DOI: https://doi.org/10.33103/uot.ijccce.22.2.7

Proposed Hybrid Ensemble Learning Algorithms for an Efficient Intrusion Detection System

Doaa N. Mhawi[1], Sokeana H. Hashim[2]
[1]Department of Computer Science, Middle Technical University and University of Technology, Baghdad, Iraq
[2]Department of Computer Science, University of Technology, Baghdad, Iraq
[1]dododuaaenteesha@mtu.edu.iq, [2]110015@uotechnology.edu.iq

Abstract—*Due to sophisticated cyber-attacks, and to produce false alarms on suspicious or unusual behavior to monitor computer resources, Intrusion Detection Systems (IDSs) are required. Hence, Many Machine Learning (ML) and data mining techniques have been proposed to increase the effectiveness of IDSs, whereas current IDS algorithms are still struggling to perform effectively while many IDSs depend on a single classifier to detect intrusions. Single-classifier IDSs cannot achieve high accuracy and low false alarm rates because of zero-day attacks. In this paper, a hybrid ensemble method using AdaBoosting and Bagging for IDS is proposed. This study aims to identify unknown (zero-day attacks) and known (well-known) attacks. So, the proposed model comprises three stages. The first stage is preprocessing. The second stage involves the application of AdaBoosting and Bagging methods by four different classifiers modifying (i.e., Naïve Bayesian (NB), Support Vector Machine (SVM), random forest (RF), and K_Nearest Neighbor (KNN)). Such a modification is performed for the AdaBoosting methods. The AdaBoosting classifier is then combined to work in the Bagging method. For attack recognition, uses the voting technique as the third stage. Experimental results reveal that using the UNSW BN15 dataset yields testing with 85.49% accuracy, 99.96% detection rate, and 0.006 false alarm rate. Therefore, the proposed Hybrid AdaBoosting and Bagging Method (HABBM) can outperform other comparable and state-of-the-art techniques across a variety of parameters.*

Index Terms—*AdaBoosting method, Bagging method, Cyber Security CS, Ensemble Method, Intrusion Detection Systems IDSs.*

I. INTRODUCTION

Nowadays, a variety of new attacks are discovered daily, and their influence is growing rapidly and dangerously. consequently, it is a difficult way to detect zero-day attacks along with potentially jeopardizing [1], [2]. Different types of attacks are complex and increasing which they are poses difficulties in detecting the intrusion. IDS is designed to defend the computer system from suspicious activities that would go undetected by a traditional packet filter[3]. Establishing high levels of cyber resilience against malicious activity and detecting unauthorized access to a computer system is critical, to scanning network packets for malicious activity signals [4]–[7]. In general, many IDSs have many disadvantages, which are including the inability to distinguish between new malicious threats, low (accuracy, and detection rate), and high false negative and positive rates. Hence, ML was applied for IDS to detect new attacks (zero-day attacks) [8]. The usage of several classifiers in place of a single one verification the idea of ensemble learning techniques, which have been the subject of numerous studies to assure high accuracy and a low false alarm rate [9]–[14].

Received 9/October/2021; Accepted 26/November/2021
 ISSN (Print) 1811-9212 ISSN (Online) 2617-3352

DOI: https://doi.org/10.33103/uot.ijccce.22.2.7

However, research using the hybrid principle in feature selection and ensemble methods (discussed in the next section) is limited. To boost accuracy, The ensemble learning classification methods usually combine a variety of fundamental classifiers in some way. These classifiers are efficient at handling the same problem by creating and integrating numerous unique models, and when combined, they result in a prediction output that is more reliable and accurate.

The best representation in the hypothesis space may not necessarily be produced by a single classifier, to begin with [1][15]. Hence, to improve prediction performance using multiple classifiers are necessary [22]. A false or inaccurate hypothesis can be improved if the training dataset for the learning algorithm is insufficient. The most well-known methods in ensemble learning are bagging and clustering [16]–[19] and boosting [20]. These algorithms achieve sufficient classification results and are commonly used to generate numerous ensemble models. In addition, learning options for improving classification performance include voting averaging, Bayesian parameters, and stacking. In various applications e.g., the detection of infiltration, ensemble techniques have been proven to enhance accuracy Ensemble [21]–[27]. For instance, having a look at the outcomes in [28]–[30] insight that their suggested ensemble models outperform single classifier models considering IDS performance. Four modified classifiers are used in this study as AdaBoosting. These classifiers are then aggregated using the principal technique of Bagging to acheive high detection rate (DR) with high accuracy (Acc.) and a low False Alarm Rate (FAR). Compared the proposed method with existing ones on an extensive testbed using the NSL-KDD dataset, the accuracy of these classifiers before and after modification is computed with runtime (complexity time). Despite keeping FAR at reasonable values, the suggested solution outperforms comparable algorithms when evaluating Accuracy (Acc.), F Measure, and classification metrics (ADR). Therefore, the contributions of this experimental work are presented as follows:

- Preprocessing step to clean the dataset and work efficiently in the next steps.
- IDSs used four modifying classifiers to detect known and unknown attacks (zero-day attacks).
- To assure the best outcome as the hybrid ensemble method, the four updated classifiers should be collected using the best machine learning (ensemble method) methodology.
- Comparing the four classifiers' performance as an AdaBoosting classifier before and after modification.
- Comparing the suggested approach to other ones already in use.

The remaining sections of this work are structured as follows; Section (II) examines the roughly comparable works. The suggested hybrid HABBM is further defined in Section III. The experimental results and comments are presented in Section IV. Section V concludes by offering suggestions for future research.

II. RELATED-WORK

A. Ensemble Classifier:

Ensemble Learning is a technique that creates multiple models and then combines them to produce improved results. It usually produces more accurate solutions than a single model would. They two most well-known methods in ensemble learning are bagging [16] This model is generated using the same machine learning algorithm with n random observations of m sub-samples of the original dataset using the bootstrap sampling method. The second step is aggregating the result generated from these models. Well-known

Received 9/October/2021; Accepted 26/November/2021
 ISSN (Print) 1811-9212 ISSN (Online) 2617-3352

DOI: https://doi.org/10.33103/uot.ijccce.22.2.7

methods, such as voting and averaging, are used for this purpose (bagging algorithm working in parallel). The term "boosting" is used to describe a family of algorithms that can convert weak models to strong models. Boosting incrementally builds an ensemble by training each model with the same dataset but where the weights of instances are adjusted according to the error of the last prediction Adaboosting is a widely known algorithm that is a boosting method [20] (these algorithms work sequentially).

In [31], authors have proposed a new IDS based on machine learning ensemble methodologies, using Bat optimization as feature selection and the Bagging approach. To increase accuracy and reduce the false positive rate, appropriate variables from the NSL-KDD dataset were chosen. The suggested ensemble method's classification accuracy, model construction time, and False Positives were all assessed. The Bagging ensemble using different classifiers provides the best classification accuracy, according to the results. The bagging method requires less time to build the model. When compared to existing machine learning techniques, the proposed ensemble method has a reduced false-positive rate.

In [32], For the new IDS paradigm, using Python package to implement boosting and bagging algorithms such as DistributedRandomForest (DRF), GradientBoostingMachine (GBM), and XGBoost. It was discovered that our model outperforms the previous Deep Neural Network after utilizing the feature selection technique Genetic Algorithm (GA). Furthermore, our results outperform several traditional machine learning algorithms.

In [33], IDS using An ensemble was proposed in this paper to reduce unwanted events in IoT networks, such as botnet assaults against DNS, HTTP, and MQTT protocols, producing new statistical flow characteristics. Then AdaBoost method was created utilizing three machine learning techniques (decision tree DT, Naive Bayes NB, and artificial neural networks ANN) to evaluate the effect of these qualities and detect detrimental occurrences quickly. The proposed system was tested using UNSW-NB15 and NIMS botnet datasets with simulated IoT sensor data, using the entropy and correlation coefficient measures. The experimental results showed that the proposed traits can be used to describe both normal and dangerous behaviors. Furthermore, the suggested ensemble technique has a higher detection rate and a lower false-positive rate as compared to other classification techniques in the framework and three other previously proposed techniques.

B. HYBRID APPROACHES:

To improve the IDS's performance, numerous hybrid approaches combining feature selection and ensemble methods have recently been developed.

[34] presented an enhanced IDS based on hybrid feature selection and two-level classifier ensembles. To reduce the feature size of the training datasets, the authors used a hybrid feature selection strategy combining three methods particle swarm optimization, ant colony algorithm, and genetic algorithm (NSL-KDD and UNSW-NB15 are considered in this paper). The classification performance of a Reduced Error Pruning Tree (REPT) classifier is used to choose features. Then, based on two meta learners, rotation forest and bagging, a two-level classifier ensemble is presented. The proposed classifier outperforms other classification algorithms that are recently proposed in the literature on the NSL-KDD dataset, with 85.8% accuracy, 86.8% sensitivity, and 88.0 percent detection rate. The results for the UNSW-NB15 dataset are also better than those obtained, using multiple approaches. Finally, a two-step statistical significance test is performed to verify the results. This has not been taken into account in previous IDS research, and hence adds value to the suggested classifier's experimental results.

Received 9/October/2021; Accepted 26/November/2021

 ISSN (Print) 1811-9212 ISSN (Online) 2617-3352

DOI: https://doi.org/10.33103/uot.ijccce.22.2.7

The authors in [35] suggested PSO-RF as a binary Particle Swarm Optimization (PSO) and random forests (RF) intrusion detection mechanism. it investigates the performance of various dimension reduction strategies as well as a set of different classifiers, including the suggested methodology. Moreover, RF employment was considered as a classifier, using binary PSO to find more acceptable properties to classify network intrusions. It presents lower dataset dimensions using various state-of-the-art dimension reduction approaches in the preprocessing step, which ultimately reduced the dataset that is provided to the proposed PSO-RF strategy, and further optimizes the data's dimensions, attaining ideal features. PSO is an appropriate optimization method that is employed for dimension optimization in this case. By using several performance measures, we conduct comprehensive testing to demonstrate the value of the proposed approach. The KDD99Cup dataset, as it contains information about numerous types of network intrusions, is used as a standard benchmark. The experimental results show that the suggested strategy outperforms existing approaches in terms of detecting all types of assaults in the dataset.

Information Gain (IG) and Principal Component Analysis (PCA) in [36] are combined with an ensemble classifier based on a Support Vector Machine (SVM), Instance-Based learning techniques (IBK), and multilayer perceptron in a new hybrid dimensionality reduction technique for intrusion detection (MLP). Three well-known datasets were used to evaluate the performance of this IG-PCA-Ensemble technique: ISCX 2012, NSL-KDD, and Kyoto. According to the experimental results, the suggested hybrid dimensionality reduction strategy with the ensemble of base learners provides more critical qualities and accordingly outperforms individual approaches in terms of accuracy and false alarm rates. By comparing our methodology to the previous work, the recommended IG-PCA-Ensemble strategy outperforms the bulk of existing state-of-the-art methodologies in terms of classification accuracy, detection rate, and false alarm rate.

Several efforts to improve the detection rate of intrusion detection systems have been developed, but such techniques still struggle to create and update the signature of new malware, in addition to producing either a large number of false alarms or low detection rates.

III. PROPOSED SYSTEM

The detection framework of the proposed ML-based IDS is depicted in *Fig. 1*, which comprises three basic phases as presented in the following sections.

- **Step 1: Preprocessing datasets**

The first step is to preprocess the source datasets to convert raw data into an analysis-ready state. The following three phases are used to demonstrate Algorithm 1: data filtering, transformation, and normalization.

Algorithm 1: Minimax Scaling (Pre-processing).

Input: Reading datasets (i.e., d1 and d2). /*Where d1 represents the NSL_KDD dataset, and d2 represents the UNSW BN15 dataset */.

Output: Dataset normalized $d1_{normalize}$, and $d2_{normalize}$.

Step 1: Data Filtering

(Removed from the anomalous datasets and redundant instances).

split datasets into parts: train 75% and test 25%.

Step 2: Data transformation

for I from 1 to n do:

if (do non-numeric) do the following:

Received 9/October/2021; Accepted 26/November/2021

 ISSN (Print) 1811-9212 ISSN (Online) 2617-3352

DOI: https://doi.org/10.33103/uot.ijccce.22.2.7

Applied Label Encoder () and One-Hot Encoding functions. /* Transform categorical features into numbers*/.
End if

Step 3: Normalization (Minimax scaling):
$d_{n\ normalize} = d_n - (d_n)_{min} / (d_n)_{max} - (d_n)_{min.}$
End for.
step 4: End

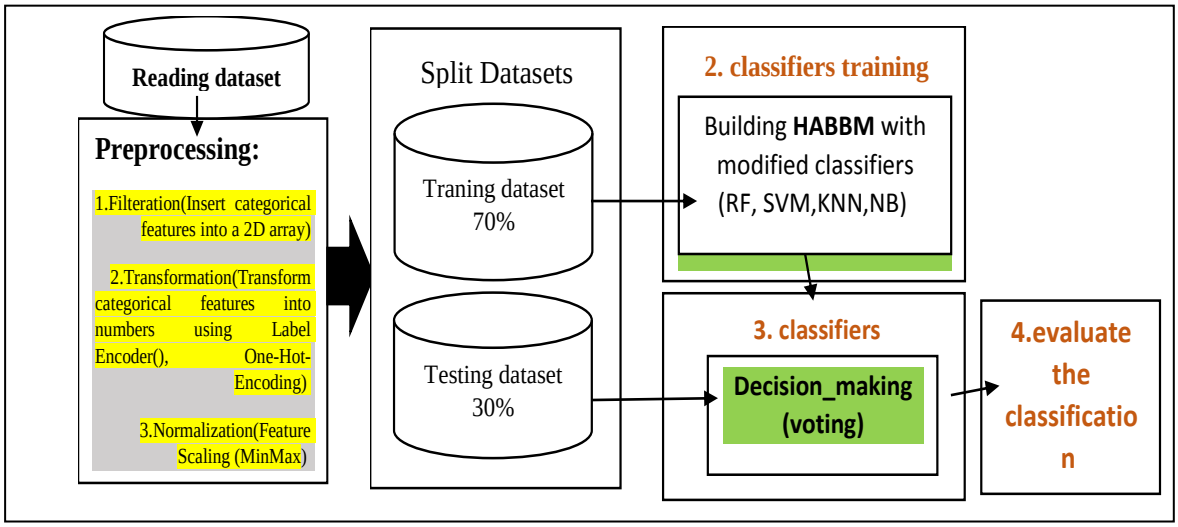

FIG. 1. GENERAL STRUCTURE OF THE PROPOSED ENSEMBLE LEARNING BASED ON ID.

- **Step 2: Training HABBM**

With the hybrid ensemble method within this step. Algorithm 2 explains this step where each classifier is modified to work as an AdaBoosting technique, which ensures the sufficient learning of each classifier. The parameters and weight are also modified to improve their efficiency to unknown attacks.

Algorithm 2: Random Forest, K_NN, SVM, and NB Algorithms used as the AdaBoosting method

1. Input: datasets after preprocessing step applied algorithm 1
   Split dataset into training and testing, /* Xi represents a feature */
   For a training set Xi, partly do

a. Algorithm: AdaBoostingRF

Generate new RF using → {h(x, θ_k),k = 1,2,…..i..} **/*RF random forest,** θ_k **is random_vector generated*/**
10 subset forest is generating.
Initialization for each feature weight value →Wi. . /* wi weight*/
Compute σ^2 for each Xi →pσ^2+ $\frac{1-p}{B}$ σ2. * σ^2 **is stander division, p population B constant*/**

b. Algorithm b: AdaBoostingSVM

Split the training set by using hyber_plain into classes: positive and negative.
Determine the support vectors using linear_SVM.
Determine important features for SVM such as: kernel = "linear," C = 1.0, random_state = 0
Compute F(x) for each support vector →(w,x) + b then update the weight. /* wi weight*/

c. Algorithm: AdaBoosting KNN

Received 9/October/2021; Accepted 26/November/2021
 ISSN (Print) 1811-9212 ISSN (Online) 2617-3352

DOI: https://doi.org/10.33103/uot.ijccce.22.2.7

Initialization→weight

Determine feature of KNN: algorithm = "auto," leaf_size = 30, metric = "minkowski," metric_params = None, n_jobs = None, n_neighbors = 5, p = 2.

Update weight

d. Algorithm: AdaBoostingNB

Training-phase:

Compute P(c) →training dataset**.**

Looping:

Compute for all classes ci($\mathbf{v_j}$).

Compute probability for each ($\mathbf{v_j}$) as follows: (Freq. $\mathbf{v_j}$)/(Freq. ci).

End Loop

Testing-phase:

Looping:

Compute $\mathbf{v_j}$ for each class (ci) in the training dataset.

P of each record applied multiplication:/* p probability*/

$\mathbf{P(E|c) = P(a_1, a_2, \dots, a_n|c) = \prod_{i=1}^{n} P(a_i|c)}$

End Loop

e. Evaluation: **(testing)**

For each algorithm (a, b, c, d) do

Compute accuracy of predicted and testing Xi

If the predicted Xi is not nearest to the tested Xi then

Algorithm an→ Update Wi and σ^2 **/*wi is weight,** σ^2 **is stander division*/**

Algorithm b→ Choose another hyber_plain

Algorithm c→ Update Wi

Algorithm d→ Update Pi

Else

For each algorithm

Compute measurements for predicted and tested Xi using Eqs. 1, 2, 3, and 4

End if

End for

2. Output: accuracy, the detection rate

The proposed model then aggregates these classifiers and uses the principle of the bagging technique to facilitate parallel operations. The best result of these classifiers is chosen using average voting techniques. Algorithm 3 explains the proposed hybrid model.

Algorithm 3: Hybrid HABBM for Intrusion Detection

1. Input: Datasets
2. Output: high detection rate, low false alarm rate
3. Preprocessing steps applied (algorithm 1).
4. For a training_set part do:
    a. Applied AdaBoosting techniques for the following algorithms: (as facing steps in the proposed model) each of these algorithms working sequentially. */ applied for training and testing*/+
    b. Algorithm 3
5. EndFor
6. After these steps, the weight of each algorithm is updated, and the four algorithms are aggregated. The principle of bagging is then applied to all to facilitate parallel operations.

Received 9/October/2021; Accepted 26/November/2021

 ISSN (Print) 1811-9212 *ISSN (Online) 2617-3352*

DOI: https://doi.org/10.33103/uot.ijccce.22.2.7

7. After the application of the bagging principle, the final results were demonstrated by using the average voting
   $\rightarrow \frac{1}{\mathrm{mj}} = \mathrm{l} \sum_{\mathrm{i}=1}^{\mathrm{l}} \mathrm{Pci}(Wix)$. */this step is related to checking the testing where PCI is the set of probability classifiers, wi weight, x features */
8. Compute accuracy for predicted Xi after voting.
9. Compute general measurement using Eqs. 1, 2, 3, and 4.
10. Else go to step 5 and update the average weighting for the classifiers using the highest probabilities
11. End if
12. Output: the best accuracy, FAR, Recall, and Precision

- **Step3: Recognition_ Attack:**

This step is utilized to test the detection mode and to follow computing the predicted and test accuracies after the voting technique. To integrate the probability distributions of base learners, a voting method is used and the combination rule to obtain classification judgments by computing the predicted and test accuracies. Finally, the ensemble classifier's results show that benign traffic and other unpreferable events may be accurately recognized and classified.

IV. EVALUATIONS AND EXPERIMENTAL RESULTS

A. THE EVALUATION RESULT OF THE PROPOSED MODEL

The primary goal, as already said, is to create trustworthy IDSs with few false alarms and high accuracy. To test the suggested system, the UNSW BN15 dataset is used. Python 3. 8 is used to execute and assess the results of the experiment on a laptop that complies with the following specifications: processor,i710510UCPU@2.80GHz,2.30GHz,6.0GBRAMwith10G,64-bitOS,andx64-basedprocessor.

The dataset used is the NSL KDD dataset and is divided into training (70%) and testing (30%) sets. The UNSW-NB15 dataset [37]. The IXIA PerfectStorm tool in the Cyber Range Lab of UNSW Canberra produced the raw network packets for the UNSW-NB 15 dataset to provide a blend of real contemporary normal activities and synthetic current assault behaviors. 100 GB of the raw traffic was captured using the tcpdump utility (e.g., Pcap files). Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode, and Worms are among the nine types of attacks in this dataset. To create a total of 49 features with the class label, 12 algorithms are built and the Argus and Bro-IDS tools are employed. The 42 characteristics of the UNSW-NB15 are compiled in Table I in a clear style. The remaining 39 of the 42 traits are numerical, while three are non-numeric categories. In the UNSW-NB15, there are two main datasets: UNSW-NB15-TEST, which is used to test the trained models, and UNSW-NB15-TRAIN, which is used to train various models. For our study, we divided the UNSW NB15 TRAIN into two sections: UNSW NB15 TRAIN-1, which accounted for 75% of the entire training set, was used training, and UNSW NB15 VAL, which made up 25% of the entire training set, was used for validation before testing. The data acquired during the training phase is contrasted with this second partition as a sanity check. When utilizing this strategy, it is imperative to refrain from training a model on the evaluation or test set as this can lead to an issue known as data leaking.

Received 9/October/2021; Accepted 26/November/2021

 ISSN (Print) 1811-9212 *ISSN (Online) 2617-3352*

DOI: https://doi.org/10.33103/uot.ijccce.22.2.7

TABLE I. FEATURES NAMES OF THE UNSW NB15 DATASET

Feature no.	Features name	Formats	Feature no.	Features name	Formats
F1	duration	Float	F2	Edwinpb	Integer
F3	proto	Integer	F4	Edwin	Integer
F5	Ct_item	Integer	F6	dwin	Float
F7	Is_sm_ips	Integer	F8	pets	Float
F9	Ct_flow	Integer	F10	dpkts	byte
F11	trans_depth	Integer	F12	dbyte	style
F13	ct_src_port	Integer	F14	sttle	Float
F15	sit	Integer	F16	loss	Float
F17	sjit	Integer	F18	depict	Float
F19	Edwin	Integer	F20	dinAjit	Float
F21	mean	Integer	F22	djit	Float
F23	smean	Float	F24	stcpb	Integer
F25	service	Float	F26	tcprtt	Binary
F27	spkts	Float	F28	synack	Integer
F29	sbyte	Float	F30	ackdat	Integer
F31	rate	Float	F32	resonse_body	Integer
F33	dttle	Categorical	F34	ct-srv-src	Categorical
F35	sloss	Integer	F36	ct-state	Integer
F37	sinpikt	Integer	F38	ct_dst.	Integer
F39	stcpb	Categorical	F40	Ct_src	Integer
F41	tcprtt	Categorical	F42	Ct_src_sport	Integer

Data leakage during training occurs when a model receives information that it shouldn't, leading to bias in the final model. As a result, the model's performance with the data it encountered is subpar [38]. *Fig. 2* depicts the features and distribution of values for each assault type among the data subsets.

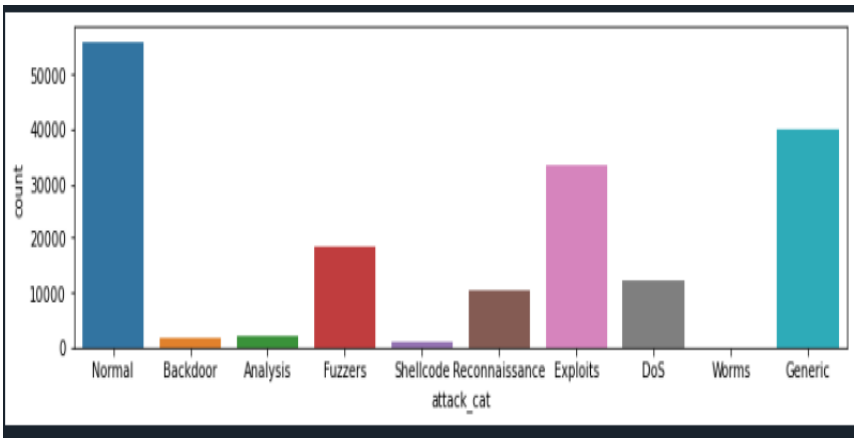

FIG.2. UNSW-NB15 REPARTITION INSTANCES.

The proposed IDS accuracy has been assessed at all levels for evaluation, with four statistical assessment measurements: F_Measure, Recall, precision, and accuracy are computed, as shown in equations 1, 2, 3, and 4 respectively. True positive values, in which incursions are appropriately classified as intrusions, are denoted by TP. The true negative values, in which normal or benign is appropriately defined as benign, are recorded as TN. FP stands for false positives, which occur when the normal is incorrectly labeled as an intrusion. FN stands for false negatives, which occur when

Received 9/October/2021; Accepted 26/November/2021
 ISSN (Print) 1811-9212 *ISSN (Online) 2617-3352*

DOI: https://doi.org/10.33103/uot.ijccce.22.2.7

incursion samples are incorrectly labeled as normal. The detailed analyses using the confusion matrix explain in Table II.

$$F\_measure = \frac{2*Recall*Precision}{Recall+Precision} \quad (1)$$

$$Recall = \frac{TP}{TP+FN} * 100\% \quad (2)$$

$$Precision = \frac{TP}{TP+FP} \quad (3)$$

$$Acc = \frac{TP+TN}{TP+TN+FP+FN} \quad (4)$$

TABLE II. CONFUSION MATRIX OF THE PROPOSED MODEL

	normal	intrusion
Normal	**16774**	**0**
Intrusion	**0**	**35829**

The running time is increased with the number of inputs. The largest input results in the class of Generic, and the smallest is found in the worms class. Therefore, the complexity time presented in *Fig. 3* explains the high runtime in the generic class, which is 14.1 min, and the lowest in the Worms is 0.9 min.

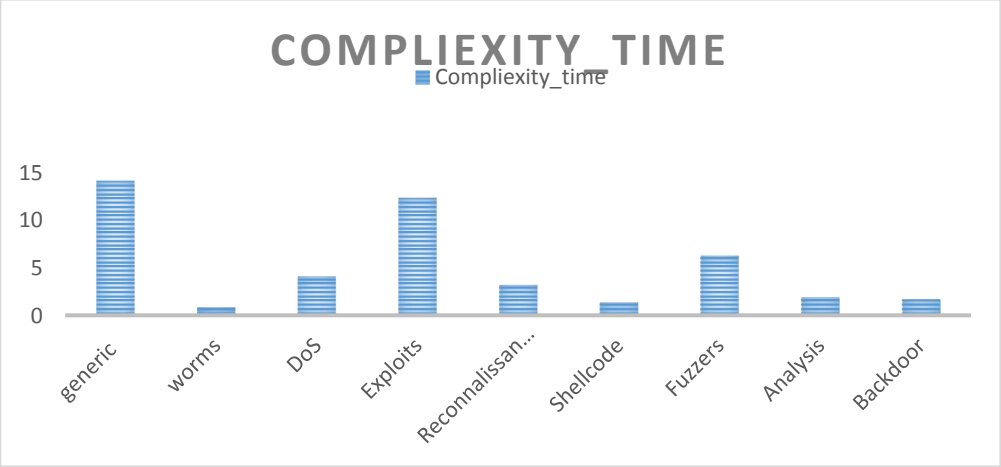

FIG. 3. COMPLEXITY TIME OF THE PROPOSED MODEL.

B. COMPARISON WITH OTHER CLASSIFICATION AND STUDIES

Table III listed the traditional classifier before and after modification using AdaBoosting in the proposed model. These classifiers modify the accuracy percentage for increasing the detection accuracy of normal attacks: the accuracy of SVM before modification was 65.50%, however, it is increased to 99.96% after modification and usage as AdaBoosting. The other classifiers demonstrated the same phenomenon, providing the best accuracy after modification.

Received 9/October/2021; Accepted 26/November/2021
 ISSN (Print) 1811-9212 ISSN (Online) 2617-3352

DOI: https://doi.org/10.33103/uot.ijccce.22.2.7

TABLE III. COMPARISON WITH TRADITIONAL CLASSIFIERS

machine-learning tech.	Acc. Before modified	Acc. After modified
NB	76.56%	98.969%
RF	80.67%	99.958%
SVM	65.50%	99.967%
KNN	79.4%	98.979%

The proposed model is also compared with other studies through the measurements of accuracy, FAR, DR, and several features with the known types of classification models. The detection accuracy using the hybrid model resulted in 0.999904% for training and 0.8549% for testing. Compared to the single-stage approach, the proposed system yields a high detection rate and a low false alarm rate. This trade-off is shown in Table IV. In comparison to previous studies, the suggested model along with testing achieved the best accuracy, DR, and FAR.

TABLE IV. COMPARISON RESULTS WITH OTHER STUDIES

ML tech.	Classification method	Features	accuracy %	DR %	F-Measure	FAR %	Time complexity in second
Ensemble learning [31]	NB_Tree	40	80.023	85	0.9472	14.8	51.43
Ensemble model with feature selection method [32]	Distributed RF and Gradient Boosting Machine	30	80.42	97.62	0.96	0.012	32.28
Ensemble classifier based on SVM [36]	IG-PCA	40	72.89	97.87	0.9201	13.53	48.01
Ensemble classifer[39]	Voting contain(C4.5,RF,Forest PA)	10	73.571	73.61	0.9302	12.92	41.67
Proposed_mode	Voting(RF,NB,KNN,SVM)	30	85.49	99.996	0.99	0.006	21.54

V. CONCLUSIONS

This paper proposed hybrid approaches of AdaBoosting and Bagging methods used for IDS. The dataset is cleaned through the preprocessing s before using that approach to increase its efficiency next steps. The four classifiers are modified andtep used as AdaBoosting to detect known and unknown attacks (zero-day attacks), then to be selected to gain the best result as a hybrid ensemble method. The results assured that the classifiers (SVM, RF, KNN, and NB) cannot effectively detect normal attacks. However, the detection robustness of the proposed hybrid method can be improved when modified and used as AdaBoosting, while the accuracy of the training dataset becomes 99.9904%. The proposed hybrid model is adopted to select the best result. Table IV shows that the proposed hybrid method attained the best results compared with other studies in this field: accuracy of 85.49% for testing, detection rate (DR) of 99.96%, and false alarm rate of 0.006. In the intrusion detection business, these results can give the proposed hybrid method a major competitive advantage as demonstrated compared to state-of-the-art techniques. Despite the efficient hybrid ensemble technique, further work is highly recommended to increase the capacity to deal with rare threats of network traffic.

Received 9/October/2021; Accepted 26/November/2021

 ISSN (Print) 1811-9212 ISSN (Online) 2617-3352

DOI: https://doi.org/10.33103/uot.ijccce.22.2.7

REFERENCES

[1] X. Sun, J. Dai, P. Liu, A. Singhal, and J. Yen, "Using Bayesian Networks for Probabilistic Identification of Zero-Day Attack Paths," IEEE Trans Inf Forensics Secur, vol. 13, no. 10, pp. 2506–2521, 2018, doi: 10.1109/TIFS.2018.2821095.

[2] M. Alazab, "Profiling and classifying the behavior of malicious codes," J Syst Softw, vol. 100, pp. 91–102, 2015, doi: 10.1016/j.jss.2014.10.031.

[3] I. Sumaiya Thaseen and C. Aswani Kumar, "Intrusion detection model using fusion of chi-square feature selection and multi class SVM," J King Saud Univ - Comput Inf Sci, vol. 29, no. 4, pp. 462–472, 2017, doi: 10.1016/j.jksuci.2015.12.004.

[4] S. Shareef and S. Hashim, "Proposed Hybrid Classifier to Improve Network Intrusion Detection System using Data Mining Techniques," Eng Technol J, vol. 38, no. 1B, pp. 6–14, 2020, doi: 10.30684/etj.v38i1b.149.

[5] S. H. Hashem, "Enhance network intrusion detection system by exploiting br algorithm as an optimal feature selection," in Handbook of Research on Threat Detection and Countermeasures in Network Security, 2014, pp. 17–32.

[6] S. Rajagopal, P. P. Kundapur, and K. S. Hareesha, "A Stacking Ensemble for Network Intrusion Detection Using Heterogeneous Datasets," Secur Commun Networks, vol. 2020, 2020, doi: 10.1155/2020/4586875.

[7] J. H. Assi and A. T. Sadiq, "NSL-KDD dataset Classification Using Five Classification Methods and Three Feature Selection Strategies," J Adv Comput Sci Technol Res, vol. 7, no. 1, pp. 15–28, 2017.

[8] S. Aljawarneh, M. Aldwairi, and M. B. Yassein, "Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model," J Comput Sci, vol. 25, pp. 152–160, Mar. 2018, doi: 10.1016/j.jocs.2017.03.006.

[9] X. Feng, Z. Xiao, B. Zhong, J. Qiu, and Y. Dong, "Dynamic ensemble classification for credit scoring using soft probability," Appl Soft Comput J, vol. 65, pp. 139–151, 2018, doi: 10.1016/j.asoc.2018.01.021.

[10] S. Sharma, R. K. Challa, and R. Kumar, "An ensemble-based supervised machine learning framework for android ransomware detection," Int Arab J Inf Technol, vol. 18, no. 3 Special Issue, pp. 422–429, 2021, doi: 10.34028/IAJIT/18/3A/5.

[11] R. Devarajan and P. Rao, "An efficient intrusion detection system by using behaviour profiling and statistical approach model," Int Arab J Inf Technol, vol. 18, no. 1, pp. 114–124, 2021, doi: 10.34028/iajit/18/1/13.

[12] A. Hnaif, K. Jaber, M. Alia, and M. Daghbosheh, "Parallel scalable approximate matching algorithm for network intrusion detection systems," Int Arab J Inf Technol, vol. 18, no. 1, pp. 77–84, 2021, doi: 10.34028/iajit/18/1/9.

[13] H. Li and J. Sun, "Predicting business failure using an RSF-based case-based reasoning ensemble forecasting method," J Forecast, vol. 32, no. 2, pp. 180–192, 2013, doi: 10.1002/for.1265.

[14] Doaa N. Mhawi and Aliaa H. Kareem "Information Retrieval Using Modified Genetic Algorithm," Al Mansur Journal, no. June, p. 15, 2017, doi: 10.36541/0231-000-027-004.

[15] M. S. Kadhm, D. E. Mhawi, and R. M. H. Zaki, "An Accurate Handwritten Digits Recognition System Based on DWT and FCT," Iraqi Journal Sci, vol. 58, no. 4B, pp. 2200–2210, 2017, doi: 10.24996/ijs.2017.58.4b.23.

[16] A. A. S. Syed and K. H. Lee, "Macroeconomic forecasting for Pakistan in a data-rich environment," Appl Econ, vol. 53, no. 9, pp. 1077–1091, 2021, doi: 10.1080/00036846.2020.1826399.

[17] P. I. Wire, "Proposed Integrated Wire/Wireless Network Intrusion Detection System," vol. 14, no. 2, pp. 9–24, 2014.

[18] H. A. Jaber and M. T. Rashid, "HD-sEMG Gestures Recognition by SVM Classifier for Controlling Prosthesis," Iraqi J Comput Commun Control Syst Eng, pp. 10–19, 2019, doi: 10.33103/uot.ijccce.19.1.2.

[19] Z. A. Mohammed, M. N. Abdullah, and I. H. Al-hussaini, "Predicting Incident Duration Based on Machine Learning Methods," Iraqi J Comput Commun Control Syst Eng, pp. 1–15, 2021, doi: 10.33103/uot.ijccce.21.1.1.

[20] K. Wang, Y. Wang, Q. Zhao, D. Meng, X. Liao, and Z. Xu, "SPLBoost: An Improved Robust Boosting Algorithm Based on Self-Paced Learning," IEEE Trans Cybern, vol. 51, no. 3, pp. 1556–1570, 2021, doi: 10.1109/TCYB.2019.2957101.

[21] J. Hu, "An approach to EEG-based gender recognition using entropy measurement methods," Knowledge-Based Syst, vol. 140, pp. 134–141, 2018, doi: 10.1016/j.knosys.2017.10.032.

[22] C. Cheng, Y. Hu, J. Wang, H. Liu, and M. Pecht, "Generalized sparse filtering for rotating machinery fault diagnosis," J Supercomput, vol. 77, no. 4, pp. 3402–3421, 2021, doi: 10.1007/s11227-020-03398-5.

[23] C. Hung and J. H. Chen, "A selective ensemble based on expected probabilities for bankruptcy prediction," Expert Syst Appl, vol. 36, no. 3 PART 1, pp. 5297–5303, 2009, doi:

Received 9/October/2021; Accepted 26/November/2021

 ISSN (Print) 1811-9212 ISSN (Online) 2617-3352

DOI: https://doi.org/10.33103/uot.ijccce.22.2.7

10.1016/j.eswa.2008.06.068.

[24] D. N. Mhawi, "Proposed Hybrid Correlation Feature Selection Forest Panalized Attribute Approach to advance IDSs," vol. 7, no. 4, 2021.

[25] D. N. Mhawi, "Proposed Hybrid CorrelationFeatureSelectionForestPanalizedAttribute Approach to advance IDSs Proposed," *Karbala international journal of modern science*, vol. 7, no. 4, 2021.

[26] D. N. Mhawi and A. Aldallal, "Advanced Feature-Selection-Based Hybrid Ensemble Learning Algorithms for Network Intrusion Detection Systems," 2022.

[27] H. W. Oleiwi and N. Saeed, "An Enhanced Interface Selectivity Technique to Improve the QoS for the Multi-homed Node," vol. 40, no. August, 2022.

[28] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in ICISSP 2018 - Proceedings of the 4th International Conference on Information Systems Security and Privacy, 2018, vol. 2018-Janua, pp. 108–116, doi: 10.5220/0006639801080116.

[29] M. Aljanabi and M. Ismail, "Improved intrusion detection algorithm based on TLBO and GA algorithms," Int Arab J Inf Technol, vol. 18, no. 2, pp. 170–179, 2021, doi: 10.34028/IAJIT/18/2/5.

[30] A. K. Hassan and D. Enteesha mhawi, "Enhance Inverted Index Using in Information Retrieval," vol. 34, no. 2, 2016.

[31] Y. Zhou, G. Cheng, S. Jiang, and M. Dai, "Building an efficient intrusion detection system based on feature selection and ensemble classifier," Comput Networks, vol. 174, 2020, doi: 10.1016/j.comnet.2020.107247.

[32] A. Rai, "Optimizing a New Intrusion Detection System Using Ensemble Methods and Deep Neural Network," in Proceedings of the 4th International Conference on Trends in Electronics and Informatics, ICOEI 2020, 2020, pp. 527–532, doi: 10.1109/ICOEI48184.2020.9143028.

[33] N. Moustafa, B. Turnbull, and K. K. R. Choo, "An ensemble intrusion detection technique based on proposed statistical flow features for protecting network traffic of internet of things," IEEE Internet Things J, vol. 6, no. 3, pp. 4815–4830, 2019, doi: 10.1109/JIOT.2018.2871719.

[34] B. A. Tama, M. Comuzzi, and K. H. Rhee, "TSE-IDS: A Two-Stage Classifier Ensemble for Intelligent Anomaly-Based Intrusion Detection System," IEEE Access, vol. 7, pp. 94497–94507, 2019, doi: 10.1109/ACCESS.2019.2928048.

[35] A. J. Malik, W. Shahzad, and F. A. Khan, "Network intrusion detection using hybrid binary PSO and random forests algorithm," Secur Commun Networks, vol. 8, no. 16, pp. 2646–2660, 2015, doi: 10.1002/sec.508.

[36] F. Salo, A. B. Nassif, and A. Essex, "Dimensionality reduction with IG-PCA and ensemble classifier for network intrusion detection," Comput Networks, vol. 148, pp. 164–175, 2019, doi: 10.1016/j.comnet.2018.11.010.

[37] N. Moustafa and J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," 2015, doi: 10.1109/MilCIS.2015.7348942.

[38] A. Khraisat, I. Gondal, P. Vamplew, J. Kamruzzaman, and A. Alazab, "Hybrid intrusion detection system based on the stacking ensemble of C5 decision tree classifier and one class support vector machine," Electron, vol. 9, no. 1, 2020, doi: 10.3390/electronics9010173.

[39] Y. Zhou, G. Cheng, S. Jiang, and M. Dai, "Building an efficient intrusion detection system based on feature selection and ensemble classifier," Comput Networks, vol. 174, no. March, 2020, doi: 10.1016/j.comnet.2020.107247.

Received 9/October/2021; Accepted 26/November/2021

 ISSN (Print) 1811-9212 ISSN (Online) 2617-3352