Received: January 4, 2020. Revised: April 10, 2020.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Detecting Intrusions in Computer Network Traffic with Machine Learning Approaches

Pascal Maniriho[1] Leki Jovial Mahoro[2] Ephrem Niyigaba[3] Zephanie Bizimana[3]
Tohari Ahmad[4]*

[1] Department of Information Technology, Christian University of Rwanda, 6821, Rwanda
[2] Department of Information Technology, Vaal University of Technology, 19000, South Africa
[3] Department of Information Technology,
Rwanda Polytechnic-Integrated Polytechnic Regional College Karongi, 85, Rwanda
[4] Department of Informatics, Institut Teknologi Sepuluh Nopember, 60111, Indonesia
* Corresponding author's Email: tohari@if.its.ac.id

Abstract: Security has been a crucial factor in this modern digital period due to the rapid development of information technology, which is followed by serious computer crimes that, in turn, led to the emergence of Intrusion Detection Systems (IDSs). Various approaches such as single machine learning classifiers and Ensemble Classifiers couple with features selection methods have been proposed to improve the performance of IDS. In this regard, in the previous work, we have used the NSL-KDD IDS dataset, Gain Ratio Feature Evaluator (GRFE), and Correlation Ranking Filter (CRF) feature selection methods coupled with various machine-learning techniques to detect intrusions in computer network traffic. While the experiment has demonstrated that GRFE selects the most relevant feature subsects over CRF, which results in different performance, the previous work can be extended as follows. First, the most relevant feature subset generated by GRFE in the previous work is employed to assess and compare the performance of a single machine learning technique (Lazy IBK, aka K-Nearest Neighbor) over an ensemble technique (Random Committee) while detecting intrusions in a computer network. Second, two distinct datasets (NSL-KDD and UNSW-NB15) are employed for better performance analysis. Third, limitations encountered in the domain of network intrusion detection are also discussed. The results reveal that the ensemble technique performs well over a single machine learning technique with a misclassification gap of 0.969% and 1.19% (obtained using NSL-KDD dataset) and 1.62% and 1.576% (obtained using UNSW-NB15 dataset).

Keywords: Computer network traffic, Feature selection, Intrusion detection, Machine learning, Network security.

1. Introduction

Network intrusion detection system (NIDS) has indeed become a valuable security tool in computer networks that is employed to discover and evaluate various attacks and security violations across different organizational networks around the world [1]. Unauthorized access to confidential resources, unsolicited duplication and alteration of records, identity theft, and any other actions targeting the destruction of the information system and network infrastructure are among security breaches. In order to respond to these issues, network intrusion-based security tools are employed to inspect each network packet, and based on the previous network traffic data, it decides on whether such traffic is trustful or malevolent. Such a category of IDS is known as misuse or signature-based IDS, as it entirely depends on the previous network audit data in order to determine or categorize the nature of each network packet [2]. The main drawback of signature-based is their failure to recognize new malevolent activities across the network. In order to overcome this challenge, Anomaly-based NIDS (A-NIDS) tools have been proposed [3]. Having been introduced and highly implemented based on the concept of network packer profiling (where each new packet is inspected by determining whether it does deviate from normal traffic or not), A-NIDS has the ability to inspect network packet and identify existing and new harmful activities. To better strengthen security, researchers have also implemented hybrid intrusion detection tools that incorporate the features of both techniques [4].

It should be noted that both NIDS and A-NIDS tools totally differ from another type of security tool, which is commonly known as intrusion prevention system (IPS) as they do not block network packets [5, 6]. However, instead, they generate an alert to the network administration team, which in turn draws the final decision. Accordingly, the credibility and performance of any network intrusion detection technique are assessed based on some evaluation parameters, as provided below [7].

- True positive rate (TPR)- the ratio of positive instances that are correctly classified as positive
- False positive rate (FPR)-the ratio of negative instances which are incorrectly classified as positive
- True negative rate (TNR)-the ratio of negative instances that are correctly classified as negative
- False negative rate (FNR)-the ratio of positive instances that are incorrectly classified as negative
- Accuracy (Acc)-the overall correct prediction rate to the sample size (total number of instances).
- Detection error rate (Mis)-the ratio of instances which are incorrectly classified concerning the total number of instances under consideration

Machine learning, statistical, and data mining techniques have gained popularity in the design and implementation of network-based intrusion detection tools over recent years [3, 6]. Nevertheless, feature selection (FS) is another important aspect that can profoundly impact the performance of the detection model [8, 9]. The main reason is that having a high volume of network traffic dataset which is made up of many features, it is crucial to identify those features which are more relevant than others as some of them are considered as useless (noise) which can drastically affect the performance, in case they are used for building and training the IDS detection model. Feature selection should, therefore, never be ignored during the design and implementation of IDS tools.

In this regard, given the significance of feature selection and the importance of implementing new methods for detecting intrusions in computer networks, this paper aims to (i) present a critical review on the existing IDS detection techniques, (ii) employ a feature subset of the well-known recent IDS datasets (NSLD-KDD and UNSWNB15) to evaluate and compare the predictive accuracy and execution time of a single machine learning classifier (K-nearest Neighbour) and ensemble classifier (Random Committee) using distinct intrusion detection datasets (NSLD-KDD and UNSWNB15) to determine if ensemble models are always better while detecting intrusion in computer networks, (iii) elaborate and discuss current limitations in the domain of IDS. In this direction, these objectives can be achieved by answering the following research questions (Qs).

Qs1. How does the existing IDS approach perform while detecting intrusions?

Qs2. How does a single machine learning classifier differ from an ensemble classifier?

Qs3. Are single machine leaning classifiers better than ensemble classifiers?

Qs4. What are the current limitations in the domain of network intrusion detection?

This paper is structured as follows: Section 2 provides a critical review of the existing method. Section 3 describes the motivation of this research, while the proposed network attack detection technique is presented in section 4. Next, section 5 depicts the details of the experiment, concerning its environment, dataset, and analysis. The discussion of the method is provided in section 6, followed by the conclusion in section 7.

2. A critical review of IDS methods

Malicious activities and incomplete signatures in computer networks can be identified using security management systems such as intrusion detection systems. Various detection techniques have been suggested, and several IDSs have been implemented to detect intrusion [10]. In 2018, Maniriho and Ahmad [2] conducted a comprehensive study on the application of machine learning techniques in the domain of network intrusion detection systems. Additionally, the effect of feature selection on the performance of the detection techniques was also investigated. Various detection models were implemented and evaluated on the network traffic dataset, which is available on [11] for public use. Desale and Roshani [12] applied a genetic algorithm to select valuable features from the NSL-KDD Network traffic dataset. Several studies addressing feature selection challenge exist in the literature [13-16].

Tavallaee et al., [17] presented a detailed study analysis on KDD Cup 99 IDS dataset. Performance evaluation of different techniques for detecting intrusive network activities was carried out in [18]. Meena and Choudhary [19] have tested many different classification techniques implemented in Waikato Environment for Knowledge Analysis (WEKA) tool. Besides, various features of both KDD Cup 99 and NSL-KDD dataset were elaborated. Naïve Bayes and J48 Classifiers were utilized to build detection models after selecting relevant features using several data feature reduction techniques. The main objective was to determine the best data reduction technique having the ability to select a feature subset that can significantly enhance the performance of the classification model. In addition, sensitivity, specificity, and accuracy were considered for comparative study [20].

Charhi et al., [21] have applied risk assessment methodologies to detect intrusions in the cloud-based deployment. Their approach reduced false alerts, and the IDS performance was improved. A new technique for performing feature reduction, which is able to optimize the process of intrusion detection by limiting the cluster size, and utilization of sub-medoids to form new suitable features was proposed in [22]. Inspired by pattern clustering concepts, a new feature representation technique was implanted in 2020 by Aldweesh et al., [23]. This approach is mainly based on the nearest neighbor and clusters center computation approaches, and it has achieved efficient computation cost in terms of testing and training of the detection algorithm. Gu et al., developed a system that detects and analyzes intrusions in network traffic payload [24]. Naïve Bayes approach was used to build a fully distributed anomaly-based network intrusion detection system where the analysis and detection run at each data collecting point [25].

Kabir and Hartmann [26] proposed two new IDS concepts and implemented them, and evaluated their performances against Snort (version 2.9.7.5) having almost 26k rules. A new dataset was collected for testing and analysis. The targeted attacks, including web cgi, policy violation, DoS and DDoS were evaluated in a real-time mode. Nevertheless, feature selection and extraction methods are not provided. The advantage is that their method performs well over Snort on the given dataset. Also, it requires only a low hardware configuration with better scalability. Consequently, the packet inspection time, along with the false positive rate for both small and large networks, were reduced. However, their proposed system still has a low detection rate for sensitive threats. Adaboost and Artificial bee (ABC) colony algorithms were used to develop a new anomaly network based to gain a low false positive rate and a high detection accuracy [27]. Feature selection was performed using the ABC algorithm.

By using KDD-Cup 99, Mirza [28] implemented various ensemble classification methods. The ensemble learning models were implemented and tested after the features were selected by using principal component analysis (PCA). Though their method cannot perform real-time detection, the normal connection and intrusive connection can be successfully identified. The method achieved better results than before, and the anomalous rate is very high. By using UNSW-NB15, Anwer et al., [29] selected features by implementing wrapper and filter features selection. In their research, J48 and Naïve Bayes Classifiers were explored to identify normal and anomalous network traffics. This method showed that J48 classifier outdoes Naïve Bayes with an accuracy of 88% in non-real time detection.

Unlike previous research, the Kyoto 2006+ dataset was employed by Park et al., [30] for evaluating IDS, IDS+shellcode, malware, and other unknown attacks. Random Forest approach was applied in the algorithm. Park et al., [30] found that the method delivered a poor performance for shellcode attack detection. Besides, there is a reduction in the overall performance for attack detection, including normal class as well. To conclude, their results showed that each attack has a different prediction rate. Similar to this research, Zaman and Lung [31] also used Kyoto 2006+ dataset for measuring their proposed method in non-real time detection. For selecting the features, [31] performed entropy computation, and seven machine learning classifiers were tested. Detection of normal traffic flows and harmful traffic flows were designed, whose results show that above 90% for precision, recall, and accuracy were achieved for most tested machine learning approaches. It also presents that good results were not achieved with the ensemble method.

For detecting R2L, U2R, DoS, and Probe attacks, a Multi-class support vector machine (SVM) was used by Ikram and Cherukuri, [32]. Before the system starts, the feature selection is applied by implementing the fusion of chi-square feature selection. The system was evaluated on NSL-KDD. Having been evaluated in non-real detection, it achieved a high detection rate, and the SVM parameters were optimized. Furthermore, the number of false alarms was decreased, and some critics on the training and testing time were discussed. By targeting the same types of attacks, Manzoor and Morgan [33] applied Support Vector Machine and Apache storm-based-IDS in the real-time mode. The features of KDD-Cup 99 were selected by exploring the operations based on statistical techniques. It was found that it processed data at high speed, where 13,600 network traffic packets can be processed in a second. Therefore, the approach can handle big network traffic data. Like the previous target attacking types, Hendrik et al. [34] proposed the use of ranker and information gain FS methods for selecting the features of the NSL-KDD dataset.

The Ant Tree Miner (ATM) classifier, whose experimental results show that the ATM classier performed well compared with other machine learning intrusion-based detection methods, was explored. Next, Ahmad [35] presented various tree-based data mining classifiers with Weka tools, which were tested using KDD Cup 99. In addition to DoS, Probe, R2L, and U2R, the normal connection was inspected. Besides, he has also suggested a combination of more than one classifier to improve detection accuracy, whereby based on the results, the ensemble method can be preferable for future work. The improvement of detection and accuracy rate for DoS, Probe, R2L, U2R was obtained by Ravale et al., [36] who applied K- Means and SVM-based RBF Kernel Function for feature selection and the detection, respectively. Hajisalem and Babaie [37] have used Artificial Fish Swarm and Artificial Bee Colony to implement a new hybrid classification technique, and irrelevant features were removed using Correlation-based Feature Selection and Fuzzy C-Means Clustering. Their techniques achieved a 99% detection rate and a 0.01% false positive rate.

Differently, Tchakoucht and Ezziyyani [9] designed the combination of different filter and wrapper methods for selecting features, Random Forest, C4.5, Naïve Bayes, and REPTree for identifying the attacks. By using the NSL-KDD dataset to evaluate probe and DoS attacks, they obtained an excellent false positive rate and detection rate. In this evaluation, there are 19 features selected to detect probe while only nine features were used for DoS attack detection. The authors suggested that this study will be extended to a real-time network traffic environment. Nadiammai and Hemalatha [38] explored KDD Cup99 by using Hybrid PSO techniques, a new decision tree-based algorithm (EDADT), and semi-supervised approaches for feature selection and detection of the DoS attack. In this research, Snort was combined with other anomaly-based approaches to minimize the workload; and low false (reduced) alarm rate and better accuracy. By using the state of the art UNSW-NB15 IDS dataset, a deep learning intrusion detection technique was proposed in [39]. Their method achieved a lower false alarm rate and high accuracy after selecting relevant features and an optimal activation function. The research applied layer configuration and feature selection to reduce the learning time, and high detection accuracy was maintained. Garg and Khurana [40] presented a comparative study on the performance of various classifications algorithms.

3. Motivations

Motivation of this research can be presented as follows.

- Most of the previous studies have evaluated the performance using a single dataset, which is actually not enough to conclude that one machine learning detection approach does perform well over the other. Therefore, this paper intends to use two distinct recent IDS datasets.
- To date, there exist few papers in the literature that discuss the current trends in the domain of intrusion detection in computer networks.
- Improve the predictive accuracy of the detection model using the most relevant feature subset from NSL-KDD and UNSWNB15 datasets.

4. Proposed detection techniques and simulation setting

This section introduces the proposed machine-learning techniques that are employed to detect network attacks and the simulation setting. The primary purpose is to evaluate how a single machine learning classifier does detect intrusions over an ensemble method. An ensemble is a set of individually trained machine learning classifiers whose classification accuracies (predictions) are combined by an algorithm. Two approaches, namely, instance-based learning (Lazy IBK) and Random Committee available in the WEKA tool, are considered for detecting network intrusions. Besides, Gain Ratio Feature Evaluator is employed to select the best feature subsets for two recent IDS datasets (NSL-KDD and UNSWNB15).

Received: January 4, 2020. Revised: April 10, 2020.

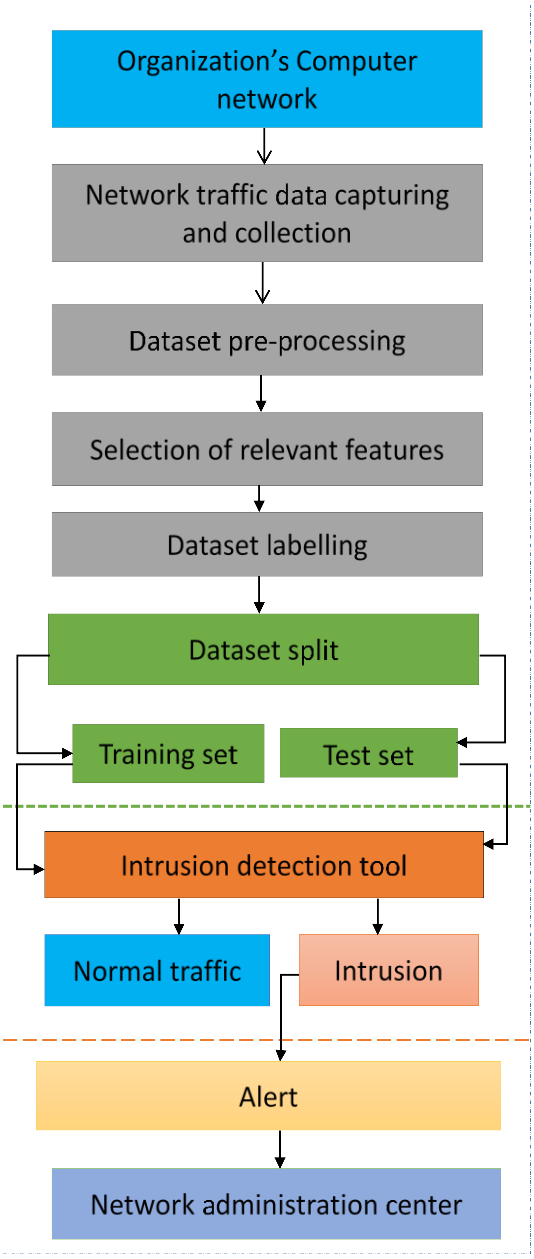
Organization's Computer network
Network traffic data capturing and collection
Dataset pre-processing
Selection of relevant features
Dataset labelling
Dataset split
Training set
Test set
Intrusion detection tool
Normal traffic
Intrusion
Alert
Network administration center

Figure 1. Various stages illustrating the functionality of a network intrusion detection system (tool)

4.1 The overall functionality of IDS

Detecting intrusions in computer network traffic involves several steps. As depicted in Fig. 1, the process begins by capturing and collecting network traffic datasets, which is then processed to remove data inconsistencies such as duplicate values, missing values, and any other kind of noise. Once the dataset has been processed, feature selection is performed to select only those features which are relevant to be used for training and testing the detection model. It is worth noting that one portion of the dataset generated after feature selection is used for training while the other portion is used for testing, which intends to see how the detection model does perform on unseen data.

Based on its nature, normal traffic will pass while whenever an intrusion is detected, the detection systems will generate an alert to notify the network administration center for decision making. The workflow for the proposed single machine learning and ensemble learning technique is depicted in Fig. 2.

4.2 Instance-based learning approach

The instance-based learning algorithm (IBK) with parameter K is also known as the K-nearest neighbor (K-NN) learning algorithm. It is in the category of regression and classification of lazy algorithms based on similarity computation between instances that are used to solve various machine learning tasks. The algorithm specifies the number of the nearest neighbors to be used while classifying a test instance. The IBK algorithm has the ability to select the appropriate value of K based on cross-validation and can also perform distance weighting. It is crucial mentioning that in the Weka tool, the IBK is implemented with a cross-validation option, which automatically helps in selecting the best value (which is equivalent to the value of K) for K-NN.

4.3 Random committee approach

The Random Committee (RC) is the implemented class for building an ensemble of randomizable-based classification approaches. Each base classification approach is built using a different random number seed but which are based on the same data. The final prediction is a straight average of the predictions generated by the individual base classification method.

5. Experimental results

The proposed methods are tested in a laptop computer having the following specifications: 64-bit Windows 10 (Professional Edition), Intel (R) Core (TM) i5-4200U CPU@ 2.30 GHz Processor with no Pen or Touch input display. The datasets we used for evaluation are described along with the analysis of the evaluation results.

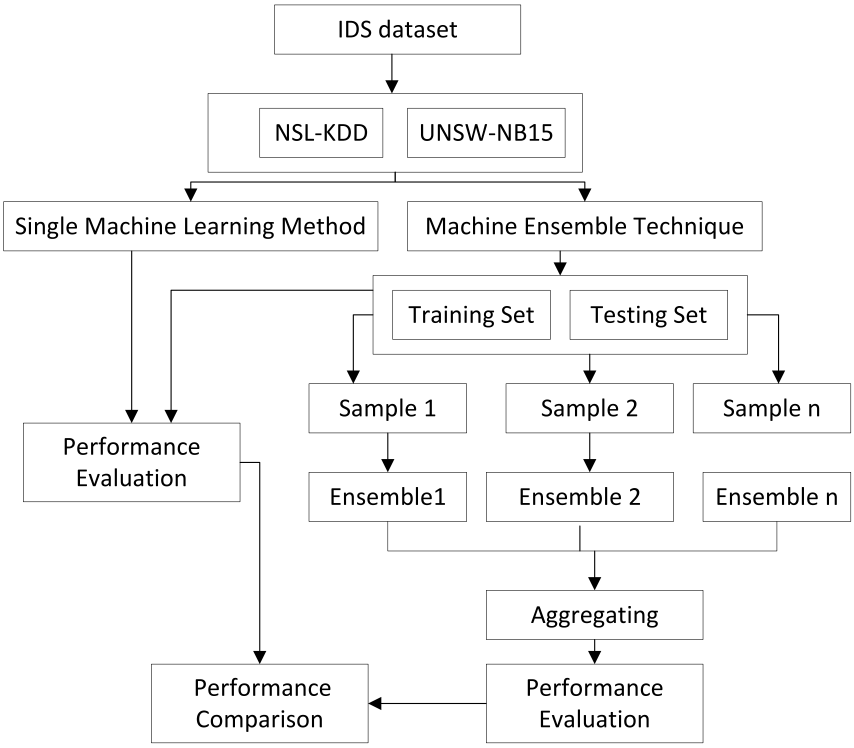

Figure. 2 Workflow for the proposed single machine learning and ensemble learning techniques

5.1 NSL-KDD and UNSW-NB15IDS Datasets

The NSL-KDD network traffic dataset for IDS in [11] contains 41 attributes and one attribute for the class label. The original data has 5 million data samples. However, in this work, the position of 20 percent training set is used for training and testing. First, as was mentioned, this is an extension of our original work presented in [2]. In the previous work, we had applied different feature selections methods to determine the best feature subset to be used for training the detection algorithms. Based on the experimental results, Gain ration Feature Evaluator (GRFE) has proven to be the best feature selection technique. In this regard, a feature subset having the best 15 attributes generated by GRFE is employed in this work to evaluate how a single machine-learning detection technique performs over an ensemble technique. It is worth noting that those attributes were selected based on their ranking scores, and the list of selected attributes is mentioned below.

Selected feature subset (15 features) from NSL-KDD = {logged_in, srv_serror_rate, flag, serror_rate, dst_host_srv_serror_rate, dst_bytes, diff_srv_rate, dst_host_serror_rate, src_bytes, same_srv_rate, service, dst_host_srv_diff_host_rate, dst_host_same_srv_rate, dst_host_srv_count, wrong_fragment}. Twenty percent (train set-20 percent), one of the NSL-KDD dataset portions, has 25192 instances, which are used for training and testing each detection technique. Accordingly, 66% of the train set equivalent to 16627 is utilized as a train split, and the remaining 34%, which is equivalent to 8565 instances, is taken as test split. For better performance analysis of both detection techniques, K-fold (also known as cross-validation) → 3, 5, and 7 folds of the entire training set are also utilized as a testing set. Cross-validation is mainly used in machine learning to estimate the skill of a machine learning approach on unseen data, i.e., a limited sample is used in order to estimate how well the model does perform in general when used to make prediction and classification on data which was not used during the training of the detection model. Therefore, it is utilized to select the best K values for the folds during the experiment.

Additionally, Gain ration Feature Evaluator (GRFE) is also used to select the best 15 features from the UNSW-NB15 IDS dataset [41]. UNSW-NB15 is one of the recent intrusion detection datasets with 44 features, including the 45th feature, which is the class for each network traffic instance. The best feature subset selected from the UNSW-NB15 after applying GRFE is presented below. Note that as it was performed into our previous work [2], relevant features are selected based on their ranking scores (from the highest to lowest score), and description for each feature can be viewed in [42, 43].

The selected feature subset (15 features) from UNSW-NB15 is {id, xState, dpkts, sbytes, dbytes, rate, sttl, dttl, dload, dinpkt, tcprtt, synack, ackdat, ct_state_ttl, is_sm_ips_ports and label (class)}. This is then used in the experiment.

The training set of the UNSW-NB15 dataset contains 175341 instances, and it is used for both training and testing. During the first experiment, 66% split and 34% of the dataset are used for training and testing, respectively. In the second experiment, 3, 5, and 7 folds of the entire training set are also utilized as a testing set to evaluate how well the detection algorithms perform on various subsets or portions of the training set, and the results for each fold are recorded.

5.2 Experiment results analysis

Throughout the experiment, the performance evaluation of these two IDS detection techniques is conducted by measuring True Positive Rate (TPR), True Negative Rate (TNR), Accuracy (Acc), Precision (Prec), and Misclassification rate (Mis). Misclassification means that normal traffic is detected as anomalous traffic and vice versa.

$$TPR = (TP)/\ (TP + FN) \qquad (1)$$

$$TNR\ = (TP)/\ (TN + FP) \qquad (2)$$

$$Acc = (TP + TN)/(TP + TN + FP + FN) \qquad (3)$$

$$Prec = (TP)/(TP + \mathrm{FP}) \qquad (4)$$

$$Mis = (FP + FN)/(TP + TN + FP + FN) \qquad (5)$$

The results of the experiment are presented in Tables 1- 6. In more specific, Tables 1, 3, 4, and 6 show the superiority of Random Committee approach over the instance based learning method. Additionally, the time taken by each model while evaluating the performance on the test split is presented in Tables 2 and 5. A comparison between the proposed method and other detection methods using NSL-KDD and UNSWNB15 Dataset is also presented in Table 7, and the overall classification results are depicted in Fig. 3, Fig. 4, Fig. 5, and also Fig. 6.

Table 1. Experimental results of both instance-based learning algorithm (IBK) and Random Committee (RC) by employing (34%) of the NSL-KDD training set as a test set

Method	**TPR**	**FPR**	**Accuracy (%)**	**Precision**	**Mis**
Lazy IBK (k=1)	0.987	0.017	98.727	0.987	1.272
Random Committee	0.997	0.003	99.696	0.997	0.303

Table 2. Time taken to test the model on test split (34%) using both techniques using NSL-KDD dataset

Detection model	**Testing time in seconds**
Lazy IBK (k=1)	7.08
Random Committee	0.11

Table 3. Experimental results generated using Random Committee (RC) by employing 3, 5, and 7 folds of the NSL-KDD training set (20-percent) as a test set

Detection Model	**Fold**	**TPR**	**FPR**	**Accuracy**	**Precision**	**Mis**
Random Committee	3	0.997	0.003	99.706	0.997	0.293
	5	0.998	0.002	99.769	0.998	0.230
	7	0.998	0.003	99.761	0.998	0.238
	Average	0.997	0.002	99.745	0.997	0.254
Lazy IBK (k=1)	3	0.985	0.015	98.535	0.985	1.464
	5	0.986	0.014	98.618	0.986	1.381
	7	0.985	0.015	98.507	0.985	1.492
	Average	0.985	0.014	98.553	0.985	1.446

Table 4. The experimental results of both instance-based learning algorithm (IBK) and Random Committee (RC) using 59, 615 instances (34%) of the UNSWNB15 Training set as a test set

Method	**TPR**	**FPR**	**Accuracy**	**Precision**	**Mis**
Lazy IBK (with K=1)	0.961	0.021	97.3346	0.979	2.665
Random Committee	0.990	0.017	98.955	0.990	1.045

Table 5. Time taken to test the model on test split (34%) using both techniques using UNSWNB15 dataset as a test set

Detection model	Testing time in seconds
Lazy IBK (k=1)	0.34
Random Committee	710.05

Table 6. Experimental results of both instance-based learning algorithm (IBK) and Random Committee (RC) by employing 3, 5, and 7 folds of the UNSWNB15 Training set as a test set

Detection Model	Fold	TPR	FPR	Accuracy	Precision	Mis
Random Committee	3	0.988	0.020	98.819	0.988	1.180
	5	0.989	0.018	98.925	0.989	1.074
	7	0.990	0.018	98.964	0.990	1.035
	Average	0.989	0.018	98.903	0.989	1.096
Lazy IBK (with K=1)	3	0.972	0.035	97.242	0.972	2.757
	5	0.974	0.033	97.350	0.974	2.649
	7	0.974	0.033	97.390	0.974	2.609
	Average	0.973	0.033	97.327	0.973	2.672

Table 7. Comparison between the proposed methods and other detection methods using NSL-KDD and UNSWNB15 dataset

Detection Methods	Classifier Method	Dataset	Detection Accuracy (%)
Detection method in [44]	Deep Neural Network	NSL-KDD	99.2000
Detection method in [40]	Random Tree	NSL-KDD	96.0800
Detection method in [37]	Hybrid ABC-AFS	UNSWNB15	98.9000
Detection method in [39]	Naive Bayes	UNSWNB15	81.2000
Detection method in [42]	Random Forest	UNSWNB15	80.9000
Proposed Method	Lazy IBK	NSL-KDD	98.7270
Proposed Method	Random Committee	NSL-KDD	99.6960
Proposed Method	Lazy IBK	UNSWNB15	97.3346
Proposed Method	Random Committee	UNSWNB15	98.9550

In more detail, Table 1 presents the overall detection accuracy, which is obtained using both Lazy IBK and Random Committee learning algorithms. For Lazy IBK, 8456 instances were correctly classified in their respective class (whether in normal traffic or anomalous traffic) with the detection (classification) accuracy of 98.73%, while 109 instances were incorrectly classified, which results in the detection error rate of 1.27%. For the RC's classification, the accuracy is 99.70%, with the detection error rate of 0.303%. Besides, TPR of 99.90% and FPR of 0.49% were also achieved. Note that both TPR and precision values must be close to one for a proper intrusion detection technique, whereas false positive rate and false negative rate should always be close to zero. Table 3 presents the results generated using the Random Committee (ensemble classifier) with 3, 5, and 7 folds of the NSL-KDD training set (20-percent) as a test set.

As it was early mentioned, the UNSWNB15 dataset was also utilized to draw a conclusion on how a single classifier does perform over an ensemble classifier while detecting intrusions in computer networks. As it could be viewed in Table 4, the overall classification results reveal that 97.3346% and 2.6654% detection accuracy and detection error rate was respectively achieved using Lazy IBK while the detection accuracy of 98.955% and 1.045% of the misclassification rate were achieved using Random Committee. Table 6 shows the results generated with cross-validation (K-folds) of 3, 5, and 7 as the test set.

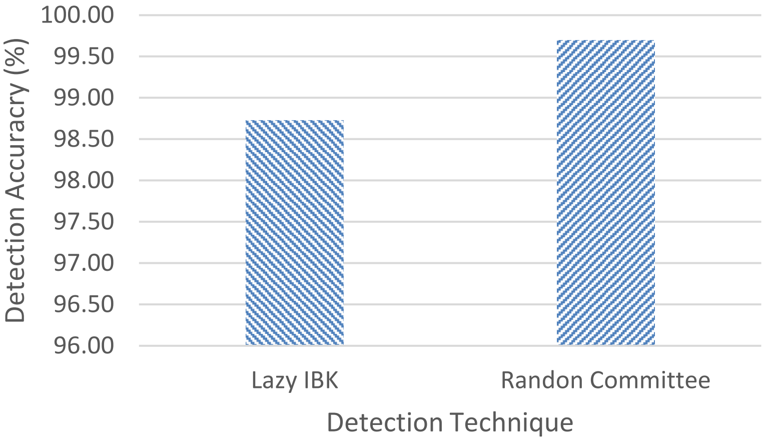

Figure. 3 Accuracy for Lazy IBK and Random Committee using NSL-KDD dataset

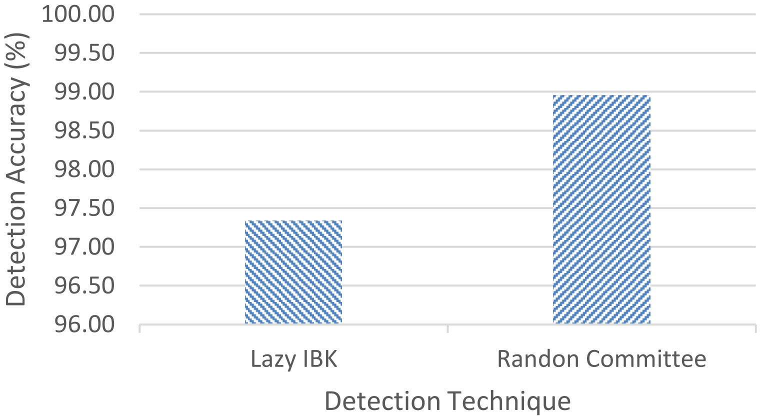

Figure. 4 Accuracy for Lazy IBK and Random Committee using UNSW-NB15 dataset

Having trained and tested both techniques with the same datasets and feature subsets, it can be concluded that the ensemble method (Random Committee) performs well over a single machine learning approach (Lazy IBK) with the misclassification gap of 0.969% → $1.272 - 0.303$ (refer to Table 1) and 1.19% → $1.444 - 0.254$ (refer to Table 3, overall classification average obtained with 3, 5, and 7 folds cross-validation) using NSL-KDD dataset. Similarly, the UNSWNB15 results also reveal the Random Committee's potential over Lazy IBK with the misclassification gap of 1.62 % → $2.665 - 1.045$ (see Table 4) and 1.576 % → $2.672 - 1.096$ refer to Table 6, overall classification average with 3, 5 and 7 folds cross-validation) using UNSWNB15 dataset. Taking into account the time it takes to test the model on the test split (NSL-KDD) it could also be seen that Random Committee takes less time (0.11 seconds) while it takes Lazy IBK 7.08 seconds to test the model which results in a testing gap of 6.97 seconds → $7.08 - 0.11$ (see Table 2). However, Lazy IBK is faster than Random Committee while using UNSWNB15, which results in a testing gap of 676.05 seconds → $710.05 - 034$ (see Table 5).

Considering the results presented in Table 7, it could be seen that the proposed methods outdo the previous ones in terms of detection accuracy with the gap of 0.496% and 2.676 % compared with the methods in [40, 44] using the NSL-KDD. Also, the detection accuracy gap of 18.055, 0.0555, and 17.755 % were achieved compared with the methods in [37, 39, 42] using the proposed UNSWNB15 feature subset and the Random Committee technique. Additionally, by utilizing the proposed feature subset of NSL-KDD and Lazy IBK technique, the accuracy gap of 2.647% was achieved (compared with the method in [40]) while comparing with the method in [42] and [39] the gap of 16.134% and 16.4334% were achieved using the selected relevant feature subset from UNSWNB15 dataset and the lazy IBK technique. However, the methods in [37,

Received: January 4, 2020. Revised: April 10, 2020.

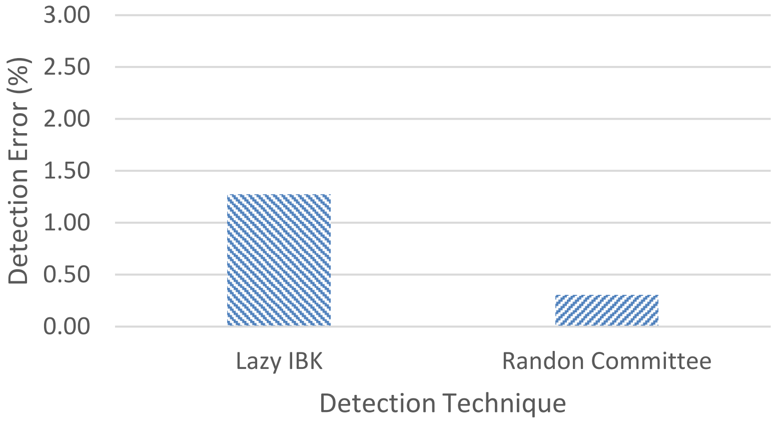

Figure. 5 Detection Error for Lazy IBK and Random using NSL-KDD dataset

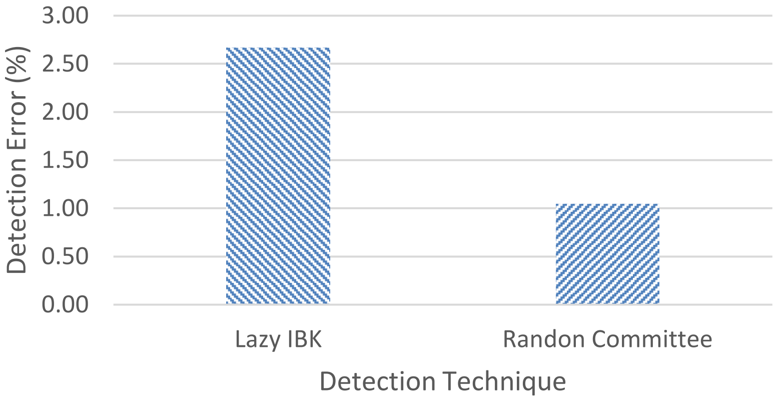

Figure. 6 Detection Error for Lazy IBK and Random using UNSW-NB15 dataset

44] outperforms the proposed single machine learning technique (Lazy IBK) with the accuracy gap of 0.473 % (using NSL-KDD) and 0.746 % (using UNSWNB15), and these gaps are mainly due to distinct features subset and optimization between these techniques.

6. Currents Limitations

Referring to the critical review, which is presented in Section 2, below, we discuss some research limitations (challenges) that are currently identified in the domain of network intrusion systems as follows.

- Large data Size. Due to network infrastructures, which are evolving and scaling rapidly, there is a massive growth of network traffic data, which, therefore, makes its mining to be very challenging.
- High Dimensionality. New feature reduction and extraction techniques are still needed to address the problem of high dimensionality, which profoundly affects the overall performance of the intrusion detection models.
- High False Positive rate (FPR). The performance of most intrusions detections techniques presented in the literature suffers from a high false positive rate, which results in many false alarms. It dramatically affects the performance of IDS as it takes a massive amount of time to generate false alarms, which in turn affects the overall detection rate.
- Standard Performance of IDS Techniques. Although there are several parameters to assess the performance of IDS detection models, as of now, there is no predefined standard accuracy rate set by the expert in the industry.
- New Intrusion Detection Dataset. Although a few network traffic datasets for intrusion detection systems exist, most of these data

are old to be used for evaluating and testing new IDS models for the currents and future network infrastructures.

- Dataset Labelling. Considering the time it takes to generate and label the dataset, there is a need to develop new methods having the ability to extract important features and label the dataset automatically.

7. Conclusion

Network intrusion detection is one of the crucial security components and defense mechanisms of network security systems. This paper proposed IDS methods to evaluate the efficiency and effectiveness of a single machine learning approach (Lazy Instance-Based Learning) and an ensemble approach (Random Committee) while detecting intrusions. The performance was tested using the best feature subset selected from the recent network traffic datasets (NSL-KDD and UNSWNB15). The results show that the ensemble technique performs well over a single machine learning technique with a misclassification gap of 0.969% and 1.19% (obtained using NSL-KDD dataset) and 1.62% and 1.576% (obtained using UNSW-NB15 dataset).

More importantly, various limitations identified in the domain of network intrusion detection are also discussed. The proposed method can help to detect intrusion in computer networks with high detection accuracy, and we believe that this work will serve as a guideline for future research in the domain of network intrusion detection systems.

References

[1] A. A. Agarkar and H. Agrawal, "LRSPPP: lightweight R-LWE-based secure and privacy-preserving scheme for prosumer side network in smart grid", *Heliyon*, Vol. 5, No. 3, 2019.

[2] P. Maniriho and T. Ahmad, "Analyzing the performance of machine learning algorithms in anomaly network intrusion detection systems", In: *Proc. of the 4th International Conference on Science and Technology*, 2018.

[3] Z. Chiba, N. Abghour, K. Moussaid, A.El Omri, and M. Rida, "Intelligent approach to build a Deep Neural Network based IDS for cloud environment using combination of machine learning algorithms", *Computers & Security*, Vol. 86, pp. 291-317, 2019.

[4] V. Hajisalem and S. Babaie, "A hybrid intrusion detection system based on ABC-AFS algorithm for misuse and anomaly detection", *Computer Networks*, Vol. 1368, pp. 37-50, 2018.

[5] M. Baykara and R. Das, "A novel honeypot based security approach for real-time intrusion detection and prevention systems", *Journal of Information Security and Applications*, Vol. 41, pp. 103-116, 2018.

[6] A. Aldweesh, A. Derhab, and A. Z. Emam, "Deep learning approaches for anomaly-based intrusion detection systems: A survey, taxonomy, and open issues", *Knowledge-Based Systems*, Vol. 18915, 2020.

[7] S. Subudhi and S. Panigrahi, "Application of OPTICS and ensemble learning for Database Intrusion Detection", *Journal of King Saud University - Computer and Information Sciences*, [in press, available online 15 May 2019].

[8] G. Hossein and H. Hamid, "A new feature selection IDS based on Genetic Algorithm and SVM", In: *Proc. of the 8th International Symposium on Telecommunications (lST'2016)*, pp. 1–6, 2016.

[9] T. A. Tchakoucht and M. Ezziyyani, "Building A Fast Intrusion Detection System For High-Speed- Building A Fast Intrusion Detection System For High-Speed- Networks : Probe and DoS Attacks Detection", In: *Proc. of the First International Conference On Intelligent Computing in Data Sciences*, Vol. 127, pp. 521–530, 2018.

[10] M. Aamir and S. M. A. Zaidi, "Clustering based semi-supervised machine learning for DDoS attack classification", *Journal of King Saud University - Computer and Information Sciences* [in press, available online 5 February 2019].

[11] "NSL-KDD | Datasets | Research | Canadian Institute for Cybersecurity | UNB." [Online]. Available:http://www.unb.ca/cic/datasets/nsl.html. [Accessed: 09-Nov-2017].

[12] K. S. Desale and A. Roshani, "Genetic Algorithm based Feature Selection Approach for Effective Intrusion Detection System", In: *Proc. of International Conference on Computer Communication and Informatics (ICCCI -2015)*, 2015.

[13] H. Alazzam, A. Sharieh, and Khair Eddin Sabri, "A feature selection algorithm for intrusion detection system based on Pigeon Inspired Optimizer", *Expert Systems with Applications*, Vol. 14815, 2020.

[14] C. Khammassi and S. Krichen, "A NSGA2-LR wrapper approach for feature selection in

network intrusion detection", *Computer Networks*, Vol. 1728, 2020.

[15] Y. Su, K. Qi, C. Di, Y. Ma, and S. Li, "Learning Automata based Feature Selection for Network Traffic Intrusion Detection", In: *Proc. of IEEE Third Int. Conf. Data Sci. Cybersp. Learn.*, pp. 622–627, 2018.

[16] K. Khan, A. Mehmood, S. Khan, M. A. Khan, and W. K. Mashwani, "A survey on intrusion detection and prevention in wireless ad-hoc networks", *Journal of Systems Architecture*, Vol. 105, 2020.

[17] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A Detailed Analysis of the KDD CUP 99 Data Set", In: *Proc. of the 2009n IEEE Symposium on Computation Intelligence in Security and Defense Applications (CISDA 2009)*, pp. 1–6, 2009.

[18] T. Ahmad, T. Hasbiya, R. M. Ijtihadie, and W. Wibisono, "Detecting Malicious Activities in a Computer Cluster for Developing Dynamic Honeypot", *ICIC Express Letters Part B: Applications*, Vol. 9, No. 3, pp. 257-264, 2018.

[19] G. Meena and R. R. Choudhary, "A Review Paper on IDS Classification using KDD 99 and NSL KDD Dataset in WEKA", In: *Proc. of International Conference on Computer, Communications and Electronics*, pp. 553–558, 2017.

[20] A. Thakkar and R. Lohiya, "Role of swarm and evolutionary algorithms for intrusion detection system: A survey", *Swarm and Evolutionary Computation*, Vol. 53, 2020.

[21] B. Charhi, M. Nada, B. Elmehdi, and R. Boubker, "Intrusion detection in cloud computing based attack patterns and risk assessment", *Adv. Sci. Technol. Eng. Syst. J.*, Vol. 2, No. 3, pp. 479–484, 2017.

[22] I. Z. Muttaqien and T. Ahmad, "Increasing Performance of IDS by Selecting and Transforming Features", In: *Proc. of Comnetsat*, pp. 85–90, 2016.

[23] A. Aldweesh, A. Derhab, and A. Z. Emam, "Deep learning approaches for anomaly-based intrusion detection systems: A survey, taxonomy, and open issues", *Knowledge-Based Syst.*, Vol. 18915, 2020.

[24] J. Gu, L. Wang, H. Wang, and S. Wang, "A novel approach to intrusion detection using SVM ensemble with feature augmentation", *Computers & Security*,Vol. 86, pp. 53-62, 2019.

[25] T. Singh and N. Kumar, "Machine learning models for intrusion detection in IoT environment: A comprehensive review", *Computer Communications* [in press, Available online 26 February 2020].

[26] F. M. Kabir and S. Hartmann, "Cyber Security Challenges: An Efficient Intrusion Detection Systems Design", In: *Proc. of International Young Engineers Forum (YEF-ECE)*, pp. 19–24, 2018.

[27] M. Mazini, B. Shirazi, and I. Mahdavi, "Anomaly network-based intrusion detection system using a reliable hybrid artificial bee colony and AdaBoost algorithms", *J. King Saud Univ. - Comput. Inf. Sci.*, Vol. 31, No. 4, pp. 541–553, 2019.

[28] A. H. Mirza, "Computer Network Intrusion Detection using various Classifiers and Ensemble Learning", In: *Proc. of the 26th Signal Processing and Communications Applications Conference (SIU)*, pp. 1–4, 2018.

[29] H. M. Anwer, M. Farouk, and A. Addel-Hamid, "A Framework for Efficient Network Anomaly Intrusion Detection with Features Selection", In: *Proc. of the 9th International Conference on Information and Communication Systems (ICICS)*, pp. 157–162, 2018.

[30] K. Park, Y. Song, and Y.-G. Cheong, "Classification of Attack Types for Intrusion Detection Systems using a Machine Learning Algorithm", In: *Proc. of IEEE Fourth International Conference on Big Data Computing Service and Applications*, pp. 282–286, 2018.

[31] M. Zaman and C.-H. Lung, "Evaluation of Machine Learning Techniques for Network Intrusion Detection", In: *Proc. of NOMS 2018 - 2018 IEEE/IFIP Network Operations and Management Symposium*, pp. 1–5, 2018.

[32] S. T. Ikram and A. K. Cherukuri, "Intrusion detection model using fusion of chi-square feature selection and multi class SVM", *J. King Saud Univ. Inf. Sci.*, Vol. 462–472, No. 29, pp. 462–472, 2017.

[33] M. A. Manzoor and Y. Morgan, "Network Intrusion Detection System using Apache Storm", *Adv. Sci. Technol. Eng. Syst. J.*, Vol. 2, No. 3, pp. 812–818, 2017.

[34] F. Hendrik, L. Leenen, and R. De La Harpe, "Ant Colony Induced Decision Trees for Intrusion Detection", In: *Proc. of the 16th European Conference on Cyber Warfare and Security ECCWS*, pp. 53–62, 2017.

[35] B. Ahmad, "Intrusion Detection With Tree-Based Data Mining Classification Techniques by Using KDD Dataset", *Eur. J. Comput. Sci. Inf. Technol.*, Vol. 5, No. 6, pp. 11–18, 2017.

[36] U. Ravale, N. Marathe, and P. Padiya, “Feature Selection Based Hybrid Anomaly Intrusion Detection System Using K Means and RBF Kernel Function”, In: *Proc. of International Conference on Advanced Computing Technologies and Applications (ICACTA),* Vol. 45, pp. 428–43, 2015.

[37] V. Hajisalem and S. Babaie, “A hybrid intrusion detection system based on ABC-AFS algorithm for misuse and anomaly detection”, *Comput. Networks,* Vol. 136, pp. 37–50, 2018.

[38] G. V. Nadiammai and M. Hemalatha, “Effective approach toward Intrusion Detection System using data mining techniques”, *Egypt. Informatics J.*, Vol. 15, No. 1, pp. 37–50, 2014.

[39] L. Zhiqiang, G. Mohi-Ud-Din, L. Bing, L. Jianchao, Z. Ye, and L. Zhijun, “Modeling Network Intrusion Detection System Using Feed-Forward Neural Network Using UNSW-NB15 Dataset”, In: *Proc. of IEEE 7th Int. Conf. Smart Energy Grid Eng.*, pp. 299–303, 2019.

[40] T. Garg and S. S. Khurana, “Comparison of classification techniques for intrusion detection dataset using WEKA”, In: *Proc. of Comparison of classification techniques for intrusion detection dataset using WEKA,* 2014.

[41] N. Moustafa and J. Slay, “The significant features of the UNSW-NB15 and the KDD99 data sets for Network Intrusion Detection Systems”, In: *Proc. of the 4th International Workshop on Building Analysis Datasets and Gathering Experience Returns for Security*, pp. 25–3, 2015.

[42] T. Janarthanan and S. Zargari, “Feature selection in UNSW-NB15 and KDDCUP’99 datasets”, In: *Proc. of IEEE Int. Symp. Ind. Electron.*, pp. 1881–1886, 2017.

[43] N. Moustafa and J. Slay, “UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)”, In: *Proc. of Mil. Commun. Inf. Syst. Conf. MilCIS 2015,* 2015.

[44] J.-H. Woo, J.-Y. Song, and Y.-J. Choi, “Performance Enhancement of Deep Neural Network Using Feature Selection and Processing for Intrusion Detection”, In: *Proc. of International Conference on Artificial Intelligence in Information and Communication (ICAIIC),* pp. 8–10, 2019.