COMPARATIVE PERFORMANCE EVALUATION OF INTRUSION DETECTION SYSTEM USING RNS - ENHANCED DATASETS AND STACK ENSEMBLE LEARNING MODELS

IDOWU Ifedotun R.[1], ASAJU-GBOLAGADE Ayisat W. [2], GBOLAGADE Kazeem A.[3]
[1] Department. of Computer Science Federal College of Animal Health &Production Technology, Moor Plantation, Ibadan, Nigeria.
[2] Dept. of Computer Science, University of Ilorin, Nigeria.
[3] Dept. of Computer Science, Kwara State University, Faculty of Information and Communication Technology, Malete, Ilorin, Nigeria.
[1] iferoseidowu@fcahptib.edu.ng,[2] aisatwuraola@gmail.com, [3]kazeem.gbolagade@kwasu.edu.ng

Key words: Extracted features, intrusion detection system, machine learning classifiers, performance evaluation, selected features, stack ensemble.

Abstract**:** *Network technologies are becoming more digitalized and vulnerable to various cyberattacks, therefore creation of effective Intrusion Detection Systems (IDSs) is essential, particularly for high network traffic volumes and need to distinguish between normal and abnormal activities. In this study, four case models of IDS with varying feature values, base classifiers (Naïve Bayes, k-Nearest Neighbor, Logistic Regression) and) Meta classifier (Random Forest) trained and tested in a stack ensemble method with UNSWNB-15 dataset are examined. The Particle Swarm Optimization algorithm (PSO) serves as the foundation for the two sets of selecting and extracting features with enhanced with Residue Number System (RNS) forward conversion. The performance of the model is evaluated using classification accuracy, error rate, precision, specificity, F-score, sensitivity, and training time. Case D (NB + LR + KNN with RF) model performs best with PSO+RNS selected features, as evidenced by its accuracy of 97.47%, compared to 95.36% for PSO-based selected features respectively.*

1. INTRODUCTION

The hardware and software components of an intrusion detection system (IDS) cooperate to identify unusual occurrences that may indicate an attack will happen, is happening, or has happened. Keep in mind that we need to consider all three tenses: some products alert users to potential attacks, some alert users when an attack is underway, and some alert users as an attack aftermath becomes apparent. Hacking has grown to be a major concern in the internet world. Various techniques are used to combat these threats, but it is more important than ever to upgrade the conventional techniques from crude approaches such as manually updating blacklists and whitelists [1]. IDS is a well-established methodology for recognizing network-based attacks, but it is still in its infancy when it comes to tracking and identifying attacks primarily aimed at network-based applications on wireless sensor networks.

Host-based IDSs (HIDS), network-based IDSs (NIDS), and web-based application IDSs are the three categories of Intrusion Detection Systems (IDSs) that can be categorized according to the kinds of activities they look at over a network [2]. HIDS monitors particular hosts and sends out alerts based on host activity, log files, system calls, and application logs. NIDS, on the other hand, monitors every bit of network traffic. Should it detect any malicious activity that coincides with known network traffic, it will raise an alarm and notify the administrator so that necessary action can be taken. The system becomes more complex as the number of characteristics increases. It is therefore challenging for the IDS to examine the enormous volume of data.

Finding important and useful features is essential for intrusion detection in the field of information security. To create an appropriate and effective IDS, important aspects must be identified prior to pre-processing. Because the dataset includes a wide range of pertinent,

superfluous, and irrelevant features, it is challenging to identify the important features, which raises the computational complexity of intrusion analysis [3,4].

Algorithms for machine learning are programs that, in the absence of human input, can learn from data and get better with time. Learning tasks can involve discovering the hidden structure in unlabeled data, learning the function that translates input to output, or instance-based learning, which creates a class label for a new instance by comparing it to instances from the training data that were saved in memory. A specific instance is not abstracted from via instance-based learning. A brief review of various ensemble and single classifier approaches is provided, along with citations to studies that employed machine learning for intrusion detection [5].

Ensemble methods are a machine learning created by combining multiple base models using stacking, bagging or voting methods in machine learning to get the final prediction and enhance overall performance. The main causes of error in learning models are due to noise, bias and variance. Ensemble methods help to minimize these factors. These techniques aim to increase the machine learning algorithms' accuracy and stability. The number that appears the most frequently in a set of numbers is referred to as the mode in statistics. Several models are employed in this method to generate predictions for every data point. Every model's prediction is regarded as a separate vote. The final prediction is derived from the predictions made by most of the models [6,7]. An ensemble is itself a supervised learning algorithm, because it can be trained and then used to make predictions. Hence, the trained ensemble stands for a single hypothesis. Nevertheless, this hypothesis may not necessarily be contained in the hypothesis space of the models that it is based on. It can be demonstrated that ensembles can represent a wider range of functions with greater flexibility. In theory, this flexibility may allow them to overfit the training set more than few numbers of model ensemble would, but in actual fact, certain ensemble techniques like stacking aim reduce problems related to over-fitting of the training data [8]

The main contributions of this paper are summarized as follows:

i.) to reduce the feature set using the hybrid approach of Particle Swarm Optimization algorithm and Residue Number System approach.
ii.) to implement stack ensemble learning models on given feature of dataset.
iii.) to analyze the classifier models on the basis of accuracy, precision, error rate, training time, specificity, sensitivity and f-score.
iv.) This study also compares the proposed work with existing work on various performance parameters.

Related Works

Authors covered a wide range of machine learning techniques, including information-theory, statistical, clustering, and classification-based methods. Different machine learning algorithms are used in intrusion detection to help distinguish between normal and abnormal activities. Issues with different network intrusion detection datasets are briefly discussed. Future directions for research, such as collaborative IDSs, are indicated. However, a number of current machine learning-based IDS concepts are not fully described in their survey. Moreover, the authors have not offered any suggestions for upcoming machine learning methods [9].

Deep Belief Network (DBN) and State Preserving Extreme Machine Learning (SPELM) algorithms were proposed [10]. DBN is used to analyze and extract attack signatures from large volumes of network data and dynamic data. By differentiating between attack and normal nodes, SPELM increases the accuracy of attack detection. The authors also concluded that Deep Belief Networks are not as effective as State Preserving Extreme Machine Learning.

A hybrid intrusion detection approach based on upgraded FCM (Firebase Cloud Messaging) and SVM (Support Vector Machine) was proposed [6]. In order to reduce the complexity of large datasets and enhance the performance of the Support Vector Machine classifier, the pre-processed training datasets were clustered using Firebase Cloud Messaging while incorporating feature information gain ratio. For every cluster whose entropy surpasses a predetermined threshold, a Support Vector Machine classifier is constructed in order to identify the attack type even more precisely

An ensemble learning framework for intrusion detection was implemented, aimed at

improving IoTs. Chi-square was used for feature selection on ToN-IoT datasets, and voting and stacking techniques were used for LR, DT, RF, and KNN classifiers [11]. Because of the meta classification, the stack ensemble method performed better than the voting ensemble method. However, the complexity of the work increases the training time.

An enhanced IDS datasets in WSN using RNS - Feature Conversion with Stack Ensemble Technique was proposed in order to effectively and optimally minimize the feature size of the data dimensions, the Particle Swamp Optimization (PSO) approach was presented, thereafter Residue Number System (RNS) was used to further convert the selected features from the dataset using moduli of $\{2^n - 1, 2^n, 2^n + 1\}$ to residues in order to reduce large weighted number to several small numbers and enhance the power consumption and improve the time complexity further [12]. The effect of RNS extraction technique cannot be withdrawn with a clearcut difference of over 7% in variation.

Using an emulated and cyclo-stationary dataset from UNSWNB-15 &UGR'16, a stacking ensemble model for network intrusion detection using heterogeneous datasets was proposed [13]. With a meta classification approach, their model produced excellent predictions with high detection accuracy, but doing so requires more computing power. The low false alarm rate identifies malicious network traffic as typical.

A multitude of machine learning (ML) models, including six classifiers for both supervised and unsupervised learning, were proposed. Information Gain was used to select features based solely on the numerical attributes of the NSL-KDD dataset [14]. Random forest classifiers produced the highest accuracy levels; their application showed that single learning implementations resulted in lower accuracy.

Hybrid Approach

The UNSWNB-15 dataset is a good option for empirical studies to propose an intrusion detection system development using data-driven approaches. However, the two major issues, namely class imbalance and class overlap, need to be addressed prior to being employed for model development.

This dataset has nine types of attacks, Reconnaissance, Backdoor, DoS, Exploits, Analysis, Fuzzers, Worms, Generic, Shellcode but due to the imbalances of most of the attack in the dataset, all the attack are grouped into one major class thereby resulting into binary classification by the ensemble learning models used. The suggested two sets, feature selection and extraction based on the PSO algorithm and RNS for IDS are covered in this section.

The Z-Score standardization technique was used to preprocess the data for the model's initial phase. The PSO Algorithm is then used to select the features, and RNS forward conversion is used to extract the residual features from the PSO Algorithm.

Lastly, we created several stack ensemble models using Random Forest as the meta classification algorithm and Naïve Bayes, K-Nearest Neighbor, and Logistic Regression as the base classification algorithm.

In addition, the primary goal is to combine and input different supervised machine learning models into the stack ensemble model, which provides a way to improve the performance of the intrusion detection system (IDS) in addition to identifying malicious and benign network activity (0 represents normal and 1 indicates abnormal). The suggested architectural design is depicted in Fig. 1.

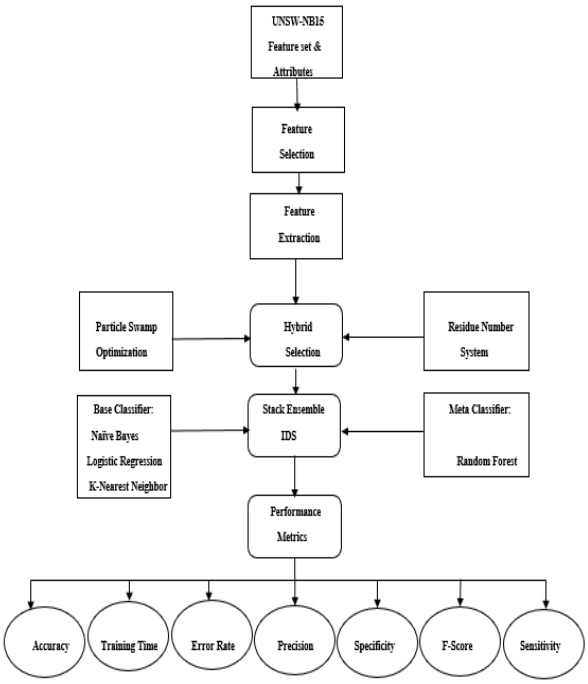

Fig. 1: Proposed Architecture Framework

We suggest classifying network traffic resulting from attacks as either abnormal or normal. In order to improve the IDS performance in the network devices, we implemented feature selection algorithms. Thus, in order to build an efficient intrusion detection system (IDS) that can identify attacks on wireless sensor networks using the UNSWNB-15 dataset, we looked at a range of machine learning algorithms and selected the most accurate and effective learner models.

2. METHODOLOGY

Using a combination of meta heuristic technique for feature selection, RNS dynamic power range method for feature extraction, and a comparative ensemble machine learning approach with a chosen combination of Base Classifiers (Naïve Bayes, K Nearest Neighborhood, Logistic Regression, and Random Forest as the dominant Meta Classifier), the stated objectives of this work are achieved in the construction of an IDS model. Four case models were compared and evaluated for the purpose of detecting and classifying intrusions in a wireless sensor network dataset. The models were based on the effects of PSO and RNS forward conversion utilizing special moduli set $\{2^n - 1, 2^n, 2^n + 1\}$ respectively, and without PSO or RNS. The extracted dataset was split into training and testing sets using two-fold cross validation. 25% of the training set were engaged, and the test passed as illustrated in figure 2, showing the training.

Fig. 2 illustrates the proposed method, which integrates feature selection and extraction classification to enhance classification performance and uses data sampling to address imbalanced data during training.

The process consists of two phases:training and prediction, the training phase consist of data normalization using the standardization method and the combined feature selection and extraction yielding 22 features and converted into residues.

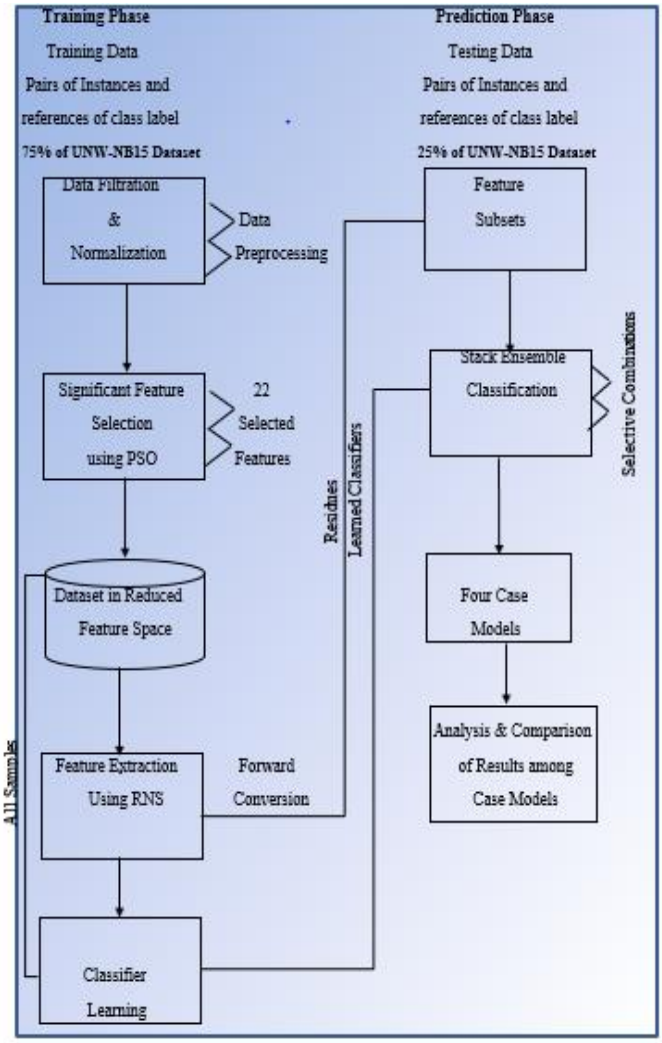

Fig. 2: Scheme of the Proposed Methodology

We built and evaluated four supervised models based on feature selection and data normalization. As the base and meta classifiers, respectively, are NB, LR, KNN, and RF.

Figure 3 shows the implementation with MATLAB version R2018b for a Meta-heuristic and RNS efficient technique for enhancing intrusion detection in wireless sensor networks using stack ensemble learning method.

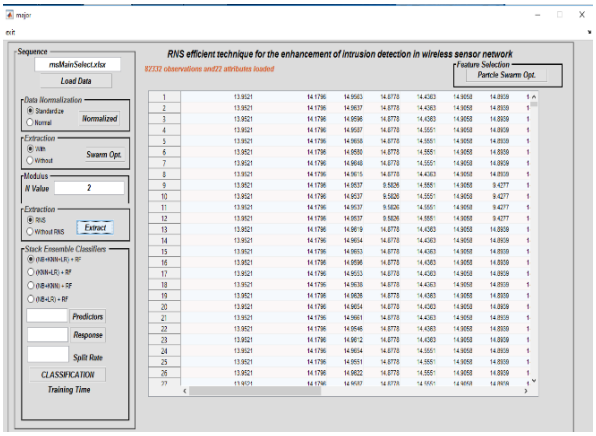

Fig. 3: RNS forward conversion into Residues

Additionally, we suggested four ensemble models to increase the suggested model's efficiency and boost the efficacy of our attack detection technique. Computational time, classification accuracy, error rate, specificity, sensitivity, f-score, and precision metrics are all included in the performance evaluation.

i. **Case A**: Naïve Bayes + KNN (Base Classifiers) with Random Forest (Meta Classifier).
ii. **Case B:** Naïve Bayes + Logistic Regression (Base Classifier) with Random Forest (Meta Classifier).
iii. **Case** C: KNN + Logistic Regression (Base Classifiers) with Random Forest (Meta Classifier).
iv. **Case D**: Naïve Bayes + Logistic Regression + KNN (Base Classifiers) with Random Forest (Meta Classifier).

3. RESULTS AND DISCUSSION

The experimental results are listed based on the various combination of the ensemble classification algorithm. The testing metrics are achieved using the True Positive rate (TP), False Positive (FP), True Negative (TN) and, False Negative (FN), accuracy and error rate as well. Machine learning statistical significance testing and training time were used to evaluate the results. For the most effective outcome for the training time, classification accuracy, error rate, specificity, sensitivity, f-score, and precision, the optimal settings are chosen. In addition to being highly interpretive, a confusion matrix can be used to estimate various other metrics. The confusion matrix shows how well a classification model performs when applied to a set of test data whose true values are known.

True Positive (TP): The instance where model correctly predict as Normal

False Negative (FN): The instance where model wrongly predict an Attack as Normal.

False Positive (FP): The instance where model wrongly predict Normal as an Attack.

True Negative (TN): The instance where model correctly predicts as Normal.

The other evaluation metrics that were used were sensitivity, also referred to as Recall or Detection rate, F-score, Accuracy, Error rate, Precision, and Specificity. The mathematical computations utilized to arrive at the evaluation parameters are explained [4,15].

Classification Accuracy is the fraction of correctly predicted samples as shown, or the model's ability to predict a test set given.

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP +TN+FP+FN}} \quad (1)$$

Error rate can be computed by dividing the total number of wrong predictions on the test set by the total number of correct predictions on the test set.

$$\text{Error Rate} = \frac{\text{FN+FP}}{\text{TP +TN+FP+FN}} \quad (2)$$

Precision is the proportion of true positives out of the total number of positives. Precision is the fraction of predicted positives outcomes that are actually positive as given,

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

Specificity is the ratio of true negatives to total negatives in the test set.

$$\text{Specificity} = \frac{\text{TN}}{TN+FP} \quad (4)$$

F-score is the harmonic mean of sensitivity and precision, with a higher score as a better model. It is similar to accuracy but stands better due to the fact that it seeks to strike a balance between precision and sensitivity particularly in cases where there is an uneven class.

$$\text{F} - \text{Score} = \frac{2(\text{Precision x Recall})}{\text{Precision +Recall}} \quad (5)$$

Sensitivity is the fraction of positive outcomes that were accurately predicted as given, or the percentage of positives that were correctly identified.

$$\text{Sensitivity} = \frac{\text{TP}}{TP+FN} \quad (6)$$

4. COMPARISON OF MODEL COMBINATIONS

The comparative analysis of the performance for each model under the three combinations is shown in table 1, graphically illustrated in figure 3.

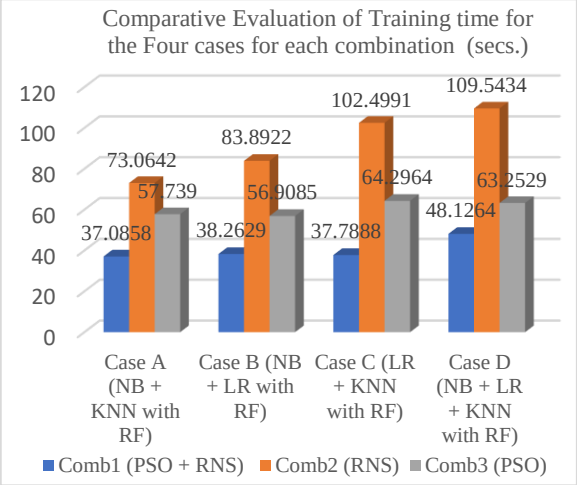

Fig. 3: Training Time of each model for three Combinations

Table 1: Training time (secs.) for each Case and three Combinations

CASE MODELS	Comb1 (PSO + RNS)	Comb2 (RNS)	Comb3 (PSO)
Case A (NB + KNN +RF)	37.0858	73.0642	57.739
Case B (NB + LR + RF)	38.2629	83.8922	56.9085
Case C (LR + KNN + RF)	37.7888	102.4991	64.2964
Case D (NB + LR + KNN +RF)	48.1264	109.5434	63.2529

Table 2: Case D (PSO+RNS) Metric results

Case D (NB + LR + KNN with RF)	Computational Time (secs.)	Classification Accuracy (%)	Error Rate (%)	Specificity (%)	Sensitivity (%)	F-Score (%)	Precision (%)
Comb1(PSO+RNS)	48.1264	97.4736	0.0253	0.97344	0.976324	0.972016	0.967745
Comb2 (RNS)	109.5434	90.6476	0.0935	0.90338	0.91027	0.897415	0.884919
Comb3 (PSO)	63.2529	95.3602	0.0464	0.943704	0.96573	0.949259	0.93334

Results indicated that Case A (NB + KNN +RF) outperformed all other models with the best computational time.

Furthermore, Case D (PSO+RNS) model has the most effective results for the following metrics with graphical illustration for specificity, sensitivity, F-score, precision in Fig. 4, classification accuracy in Fig. 5 and error rate in Fig. 6 as shown in table 2 respectively.

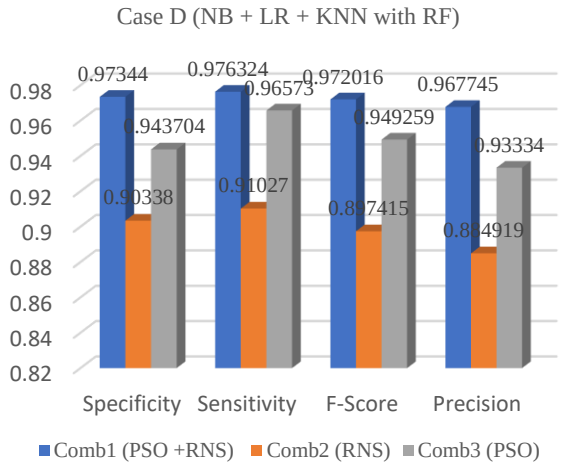

Fig. 4: Case D Metrics for three Combinations

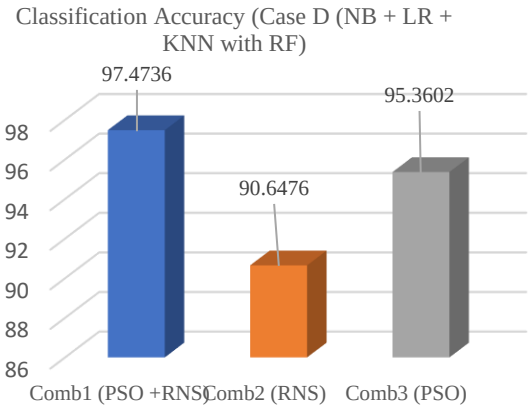

Fig. 5: Case D Classification Accuracy for three Combination

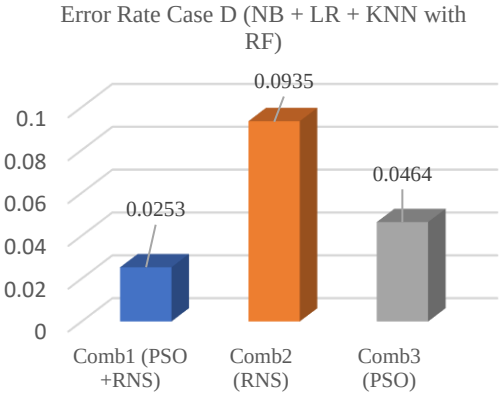

Fig. 6: Case D Error rate for three Combinations

5. CONCLUSION

The paper focuses on four supervised classifiers with different selected feature values on the UNSWNB-15 dataset evaluated.

The feature values were derived from Particle Swarm Optimization (PSO) and Residue Number System (RNS) feature selection and extraction approaches respectively.

The experimental results show the effectiveness of the combined approaches rather than a single approach may not necessarily improves the accuracy on the optimized and significant features, even with RNS approach alone, it could enhance the performance of the machine learning models.

The 22 features selected with the Inclusion of standardized data + PSO + RNS + Ensemble classifier shows a better performance compare to the absence of RNS.

6. REFERENCES

[1]. Smitha Rajagopal, Poornima Panduranga Kundapur,Katiganere Siddaramappa Hareesha, (2020) A stacking ensemble for network intrusion detection using heterogeneous dataset .Security and communication networks, vol. 20200, article ID 4586875,9 pages, 2020. Https://doi.Org/10.1155/2020/4586875

[2]. Rahman,Mashuqur&Kamruzzaman,Niton &Akter, Nasrin &Arbe, Nafija &Rahman,Md. Mahbubur (2021). Network Intrusion Detection using Hybrid Machine Learning Model,1-8. 10.1109/ICAECT49130.2021.9392483.

[3]. Mahmood RAR, Abdi A, Hussin M. Performance Evaluation of Intrusion Detection System using Selected Features and Machine Learning Classifiers. Baghdad Sci.J 2021 Jun. 20; 18(2(Suppl.):0884. https://bsj.uobaghdad.edu.iq/index.php/BSJ/article/view/6210

[4]. Alqahtani, H., Sarker, I. H., Kalim, A., Minhaz Hossain, S. M., Ikhlaq, S., & Hossain, S. (2020). Cyber intrusion detection using machine learning classification techniques. In N. Chaubey, S. Parikh, & K. Amin (Eds.), Computing Science, Communication and Security: First International Conference, COMS2 2020 Gujarat, India, March 26–27, Revised Selected Papers (pp. 121-131). (Communications in Computer and Information Science; Vol. 1235 CCIS). Springer, Springer Nature. https://doi.org/10.1007/978-981-15-6648-6_10.

[5]. Atawodi, S.I., (2019). Machine Learning Approach to Network Intrusion Detection System using K Nearest Neighbor and Random Forest, Master's Thesis

[6]. Islabudeen, M., Kavitha Devi, M.K. (2020). A Smart Approach for Intrusion Detection and Prevention System in Mobile Ad Hoc Networks Against Security Attacks. Wireless Pers Commun 112, 193–224. https://doi.org/10.1007/s11277-019-07022-5

[7]. Bing Xue, Mengjie Zhang, Will N. Browne (2014). Particle swarm optimisation for feature selection in classification: novel initialisation and updating mechanisms. Appl. Soft Comput. 18: 261–276

[8]. Singh, Amanpreet & Goyal, Akhil. (2018). Intrusion Detection System Based on Hybrid Optimization and using Neural Network: A Review. 10.13140/RG.2.2.23285.83681

[9]. Hamed. Bonab and Fazli. Can (2019) ''Less Is More: A Comprehensive Framework for the Number of Components of Ensemble Classifiers, IEEE Trans. neural networks Learn. Syst., vol. 30, no. 9, pp. 2735–2745, Sep. 2019, doi: 10.1109/TNNLS.2018.2886341''

[10]. Sun, Z.; Xu, Y.; Liang, G.; Zhou, Z. (2018). An Intrusion Detection Model for Wireless Sensor

Networks with an Improved V-Detector Algorithm. IEEE Sens. J. 2018, 18, 1971–1984.

[11]. Kunal Singh,James Mathai, (2019).“Performance Comparison of Intrusion
Detection System between DBN and SPELM Algorithm” at National Institute of Technical Teacher Training $ Research, Bhopal India.

[12]. Idowu, I.R. Asaju-Gbolagade, A.W and Gbolagade, K. A. (2023). Enhancement of Intrusion Detection Dataset in Wireless Sensor Network using RNS - Feature Conversion with Stack Ensemble Technique. University of Ibadan Journal of Science and Logics in ICT Research (UIJSLICTR), Vol. 10 No. 1, pp. 22 - 36. ©U IJSLICTR Vol. 10, No. 1, June 2023

[13]. Zhiyou Zhang, Peishang Pan (2019) “A Hybrid Intrusion Detection Method Based on Improved Fuzzy C-Means and SVM” at International Conference on Communication Information System and Computer Engineer [CISCE]

[14]. Preeti Mishra, Vijay Varadharajan, (2018). “A Detailed Investigation and Analysis of using Machine Learning Techniques for Intrusion Detection” Senior Member, IEEE, Uday Tupakula, Member, IEEE and Emmanuel S. Pilli, Senior Member, IEEE.

[15]. Alotaibi, Yazeed, and Mohammad Ilyas. (2023) "Ensemble-Learning Framework for Intrusion Detection to Enhance Internet of Things’ Devices Security" Sensors 23, no. 12: 5568. https://doi.org/10.3390/s23125568