An Efficient Intrusion Detection Framework Based on Embedding Feature Selection and Ensemble Learning Technique

Fawaz Mokbal
Fan Gongxiu Honors College, Beijing University of Technology, China
fawaz@emails.bjut.edu.cn

Wang Dan*
Faculty of Information Technology, Beijing University of Technology, China
wangdan@bjut.edu.cn

Musa Osman
Faculty of Information Technology, Beijing University of Technology, China
msuliman@emails.bjut.edu.cn

Yang Ping
School of Economics and Management, Beijing Information Science and Technology University, China
yangping@bistu.edu.cn

Saeed Alsamhi
Athlone Institute of Technology, Ireland
Salsamhi@ait.ie

Abstract: *Network security has emerged as a crucial universal issue that affects enterprises, governments, and individuals. The strategies utilized by the attackers are continuing to evolve, and therefore the rate of attacks targeting the network system has expanded dramatically. An Intrusion Detection System (IDS) is one of the significant defense solutions against sophisticated cyberattacks. However, the challenge of improving the accuracy, detection rate, and minimal false alarms of the IDS continues. This paper proposes a robust and effective intrusion detection framework based on the ensemble learning technique using eXtreme Gradient Boosting (XGBoost) and an embedded feature selection method. Further, the best uniform feature subset is extracted using the up-to-date real-world intrusion dataset Canadian Institute for Cybersecurity Intrusion Detection (CICIDS2017) for all attacks. The proposed IDS framework has successfully exceeded several evaluations on a big test dataset over both multi and binary classification. The achieved results are promising on various measurements with an accuracy overall, precision, detection rate, specificity, F-score, false-negative rate, false-positive rate, error rate, and The Area Under the Curve (AUC) scores of 99.86%, 99.69%, 99.75%, 99.69%, 99.72%, 0.17%, 0.2%, 0.14%, and 99.72 respectively for abnormal class. Moreover, the achieved results of multi-classification are also remarkable and impressively great on all performance metrics.*

Keywords: *Network security, intrusion detection, ensemble learning, xgboost algorithm, features selection.*

Received February 18, 2020; accepted August 29, 2021
https://doi.org/10.34028/iajit/19/2/11

1. Introduction

As the developing technological revolution, besides the widespread acceptance of the Internet has grown, the world is becoming a global village. Therefore, the Internet and computer networks have become important prerequisite components of cutting-edge life indispensable. Further, these techniques are reserving appreciable and sensitive private information in various aspects, whether for organizations (e.g., email systems, e-commerce transactions, and much more) or individuals (e.g., online social activities, online shopping, and online gaming) as samples of an aspect of many ' 'people's lives.

This extravagant dependence along with precious information on networks is motived to cause underlying security problems, which for sure be to have an impact on our traditional activities and weaken the protection of our private info, as soon as attacks or intrusions occur. Such penetrations may conjointly cause heavy economic losses or perhaps irreplaceable damage. Therefore, network security, conjointly known as cybersecurity, has increased interest [14, 25]. Many security defense strategies, including firewalls, data encryption, user authentication, and other technologies, have been proposed [15]. However, when confronted with new and more sophisticated intrusion techniques, these traditional methods demonstrate a limited capability to detect intrusion and occasionally fail altogether [28]. Thus, Intrusion Detection Systems (IDSs) have received a lot of interest in the research domain [13], and considering as one of the paramount security issues in today's cyber-world.

The original unbalanced data, low detection rates, and high false alarms rates are challenges facing IDSs [14, 29], which greatly interest information security experts and researchers alike [23]. In addition, IDS demands a comprehensive benchmarking dataset containing network traffic up-to-date of normal and abnormal behaviors [19]. Furthermore, most proposed systems are either built based on out-of-date data or conducted on partials of the dataset being studied. That is, choosing a specific type of attack.

In this research, a robust and effective XGBoost-based intrusion-detection framework is proposed. The CICIDS2017 most up-to-date real-world intrusion dataset is used [22], including the most cutting-edge and common attacks and the benign samples. All dataset files are integrated into one dataset and derivative a uniform subset of features representing all attacks. The key contributions of this research are as follows:

- We proposed the IDS approach based on multi-datasets aggregation learning. Unlike most previous works, all data files in CICIDS2017 are integrated into one dataset include 2, 300, 825 samples, and all attacks are considered.
- We introduce an approach that focused on extracting features that could be calculated quickly, correctly based on the relative importance (i.e., Gain) method that analytically selects the optimal features uniformly and comprehensively representing all attacks rather than selected features for each attack separately.
- We proposed a predictive model (ensemble method) that takes the form of a set of decision models, including 300 models to build the intrusion-detection framework, instead of relying on one model. The framework is designed in a way to address several important requirements for machine learning model in real-world information security domains, i.e., extremely low false-positive rate, model interpretability and robustness to a potential intrusion.

The proposal IDS performance is evaluated in both multi-classification (8 class) and binary (2 class) problems. The proposed framework has successfully exceeded several evaluations on a big testing dataset and achieved remarkable and significant results with an accuracy overall, precision, detection rate (sensitivity), specificity, F-score, false negative rate, false positive rate, error rate of 99.90%, 99.90%, 99.97%, 99.90%, 99.90%, 5%, 0.2%, and 0.1%, respectively, in terms of the weighted average of multi-classification. Moreover, the achieved results of binary-classification are also remarkable. The evaluation results show that the strategy used with the IDS achieves noteworthy results in both the multi-classification problem and binary problem. Moreover, results show that the proposed IDS framework is able to handle the huge and imbalanced dataset problem effectively and efficiently.

The rest of this paper is organized as follows. Section 2 presents a brief background of IDS. Section 3 discusses the related work of the most recently proposed approaches in the intrusion detection domain using the CICIDS2017 dataset and highlights research gaps that our proposed framework focuses on. Section 4 presents the proposed intrusion detection framework's key specifics and the network traffic dataset used. Section 5 introduces the experimental results, discussion, and comparison with the previous studies, while section 6 presents a conclusion and highlights future work.

2. Intrusion Detection System

A remarkable variety of IDS approaches have been developed based on machine learning techniques that can be seen in [17].

The purpose of the IDS is to observe user activities and classify them as to whether they are normal or abnormal based on particular rules or models. Such a security system is ordinarily applied together with firewalls and complement for them [20]. In general, IDSs can be classified into two primary approaches based on the strategy they use to spot the attacks include:

- Misuse detection is well-known as a signature-based system, wherever each signature represents an activity pattern that matches to signified attack. The IDS recognizes the attack by trying to find those patterns in the records. However, misuse detection cannot become aware of the novel assaults or minor variations of better-known attacks.
- Anomaly detection works by investigating the activity transactions, ongoing traffic, or behavior for anomalies on networks that identify an intrusion. However, a demerit of this approach is that there is no precise method for determining normal behavior. In contrast, it has the potential to detect anonymous attacks; consequently, it has enticed growing attention in industry and research domains.

3. Related Work

Many studies were engaged to improve the performance of the IDS system in different ways. However, the vast majority of them are based on the out-of-date and incomprehensive dataset [11, 24]. Furthermore, the majority of them are still suffering from poor detection and/or high false alarms. However, we focus on those IDS using up-to-date real-world intrusion CICIDS2017 dataset for IDS system that includes the most cutting-edge common attacks along with the benign samples [4, 22].

The Ustebay *et al*. [26] proposed IDS for Distributed Denial-Of-Service (DDoS) using 70% of the DDoS dataset, which belongs to CICIDS2017. The ten important features are selected using Recursive Feature Elimination (RFE) and utilizing Random Forest (RF) and Multilayer Perceptron (MLP) for detection tasks. Their results obtained on binary classification with accuracy and Receiver Operating Characteristic (ROC) score of 91% and 97%. However, our proposed achievements on binary classification with accuracy and ROC score of

99.86% and 99.72%. Furthermore, our IDS applied to all attacks embedding in the dataset.

The Jiang *et al*. [12] proposed an application layer IDS to detect DDoS attacks (ALDDoS) that utilized 30.8K of the CICIDS2017 dataset. The result obtained with their proposed system is able to detect DDoS attacks with an accuracy rate of 99.8%. However, our proposed IDS framework achieved accuracy up to 99.99% to detect DDoS attacks. Furthermore, our proposed IDS targets all the attack families existing in the dataset.

A study in [1] proposed IDS by employ two feature dimensionality reduction methods, include Principle Component Analysis (PCA) and Auto-Encoder (AE). They applied different machine algorithms such as RF, Quadratic Discriminant Analysis (QDA), Bayesian Network, and Linear Discriminant Analysis (LDA) individually for the detection task. The RF achieved the best results on the CICIDS2017 dataset with accuracy, FP rate, and detection rate score of 99.50% 0.2% 98.5 respectively for binary-class with 59 features using PCA and the accuracy of 99.60% for multi-class. However, our proposed IDS results achieved an accuracy, FP rate, and detection rate score of 99.86%, 0.2%, and 99.75% for binary-class (abnormal) with 50 features and an accuracy score of 99.90% for multi-class.

In [27], Authors proposed IDS using multiple-SVM detectors that trained on features selected by genetic algorithm-based. The method relies upon that each detector is training to detect a specific attack class utilizing selected features by the GA. However, only a small part of CICIDS2017 dataset samples are utilized to estimate their IDS, and the results achieved for accuracy, precision, detection rate, and specificity score of 99.39 %, 99.43%, 96.04%, and 99.67%, respectively. However, our proposed IDS results achieved accuracy, precision, detection rate, and specificity score of 99.90%, 99.90%, 99.97%, and 99.90%, respectively. Furthermore, in this paper, almost all the samples included in the CICIDS2017 dataset are used.

The study presenting in [2] proposed DoS-IDS to detect denial of service. The Fisher Score method is utilized for feature selection. The Nearest Neighbor (KNN), Support Vector Machine (SVM), and Decision Tree (DT) algorithms are applied individually to use as the classifiers. The accuracy results achieved with SVM, KNN, and DT algorithms were 99.7%, 57.76%, and 99%, respectively. In contrast, our proposed IDS targets all the attack families existing in CICIDS2017, further, as shown within confusion matrix results, the achieved results 100% for DoS and also100% for DDoS attacks.

4. Research Methodology

In this research, an efficient intrusion detection framework is presented. The proposed detection framework is composed of four stages:

1. Data Preparation including (Multi-datasets aggregation and data wrangle).
2. Feature selection using an embedded method to determine the most relevant features for all attacks.
3. Handling Imbalance dataset.
4. Construct and train an ensemble-based XGBoost framework with hyper-parameter optimization.

The detection framework has undergone rigorously extensive experimentations of hyper-parameter optimization, including the numbers of models (trees), size of the tree (depth), learning rate, and row subsampling, much more. Figure 1 demonstrates the schematic detection framework proposed. More detailed information on the proposed IDS is in the coming subsections.

4.1. Data Preparation

The data preparation phase is a set of tasks that are very crucial before doing any analytical process or building a machine-learning model. Therefore, most of these tasks are taken into consideration to build the proposed IDS framework as follows.

4.1.1. Traffic Dataset

CICIDS2017 is an up-to-data dataset that contains 80 features includes the majority of common attacks developed by the Canadian Institute for Cybersecurity to overcome the existing gaps in the current datasets, such as out-of-date, lack of features, metadata, and a lack of diversity of known attacks. The dataset is built to fulfill the eleven criteria such as complete interaction, Complete Network configuration, complete traffic, labeled dataset, complete capture, available protocols, attack diversity, heterogeneity, feature set, and meta-data as described in [4, 22], which make it an accurate and the complete dataset to date. The implemented attacks in CICIDS2017 involve Brute Force (File Transfer Protocol (FTP) and Secure Shell Protocol (SSH)), Web Attack (BForce, Cross-Site Scripting (XSS) and Structured Query Language SQL (SQL) Inject.), DoS, Heartbleed, Infiltration, Botnet, and DDoS, etc., The details are showing in

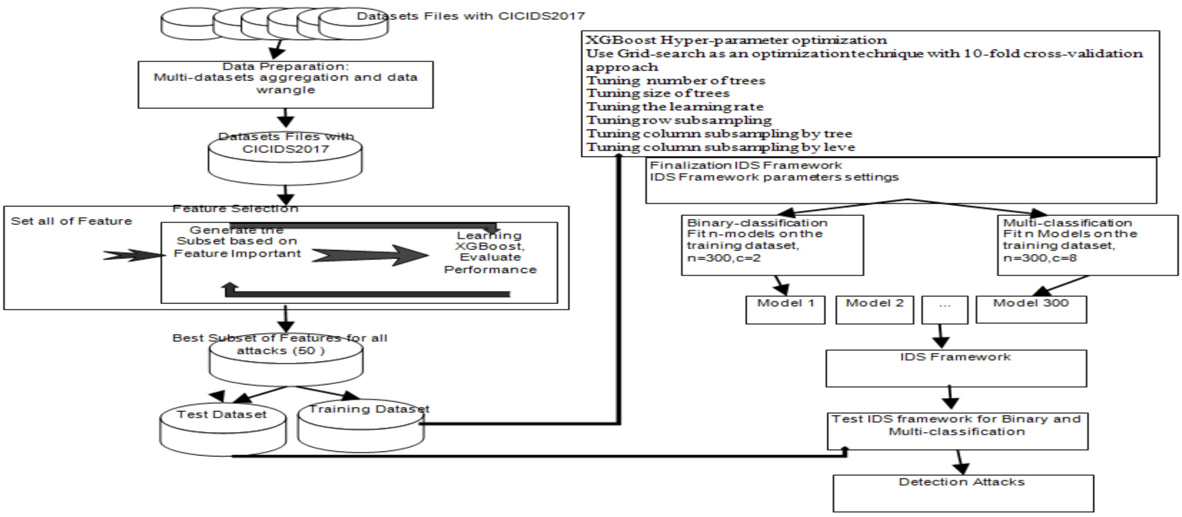
Datasets Files with CICIDS2017
Data Preparation:
Multi-datasets aggregation and data wrangle
Datasets Files with CICIDS2017
Feature Selection
Set all of Feature
Generate the Subset based on Feature Important
Learning XGBoost, Evaluate Performance
Best Subset of Features for all attacks (50)
Test Dataset
Training Dataset
XGBoost Hyper-parameter optimization
Use Grid-search as an optimization technique with 10-fold cross-validation approach
Tuning number of trees
Tuning size of trees
Tuning the learning rate
Tuning row subsampling
Tuning column subsampling by tree
Tuning column subsampling by leve
Finalization IDS Framework
IDS Framework parameters settings
Binary-classification
Fit n-models on the training dataset,
n=300, c=2
Multi-classification
Fit n Models on the training dataset,
n=300, c=8
Model 1
Model 2
...
Model 300
IDS Framework
Test IDS framework for Binary and Multi-classification
Detection Attacks

Figure 1. The schematic IDS framework.

Table 1. The CICIDS2017 dataset description.

CSV File name	Day	Traffic Type	Size	Description
Monday-WorkingHours.pcap_ISCX.csv	Monday	Benign only	529,918	Normal traffic activities (Excluded)
Tuesday-WorkingHours.pcap_ISCX.csv	Tuesday	Benign	432,074	Normal traffic activities
		FTP-Patator	7,938	The attacker tries to get the FTP login password using FTP Patator.
		SSH-Patator	5,897	The attacker tries to get the SSH login password using SSH-Patator
Wednesday-workingHours.pcap_ISCX.csv	Wednesday	Benign	440,031	Normal traffic activities
		DoS GoldenEye	10,293	generating numbers of unique and obfuscated traffic by using the GoldenEye tool to carry out a DoS attack
		DoS Hulk	231,073	Generating numbers of unique and obfuscated traffic by using the HULK tool to consume caching engine capacity on a webserver to carry out a DoS attack
		DoS Slowhttptest	5,499	The attacker intentionally opens numerous HTTP connections to the same server exploiting the HTTP-Get request to carry out a DoS attack.
		DoS slowloris	5,796	Denial of service attack executed by using the SlowLoris tool.
		Heartbleed	11	Authorizing access to valuable information by reading the memory of the systems protected by the vulnerable versions of the OpenSSL.
Thursday-WorkingHours-Morning- WebAttacks.pcap_ ISCX.csv	Thursday	Benign	168,186	Normal traffic activities
		Web Attack – Brute Force	1,507	An attacker using trial-and-error to get privilege information (e.g., password and Personal Identification Number (PIN))
		Web Attack – Sql Injection	21	SQL Injection includes the crafting of users' inputs so as to carry out actions beyond the intended function of a web application
		Web Attack – XSS	652	Injecting malicious scripts into otherwise trusted and benign websites to stolen users ' sensitive information.
Thursday-WorkingHours-Afternoon-Infilteration.pcap_ISCX.csv	Thursday	Benign	288,566	Normal traffic activities
		Infiltration	36	Access to the networked system data by infiltrating and gain full authorization through uses infiltration methods and tools.
Friday-WorkingHours-Morning.pcap_ISCX.csv	Friday	Benign	189,067	Normal traffic activities
		Botnet	1,966	The attacker utilizes Trojans to break down the security of several target machines, taking control of these machines and manage remotely all machines in the network of Bot.
Friday-WorkingHours-Afternoon-PortScan.pcap_ISCX.csv	Friday	Benign	127,537	Normal traffic activities
		PortScan	158,930	gathering information about victim machine, and execute service by dispatching packets with different destination ports
Friday-WorkingHours-Afternoon- DDos.pcap_ISCX.csv	Friday	Benign	97,718	Normal traffic activities
		DDoS	128,027	using multiple machines in a different location which operate together to attack one target machine

4.1.2. Data Integration and Handling Ambiguous Feature Values

The process is mainly aimed to append several attacks found in different and separate dataset files (Comma-Separated Values (CSV) files) that have the same features by combining them as one big dataset. Accordingly, extracted a significant subset of the most relevant and applicable features that cover all attacks simultaneously. We integrated seven CSV files (Tuesday to Friday) that include attacks and benign, while excludes one file (Monday), including only benign samples. As a result, (2, 300, 825) samples as one dataset file including 1, 743, 179 benign and 557, 646 attack samples were Integrated. Ambiguous feature values existing in a dataset such as NaN and infinite values are handled by transferring their values to the mean value of the feature. Furthermore, feature values within the dataset that have a very variance between the minimum and maximum values are normalized. Min-max scaling is used to replace every value in features such as 'Flow Duration', 'Flow Bytes/ s', 'Fwd Packet Length Std', 'Flow Packets/s', 'Flow IAT 'Mean', and 'Flow', with a new value using Equation (1), to get the values within the range [0, 1].

$$x_{sc} = \frac{(x - x_{\min})}{(x_{\max} - x_{\min})} \tag{1}$$

Where x_{sc} is a new value after scaling, x is the original feature value, $x_{\min}$ is the feature's minimum value, and $x_{\max}$ is the feature's maximum value.

4.2. Feature Selection

Due to the massive amount of data employed over the real-time network, intrusion detections are a challenging and costing task. Therefore, having the right features are crucial for improving model performance. Thus, the feature selection methods come as efficient ways to cut back computation time and complexness by select the most relevant features subset, casting off the irrelevant and surplus features from the dataset to establish an efficient learning method, and inferring realistic patterns. Besides, the system gains other benefits such as reducing the overfitting problem, improving accuracy, reducing training and testing time, and much more. The feature selection process is assessed according to two major phases that are generation procedure and assessment function [9]. The first phase produces the subset of features, while a second phase looks after evaluating this produced feature subset. Three methods are used to evaluate a subset of features involve the filter method, the wrapper method or hybrid method [30].

In this research, our main purpose is to discover and extract valuable information (features) from large-scale network traffic datasets, so that the proposed ensemble-based intrusion detection framework will attain lower generalization faults and minimal false alarm along with significant accuracy and detection rates. Thus, they facilitate the computational complexity and improve the performance of the intrusion detection framework simultaneously. For that, a hybrid approach combines tree-based relative importance and model-based feature selection methods to deal with feature selection over huge and imbalanced data. It consists of two basic steps:

1. An ensemble-based Gradient Boost Trees is utilized as a filter method by evaluating each feature upon its importance. The goal is to reduce the effect of the noise as well as minimize the computational cost during the warping stage. The relative importance of a feature calculates based on the number of times a feature is chosen for splitting, weighted by the squared improvement to the tree as a result of each split, as in Equation (2) [6].

$$\hat{I}_j^2(T) = \sum_{t=1}^{J-1} \hat{\imath}_t^{\,2}(v_t = j) \tag{2}$$

Where the sum is over the non-terminal nodes t of the J-terminal node tree T, v_t is the splitting variable related to node t, and $\hat{\imath}_t^2$ is the corresponding to improvement criterion in squared error in space of the regression tree R as a result of the split into two sub-regions R_l, R_r can be defined as Equation (3).

$$t^2(R_l, R_r) = \frac{w_l w_r}{w_l + w_r}(\bar{y}_l - \bar{y}_r)^2 \tag{3}$$

Where $\bar{y}_l, \bar{y}_r$ are the left and right child's reacts means and w_l, w_r are summation of the corresponding weights. Having $\{T_m\}_1^M$ decision trees, acquired through boosting, the generalization can be averaged over all of the trees as Equation (4).

$$\hat{I}_j^2(T) = \frac{1}{M}\sum_{m=1}^{M} \hat{\imath}_t^{\,2}(T_m) \tag{4}$$

In case multi-classification (N-classes), there are N logistic (regression) functions $\{T_{nm}(x)\}_{n=1}^N$, where each has a sequence of M trees. Like that, the generalization be as Equation (5).

$$\hat{I}_{jn}^2(T) = \frac{1}{M}\sum_{m=1}^{M} \hat{\imath}_t^{\,2}(T_{mn}) \tag{5}$$

Where T_{mn} is the tree crated for the n^{th} class at epoch/iteration m. The relevance of the predictor variable x_j in separating class n from the other classes can be interpreted as a relative contribution I_{jn}. The aggregate relevance of x_j can be gained by averaging x overall classes as Equation (6).

$$\hat{I}_j = \frac{1}{N}\sum_{k=1}^{K} \hat{I}_{jn} \tag{6}$$

2. The important features score are used as input to the model-based feature selection through the warping method. This way decreases the computational requirements by warp method especially we deal with big data.

The model-based feature selection is applied by using a meta-transformer method along with the XGBoost estimator for evaluation. The dataset therefore, is reconstructed into a subset of selected features by using importance scores as thresholds within a model. Hence, different subsets of features are consistently selected and evaluated based on its significance in model performance (i.e., starting with all and ending with essential features subsets). Therefore, it essentially allowed us to evaluate every feature subsets in the dataset based on its real contribution to model performance. Figure 2 shows the best subset of feature.

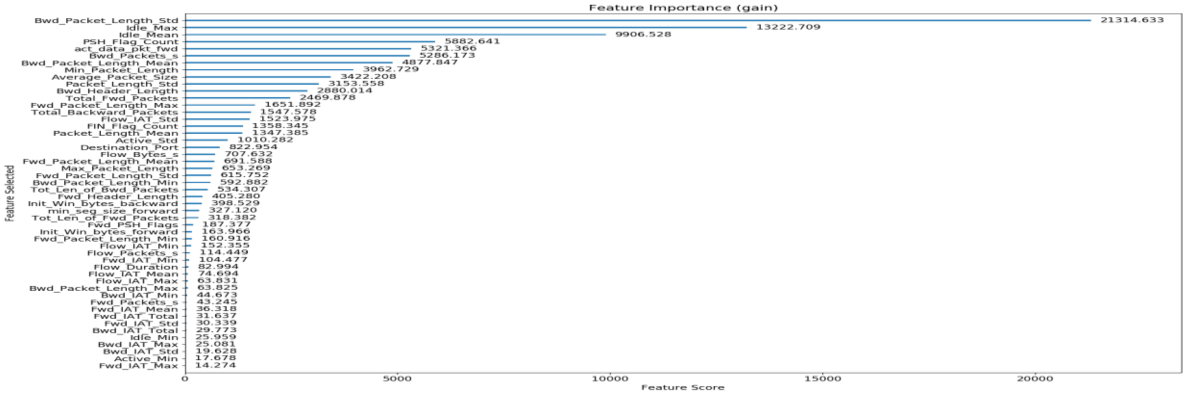

Figure 2. Best subset of features with a gain score.

4.3. Handling Imbalance Data

An unbalanced dataset is a crucial issue for classification algorithms causing a weak classifier performance [18]. The CICIDS2017 dataset are originally imbalanced. Furthermore, during the data integration process in the previous section, all-benign samples of each dataset file are combined into one class, which leads to forming the vast majority class. To handle the imbalanced dataset there are two main approaches:

- Resampling method that handling imbalanced data by resampling original data.
- The ensemble method, which based on modify existing classification algorithms to make them fitting for unbalanced datasets.

Fundamentally, the resampling approach includes undersampling and oversampling methods. The aim of this approach is to balancing classes by either decrease the distribution of the majority class as in the undersampling method or increasing the distribution of the minority class as in the oversampling method. However, each method has its own set of problematic repercussions that could potentially obstruct learning [5, 10, 16]. Eliminating examples from the majority class may lead to the classifier miss critical principles pertaining to the majority class. In the case of oversampling, appending replicated data to the original dataset, leading to multiple instances of certain samples become “tied,” resulting in overfitting [18]. On the other approach, the aim of the ensemble technique is to improve the performance of single classifiers through constructing numerous two-level classifiers from the unique records and thereafter aggregate their predictions [7].

In this research, two-step have been taken to handling imbalance data as follows:

- Classes combination, that is, merge a few minority attack classes having semi-characteristics and behaviour.
- Adopt ensemble XGBoost learning method.

In the first step, the DoS GoldenEye, DoS Hulk, DoS Slowhttptest, DoS slowloris have been combined into one DOS class. Similarly, Web Attack-Brute Force, Web Attack-Sql Injection, Web Attack-XSS are combined into one class called Web Attack. Moreover, the FTP-Patator, SSH-Patator are also combined. In the end, eight classes are adopted (i.e., Normal, WebAttack, Infiltration, BrutwForc, Dos, Botnet, PortScan, and DDoS). In the second step, the first classifier is built on the training dataset to predict the instances using the XGBoost algorithm. In the next iteration/epoch, the new classifier focuses on those wrongly classified instances in the previous round, and places more weights on them. This procedure is implemented through construct the first classifier to expect the samples. Then, it calculates the log loss function and uses this loss to construct an improved classifier in the second iteration. At each iteration, the residual of the loss function is calculated using the gradient descent approach to become the new objective variable for the next epoch/iteration.

4.4. Construct the Model

Applying multi-machine learning algorithms aka ensemble learning allows for creating a set of hypotheses for a particular problem. Accordingly, the hypotheses of each detector are amalgamated to produce a collective result, leading to a stronger generalization capability in comparison to individual base learners [17]. One of the advanced ensemble boosting techniques that use homogeneous base learners is XGBoost. The multiple learners of the same machine-learning algorithm (i.e., decision tree) are employed to establish a fixed of hypotheses over different sub-instances of the same training dataset. XGBoost is an open-source optimized gradient boosting system, crated to be tremendously efficient, flexible, accurate, and smooth to use [3, 20].

In this research, a scalable system is implemented using the XGBoost-based ensemble learning technique. An ensemble-learning introduces a systematic resolution to aggregate prognosticative power of multiple classifiers (learners) to obtain sustainable results instead of relying on a single classifier, thus, establishing a powerful and unique model. The proposed model includes multiple weak learners. Each weak learner is a decision tree. The trees are constructed in sequential form, such that each following tree objectives to scale back the faults of the preceding tree. Every tree learns from its predecessors and modernizes the residual faults. Hence, the tree that grows next in the sequence will learn from a modernized version of the residuals. Using XGBoost, we trained an intrusion detection framework to include multiple learners; at each iteration, one learner (i.e., tree) is added to the intrusion framework until it reaches the maximum number of predefined trees.

However, establishing the most effective detector model using an XGBoost algorithm requires parameter tuning [3]. Thus, we subdivided the experimentations to tune model parameters using two steps. Initially, we started with the default configuration using the k-fold cross-validation method as a baseline estimate of the proposed detection framework on the training dataset to establish the basis of the estimation of the proposed detection framework. We furnished a dictionary of hyper-parameters to evaluate in the param-grid argument as a map of the name parameter call and an array of values to inspect. These parameters are the number of estimators (trees) within values from 100 to 700 with an increment of 100 at every reestablishment, size of estimators (max-depth) with values of (3, 5, 7, and 9), learning rate within values of (0.001, 0.01, 0.1, 0.2, and 0.3), row subsampling (subsample) within range values of 0.1 to 1.0, and column subsampling within values between 0.1 and 1.0 incremented by 0.1. Besides, all tuning parameters are applied along with the logloss function as the calibration statistic function to be minimized for training the model.

We found the best number of trees is 300, with a depth value of 3 and a learning rate of 0.1. At the same time, row subsampling and column subsampling are 0.30 and 1.0, respectively. These parameters give us a vision that the detector model is less computational complexity and lighter (n_estimators=300 and depth =3) and therefore faster. These advantages of the detector model will discuss in the results section.

5. The Results and Discussion

In this section, performance evaluation criteria used for evaluating the proposed framework, the results of two experiments (multi-classification and binary classification) obtained by the proposed methodology with its discussion are presented. At the end, a comparison of the proposed framework with some recently proposed methods is also presented as given below.

5.1. Performance Evaluation Criteria

In this research, nine performance metrics to estimate the efficiency of the proposed classifier are applied. The detailed derivation of the selected performance metrics based on the confusion matrix [8] are formulated from (7) to (15).

$$\Pr ecision = \frac{TP}{(TP + FP)} \tag{7}$$

$$Detection_{Rate}(DR) = \frac{TP}{(TP + FN)} \tag{8}$$

$$Specificiy = \frac{TN}{(TN + FP)} \tag{9}$$

$$FP_{Rate} = \frac{FP}{(TN + FP)} \tag{10}$$

$$FN_{Rate} = \frac{FN}{(TP + FN)} \tag{11}$$

$$F - Score = 2\left(\frac{Detection_{Rate} \times \Pr ecision}{DetectionRate + \Pr ecision}\right) \tag{12}$$

$$Accurcy_{overall} = \frac{(TP + TN)}{(TP + TN + FP + FN)} \tag{13}$$

$$Error_{Rate} = \frac{(FP + FN)}{(TP + TN + FP + FN)} \tag{14}$$

$$AUC = \frac{1}{2}\left(\frac{TP}{TP + FN} + \frac{TN}{TN + FP}\right) \tag{15}$$

5.2. Experiments Result

The final detector framework is configured to attain optimal results after Hyper-parameters calibrating, as showing in Table 2. The entire experiments are executed on LinuxMint-19-tara operating platform, Kernel 4.15.0-42-generic, with 16 GB RAM, Intel Xeon CPU E-5-2620 v3@2.40GHz, GPU NIVIDA (Quadro K220). The XGboost version is 0.82, and the Python framework version is 3.6.7. The dataset has been split randomly and separately into two parts with a partition ratio of 70%: 30% for training and testing sets. The samples of 1, 610, 577 are allocated for detector training, and the samples of 690, 248 are allocated for detector testing. All detector parameters are executed along with log loss and error functions as the objectives to optimize during training our detection framework. Figure 3 shows the log loss of training and testing of intrusion detector framework, while Figure 4 shows the error of intrusion detector. Further, the first and last models (trees) in the proposed framework are shown in Figures 5 and 6 for deep insight.

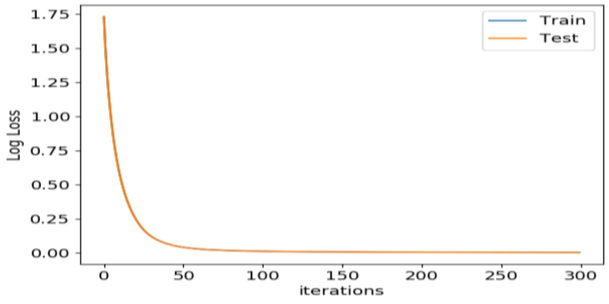

Figure 3. Log loss of intrusion detection framework.

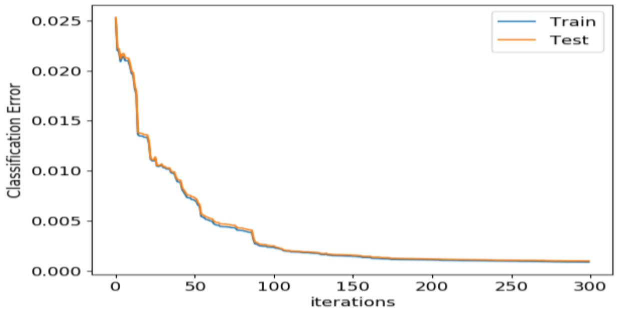

Figure 4. Error of intrusion detection framework.

Table 2. Parameters of intrusion detector framework.

Parameters	Values	Parameters	Values
n_estimators	300	max_depth	3
learning_rate (eta)	0.1	gamma	0
Subsample	0.30	colsample_bytree	1.0
Objective (Multi-class)	multi:softmax	eval_metric	mlogloss/merror
Objective (Binary-class)	binary: logistic	eval_metric	logloss/error

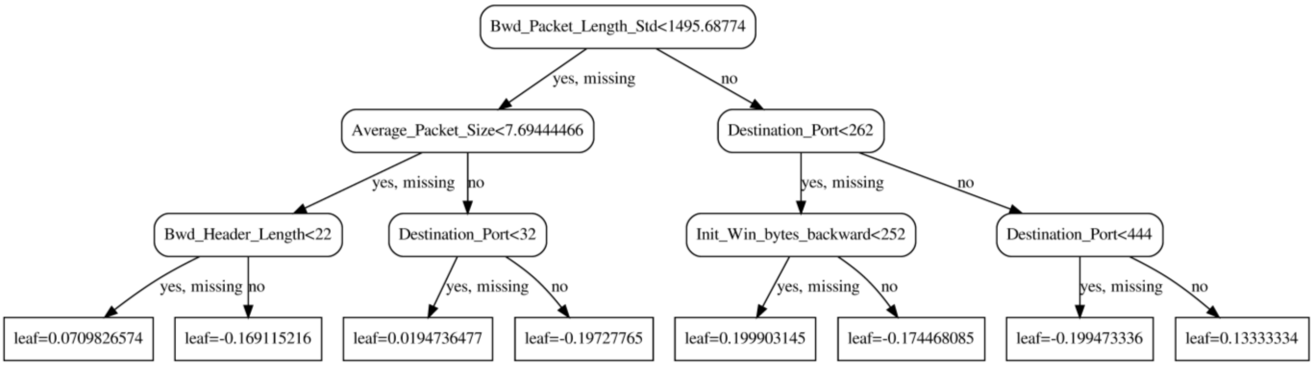

Figure 5. The model (tree) No.1 in the proposed framework.

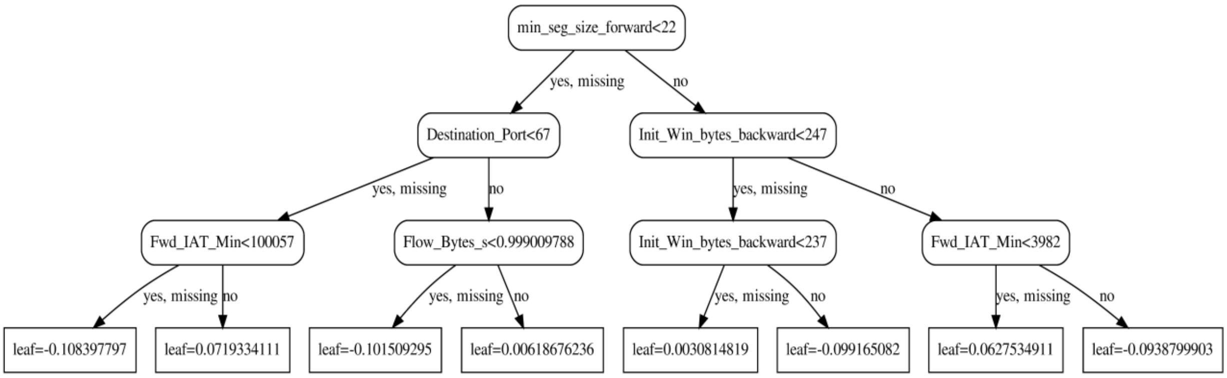

Figure 6. The model (tree) No.300 in the proposed framework.

5.2.1. Multi-Classification Result

For the multi-classification experiment, the objective argument is set to multi:softmax, and evaluation metric is defined as mlogloss, while the num_class parameter is 8. As a result of estimating the proposed detection framework on the testing dataset, the overall accuracy of 99.90% is achieved. However, extensive analyses have been done with the eight performance metrics to obtaining comprehensive results, as shown in Table 3. Furthermore, a weighted averaging metric is calculated to consider label unbalance, where the occurrence ratio is considered in the calculation. We achieved an accuracy overall, precision, detection rate (sensitivity), specificity, F-score, FN rate, FP rate, error rate of 99.90%, 99.90%, 99.97%, 99.90%, 5%, 0.2% and 0.1%, respectively in terms of the weighted average.

These results established the efficiency and effectiveness of the proposed detection framework with significant perfection and harmony of performance on multiple measurements to multiple classes. Besides, an in-depth experimental evaluation and analyses using various performance evaluation metrics for each class are shown in Table 3. Furthermore, Figure 7 clearly shows the efficacy of the proposed detector alongside the perfect statistical analysis based on the confusion matrix.

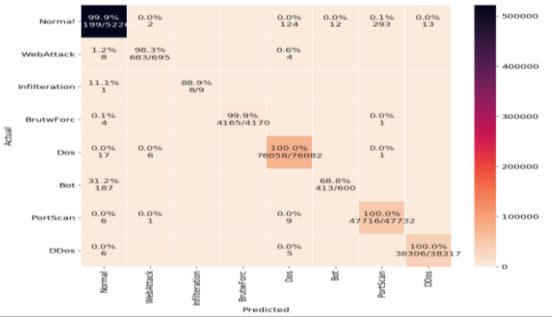

Figure 7. Confusion matrix with a statistical illustration for multiple-classes.

5.2.2. Binary-Classification Result

Although the proposed detection framework has a perfect performance in multi-class, we conducted another experiment on two classes (i.e., benign or attack) to strictly verify that the proposed detection framework fulfills a perfect performance in various cases. The dataset were configured entirely to form two classes include 1.743,179 samples for the normal class and 557,646 samples for the abnormal class. The dataset have been separated randomly into the training dataset, including 1,610,577 samples, and the testing dataset includes 690, 248 samples.

The achieved results of binary-classification have been remarkable and impressively great with an accuracy overall, precision, detection rate (sensitivity), specificity, F-score, false negative rate, false positive rate, error rate, and AUC scores of 99.86%, 99.69%, 99.75%, 99.69%, 99.72%, 0.17%, 0.2%, 0.1% and 99.72 respectively for abnormal class. More details are presented in Table 4, and the confusion matrix with statistical analysis is shown in Figure 6. Further, Figure 9 shows the ROC of the proposed IDS framework's performance, which is considered the most significant estimation metric for evaluating IDS performance. The area under the curve is 99.72%, which establishes the advantages and skills of the proposed IDS framework.

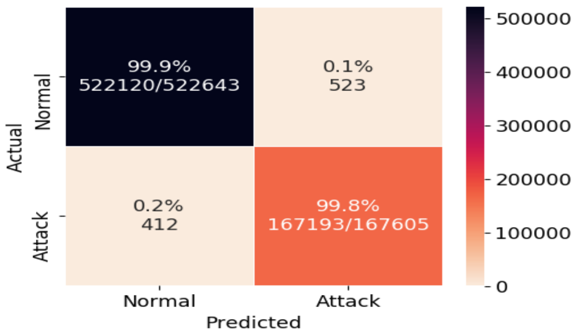

Figure 8. Confusion matrix with a statistical illustration for binary-classes.

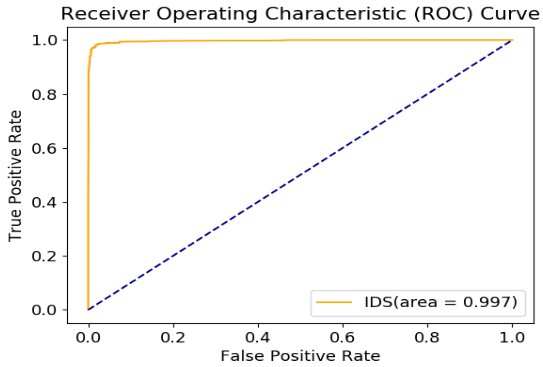

Figure 9. Area under the ROC carve.

5.3. Compression with Previous Methods

To clarify the advantages of the proposed detection model, the most freshly proposed approaches within the IDS domain that use the CICIDS2017 dataset are selected to conduct comparisons, as shown in Table 5. Although the results in Table 5 clearly indicate that our proposed framework continually outperforms different detection approaches in all different measurements, the results in Table 5 provide a brief overview estimation of quantitative measures between our proposed framework and the other currently proposed approaches. Thereby, this comparison might additionally have limitations. We cannot make an absolute assertion that our proposed IDS performs higher when compared to other IDSs within the context of intrusion detection. However, consistent with the consequences proven in Table 5, our proposed intrusion detection framework has robust confront benefits and advantages in the intrusion detection domain.

Table 3. Results of intrusion detection framework for multi-classification.

Class	Precision	Detection Rate	Specificity	F-score	FN Rate	FP Rate	Error Rate
Normal	0.999562	0.999151	0.998634	0.999356	0.001	0.0014	0.000975
WebAttack	0.986994	0.982734	0.999987	0.984859	0.017	0.0000	0.00003
Infilteration	1.000000	0.888889	1.00000	0.941176	0.111	0.0000	0.000001
BrutwForc	1.000000	0.998801	1.00000	0.999400	0.001	0.0000	0.000007
Dos	0.998136	0.999685	0.999769	0.998910	0.000	0.0002	0.000240
Botnet	0.971765	0.688333	0.999983	0.805854	0.312	0.0000	0.000288
PortScan	0.993856	0.999665	0.999541	0.996752	0.000	0.0005	0.000451
DDos	0.999661	0.999713	0.999980	0.999687	0.0003	0.0000	0.000035
Weight -avg	**0.998981**	**0.998986**	**0.999737**	**0.998962**	**0.055**	**0.0021**	**0.0010**

Table 4. Results of intrusion detection framework for binary classification.

Class	Precision	Detection Rate	Specificity	F-score	Accuracy Overall	FN Rate	FP Rate	Error Rate	AUC	Training Time (M)	Test Time (M)
Attack	0. 9969	0. 9975	0.9969	0. 9972	0.9986	0.0017	0.0020	0.0014	0.9972	27.72	0.25
Normal	0. 9992	0. 9990	0.9992	0. 9991							
Macro-avg	0.9980	0.9983	0.9981	0.9982							
Weight -avg	0.9986	0.9986	0.9985	0.9986							

Table 5. Result comparison proposed detection framework with other previous methods.

Detectors	Attack type	Classification Type	Accuracy Overall	Precision	Detection Rate	Specificity	F-score	FP Rate	FN Rate	Error Rate	AUC
Ustebay *et al*. [26]	DDoS	Binary	91.00 %	N/A	N/A	N/A	N/A	N/A	N/A	N/A	97.00 %
Jiang *et al*. [12]	DDoS	Binary	99.23 %	99.87 %	99.60 %	N/A	99.41 %	N/A	N/A	N/A	N/A
Abdulhammed *et al*. [1]	All	Binary	99.50 %	N/A	98.50 %	N/A	99.60 %	0.20 %	N/A	N/A	N/A
Abdulhammed *et al*. [1]	All	Multi	99.60 %	96.50 %	99.60 %	N/A	N/A	1.00 %	N/A	N/A	N/A
Vijayanand *et al*. [27]	All	Multi	99.39 %	99.43 %	96.04 %	99.67 %	N/A	0. 53 %	2.43 %	N/A	N/A
This paper	**All**	**Binary**	**99.86 %**	**99.69 %**	**99.75 %**	**99.69 %**	**99.72 %**	**0.2 %**	**0.17 %**	**0.14 %**	**99.72 %**
This paper	**All**	**Multi**	**99.90 %**	**99.90 %**	**99.97 %**	**99.90 %**	**99.90 %**	**5.5 %**	**0.2 %**	**0.10 %**	**N/A**

6. Conclusions

An intrusion detection system using the AI technique is one of the crucial solutions to cybersecurity attacks. However, the challenge of constantly improving accuracy, detection rate, and minimal false alarm of IDS is ongoing. The reliability and quality of the training dataset highly affect the performance of such systems. This research developed state-of-the-art and sophisticated cyber-defense approaches using the latest machine learning technique using XGBoost and an embedded feature selection method. Further, the best uniform feature subset is extracted using the up-to-date real-world intrusion dataset CICIDS2017 for all attacks. The proposed detection method was evaluated in both multi-classification and binary problems using a huge unbalanced dataset. Numerous analyses were performed to evaluate the proposed framework at various stages with diverse and comprehensive measurements.

The experiments results on the test dataset exhibited that our proposed detection framework attains a powerful performance over the nine different measurements used. When as compared to other freshly proposed approaches related to intrusion detection problems, our proposed detection framework demonstrated robust benefits, advantages, and capability to be extraordinarily competitive. Furthermore, it also has less complexity and able to process large amounts of data, which makes it easy to deploy and gain additional advantages compared to other approaches. However, this strong performance, capability, and efficiency of our proposed framework could be additionally advanced to deal with the massive data traffic in real-time; therefore, we will be considered this issue in our future work.

References

[1] Abdulhammed R., Musafer H., Alessa A., Faezipour M., and Abuzneid A., "Features Dimensionality Reduction Approaches for Machine Learning Based Network Intrusion Detection," *Electronics*, vol. 8, no. 3, pp. 332, 2019.

[2] Aksu D., Üstebay S., Aydin M., and Atmaca T., "Intrusion Detection With Comparative Analysis of Supervised Learning Techniques and Fisher Score Feature Selection Algorithm," *in Proceedings of International Symposium on Computer and Information Sciences*, pp. 141-149, 2018.

[3] Chen T. and Guestrin C., "XGBoost: A Scalable Tree Boosting System," *in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, pp. 785-794, 2016.

[4] CIC, "Intrusion Detection Evaluation Dataset (CICIDS2017)," *Canadian Institute for Cybersecurity*, https://www.unb.ca/cic/datasets/ids-2017.html, Last Visited, 2019.

[5] Drummond C. and Holte R., "C4.5, Class Imbalance, and Cost Sensitivity: Why Under-Sampling Beats Over-Sampling," *Workshop on Learning from Imbalanced Datasets II, ICML, Washington DC*, pp. 1-8, 2003.

[6] Friedman J., "Greedy Function Approximation: A Gradient Boosting Machine," *The Annals of Statistics*, vol. 29, no. 5, pp. 1189-1232, 2001.

[7] Galar M., Fernandez A., Barrenechea E., Bustince H., and Herrera F., "A Review on Ensembles for The Class Imbalance Problem: Bagging-, Boosting-, and Hybrid-Based Approaches," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 4, pp. 463-484, 2012.

[8] Gareth J., Daniela W., Trevor H., and Robert T., *An Introduction to Statistical Learning with Applications in R*, Springer, 2013.

[9] Ghanem W. and Jantan A., "Novel Multi-Objective Artificial Bee Colony Optimization for Wrapper Based Feature Selection in Intruction Detectoin," *International journal of advance soft computing applications*, vol. 8, no. 1, pp. 70-81, 2016.

[10] Ivanciuc O., "Weka Machine Learning for Predicting the Phospholipidosis Inducing Potential," *Current Topics in Medicinal Chemistry*, vol. 8, no. 18, pp. 1691-1709, 2008.

[11] Jayakumar K., Revathi T., and Karpagam S., "Intrusion Detection Using Artificial Neural Networks with Best Set of Features," *The International Arab Journal of Information Technology*, vol. 12, no. 6A, pp. 728-734, 2015.

[12] Jiang J., Yu Q., Yu M., Li G., Chen J., Liu K., and Huang W., "ALDD: A Hybrid Traffic-User Behavior Detection Method for Application

Layer DDoS," *in Proceedings of 17th IEEE International Conference on Trust, Security and Privacy in Computing and Communications/12th IEEE International Conference on Big Data Science and Engineering*, New York, pp. 1565-1569, 2018.

[13] Liao H., Lin C., Lin Y., and Tung K., "Intrusion Detection System: A Comprehensive Review," *Journal of Network and Computer Applications*, vol. 36, no. 1, pp. 16-24, 2013.

[14] Luo B. and Xia J., "A Novel Intrusion Detection System Based on Feature Generation with Visualization Strategy," *Expert Systems with Applications*, vol. 41, no. 9, pp. 4139-4147, 2014.

[15] Marir N., Wang H., Feng G., Li B., and Jia M., "Distributed Abnormal Behavior Detection Approach Based on Deep Belief Network and Ensemble SVM Using Spark," *IEEE Access*, vol. 6, pp. 59657-59671, 2018.

[16] Mease D., Wyner A., and Buja A., "Boosted Classification Trees and Class Probability/Quantile Estimation," *Journal of Machine Learning Research*, vol. 8, pp. 409-439, 2007.

[17] Mishra P., Varadharajan V., Tupakula U., and Pilli E., "A Detailed Investigation and Analysis of Using Machine Learning Techniques for Intrusion Detection," *IEEE Communications Surveys and Tutorials*, vol. 21, no. 1, pp. 686-728, 2019.

[18] Moayedikia A., Ong K., Boo Y., Yeoh W., and Jensen R., "Feature Selection for High Dimensional Imbalanced Class Data Using Harmony Search," *Engineering Applications of Artificial Intelligence*, vol. 57, pp. 38-49, 2017.

[19] Moustafa N. and Slay J., "The Evaluation of Network Anomaly Detection Systems: Statistical Analysis of The UNSW-NB15 Data Set and The Comparison with The KDD99 Data Set," *Information Security Journal: A Global Perspective*, vol. 25, no. 1-3, pp. 18-31, 2016.

[20] Nielsen D., *Tree Boosting With XGBoost*, Ntnu, 2016.

[21] Raman M., Somu N., Kirthivasan K., Liscano R., and Sriram V., "An Efficient Intrusion Detection System Based on Hypergraph-Genetic Algorithm for Parameter Optimization and Feature Selection in Support Vector Machine," *Knowledge-Based Systems*, vol. 134, pp. 1-12, 2017.

[22] Sharafaldin I., Habibi Lashkari A., and Ghorbani A., "Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization," *in Proceedings of the 4th International Conference on Information Systems Security and Privacy*, Funchal, pp. 108-116, 2018.

[23] Singh R., Kumar H., and Singla R., "An Intrusion Detection System Using Network Traffic Profiling and Online Sequential Extreme Learning Machine," *Expert Systems with Applications*, vol. 42, no. 22, pp. 8609-8624, 2015.

[24] Tabash M., Allah M., and Tawfik B., "Intrusion Detection Model Using Naive Bayes and Deep Learning Technique," *The International Arab Journal of Information Technology*, vol. 17, no. 2, pp. 215- 224, 2020.

[25] Tjhai G., Furnell S., Papadaki M., and Clarke N., "A Preliminary Two-Stage Alarm Correlation and Filtering System Using SOM Neural Network And K-Means Algorithm," *Computers and Security*, vol. 29, no. 6, pp. 712-723, 2010.

[26] Ustebay S., Turgut Z., and Aydin M., "Intrusion Detection System with Recursive Feature Elimination by Using Random Forest and Deep Learning Classifier," *in Proceedings of International Congress on Big Data, Deep Learning and Fighting Cyber Terrorism*, Ankara pp. 71-76, 2019.

[27] Vijayanan R., Devaraj D., and Kannapiran B., "Intrusion Detection System for Wireless Mesh Network Using Multiple Support Vector Machine Classifiers with Genetic-Algorithm-Based Feature Selection," *Computers and Security*, vol. 77, pp. 304-314, 2018.

[28] Wang H., Gu J., and Wang S., "An Effective Intrusion Detection Framework Based on SVM with Feature Augmentation," *Knowledge-Based Systems*, vol. 136, pp. 130-139, 2017.

[29] WhiteHat, "2018 Application Security Statistics Report," 2018.

[30] ZorarpacI E. and Özel S., "A Hybrid Approach of Differential Evolution and Artificial Bee Colony for Feature Selection," *Expert Systems with Applications*, vol. 62, pp. 91-103, 2016.

Fawaz Mokbal received a B.S. degree in computer science from Thamar University, Yemen, an M.S degree in Information Technology from the University of Agriculture, Pakistan, and a Ph.D. degree in Computer Science and Technology from Beijing University of Technology, China. Currently, he is a Teacher and Researcher with Fan Gongxiu Honors College, Beijing University of Technology, China. For one and half years, he served as a Research Associate with the Faculty of Computer Science, ILMA University, Pakistan, for five years he was the Manager of Information Systems with the Ministry of Local Administration in Yemen, and for two years, he was the Head of the Technical Team of the Information Center Project for the local authority. He is the author and a reviewer of various SCI, EI, and Scopus indexed journals. His research interests include machine and deep learning, medical images, brain-computer interface, Web application security, and the IoT security issues.

Wang Dan received the B.S. degree in computer application, the M.S. degree in computer software and theory, and the Ph.D. degree in computer software and theory from Northeastern University, China, in 1991, 1996, and 2002, respectively. She is currently a professor and doctoral supervisor in Computer Science and Technology. She has been engaged in teaching for over 20 years. She has presided several Beijing Municipal Natural Science Foundation and research projects commissioned by enterprises. She has published more than 50 papers in journals and conferences and finished two textbooks. She used to be visiting scholar at the University of California, Riverside, and the University of Illinois at Urbana Champaign in the U.S.A. Her major areas of interest include trusted software, web security, and big data.

Musa Osman is a Ph.D. student at Beijing University of Technology (BJUT), China. He received his BSc in computer science at the University of Gazira, Sudan, and MSc in Information System at Osmania University, India. His main research interests are security issues in the Internet of Things mostly based on RPL protocol, Machine.

Yang Ping received the Ph.D. degree in computer science and technology from Beijing University of Technology, in 2020, under the supervision of Professor Dan Wang. She is currently a Lecturer with the School of Economics and Management, Beijing Information Science and Technology University. Her research interests include artificial intelligence in biomedical engineering, intelligent information processing, machine learning, and information security.

Saeed Alsamhi received the B.Eng. degree from the Department of Electronic Engineering (Communication Division), IBB University, Yemen, in 2009, and the M.Tech. degree in communication systems and the Ph.D. degree from the Department of Electronics Engineering, Indian Institute of Technology (Banaras Hindu University), IIT (BHU), Varanasi, India, in 2012 and 2015, respectively. In 2009, he worked as a Lecturer Assistant in Engineering's faculty at IBB University. He held a postdoctoral position with the School of Aerospace Engineering, Tsinghua University, Beijing, China, in optimal and smart wireless network research and its applications to enhance robotics technologies. Since 2019, he has been an Assistant Professor. He has published 30 articles in high reputation journals in IEEE, Elsevier, Springer, Wiley, and MDPI publishers. His areas of interest include green communication, green Internet of Things, QoE, QoS, multi-robot collaboration, blockchain technology, and space technologies (high altitude platform, drone, and tethered balloon technologies). He is currently MSCA SMART 4.0 FELLOW at the Athlone Institute of Technology, Athlone, Ireland.