_{tesst} is an unknown input to the trained model from the testing dataset and Y is the corresponding class label. Here, $M(\cdot,\cdot\cdot)$ denotes mutual or shared information in training data and testing data samples. So, training a model on biased data always results in biased prediction, numerically expressed Eq. 2:

$$M\left(b\left(X\right);c\left(f\left(X\right)\right)\right)>>0 \quad (2)$$

Equation 2, $b(X)$ is the biases associated with training features and $c(f(X))$ is the classifier model that maps input observation to the output class label such that: c: $X^K \rightarrow Y$ and K denote the dimension of the feature map in the hidden layer of the learning model.

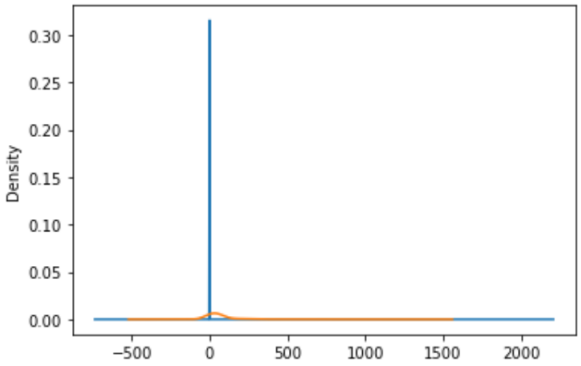

Fig. 2: The density plot exhibiting the dataset is highly biased as there is no similar pattern between the packet length (blue) graph of flow duration (orange)

Table 1: SYN and ACK flag count

SYN flag count			ACK flag count		
Label	Benign	SYN	Label	Benign	SYN
0	390	1582090	0	333	825
1	2	199	1	59	1581464

The model trained with the biased dataset $b(X)$, becomes biased towards the detection majority classes. In other words, the model tends to predict the class with the majority of samples more frequently. For example, if one class has 99% of the samples and the other class has only 1% of the samples, a model trained on this data will have a high accuracy by simply always predicting the majority class. However, it may not perform well on samples belonging to the minority class, resulting in poor performance in real-world applications. Therefore, handling biases in the training model becomes an optimization problem, which needs an effective calibration in adjusting hyperparameters and minimizing the loss so that the classifier model *c(x)* should not get a biased feature map as its input from the hidden layer of the model *f(x)*. The problem of optimizing the training by minimizing the effects of biases is considered the problem of regularization loss numerically expressed in Eq. 3:

$$\min_{\theta_f,\theta_c} \mathbb{E}_{\tilde{x}\sim P_{\tilde{X}}(\cdot)}\left[\mathcal{L}_c\left(y_{\tilde{x}}, C\left(f\left(\tilde{x}\right)\right)\right)\right] + \lambda M\left(b\left(X\right); f\left(x\right)\right) \quad (3)$$

Equation 3, θ_f denotes learning parameters or feature maps of the main learning model and θ_c represents the features at the classifier layer to map learned observation (data distribution) to class label, L_c describes mean square error loss and λ is the hyperparameter to balance the bias in the input data samples.

The proposed work introduces a novel regularization algorithm that can avoid the biased learning of the model and increase the detection performance with higher accuracy, even if the training data is heavily biased. Regularization is achieved by adding a penalty term, also known as a regularization loss, to the loss function during training. Hence in the proposed work, the algorithm will have knowledge about 'what not to learn' along with regularization 'what to learn' parameters. The architecture of the proposed learning model is presented in Fig. 3, which consists of two multi-layer Artificial Neural Networks (ANN). The study considers that the proposed model takes input data as $xi \in X$ and the corresponding output $y_x \in Y$, Also, the system design considers β as bias caused by the variables such as flow duration (f_d) and packet size (p_s) such that:

$$\left\{f_d, p_s\right\} \in X$$

where the objective function approximator can be represented as follows:

$$g\left(f\left(X\right)\right) \rightarrow Y$$

The above model $g(f(X))$ is the main learning network with one input layer that holds 86 input data variables, three hidden layers, each with 50 neurons, and one output layer with a single neuron. The second model, i.e., regularizer, is configured with three hidden layers with 50, 50, and 25 neurons, respectively. The output layer of this model contains a single neuron which predicts the optimal bias value. This model acts as a regularizer and is represented as $h(f(X))\rightarrow\beta$. The proposed learning model uses a sigmoid (σ) activation function and an Adam optimizer is used in the training to regulate the learning rate and reduction in the loss.

The overall architecture of the proposed DNN model can be described as follows:

$$model = h(x) \in g\left(f\left(X\right)\right) \quad (4)$$

$$g\left(f\left(X\right)\right) \rightarrow \left\{f\left(x_i\right) + g\left(x\right)\right\} \quad (5)$$

Equation 4-5, the *f(x)* and *g(x)* are the set of objective function approximator $g(f(X))$ to predict attack class. On the other hand, another function approximator *h(x)* takes intermediate values of *f(x)* to predict optimal bias (β). Subsequently, (β) is then backpropagated to the main function approximators $g(f(X))$. In this regard, when input x_i is given to the neural network, it first gives intermediate output such that: $f : x_i \rightarrow O_H^k$, where O is the output of the third hidden (H) layer neurons of the first function approximator, where hidden layer $H = (H_1, H_2, H_3)$ and k denotes the dimension of output layer (O).

The output O_H^k of the last hidden layer, i.e., H_3 is then processed via adversarial function approximator $h(x)$, which predicts optimal bias and uses backpropagation with inverse gradient by measuring the variation in the biases concerning variation in the error, thereby adjusting the learning parameters of the first function approximators towards improving the accuracy of the classification layer $g\left(O_H^k\right)$. The attributes of each function approximator are represented in the following equations:

$$f : X \rightarrow O_H^k \quad (6)$$

$$h : O_H^k \rightarrow \beta \quad (7)$$

$$g : O_H^k \rightarrow Y \quad (8)$$

The model design is carried out in such a way that it takes input data in the form of a vector and performs intrusion detection robustly in the real time scenario, even with unbiased data in the observation state. However, to accomplish such processes, both the function approximators need to be trained with the suitable feature vector. The next subsection describes the training process of the proposed I-IDM.

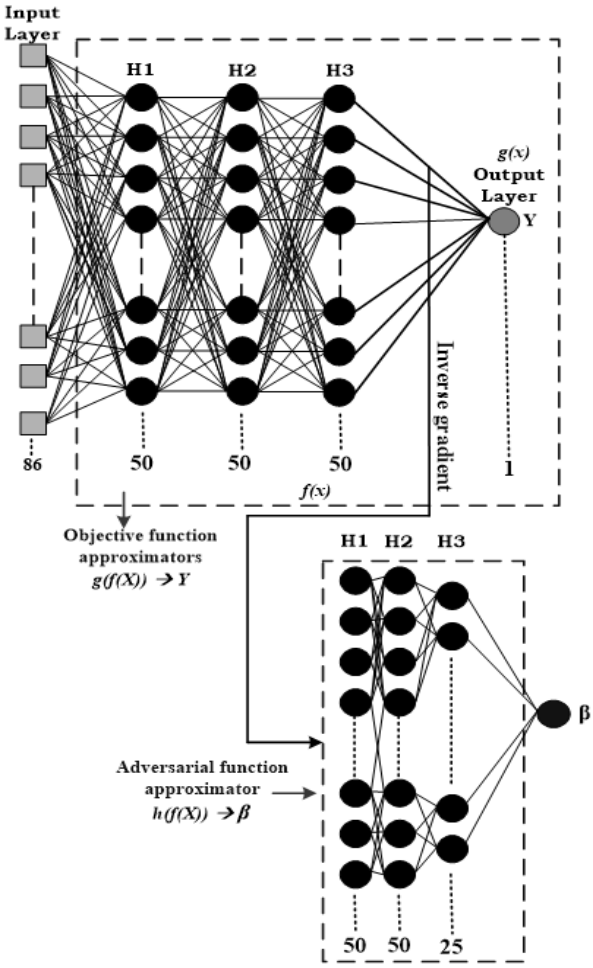

Fig. 3: Architecture of learning model with regularization mechanism

In order to train the proposed model, I-IDM first needs to separate the dataset into training and testing sets so that appropriate evaluation is carried out to justify its scope. The study considers a train-test split with a ratio of 80:20, respectively. Moreover, the training set (X_{Tr}) is used to perform model training and the testing set (X_{Ts}) is used to assess the performance of the trained model unbiasedly. Moreover, the training set also provides a scope for improving trained model performance by adjusting their hyperparameters. The characteristic of biased-dataset X_b with respect to X_{Tr} and X_{Ts} can be represented as follows in Eq. 9:

$$f\left(X_{Tr}^{b}\right);T''f\left(X_{Ts}^{b}\right);Y \cong \tag{9}$$

The above equation shows that model trained with biased data X_{Tr}^{b} will always lead to a biased model and when it is evaluated with biased testing data X_{Ts}^{b}, the result will be approximately equal to zero. Therefore, the proposed study focuses on minimizing loss on the intermediate data O_H^k of f:$X \in g(f(X))$ with the help of $h(f(X))$. Here, no other operation is carried out on the $g\left(O_H^k\right)$ since it takes the output of f: X as its input. Therefore, the training of the model is carried out as follows in Eq. 10:

$$L_\beta\left(O_H^k\right) = E_{\bar{X}}\left(\beta\left(H_u\right), h\left(O_H^k\right)\right) \tag{10}$$

where, L_β bias loss, $E_{\bar{X}}$ denotes loss function Mean Square Error (MSE). It is to be noted that the training process for both the function approximator and model is carried out differently. The first model f: X gets training based on the input feature of X_{Tr} and its output O_H^k further gets regularized through the second model $h(x)$ which gets trained with the value of the harmonic mean (H_u) computed using four different values of β as follows:

$$\beta_1 = \frac{present\ f_d}{hight\ f_d\ in\ attack\ packet} \tag{11}$$

$$\beta_2 = \frac{present\ f_d}{hight\ f_d\ in\ normal\ packet} \tag{12}$$

$$\beta_3 = \frac{present\ p_s}{hight\ p_s\ in\ attack\ packet} \tag{13}$$

$$\beta_4 = \frac{present\ p_s}{hight\ p_s\ in\ normal\ packet} \tag{14}$$

where, f_d is the flow data and p_s refers to packet size or length. The harmonic-mean H_u from Eq. 11-14 is computed as follows in Eq. 15:

$$H_u = \frac{n}{\frac{1}{\beta_1}+\frac{1}{\beta_2}+\frac{1}{\beta_3}+\frac{1}{\beta_4}} \tag{15}$$

where, n is equal to 4.

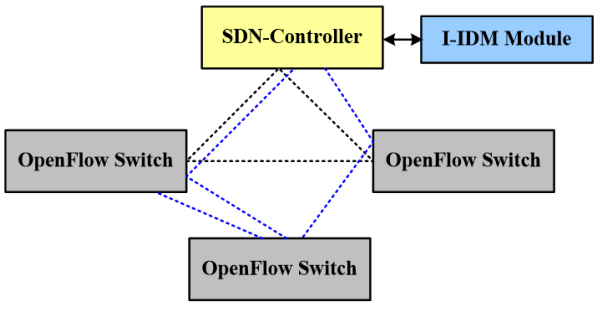

Fig. 4: Contextual architecture of I-IDM in SDN

Based on the traffic features, the optimal bias will be computed and training of the model can be done effectively to detect DDoS attacks unbiasedly without incurring any computational overhead on SDN components. The scope of implementation of the proposed I-IDM in the SDN controller is the SYN flood countermeasure module, which actively mitigates attacks at its origin, ensuring that "normal traffic flows" are further processed towards the controller. Figure 4 shows the implementation of I-IDM in the SDN system, where the proposed I-IDM is implemented as an upgraded security module in the SDN controller.

The SDN controller can monitor all OpenFlow switches and request all network traffic data to determine whether the network's traffic flow is normal or malicious. The proposed model IIDM uses the function of the SDN controller to get insight into the global view of the entire network to detect intrusions. The SDN controller uses a fixed time frame, based on which it requests to open the flow switch to provide the network flow status.

Results and Discussion

The proposed system discusses a technique to resist potential DDoS attacks on the SDN system using a deep learning approach. The resistivity mechanism discussed in the prior section is assessed using a simulation-based approach, while the obtained outcomes are discussed in this section. This section elaborates on the test environment considered for the study, along with illustrating the results obtained.

Performance Metrics

The proposed study considers three performance metrics Precision (P), Recall (R), and F-measure (F), to evaluate the performance of the model:

- Precision (P): It represents true detections over the false detection rate. The higher precision leads to a lower false alarm rate:

$$Precision = \frac{TP}{TP + FP} \tag{16}$$

- Recall (R): This metric presents measures of predicted intrusions vs. all intrusions presented in the dataset:

$$Recall = \frac{TP}{TP + FN} \tag{17}$$

- F1-score: This metric measures the weighted correlation of both *P* and *R*:

$$F1 = \frac{2(Recall \times Precision)}{(Recall + Precision)} \tag{18}$$

Test Environment

The implementation of the proposed study is carried out using a python programming language in an Anaconda distribution. The proposed IIDS scheme is validated against the CIC-DDoS2019 dataset to justify the scope of the proposed scheme. The performance of the proposed system is evaluated considering different test cases where different ML classifiers and deep learning models are implemented and executed over the same dataset, which is balanced using smote oversampling algorithm.

Table 2 presents the numerical outcome for the comparative analysis of the proposed system with existing machine learning techniques with smote.

Figure 5, demonstrates the performance of deep learning with smote and the proposed system (i.e., deep learning model with proposed regularization technique). The graph trend exhibits better performance achieved by the proposed system, Table 2.

Figure 6 is a comparative analysis of the proposed learning model with existing machine learning techniques KNN, Naïve Bayes, and deep learning with smote. The graph trend exhibits that the proposed model outperforms the existing machine learning technique KNN, Naïve Byes, and deep learning in precision rate, recall rate, and F1-score.

The reason behind achieving better performance by the proposed system is that the proposed model is trained with a better regularization mechanism, which provides better feature representation and leads to better attack class prediction in an unbiased manner. The proposed regularization mechanism makes the learning model generalize features unbiasedly, monitoring biased network behavior to prevent the model from overfitting. On the other hand, the generic deep learning technique outperforms the KNN and NB in terms of precision, recall, and F1-score. Furthermore, due to the immense amount of artificial data generated by smote, these classifiers also experience delays in training. In the case of the proposed work, the study does not focus on up-sampling the data; instead, it focuses on avoiding bias factors in the learning process of the model. The proposed scheme uses dual multi-layer ANN, it took a large training time, but training time is similar to generic deep learning and KNN. However, the NB classifier trained faster. Also, there is a good scope for optimizing the proposed learning-based regularization scheme for multi class classification problems, which we will do in our future work.

Table 2: Numerical outcome of classification models

	KNN + smote (%)	NB + smote (%)	Deep learning + smote (%)	Proposed (%)
Precision	89.37	88.76	91.52	96.71
Recall	90.97	86.62	90.65	97.14
F1-score	89.16	86.30	91.08	96.92

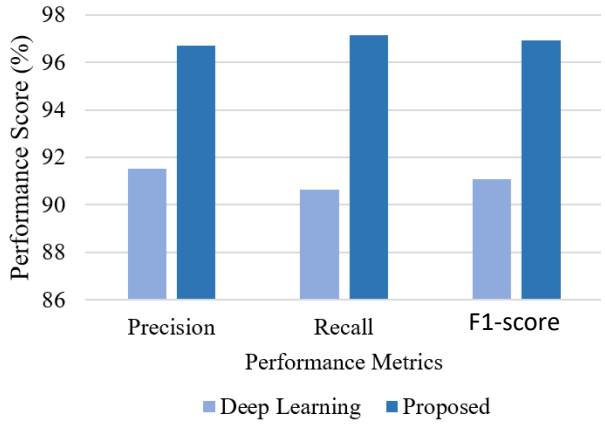

Fig. 5: Performance analysis of normal deep learning model and proposed regularization based

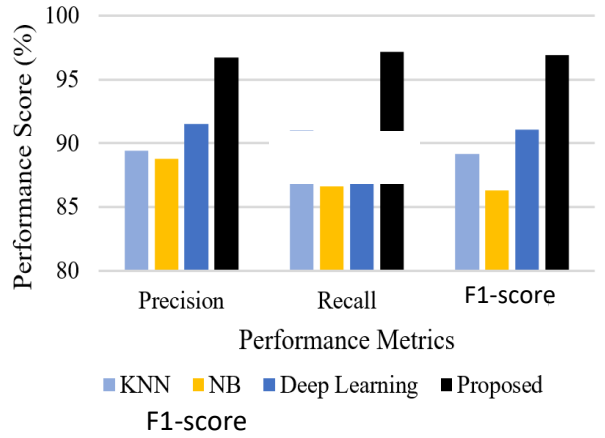

Fig. 6: Comparative analysis of the proposed system with other classification models

Conclusion

This study proposed study has implemented a deep learning model trained with a novel regularization mechanism for detecting network intrusion in the SDN oriented network system. The adopted dataset is associated with biases and the proposed study considered dataset biases as the problem of regularization loss. Therefore, an additional neural network as an adversarial model is employed in the implementation to perform unbiased feature learning and accurate prediction. Experimental results showed that the proposed regularization scheme achieves better prediction outcomes than the existing machine learning method implemented with the popular oversampling technique smote. Moreover, the proposed model has better scope for implementation in the real time scenario as it has been trained with a new dataset that contains signatures of modern attacks, making it suitable for real world scenarios such as SDN and IoT. In future work, the proposed model will be extended to detect multiple classes of attacks considering other recent datasets and deep learning-based classification models.

Acknowledgment

The authors of this manuscript would like to express their gratitude to the department of computer science and engineering, national institute of engineering, for their efforts in guiding in the context of the current research work with for the constructive feedback which improved the submission. No funding was received to assist with the preparation of this manuscript. The authors have no conflicts of interest to declare relevant to this article's content.

Funding Information

The authors have not received any financial support or funding to report.

Author's Contributions

Narender M: The experiment was carried out and completed. Revised and proofread the manuscript.

Yuvaraju B N: Guided the designed route and provided experimental guidance for this manuscript.

Ethics

This article is original and contains unpublished material. The corresponding author confirms that all other authors have read and approved the manuscript and no ethical issues are involved.

References

Aladaileh, M. A., Anbar, M., Hasbullah, I. H., Chong, Y. W., & Sanjalawe, Y. K. (2020). Detection techniques of distributed denial of service attacks on software-defined networking controller-a review. *IEEE Access*, *8*, 143985-143995. https://doi.org/10.1109/ACCESS.2020.3013998

Alamri, H. A., & Thayananthan, V. (2020). Bandwidth control mechanism and extreme gradient boosting algorithm for protecting software-defined networks against DDoS attacks. *IEEE Access*, *8*, 194269-194288. https://doi.org/10.1109/ACCESS.2020.3033942

Alao, O. D., Alimi, S., Kuyoro, S. O., Amanze, R. C., Adio, A. K., & Agbaje, M. O. (2022). An ensemble of Gaussian mixture model and support vector machines for network intrusion detection. *Journal of Computer Science*, *18*(9), 868-876. https://doi.org/10.3844/jcssp.2022.868.876

Alfrhan, A. A., Alhusain, R. H., & Khan, R. U. (2020, September). SMOTE: Class imbalance problem in intrusion detection system. In *2020 International Conference on Computing and Information Technology (ICCIT-1441)* (pp. 1-5). IEEE. https://doi.org/10.1109/ICCIT-144147971.2020.9213728

Alqarni, A. A., & El-Alfy, E. S. M. (2022). Improving Intrusion Detection for Imbalanced Network Traffic using Generative Deep Learning. *International Journal of Advanced Computer Science and Applications*, *13*(4). https://doi.org/10.14569/IJACSA.2022.01304109

Bagui, S., & Li, K. (2021). Resampling imbalanced data for network intrusion detection datasets. *Journal of Big Data*, *8*(1), 1-41. https://doi.org/10.1186/s40537-020-00390-x

Bannour, F., Souihi, S., & Mellouk, A. (2017). Distributed SDN control: Survey, taxonomy and challenges. *IEEE Communications Surveys & Tutorials*, *20*(1), 333-354. https://doi.org/10.1109/COMST.2017.2782482

Bedi, P., Gupta, N., & Jindal, V. (2020). Siam-IDS: Handling class imbalance problem in intrusion detection systems using siamese neural network. *Procedia Computer Science*, *171*, 780-789. https://doi.org/10.1016/j.procs.2020.04.085

Benarous, L., Kadri, B., & Bouridane, A. (2017). A survey on cyber security evolution and threats: Biometric authentication solutions. *Biometric Security and Privacy: Opportunities & Challenges in The Big Data Era*, 371-411. https://doi.org/10.1007/978-3-319-47301-7_15

Benzekki, K., El Fergougui, A., & Elbelrhiti Elalaoui, A. (2016). Software-Defined Networking (SDN): A survey. *Security and Communication Networks*, *9*(18), 5803-5833. https://doi.org/10.1002/sec.1737

Chica, J. C. C., Imbachi, J. C., & Vega, J. F. B. (2020). Security in SDN: A comprehensive survey. *Journal of Network and Computer Applications*, *159*, 102595. https://doi.org/10.1016/j.jnca.2020.102595

Cihan, A. T. A. Ç., & Akleylek, S. (2019). A survey on security threats and solutions in the age of IoT. *Avrupa Bilim ve Teknoloji Dergisi*, (15), 36-42. https://doi.org/10.31590/ejosat.494066

Dong, S., & Sarem, M. (2019). DDoS attack detection method based on improved KNN with the degree of DDoS attack in software-defined networks. *IEEE Access*, *8*, 5039-5048. https://doi.org/10.1109/ACCESS.2019.2963077

Dong, S., Abbas, K., & Jain, R. (2019). A survey on Distributed Denial of Service (DDoS) attacks in SDN and cloud computing environments. *IEEE Access*, *7*, 80813-80828. https://doi.org/10.1109/ACCESS.2019.2922196

El Mrabet, Z., Kaabouch, N., El Ghazi, H., & El Ghazi, H. (2018). Cyber-security in smart grid: Survey and challenges. *Computers & Electrical Engineering*, *67*, 469-482. https://doi.org/10.1016/j.compeleceng.2018.01.015

Gaffer, S. M., Yahia, M. E., & Ragab, K. (2012, December). Genetic fuzzy system for intrusion detection: Analysis of improving of multiclass classification accuracy using KDDCup-99 imbalance dataset. In *2012 12th International Conference on Hybrid Intelligent Systems (HIS)* (pp. 318-323). IEEE. https://doi.org/10.1109/HIS.2012.6421354

Haider, S., Akhunzada, A., Mustafa, I., Patel, T. B., Fernandez, A., Choo, K. K. R., & Iqbal, J. (2020). A deep CNN ensemble framework for efficient DDoS attack detection in software defined networks. *IEEE Access*, *8*, 53972-53983. https://doi.org/10.1109/ACCESS.2020.2976908

Han, T., Jan, S. R. U., Tan, Z., Usman, M., Jan, M. A., Khan, R., & Xu, Y. (2020). A comprehensive survey of security threats and their mitigation techniques for next-generation SDN controllers. *Concurrency and Computation: Practice and Experience*, *32*(16), e5300. https://doi.org/10.1002/cpe.5300

Hussain, S. N., & Singha, M. N. R. (2020). A survey on cyber security threats and their solutions. *Int. J. Res. Appl. Sci. Eng. Technol*, *8*(7), 1141-1146. https://doi.org/10.22214/ijraset.2020.30449

Khudhu, A. R., & Samsudin, K. (2022). IoT intrusion detection using auto-encoder and machine learning techniques. *Journal of Computer Science*, *18*(10), 904-912. https://doi.org/10.3844/jcssp.2022.904.912

Manjula, M., Venkatesh, & Venugopal, K. R. (2023). Cyber security threats and countermeasures using machine and deep learning approaches: A survey. https://doi.org/10.3844/jcssp.2023.20.56

Mbow, M., Koide, H., & Sakurai, K. (2022). Handling class Imbalance problem in Intrusion Detection System based on deep learning. *International Journal of Networking and Computing*, *12*(2), 467-492. https://doi.org/10.15803/ijnc.12.2_467

Novaes, M. P., Carvalho, L. F., Lloret, J., & Proença, M. L. (2020). Long short-term memory and fuzzy logic for anomaly detection and mitigation in software defined network environment. *IEEE Access*, *8*, 83765-83781. https://doi.org/10.1109/ACCESS.2020.2992044

Ouiazzane, S., Addou, M., & Barramou, F. (2021). A Multiagent and Machine Learning based Hybrid NIDS for Known and Unknown Cyber-attacks. *International Journal of Advanced Computer Science and Applications*, *12*(8). https://doi.org/10.14569/IJACSA.2021.0120843

Phan, T. V., Nguyen, T. G., Dao, N. N., Huong, T. T., Thanh, N. H., & Bauschert, T. (2020). DeepGuard: Efficient anomaly detection in SDN with fine-grained traffic flow monitoring. *IEEE Transactions on Network and Service Management*, *17*(3), 1349-1362. https://doi.org/10.1109/TNSM.2020.3004415

Ravi, N., & Shalinie, S. M. (2020). Learning-driven detection and mitigation of DDoS attack in IoT via SDN-cloud architecture. *IEEE Internet of Things Journal*, *7*(4), 3559-3570. https://doi.org/10.1109/JIOT.2020.2973176

Setiawan, B., Djanali, S., & Ahmad, T. (2020). Analyzing the performance of intrusion detection model using weighted one-against-one support vector machine and feature selection for imbalanced classes. *Int. J. Intell. Eng. Syst*, *13*, 151-160. https://doi.org/10.22266/ijies2020.0430.15

Stellios, I., Kotzanikolaou, P., Psarakis, M., Alcaraz, C., & Lopez, J. (2018). A survey of IoT-enabled cyberattacks: Assessing attack paths to critical infrastructures and services. *IEEE Communications Surveys & Tutorials*, *20*(4), 3453-3495. https://doi.org/10.1109/comst.2018.2855563

Wagner, T. D., Mahbub, K., Palomar, E., & Abdallah, A. E. (2019). Cyber threat intelligence sharing: Survey and research directions. *Computers & Security*, *87*, 101589. https://doi.org/10.1016/j.cose.2019.101589

Xu, Y., Sun, H., Xiang, F., & Sun, Z. (2019). Efficient DDoS detection based on K-FKNN in software defined networks. *IEEE Access*, *7*, 160536-160545. https://doi.org/10.1109/ACCESS.2019.2950945

Yan, Q., Yu, F. R., Gong, Q., & Li, J. (2015). Software-Defined Networking (SDN) and distributed denial of service (DDoS) attacks in cloud computing environments: A survey, some research issues and challenges. *IEEE Communications Surveys & Tutorials*, *18*(1), 602-622. https://doi.org/10.1109/COMST.2015.2487361