MACHINE LEARNING MODEL FOR ANOMALY-BASED INTRUSION DETECTION USING RANDOM FOREST CLASSIFIER

Ebiesuwa Seun
Department of Computer Science, Babcock University, Ilisan-Remo, Nigeria
ebiesuwao@babcock.edu.ng

Nwachukwu Victor
Department of Computer Science, Babcock University, Ilisan-Remo, Nigeria
nwachukwu0116@pg.babcock.edu.ng

Falana Taye
Department of Computer Science, Babcock University, Ilisan-Remo, Nigeria
falana0097@pg.babcock.edu.ng

Adegbenjo Aderonke
Department of Computer Science, Babcock University, Ilisan-Remo, Nigeria
adegbenjoa@babcock.edu.ng

Dipo Tepede
Department of Computer Science, Babcock University, Ilisan-Remo, Nigeria
akin-tepede0115@pg.babcock.edu.ng

Adio Adesina
Department of Basic Sciences, Babcock University, Ilisan-Remo, Nigeria
adioa@babcock.edu.ng

Abstract

The dynamic nature of cyber threats creates a gap between the detection capabilities of existing anomaly-based Intrusion Detection System (IDS) and their inability to quickly adjust to new threat vectors. One of the most important challenges is achieving high accuracy in detecting anomalous signatures while mitigating false alarms. This proposed approach seeks to enhance the model's ability to recognize abnormal patterns in network behaviour by fusing the power of Random Forest with anomaly detection capabilities. The ensemble technique helps tackle the ever-changing nature of cyber threats. The model exhibits robustness to various intrusion scenarios and achieves excellent accuracy by combining predictions from different decision trees. By adding Random Forest to the ensemble, the protection system against changing cyber threats becomes more resilient and adaptable. Model evaluation was carried out using the NSL-KDD dataset, showcasing its effectiveness in detecting anomalies within network traffic. The results emphasize the model's potential to protect digital ecosystems from advanced cyberattacks by highlighting its capacity to identify minute departures from typical behaviour.

Keywords: Random Forest; Recursive feature elimination; Bootstrap

1. Introduction

Security has grown in importance as a concern in the online world due to the fast-growing number of individuals using the internet [Ishaque *et al.* (2023)]. Data has become increasingly valuable in today's world, making information security essential. It is now crucial to protect data and information against threats. This refers to preserving the data and information's Confidentiality, Integrity, and Availability [McCarthy (2023)]. The majority of cyberattacks aim to get past security measures and take advantage of important data and information [Nwachukwu *et al.* (2021)]. Network intrusion detection systems have used a variety of methods up to this point to identify anomalies in the network, including anomaly-based, signature-based, and hybrid approaches [Agrawal *et al.* (2022)] as shown in Figure 1.

IDSs that genuinely rely on heuristic rules are signature-based systems, such as Snort. NIDS vendors maintain and update a large database of known attack patterns, which is the foundation of signature-based intrusion detection. The system compares network traffic flowing across it with certain attack signatures. Security staff are alerted by the system if they find a match between the incoming traffic and a known attack signature. Because signature-based detection is based on a large historical attack pattern library, it is very good at identifying threats that have already been reported. This method's weakness is that it is unable to recognize new attacks for which a footprint is not known yet, which results in a high warning rate. The primary drawback of signature-depending approaches is that they necessitate constant database maintenance and can only identify known assaults [Thein *et al.* (2023).]. Behavior-oriented IDSs measure how much a system's operation deviates from what is deemed typical in order to analyze system operation and try to solve this issue. The first step in anomaly-based intrusion detection is to establish a baseline for typical network behavior. Over time, network traffic patterns are observed and analyzed to establish this reference. After setting up the baseline, the IDS looks out for changes to this predetermined standard on the network. An unusual or suspicious activity in network traffic may trigger alerts from the intrusion detection system. These alerts are generated when network traffic deviates significantly from the baseline.

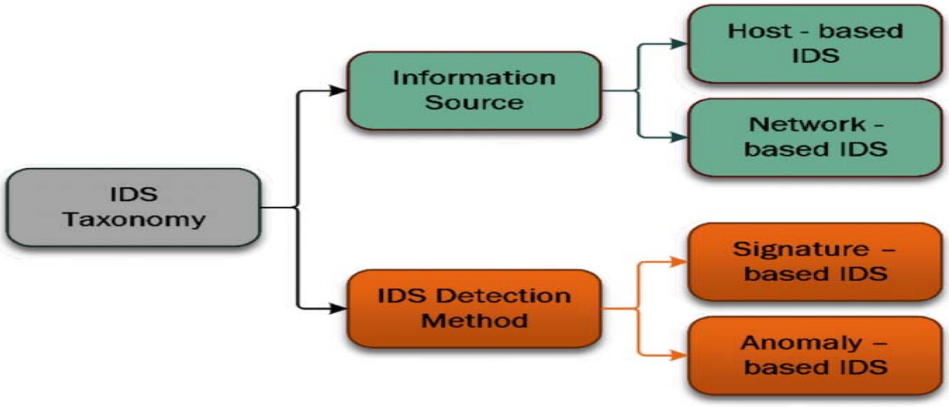

Figure 1: IDS Taxonomy [Al-Amiedy *et al.* (2019)]

Systems that can automatically identify which events correlate to normal or aberrant behaviors can be designed using statistical approaches and recently, machine learning (ML) algorithms [Khraisat *et al.* 2019).]. The technique of gaining knowledge from a vast volume of data is referred to as machine learning (ML). ML is the process of extracting knowledge from extensive datasets [Dua and Du (2016)]. The area of Anomaly-based IDS (AIDS) has made substantial use of machine-learning techniques. Numerous methodologies and algorithms, including nearest-neighbor approaches, neural networks, decision trees, association rules, clustering, and genetic algorithms, have been deployed to derive insights from intrusion datasets [Xiao *et al.* (2018), Tahsien *et al.* (2020)]. ML has been utilized to classify and identify traffic characteristics from network traffic datasets, and assist security experts in taking preventative action against malevolent attacker behavior [Zhang *et al.* (2023).]. Administrators take action depending on the classification of data, which is obtained from an external network and categorized as either attack or normal data using an IDS technique.

2. Related Works

Support Vector Regression (SVR) algorithm driven by a spare auto-encoder (SAE) was by [Preethi and Khare (2020)] as a model to predict network intrusion. The SAE effectively and significantly improved SVR's accuracy concerning attacks while requiring the least amount of training time. NSL-KDD was utilized in the studies designed to anticipate intrusions in order to investigate and quantify the efficacy of SAE-SVR. The SAE-SVR was found to have 97% better accuracy than the other models that were taken into consideration for comparison.

Autoencoder was deployed in [Ball and Drevin (2023)], for anomaly detection with network analysis features. There are 120,676 records total in the regular transactional features, 439 of which have been verified as fake. This means that only 0.36% of the data are attributable to the minority class, indicating a class imbalance. The disparity arises from the fact that an organization typically processes a far higher number of legitimate transactions than fraud. The imbalance in favor of the valid transactions results from the fact that significantly more data are gathered on valid transactions than on anomalous ones. A recall score of 100% and a precision score

of 26% were the final results. The model was expected to yield low precision scores because of the highly unbalanced distribution of the data.

To create effective coding for unlabeled data, the auto-encoder was utilized in [Manjunatha *et al.* (2023)]. In order to achieve dimensionality reduction, the auto-encoder learns a portrayal or encoding for a vast set of data by modeling the network to ignore irrelevant (noise) input data. On the KDDcup99 dataset, the Auto-encoder got a percentage of 85.78 for accuracy and $7\frac{1}{2}\%$ as false positive rate. With 99.24 percent accuracy, U2R packets were identified as malicious packets.

To reduce the extensive training duration, researchers introduced a Kernel-based Extreme Learning Machine (KELM) [Wang *et al.* (2021)]. Addressing the challenge of suboptimal classification results, often stemming from the random initialization of KELM's kernel parameters, an enhanced grey wolf optimizer (EGWO) was developed to fine-tune these attributes. Experiments conducted on the model demonstrated an accuracy score of 98% with the devised DBN-EGWO-KELM algorithm. The study in [Al-Hawawreh *et al.* (2018)] introduced a detection model for Control Systems capable of learning as well as validating using Internet packet data. This method involves multiple learning phases utilizing a deep feedforward neural network architecture and deep auto-encoder.

In [Andresini *et al.* (2021)], a deep learning approach was presented, providing an efficient technique to examine network traffic for activities that are not normal through the deployment of convolutional neural networks (CNNs). The core idea involves training a 2D CNN architecture by representing network flows as 2D images. An IDS working on a Fast Hierarchical Deep CNN was proposed by [Mendonça *et al.* (2021)]. First, packet data from a campus network was gathered. The training and preliminary testing stages were conducted using this dataset. Subsequently, the ML model was fed by the result of a PCA approach that used the Tree-CNN and the SRS activation function. Ultimately, the traffic was categorized as abnormal or not, and its validity was checked. The model got an accuracy of 98%.

To develop an IDS model capable of identifying various malicious attack types, [Samunnisa *et al.* (2023)] introduced an efficient hybrid model combining clustering and classification with threshold-based functions. The experiments involved testing the outcomes with two distinct threshold values, 0.01 and 0.5. The performance evaluation considered accuracy, false alarm ratio, and rate of detection. The suggested method demonstrated enhanced classification accuracy, reaching 92.7%. By combining the innate immune system and the adaptive immune system, the model put forward by [Dutt *et al.* (2020)] replicates the structure of the natural immune system (IS). Statistical Modeling based Anomaly Detection (SMAD), the first tier of the intrusion detection system (IDS), serves as an interface between the system and the innate immune system (IIS). In order to quickly find any flaws, it records the first network activity. Similar to how T-cells and B-cells work in the adaptive immune system, the second layer, known as adaptive immune-based anomaly detection (AIAD), examines the properties of dubious network packets to find anomalies. For efficient intrusion detection, the system retrieves pertinent data from the payload and header portions. Standard intrusion detection datasets KDD99 and UNSW-NB15 were used in experimental evaluations, coupled with live network traffic, achieving a true positive score of 96.04%.

Researchers are examining the potential for developing effective IDS as evidenced by a novel image-processing-centric NIDS framework introduced in [Siddiqi and Pak (2022)]. The framework proposes a three-tiered structure to create a more precise dataset. By reducing the number of features, this framework achieves a balance of low computational cost and impressive precision value. Data normalization was also achieved to enhance the interpretation. A Convolutional Neural Network (CNN) leveraging this representation can better grasp intricate and practical patterns in the dataset. To examine the potency as well as the versatility of this technique, three distinct datasets were employed, yielding an average accuracy of 98%. In the context of [Liang *et al.* (2023)], an intrusion detection system for Advanced Metering Infrastructure (AMI) systems was introduced based on federated learning (FL). The unique aspect of this approach involves transmitting only the model parameters to the data center during the training phase when the IDS is activated. The data center utilizes weight assignments and aggregation to disseminate the learning to each data concentrator, enabling collaborative learning. This method employs an enhanced deep neural network (DNN), and thorough testing yielded 99% score for accuracy.

In the research outlined in study [Kandhro *et al.* (2023)], a generative adversarial network was introduced as a method for detecting cyber threats in IoT. The findings indicated a performance enhancement ranging from 95% to 97% in accuracy when identifying various types of attacks. In order to combat ping flood assaults, [Almorabea *et al.* (2023)] developed an IoT network intrusion detection system. Using embedded devices, this framework creates an Internet of Things testbed that mimics two datasets: malicious ping flood attack traffic and legitimate ping traffic. The Zeek tool is used to extract features from the traffic that has been collected. The K-nearest neighbor approach, was found to have the highest detection accuracy on the testbed, with a 99.67% F1-score and an error rate of 0.33%.

This study [Dehlaghi-Ghadim *et al.* (2023)] utilized the ICS-Flow dataset, the dataset is designed for the evaluation of IDS using ML approaches. The network data comprises both typical and unusual network flows and packets that were taken from Industrial Control System (ICS) components that were simulated as well as networks that were emulated. The anomalies were introduced into the system through different types of cyberattacks. Three ML techniques were deployed to model an IDS utilizing the ICS-Flow dataset and experimental results show an accuracy of 98.1% for decision tree, 98.4% for RF, and 98.2% for ANN. The study presented in [Korium *et al.* (2023)] introduces a unique intrusion detection methodology tailored for cyberattacks. The authors propose an ML-based IDS designed to detect anomalous activities by analyzing unusual data flows in network traffic. Through extensive numerical experiments conducted on widely recognized datasets, both individually and in combination, the efficiency of the proposed technique is demonstrated, with an accuracy value of 99%.

A subset of the bagging technique, the RF performs better when there is noise and poor discrimination data, and it is not affected by parameter initialization [Dong *et al.* (2020), Reddy and Parvathy (2022)]. A collection of models is trained on various dataset subsets in the bagging process, and the result is produced by a combination of the outputs from each model [Sruthi (2023)] as illustrated in Figure 2 below. Decision trees serve as the foundational model in the random forest scenario [Raja *et al.* (2022)].

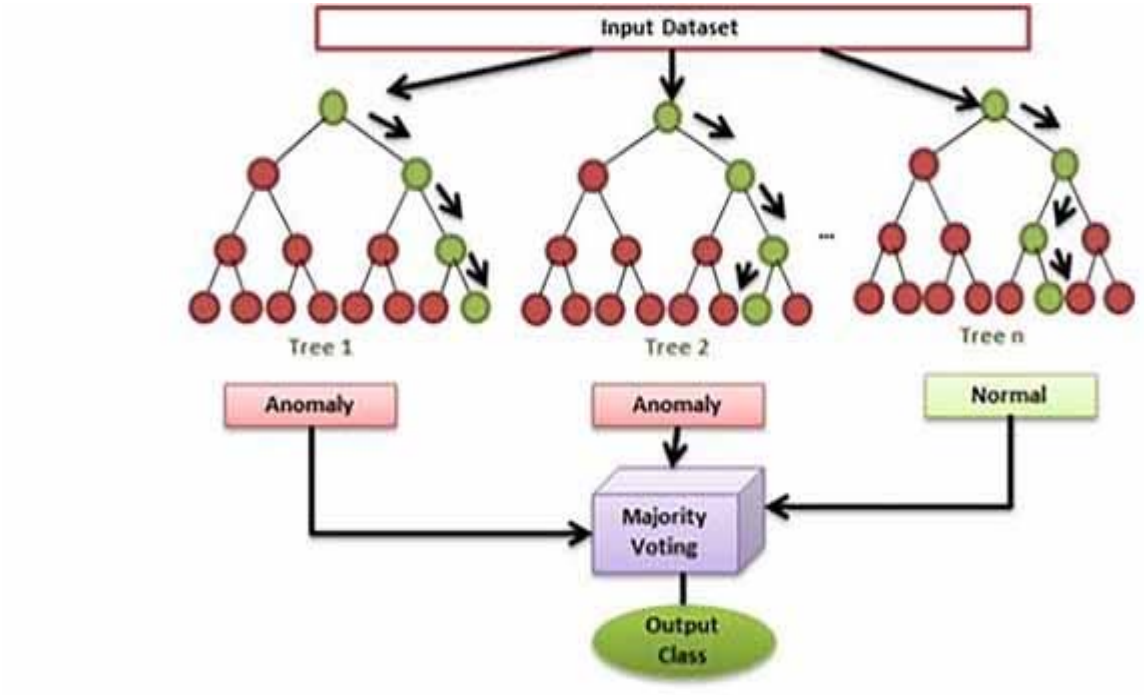

Figure 2: Random Forest Classifier [Rani *et al.* (2023)]

Intrusion detection systems using the Random Forest algorithm (RF) have been suggested by a number of authors, [Chen and Yuan (2022)] used RF in identifying malicious wireless network intrusions and the simulation results show a high probability of accurate intrusion identification. In [Marteau (2021)], a DiFF-RF algorithm was proposed for anomaly detection, and it was evaluated using the UCI repository dataset, the experiments show that DiFF-RF is computationally better than the SVM (support vector machines) baseline and auto-encoder algorithms. RF algorithm coupled with grid search was used in [Subbiah *et al.* (2022)] to develop a model for the detection of intrusion. The algorithm was implemented and tested using NSL-KDD. The authors concluded that the RF-based algorithm outperformed LDA (Linear Discriminant Analysis) as well as the CART (Classification and Regression Tree). RF classification was also implemented in [Tudosi *et al.* (2023)] for network filtering and detection of anomalous traffic. Performance evaluation shows the model's effectiveness in identifying threats such as SQL injection, port scanning, and DoS attacks with significant accuracy. RF classifier was found to be the most effective in [Choubisa *et al.* (2022)] for attack classification and in [Duraibi (2022)] combined with snake optimizer for intrusion detection as well as in [Varghese and Vivek (2023)] where the KDDcup 99 dataset was deployed to examine the efficiency of ensemble ML models as well as in [Serinelli *et al.* (2020)], RF with an accuracy of 97.93% outperformed SVM, XGBoost utilizing the NSL-KDD.

3. Methodology

This section presents and discusses the proposed architecture and its major components. The proposed architecture in figure 4 shows the key components: recursive feature elimination and the bagging algorithm which play vital roles in enhancing the overall efficiency of the model.

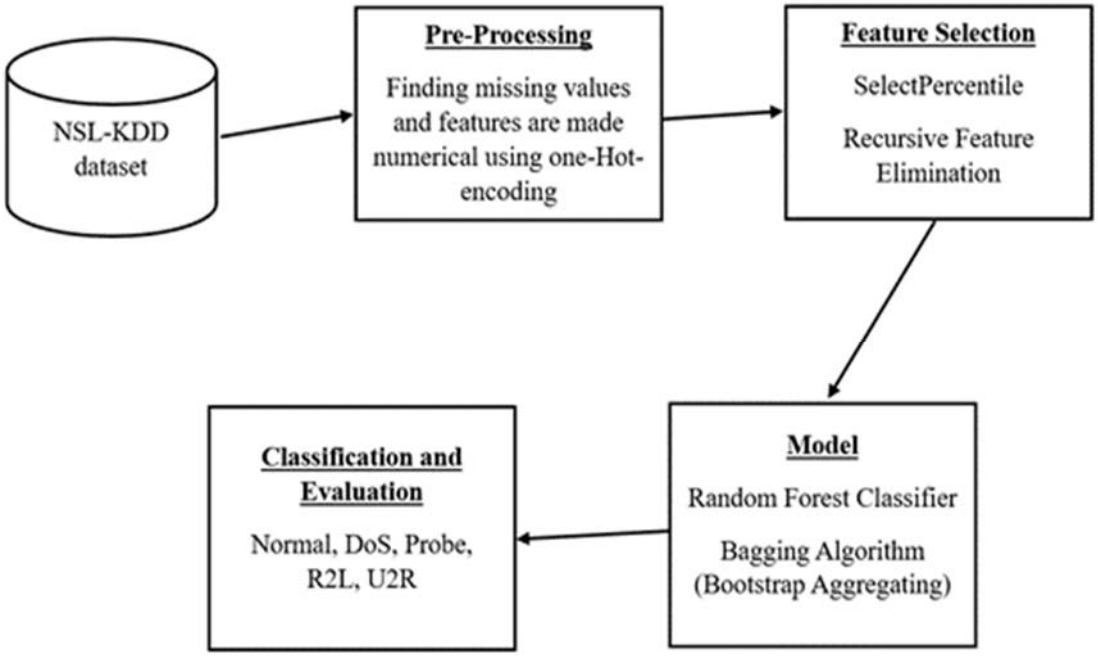

Figure 3: Proposed Model

3.1 *NSL-KDD Dataset*

In cybersecurity and network intrusion detection, the NSL-KDD is used more often. NSL-KDD is an improvement in the KDD 99 data set, largely employed in IDS assessments. Utilizing internet packet statistics as a basis, the KDD Cup 1999 data collection included both common and unique attack methods. The original KDD Cup 1999 data set had several shortcomings, including significant redundancy and low attack type variety. To overcome these issues, the NSL-KDD data set was created. Machine learning models and intrusion detection systems that are intended to identify network intrusions or cyberattacks are frequently tested and evaluated using the NSL-KDD data set. This makes it possible to assess how well intrusion detection systems identify particular kinds of attacks. There are 41 attributes total in the NSL-KDD data collection, comprising both features and labels. Models for intrusion detection are trained and assessed using these attributes. The labels show if a network connection is normal or connected to a certain kind of attack, while the features depict different aspects of network traffic.

3.2 *Pre-Processing*

One-hot encoding was utilized to transform all features to numbers. The features are scaled to prevent large-value features from overly influencing the outcomes.

3.3 *Feature Selection*

Starting with the complete dataset, the Recursive Feature Elimination (RFE) methodology is a wrapper feature selection method. Depending on which salient features are chosen, the dataset is ranked from best to worst using a ranking system that is essential to the RFE methodology. It updates the dataset after removing the least significant features from each iteration, repeating the procedure until the most significant features are chosen. The RFE method is cross-validated to ascertain the ideal feature count using Recursive Feature Elimination Cross-Validation (RFECV) using Algorithm I.

Algorithm I: RFECV

Input:
X: Dataset (n_samples, n_features) y: Target vector of shape (n_samples,) Scoring_Metric: Accuracy Stopping_Criterion: 13
Output: Selected_Features: Subset of features selected by RFECV
Pseudocode I: Recursive Feature Elimination Cross-Validation (RFECV)
1. Initialize a list to keep track of selected features: Selected_Features = [ ] 2. Initialize the best score as a negative infinity: Best_Score = -∞ 3. Initialize a variable to keep track of the optimal number of features: Optimal_Num_Features = 0 4. while (number of selected features < n_features) and (number of selected features < Stopping_Criterion): a. For each feature in X: i. If the feature is not in Selected_Features: - Train the model using the features in Selected_Features + [the current feature] - Calculate the cross-validated score using Scoring_Metric - If the score is better than Best_Score: - Update Best_Score to the current score - Update Optimal_Num_Features to the number of features in Selected_Features + 1 b. Select the feature that achieved the best score. c. Add the selected feature to Selected_Features. 5. The process continues until the number of selected features reaches the stopping criterion or all features are selected. 6. The final Selected_Features list contains the subset of features that achieved the best cross-validated score. 7. Return Selected_Features.

13 features were selected and this will now be used to train the model.

3.4 *Model*

Bootstrap Aggregation (Bagging) is the ensemble ML technique deployed in building the model using a Random Forest classifier. The key principle behind bootstrap aggregation is to train numerous instances of a particular model using randomized datasets and then collate their predictions to compute a final prediction.

3.5 Sampling

A set of bootstrap samples $B_1, B_2, B_3, \dots, B_{10}$ from the NSL-KDD dataset was created. Each bootstrap sample created has randomly selected N data points from the dataset shown in "Eq. (1)"

$$B_i = \{D_{i1}, D_{i2}, \dots, D_{iN}\} \quad (1)$$

For each sample B_i, a decision tree T_i with feature randomization was trained. At each split of T_i, a set of m features from F will be selected using "Eq. (2)".

$$m = |F_i| = \sqrt{F}; F = 13 \quad (2)$$

The feature selection f where $f \in F_i$, it uses the Gini impurity reduction (information gain) given in "Eq. (3)" as

$$split(node) = argmax(f \in F_i)\ Gain(node, f) \quad (3)$$

3.6 *Prediction Aggregation*

After training the 10 trees, their predictions are collated and a final prediction is made using "Eq. (4)". The class with the majority vote becomes the final prediction.

$$c_{final} = argmax(class_{count}(T_i(x)\ for\ i = 1\ to\ 10)) \quad (4)$$

3.7 k-fold cross-validation

The NSL-KDD dataset's performance evaluation employs the K-fold cross-validation technique where k = 10. The model undergoes a training phase on $k-1$ of these subsets and is then assessed on the subsets left. This iterative process is done 10 times, with each round using a different subset for the testing phase. To obtain a more precise measure of the technique's capability to examine new data, the performance metric, be it accuracy, precision, recall, or F1-Score (represented as n), is averaged over the k folds using the formula in "Eq. 5".

$$val_n = \frac{1}{k}\sum_{i=1}^{k} n_i \qquad (5)$$

4. Results

4.1 *Model Performance (DoS, Probe, R2L, U2R)*

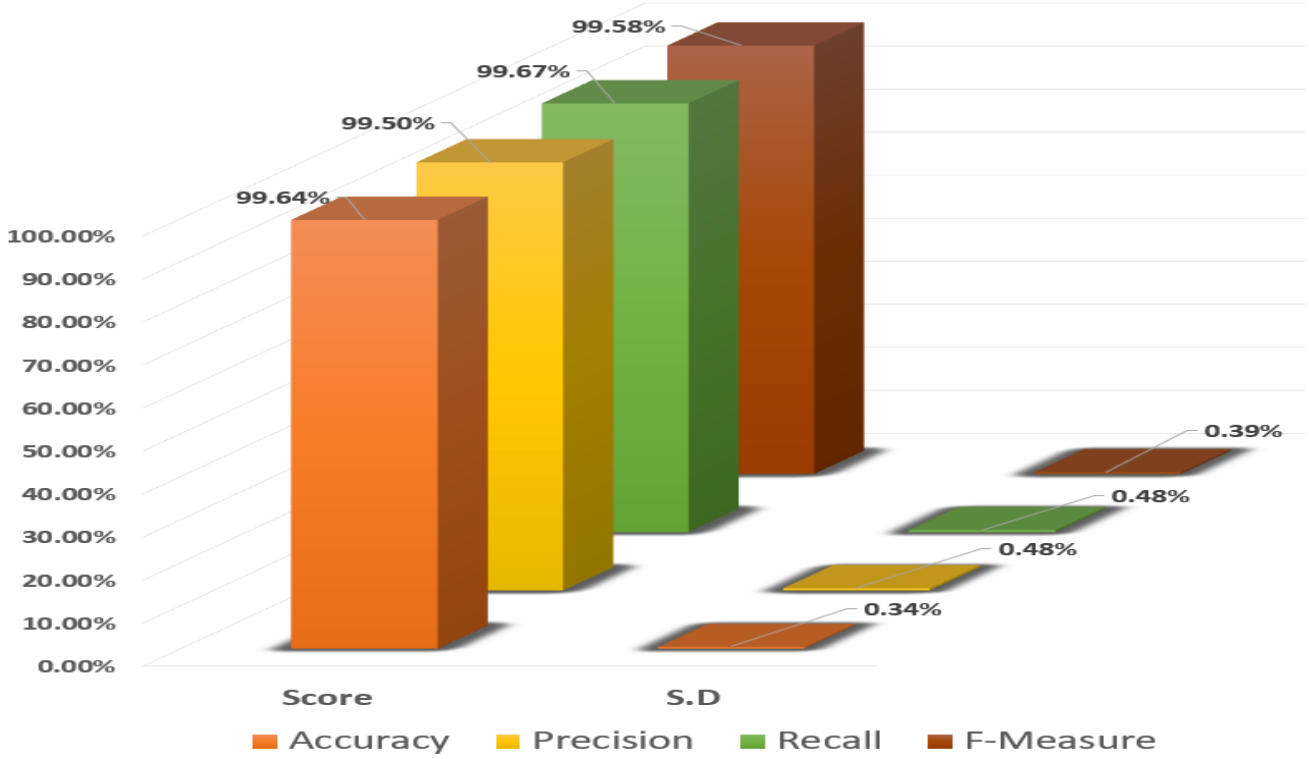

Figure 4: Model performance for DoS

Figure 3 shows the model's performance when after k-fold validation. The scores for accuracy, precision, recall and f-measure are plotted alongside their respective standard deviations. With a standard deviation of 0.34%, the accuracy is 99.64%. This shows that there is very little variation in accuracy across different folds and that the model has an impressive value for DoS prediction accuracy. Having a standard deviation of 0.48%, and precision of 99.5%. This shows that the model has excellent accuracy performance, which is crucial for reducing false positive errors. The model exhibits a high recall of 99.67%, with a minimal standard deviation of 0.48%. This indicates the model's effectiveness in minimizing false negative errors, a critical aspect in intrusion detection. The F1 score, with a low standard deviation of 0.39%, reaches 99.58%. This balanced statistic, considering both precision and recall, showcases the model's consistent performance with minimal variability.

The performance of the model for specific attack types—Probe, R2L, and U2R—is visually represented in Figure 4, Figure 5, and Figure 6, respectively.

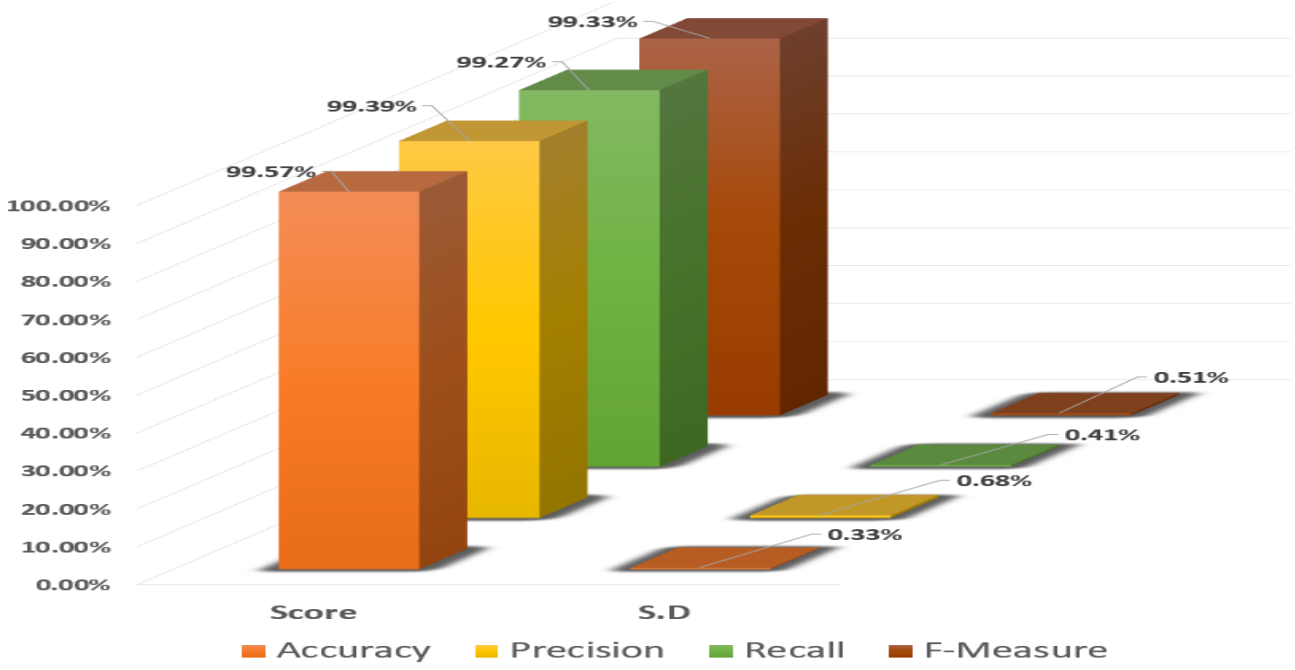

Figure 5: Model performance for probe

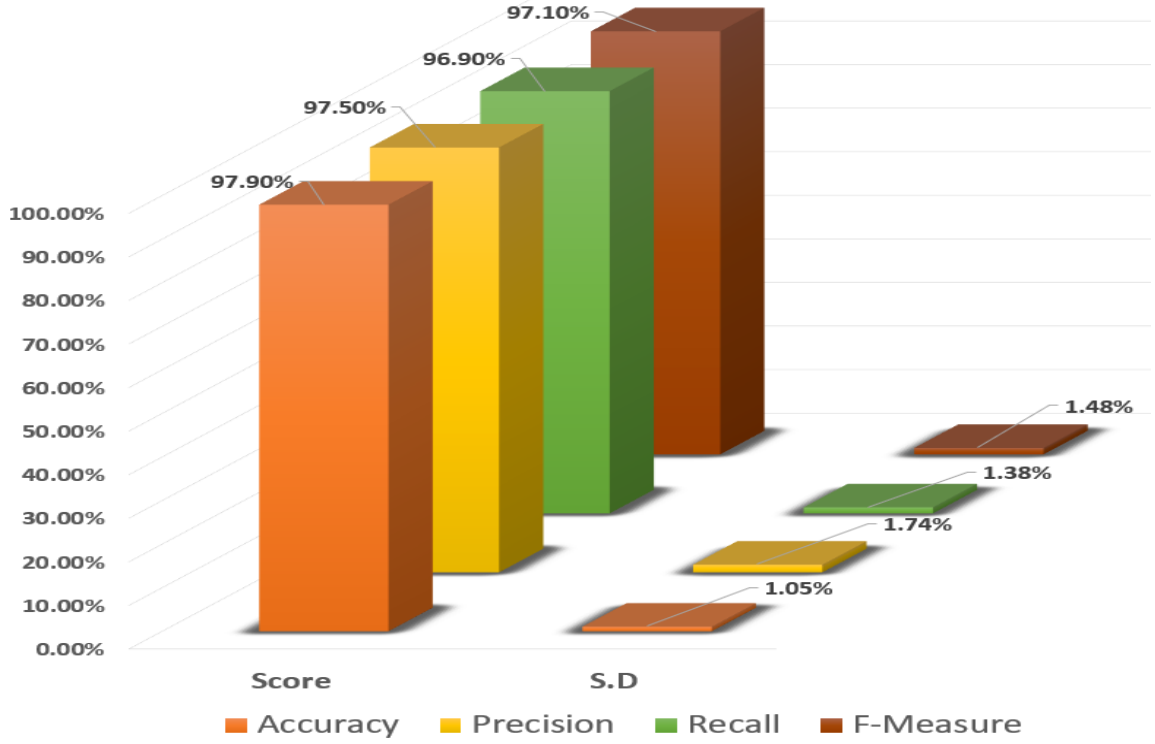

Figure 6: Model performance for R2L

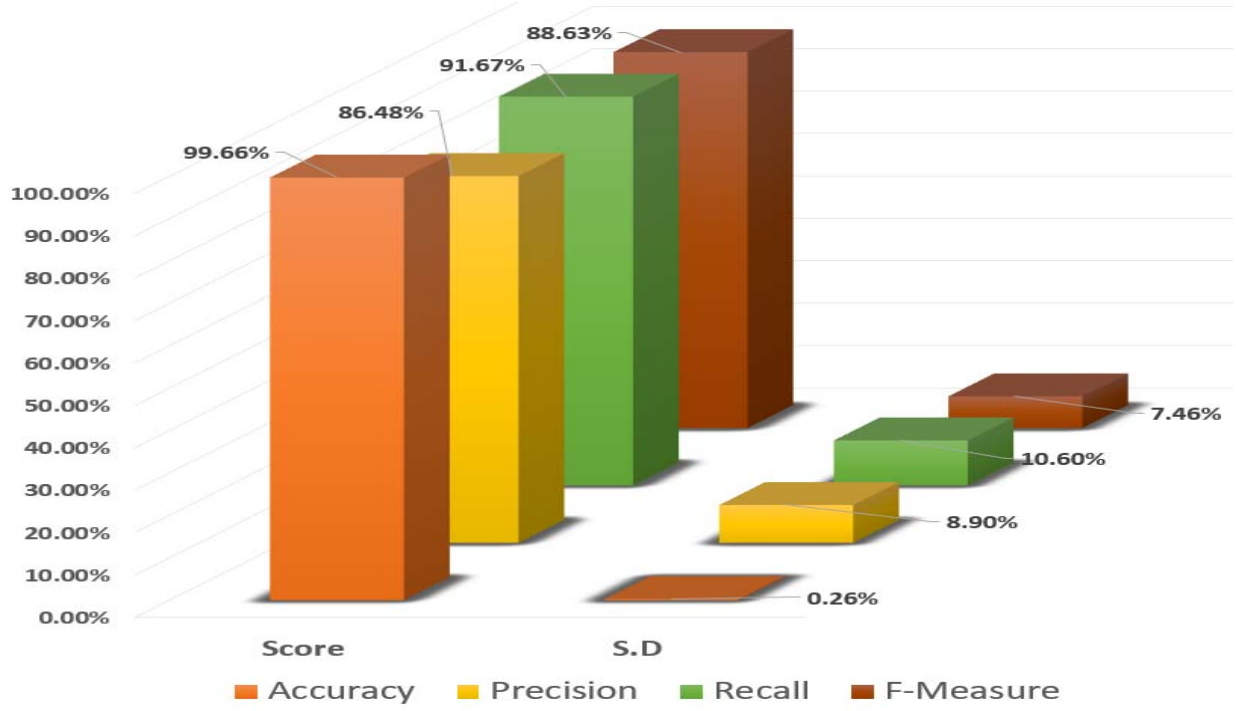

Figure 7: Model performance for U2R

4.2 Recursive Feature Elimination with Cross-Validation (RFECV) graph

The model's performance across varying numbers of features in the dataset is visually represented in Recursive Feature Elimination with Cross-Validation (RFECV) graphs. Specifically, Figure 8, Figure 9, Figure 10, and Figure 11 illustrate this for DoS, Probe, U2R, and R2L, respectively. The horizontal X-axis displays the number of selected features, while the vertical Y-axis represents the cross-validation score for accuracy. These graphs provide insights into how the model's accuracy is influenced by the number of features for different types of attacks.

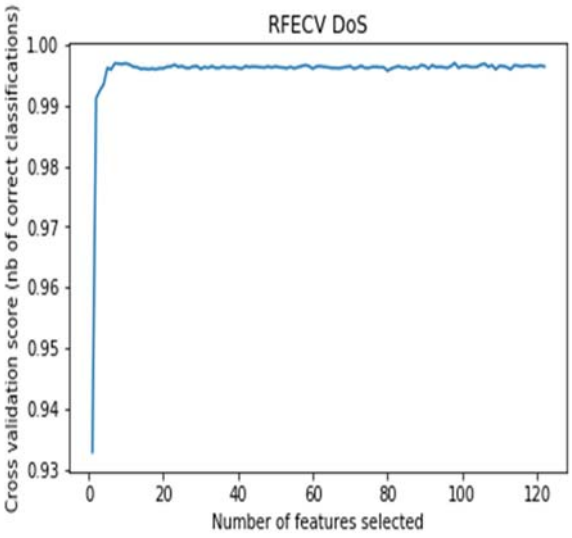

Figure 8: RFECV for DoS

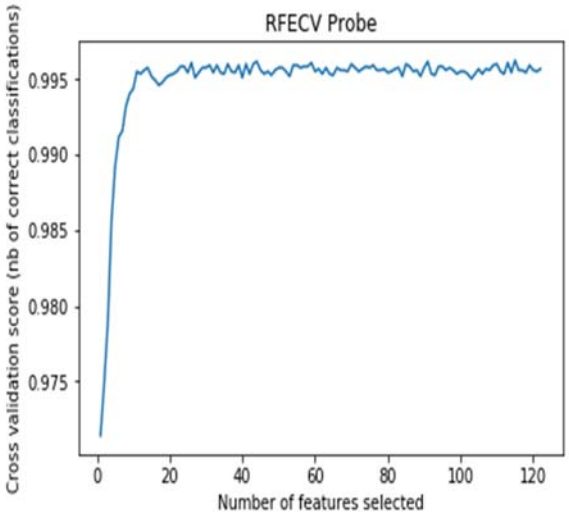

Figure 9: RFECV for Probe

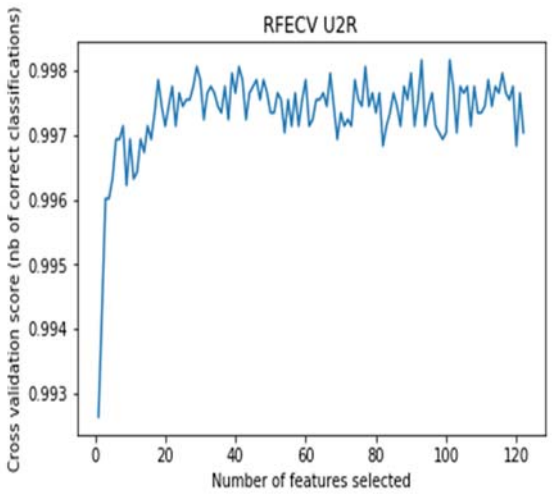

Figure 10: RFECV for U2R

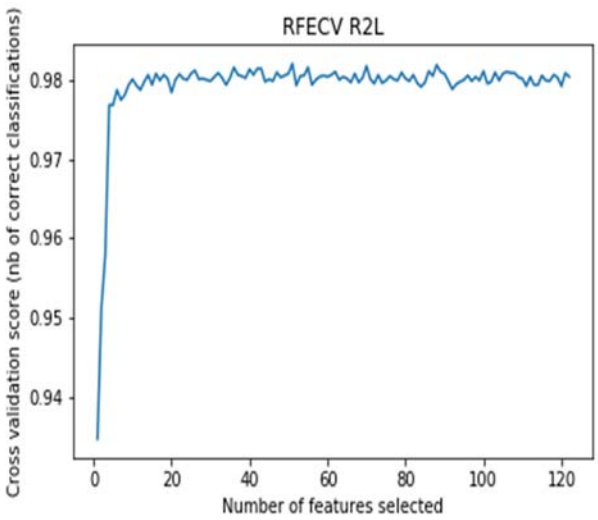

Figure 11: RFECV for R2L

5. Conclusion and future work

By combining the advantages of each individual decision tree, Random Forest's integration into the ensemble framework demonstrates a synergistic method that produces an intrusion detection system that is both accurate and resilient. The empirical assessments validate the model's ability to identify anomalies in network data, highlighting its capacity to identify minute variations that may be signs of impending cyberattacks. Because of the ensemble's flexibility in responding to various intrusion situations and Random Forest's resilience, the proposed model is positioned to be a useful and proactive instrument for cybersecurity professionals.

It is impossible to overestimate the significance of dynamic and adaptive intrusion detection systems given the ongoing evolution of cyber threats in the area of sophistication as well as complexity. This presented Machine Learning technique not only fills this gap but also adds significant knowledge to the continuing discussion on the usefulness of using ensemble methods in practice, especially with Random Forest.

The foundation for future efforts to improve anomaly-based intrusion detection systems is laid by this research, which also highlights the need for more investigation into ensemble approaches for cybersecurity resilience.

Conflicts of interest: All authors have no conflicts of interest in this paper.

Reference

[1] Agrawal, S., Sarkar, S., Aouedi, O., Yenduri, G., Piamrat, K., Alazab, M., Bhattacharya, S., Maddikunta, P. K. R., & Gadekallu, T. R. (2022). Federated Learning for intrusion detection system: Concepts, challenges and future directions. Computer Communications, 195, 346–361. https://doi.org/10.1016/j.comcom.2022.09.012

[2] Al-Amiedy, T. A., Anbar, M., Alqattan, Z. N. M., & Alzubi, Q. M. (2019). Anomaly-based intrusion detection system using multi-objective grey wolf optimisation algorithm. Journal of Ambient Intelligence and Humanized Computing, 11(9), 3735–3756. https://doi.org/10.1007/s12652-019-01569-8

[3] Al-Hawawreh, M., Moustafa, N., & Sitnikova, E. (2018). Identification of malicious activities in industrial internet of things based on deep learning models. Journal of Information Security and Applications, 41, 1–11. https://doi.org/10.1016/j.jisa.2018.05.002

[4] Almorabea, O. M., Khanzada, T. J. S., Aslam, M. A., Hendi, F. A., & Almorabea, A. M. (2023). IoT Network-Based Intrusion Detection Framework: a solution to process ping floods originating from embedded devices. IEEE Access, 11, 119118–119145. https://doi.org/10.1109/access.2023.3327061

[5] Andresini, G., Appice, A., & Malerba, D. (2021). Nearest cluster-based intrusion detection through convolutional neural networks. Knowledge Based Systems, 216, 106798. https://doi.org/10.1016/j.knosys.2021.106798

[6] Ball, R. D., & Drevin, L. (2023). Anomaly detection using autoencoders with network analysis features. ORiON, 39(1). https://doi.org/10.5784/39-1-711

[7] Chen, Y., & Yuan, F. (2022). Dynamic detection of malicious intrusion in wireless network based on improved random forest algorithm. 2022 IEEE Asia-Pacific Conference on Image Processing, Electronics and Computers (IPEC). https://doi.org/10.1109/ipec54454.2022.9777557

[8] Choubisa, M., Doshi, R., Khatri N., & Kant Hiran, K. (2022). A Simple and Robust Approach of Random Forest for Intrusion Detection System in Cyber Security. 2022 International Conference on IoT and Blockchain Technology (ICIBT), Ranchi, India, 2022, pp. 1-5, doi: 10.1109/ICIBT52874.2022.9807766

[9] Dehlaghi-Ghadim, A., Moghadam, M. H., Balador, A., & Hansson, H. (2023). Anomaly Detection Dataset for industrial control systems. IEEE Access, 11, 107982–107996. https://doi.org/10.1109/access.2023.3320928

[10] Dong, L., Xing, L., Liu, T., Du, H., Mao, F., Han, N., Li, X., Zhou, G., Zhu, D., Zheng, J., & Zhang, M. (2020). Very high resolution remote sensing imagery classification using a fusion of random forest and Deep Learning Technique—Subtropical area for example.

IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 13, 113–128. https://doi.org/10.1109/jstars.2019.2953234
[11] Dua, S., & Du, X. (2016). Data mining and machine learning in cybersecurity. In Auerbach Publications eBooks. https://doi.org/10.1201/b10867
[12] Duraibi, S. (2022). An Intelligent Approach for Intrusion Detection using Snake Optimizer and Random Forest. 2022 International Conference on Computational Science and Computational Intelligence (CSCI), Las Vegas, NV, USA, 2022, pp. 1016-1021, doi: 10.1109/CSCI58124.2022.00180.
[13] Dutt, I., Borah, S., & Maitra, I. K. (2020). Immune System Based Intrusion Detection System (IS-IDS): a proposed model. IEEE Access, 8, 34929–34941. https://doi.org/10.1109/access.2020.2973608
[14] Ishaque, M., Johar, M., Khatibi, A., & Yamin, M. (2023). A novel hybrid technique using fuzzy logic, neural networks, and genetic algorithm for intrusion detection system. *Measurement: Sensors*, 100933. https://doi.org/10.1016/j.measen.2023.100933
[15] Kandhro, I. A., Alanazi, S. M., Ali, F., Kehar, A., Fatima, K., Uddin, M., & Karuppayah, S. (2023). Detection of Real-Time malicious intrusions and attacks in IoT-empowered cybersecurity infrastructures. IEEE Access, 11, 9136–9148. https://doi.org/10.1109/access.2023.3238664
[16] Khraisat, A., Gondal, I., Vamplew, P., & Kamruzzaman, J. (2019). Survey of intrusion detection systems: techniques, datasets and challenges. Cybersecurity, 2(1). https://doi.org/10.1186/s42400-019-0038-7
[17] Korium, M. S., Saber, M., Beattie, A., Narayanan, A., Sahoo, S., & Nardelli, P. H. J. (2023). Intrusion detection system for cyberattacks in the Internet of Vehicles environment. Ad Hoc Networks, 153, 103330. https://doi.org/10.1016/j.adhoc.2023.103330
[18] Liang, H., Liu, D., Zeng, X., & Ye, C. (2023). An intrusion detection method for advanced metering infrastructure system based on federated learning. Journal of Modern Power Systems and Clean Energy, 11(3), 927–937. https://doi.org/10.35833/mpce.2021.000279
[19] Manjunatha, B. A., K, A., Naresh, E., Pareek, P. K., & Reddy, K. T. (2023). A network intrusion detection Framework on sparse deep denoising autoencoder for dimensionality reduction. Research Square (Research Square). https://doi.org/10.21203/rs.3.rs-3107463/v1
[20] Marteau, P. (2021). Random Partitioning Forest for Point-Wise and Collective Anomaly Detection—Application to network Intrusion Detection. IEEE Transactions on Information Forensics and Security, 16, 2157–2172. https://doi.org/10.1109/tifs.2021.3050605
[21] McCarthy, J. A. (2023). Foundational PNT Profile: https://doi.org/10.6028/nist.ir.8323r1
[22] Mendonça, R. V., Teodoro, A. a. M., Rosa, R. L., Saadi, M., Carrillo, D., & Nardelli, P. H. J. (2021). Intrusion detection system based on fast hierarchical deep convolutional neural network. IEEE Access, 9, 61024–61034. https://doi.org/10.1109/access.2021.3074664
[23] Nwachukwu V, Ikerionwu C., John-Otumu A. (2021). An Enhanced Model for Mitigating DoS Attacks on Linux Servers using IPTables and Bash scripts. International Journal of Advanced Trends in Computer Applications (IJATCA) Volume 8, Number 2, July - 2021, pp. 68-74 ISSN: 2395-3519.
[24] Preethi, D., & Khare, N. (2020). Sparse auto encoder driven support vector regression based deep learning model for predicting network intrusions. Peer-to-Peer Networking and Applications, 14(4), 2419–2429. https://doi.org/10.1007/s12083-020-00986-3
[25] Raja, S. P., Sawicka, B., Stamenković, Z., & Mariammal, G. (2022). Crop prediction based on characteristics of the agricultural environment using various feature selection techniques and classifiers. IEEE Access, 10, 23625–23641. https://doi.org/10.1109/access.2022.3154350
[26] Rani, D., Gill, N. S., Gulia, P., Arena, F., & Pau, G. (2023). Design of an intrusion detection model for IoT-Enabled smart Home. IEEE Access, 1. https://doi.org/10.1109/access.2023.3276863
[27] Reddy, P. D., & Parvathy, L. R. (2022). Prediction Analysis using Random Forest Algorithms to Forecast the Air Pollution Level in a Particular Location. 2022 3rd International Conference on Smart Electronics and Communication (ICOSEC). https://doi.org/10.1109/icosec54921.2022.9952138
[28] Samunnisa, K., Kumar, G. S. V., & Madhavi, K. (2023). Intrusion detection system in distributed cloud computing: Hybrid clustering and classification methods. Measurement: Sensors, 25, 100612. https://doi.org/10.1016/j.measen.2022.100612
[29] Serinelli, B. M., Collen, A., & Nijdam, N. A. (2020). Training Guidance with KDD Cup 1999 and NSL-KDD Data Sets of ANIDINR: Anomaly-Based Network Intrusion Detection System. Procedia Computer Science, 175, 560–565. https://doi.org/10.1016/j.procs.2020.07.080
[30] Siddiqi, M. A., & Pak, W. (2022). Tier-Based optimization for synthesized network intrusion detection system. IEEE Access, 10, 108530–108544. https://doi.org/10.1109/access.2022.3213937
[31] Sruthi, E. R. (2023, October 26). Understand random forest algorithms with examples (Updated 2023). Analytics Vidhya. https://www.analyticsvidhya.com/blog/2021/06/understanding-random-forest/
[32] Subbiah, S., Anbananthen, K. S. M., Thangaraj, S., Kannan, S., & Chelliah, D. (2022). Intrusion detection technique in wireless sensor network using grid search random forest with Boruta feature selection algorithm. Journal of Communications and Networks, 24(2), 264–273. https://doi.org/10.23919/jcn.2022.000002
[33] Tahsien, S. M., Karimipour, H., & Spachos, P. (2020). Machine learning based solutions for security of Internet of Things (IoT): A survey. Journal of Network and Computer Applications, 161, 102630. https://doi.org/10.1016/j.jnca.2020.102630
[34] Thein, T. T., Shiraishi, Y., & Morii, M. (2023). Personalized federated learning-based intrusion detection system: Poisoning attack and defense. Future Generation Computer Systems. https://doi.org/10.1016/j.future.2023.10.005
[35] Tudosi, A. D., Graur, A., Balan, D. G., Dan Potorac A. & Tarabuta, R. (2023). Distributed Firewall Traffic Filtering and Intrusion Detection Using Snort on pfSense Firewalls with Random Forest Classification. 2023 46th International Conference on Telecommunications and Signal Processing (TSP), Prague, Czech Republic, 2023, pp. 101-104, doi:10.1109/TSP59544.2023.10197784
[36] Varghese, N., & Vivek, R. (2023). The Efficiency of Ensemble Machine Learning Models on Network Intrusion Detection using KDDCup 99 Dataset. 2023 IEEE International Conference on Contemporary Computing and Communications (InC4), Bangalore, India, 2023, pp. 1-5, doi: 10.1109/InC457730.2023.10263037.
[37] Wang, Z., Zeng, Y., Liu, Y., & Li, D. (2021). Deep Belief Network Integrating Improved Kernel-Based Extreme Learning Machine for network Intrusion detection. IEEE Access, 9, 16062–16091. https://doi.org/10.1109/access.2021.3051074
[38] Xiao, L., Wan, X., Lu, X., Zhang, Y., & Wu, D. (2018). IoT Security Techniques based on Machine learning: How do IoT devices use AI to enhance security? IEEE Signal Processing Magazine, 35(5), 41–49. https://doi.org/10.1109/msp.2018.2825478
[39] Zhang, L., Liu, K., Xie, X., Bai, W., Wu, B., & Dong, P. (2023). A data-driven network intrusion detection system using feature selection and deep learning. Journal of Information Security and Applications, 78, 103606. https://doi.org/10.1016/j.jisa.2023.103606

Authors' Profile

Dr. Seun Ebiesuwa holds a first degree in Computer Engineering from the Lagos State University. Dr. Ebiesuwa bagged a Master's and a Ph.D degree in Computer Science from Babcock University. He is a lecturer in the Department of Computer Science, Babcock University and he specializes in Information Systems. Dr. Seun Ebiesuwa has over forty-five academic publications in internationally peer-reviewed journals of Computer Science in the areas of Artificial Intelligence, Data Analytics, Machine Learning and Medical Informatics. He has made several academic presentations in Workshops, Seminars, Symposiums and Conferences.

Nwachukwu Victor is a PhD scholar in Computer Science (Artificial Intelligence) at Babcock University, Ilishan-Remo, Nigeria. He bagged his First degree and M.Sc. degree in Information Technology from Federal University of Technology, Owerri, Nigeria. His research interests include, Machine learning algorithms, deep learning techniques, cybersecurity.

Taye Oluwaseun Falana is a Post Graduate student at Babcock University, Ilishan-Remo, in Ogun State, Nigeria in Computer Science Department. His area of specialization is Network and Telecommunication with interest in the Internet of Things. He is a Senior Manager, Operations and Technologies in a Fintech Company located in Lagos, Nigeria. He has over 20 years professional experience in Teaching, Banking/Finance, Information/IT Security, Cybersecurity and particularly, IT Project Management. He bagged his first degree in Electrical and Electronics Engineering from the Federal University of Technology, Akure, Ondo State, Nigeria.

Adegbenjo Aderonke A. is working as a Lecturer in the Department of Computer Science at Babcock University, Ilishan Remo, Nigeria. A graduate of Babcock University, Master's degree in Computer Science department at the University of Ibadan. A Master in Philosophy in Computer Science at Babcock University. Also, Doctor of Philosophy in Computer Science at Babcock University. Her research interests are Telecommunication and Networking. She is a member of Computer Professionals (Registration Council of Nigeria).

Dipo Tepede is a recognized expert in Process and Project Management, holding a Master Black Belt in Lean Six Sigma, along with certifications as a Professional Project Manager and Business Analyst from the Project Management Institute, and as a SAFe Scrum Master from Scaled Agile. He obtained his undergraduate degree from Obafemi Awolowo University and an MBA from the University of Gavle, Sweden. His academic journey expanded to Arden University in the United Kingdom, where he explored a Master of Science in Data Analysis. Currently pursuing a Ph.D. in Cybersecurity at Babcock University, his research focuses on Machine Learning, Data Analytics, and Artificial Intelligence, showcasing his commitment to advancing knowledge in the field.

Adio Adesina PhD is an Applied Mathematician from Nigeria. He currently works as an Associate Professor in the Department of Basic Sciences, Babcock University Nigeria, since 2020. He has published over thirty research papers in local and international journals and conferences. His main research interests include Differential & Integro-differential equations and Operations Research. He has twenty-six years of university teaching and research experience.