International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Network Intrusion Detection by Optimize Feature Engineering Using Hybridization of GWO and Nonlinear Activation Function

[1]Rupali*, [2]Kamal Malik

Submitted: 06/02/2024 **Revised**: 13/03/2024 **Accepted**: 20/03/2024

Abstract: Social networks, connections with others, and a revolution in our lives have all been made possible by the internet. Sharing business and personal information, however, puts people and organizations at risk. An important problem is the security of data, and intrusion detection systems (IDS) are essential for shielding users from malevolent network attacks. Traditional rule-based systems find it difficult to adjust to evolving cyber threats. Techniques for machine learning (ML) have become a practical way to increase the efficacy and efficiency of intrusion detection. A thorough understanding of machine learning (ML)--based intrusion detection is heavily sought after by researchers and practitioners who want to build stronger, more effective defenses against cyber attacks. This research improves the class imbalance problem in the KDD-99 dataset by optimizing non-linear feature weights using Grey Wolves and activating Leaky RElu with a back propagation technique. Weighted features boost feature information while reducing noise across all features. During experiment analysis, class-wise accuracy is represented by a confusion matrix and an ROC curve for complete performance analysis. In terms of results, accuracy improves by 2-3%, precision by 3-4%, and precise recall improves by 5-6% on average across classes. In the experiment, a class imbalance in three classes improved by 3-4%.

Keywords: *Intrusion detection, Machine Learning.*

1. Introduction

Owing to the rapid progress in Internet technology, there has been a growing concern regarding the surge in cyber-attacks in recent years. In 2019, over 32% of corporations and 22% of nonprofits in the United Kingdom reported encountering a cyber breach or assault [1]. An Intrusion Detection System (IDS) is a method used to detect and identify various types of attacks. Despite the notable achievements of the implemented Intrusion Detection methods, there is a growing apprehension regarding the enhancement of current approaches or the introduction of novel ones [1,2].IDS has been utilized for several years to examine network traffic and promptly detect any malevolent activity or potential dangers. An Intrusion Detection System (IDS), similar to a firewall, serves the purpose of safeguarding confidentiality, integrity, and availability. These are the primary objectives that potential attackers aim to compromise. To assess the effectiveness of an IDS, specific criteria have been established to define the desirable attributes that a competent IDS should contain. The mentioned criteria encompass correctness, little overhead, excellent efficiency, and completeness [3,4].IDS can be categorized into two primary types according to their detection methodology: Signature-Based IDS (likewise called Misuse Detection) and Anomaly-Based IDS (sometimes referred to as Behavioural Detection). Signature-based IDS analyzes system activity or network traffic by comparing it to pre-established signatures or patterns of known assaults. When a match is detected, these systems generate alerts or execute specified actions [5,6].These systems exhibit efficiency in detecting familiar attacks but may encounter difficulties in identifying innovative or previously undiscovered threats. Anomaly-based intrusion detection systems (IDS) employ statistical techniques, ML algorithms, or expert knowledge to identify anomalies from established patterns of normal behavior. Alerts are sent when actions depart significantly from these models. Hybrid Intrusion Detection Systems (IDS) integrate components from both signature-based and anomaly-based detection methods to enhance the accuracy and scope of detection.IDS can also be categorized according to their extent of deployment. To identify and recognize any potentially harmful or unauthorized activity, like denial-of-service attacks, port scanning, or unauthorized entry attempts, network-

[1]*Research Scholar, Rupali_dhir@hotmail.com*
CT University, Ludhiana
[2]*Professor Department of CSE*
Kamal.malik91@gmail.com
CT University, Ludhiana

based intrusion detection systems, or IDS, are made to actively monitor network traffic. They are commonly positioned in key locations in the network's framework, like switches, routers, or networking gateways. Host-based IDS operate on individual hosts or endpoints, overseeing system logs, file integrity, and other activities related to the host. They offer insight into the activity taking place on particular hosts, enhancing the network-focused perspective provided by NIDS. Distributed IDS are comprised of a network of interconnected IDS sensors strategically placed throughout an organization's infrastructure. These sensors work together to identify and respond to security occurrences. Organizations frequently utilize a blend of these solutions to offer extensive coverage and multi-layered defense against cyber threats.

Machine learning (ML) plays a dynamic role in Intrusion Detection Systems (IDS) because it can adjust to changing threats, identify unfamiliar attacks, minimize false positive rates, and efficiently process extensive datasets. Machine learning algorithms have the ability to detect abnormalities and deviations from typical patterns, hence minimizing instances of incorrect identifications [7-10]. Additionally, they possess the capability to manage intricate feature areas, rendering them appropriate for high-speed networks and expansive enterprise contexts. Machine learning has the ability to generate models that represent typical patterns based on past data, and can detect anomalies as potential instances of unauthorized access.

Machine learning-based IDS can be combined with automated response systems to promptly address identified threats. ML models possess the capability to undergo continual updates and retraining in order to effectively adjust to alterations in the threat landscape. In general, machine learning improves the precision, flexibility, and effectiveness of detecting various cyber threats, hence assisting organizations in their defense.ML algorithms play a crucial role in IDS by identifying abnormal or harmful actions in computer networks or systems. Some often used algorithms in machine learning are Decision Trees (DTs) [11,12], Random Forests (RFs) [13,14], Support Vector Machines (SVMs) [15,16], Neural Networks (NNs) [17,18], Ensemble Learning [19,20], and Clustering Algorithms [21,22]. Decision Trees are hierarchical structures utilized for classification tasks, whereas RFs are ensemble learning techniques that amalgamate numerous decision trees and predictions. Support Vector Machines are resilient and efficient at identifying incursions in intricate and multi-dimensional data. Neural Networks, precisely Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs), are being more and more utilized for intrusion detection, as evidenced by references [23-25]. Clustering methods, like DBSCAN and K-means clustering, detect groups of similar data points and categorize exceptional values as anomalies. Ensemble Learning techniques, such as boosting and bagging, amalgamate many base learners to enhance predictive performance. The selection of an algorithm is contingent upon the characteristics of the data, its intricacy, and the desired level of performance.

Challenges and future opportunities

Problems like data quantity and quality, adversarial attacks, imbalanced datasets, explain ability, scalability, interpretability, and performance are encountered in machine learning for IDS. These challenges may impede confidence in IDS and complicate the process for security analysts to verify and respond to alerts that are generated. Potential future areas of focus encompass hybrid methodologies, elucidating explainable artificial intelligence (XAI), enhancing adversarial robustness, facilitating incremental updates and online learning strengthening edge computing and Internet of Things (IoT) security. Ensuring both the accuracy and abundance of data is essential for constructing a resilient IDS based on ML. Imbalanced datasets have the potential to introduce bias in ML models, favoring the majority class and resulting in subpar detection capabilities for minority intrusions. Explain ability and Interpretability are crucial for security analysts to verify and respond to generated alarms. Continuous online learning and incremental upgrades are crucial for ensuring the long-term efficacy of IDS. The significance of IoT security and edge computing is growing, necessitating future research to concentrate on creating streamlined and effective models for edge devices with limited resources.

Gaps in previous work

The fundamental challenge in intrusion detection is detecting unnoticed attacks, thus it is evident not to learn about attacks that will occur in the future, therefore effective learning and feature engineering are highly crucial for the classification and detection of intrusions according to the following gaps found

in prior work.

- Previous studies ignored class imbalance and used binary classification, which is not a realistic condition.
- Feature engineering reduces accuracy by increasing entropy through feature selection and feature loss.

Contribution of research

In this research, we do not claim to identify unseen attacks; instead, we enhance imbalanced learning classes by the following.

- Improve feature weighting by AI-based non-linear mapping using an activation function to identify deep non-linear features without increasing resources.
- Improve class overlap using random forest and voting-based optimization.

This work is divided into four parts: an introduction which briefly reviews the topic, motivation, challenge, gaps, and research contribution. The second half includes a review of past studies. The final section explains the proposed approach using an algorithm, flow charts, and a mathematical model. The fourth chapter examines the findings of various performance scenarios using the ROC curve and confusion matrix.

2. Background and Related Work

The significance of cybersecurity in diverse fields has led to substantial research on IDS that employ machine learning approaches. Below is a compilation of relevant literature in the domain of intrusion detection employing ML techniques.Sajja, et.al [5] improve the efficacy of IDS by employing learning-based algorithms and rule-based approaches for the classification and detection of intrusions. ML algorithms and rule-based approaches are evaluated through the utilization of standardized datasets such as KDDcup 99. Kilincer, et.al [7] conducted a comprehensive analysis of Intrusion Detection Systems (IDS) by examining multiple datasets such as UNSW-NB15, ISCX-2012, CSE-CIC IDS-2018, NSL-KDD, and CIDDS-001. The utilization of ML techniques such as DT, SVM, and KNN has led to favorable outcomes and holds promise for the application of Artificial Intelligence (AI)-based IDS.Mousavi, et.al [11] employ a stepwise feature elimination technique to identify 16 essential features for networking visits. By integrating an ant colony algorithm with an ensemble of decision trees, this approach obtains a remarkable average Matthews correlation coefficient of 0.91, and an accuracy rate of 99.92%.Maseer, et.al [8]showed a comprehensive analysis of existing research on AIDS, utilizing various datasets and methodologies to discover appropriate ML algorithms for AIDS. The study utilizes 10 widely used supervised and unsupervised machine learning methods, such as ANN, DT, k-NN, NB, RF, and SVM. The performance of 31 machine learning models for AIDS prediction is evaluated by assessing precision, accuracy, recall, and F-Score. Additionally, the study considers testing and training time as an important factor.

Table1. A review of recent research COVID-19 prediction using deep leaning

Reference/Year	ML Models	Prediction category	Dataset	Limitation	Results
Rawat, et.al [5]/2022	DNN, Unsupervised feature learning	ML intrusion detection methods employing deep neural networks and unsupervised feature learning.	NSL-KDD dataset	Conventional ML-IDS rely on manual feature engineering. However, the use of deep neural networks and unsupervised feature learning can alleviate this requirement. Nevertheless, the process of designing,	The DNN utilizing 15 features obtained by Principal Component Analysis (PCA) proved to be the most efficient approach for modeling.

				training, and tuning these systems can present additional difficulties.	
Mousavi, et.al [11]/2022	Decision Trees (DTs)	Intelligent intrusion detection	KDD99 dataset	An IDS depends on a substantial volume of data, frequently marked by redundancies and interference, hence diminishing its stability and precision.	A highly effective classifier is developed by employing an ant colony (ACO) algorithm and an ensemble of decision trees, resulting in an impressive average Matthews correlation coefficient of 0.91 and accuracy rate of 99.92%.
Deore & Bhosale [24]/2022	RNN	Intrusion detection system for feature reduction	NSL-KDD dataset	Designing and training RNNs for feature reduction in intrusion detection requires meticulous attention to capture pertinent information. Nevertheless, established methodologies for feature selection and dimensionality reduction in the domain of IDS are presently limited.	The algorithm categorizes data into attack and non-attack groups, improving the ranking of features and reducing the number of features. This, in turn, decreases the time required for preprocessing tasks such as information acquisition and dataset correlation.
Jakka & Alsmadi [22]/2022	Ensemble Model: Bagging and Boosting	Intrusion Detection System Classification	KDD cup 99 datasets	The primary obstacles in this domain pertain to formulating regulations for forecasting malware in	This study presents a novel ensemble model for IDS that achieves an impressive overall accuracy

				unfamiliar regions, handling intricacy, and striking a balance between stringent detection precision and efficiency demands.	of approximately 99.49%. The model was constructed and assessed utilizing the data set KDD99, which includes an assemblage of 42 unique features.
Sajja, et.al [5]/2021	Neural Networks (NN), Random Forest and SVM	Intrusion detection and classification	KDD 99 dataset	Several machine learning algorithms are trained on historical data, which may not encompass the complete spectrum of potential future dangers.	The SVM algorithm has demonstrated excellent performance. The accuracy of SVM surpasses that of Neural Networks and Random Forest.
Kilincer, et.al [7]/2021	SVM, KNN, DT	IDS prediction and classification	UNSW-NB15, NSL-KDD ISCX-2012, CIDDS-001, and CSE-CIC IDS-2018data sets	The restricted quantity of data sets included in these investigations hampers the capacity to precisely assess the efficacy of different attack categories.	The UNSW-NB15 dataset exhibits performance rates for all classifiers that surpass those documented in the literature, thereby providing evidence of an effective categorization procedure.
Maseer, et.al [8]/2021	Supervised: DT, ANN, NB, k-NN, RF, CNN, and SVM. Unsupervised: expectation-maximization (EM), k-means, and self-organizing	Anomaly-based intrusion detection systems	CICIDS2017 dataset	The CICIDS2017 dataset, similar to numerous others, might not encompass all possible attack scenarios that occur in real-world situations.	The DT-AIDS, NB-AIDS, and k-NN-AIDS models get superior results and exhibit a higher proficiency in identifying online attacks compared to other models that display

	maps (SOM)				inconsistent and worse outcomes.
Ajdani & Ghaffary [14]/2021	Random Forest (RF)	Enhancing intrusion detection	KDD-Cup'99 andUNSW-NB15Datasets	The PSO and Random Forest algorithms can encounter scaling problems when applied to data streams with enormous volumes, mostly due to the computational burden and memory limitations.	The suggested approach greatly enhances the precision and rate of learning of the PSO algorithm, attaining a detection rate of 97% and 75.94% of accuracy rate
Gu & Lu [16]/2021	SVM with naïve Bayess	Intrusion detection framework feature embedding.	UNSW-NB15, CICIDS2017, NSL-KDD, Kyoto 2006+	Data sets often exhibit class imbalance, a common occurrence in real-world circumstances.	The suggested intrusion detection approach has exhibited strong performance on many datasets, attaining high accuracy rates of 98.92% on CICIDS2017, 98.58% on Kyoto 2006+, 99.35% on NSL-KDD, and 93.75% on UNSW-NB15.
Shaukat,et.al [9]/2020	NB and Decision Tree (J48) algorithm	Anomaly detection in IDS	ISCX 2012 dataset	Every method possesses unique advantages and disadvantages, rendering it impractical to generalize the optimal technique for detecting IDS.	The results indicate that J48 exhibits increased accuracy and a reduced false alarm rate, albeit at the cost of longer training time. J48 surpasses NB in terms of precision, recall, accuracy, and f1-score. The J48 model achieved accuracies of

	0.9999 and 0.9993 when using 68 and 8 features, respectively.

Research Methodology

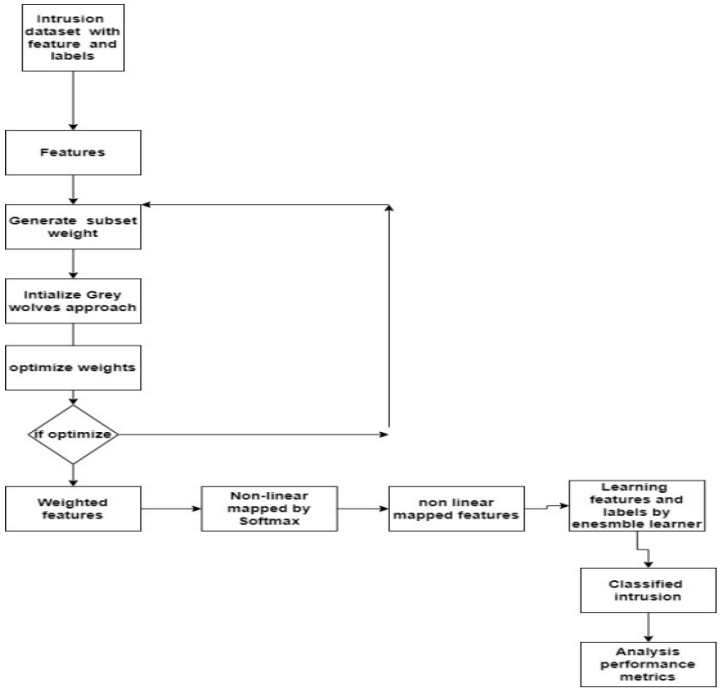

Figure 1: Proposed Flowchart

GWO Mapping

For client (i), the GWO update can be represented as:

$$W_i^{(t+1)} = W_i^{(t)} - \eta \nabla L\left(W_i^{(t)}, D_i\right) * t_l * h_i \quad \text{.....(1)}$$

$$t_l = W_{hxxi} + W_{hhhi-1+\,b_h} \quad \text{.....(2)}$$

$$h_i = sigmoid\ (t_i) \quad \text{.....(3)}$$

- $W_i^{(t)}$: Weights of the local model for client (i) at iteration (t).

-η: Learning rate.

- $\nabla L\left(W_i^{(t)}, D_i\right)$: Gradient of the loss function (L) with respect to the weights, evaluated on the local dataset (D_i).

$t_l * h_i$:non -linear distributed mapping of features

2. updated features mapped Optimization

-Weight Aggregation: The central server aggregates the updated weights from all clients.

- The aggregated update can be represented as:

$$- W^{(t+1)} = \frac{1}{N}\sum_{i=1}^{N} W_i^{(t+1)} * L(y_i : \hat{y}_i) \quad \text{.....} \quad \text{...(4)}$$

$$L(y_i : \hat{y}_i) \leftarrow -\sum_i \sum_j y_{ij} \log\left(\hat{y}_{ij}\right) + \left(1 - y_{ij}\right) \log\left(1 - \hat{y}_{ij}\right) \text{....(5)}$$

Where (N) is the number of Features.

$L(y_i : \hat{y}_i)$: loss function which depend on different aggregation of attacks

3. Features Weight Update

- **Global Model Update**: The global model weights are updated.

$$- W_{global}^{(t+1)} = W^{(t+1)} * \delta_i * \theta_i \quad \text{.....(7)}$$

$$\delta_i \leftarrow \text{dL/d } \theta_i \text{.....(8)}$$

$$\theta_i \leftarrow \theta_i \eta + \delta_i \text{.....(9)}$$

Weight update: Calculate the partial derivative with respect to θ_i.

4. Weight inundation by Random Forest

$$f_t = f\ (\sigma\big(W_f[h_{t-1}, x_t] + b_f\big) * tanhtanh(W_c[h_{t-1}, x_t] + b_c)\ \ ) \text{.....(10)}$$

$$h_t = f(W_{hh}h_{t-1} + W_{xh}x_t + b_h) \text{.....(11)}$$

Here, the bias term b_f is applied to the hidden layer, W_f represents the weight on the hidden layer, x_t is the input-based vector, and f_t represents the forget gate vector.

$$y_t = W_{hy}h_t + f_t. \text{ .....(12)}$$

ALGORITHM: Intrusion Detection Using Grey Wolf Optimizer (GWO) and Ensemble Learning

INPUT: Intrusion Detection Dataset with features (X) and labels (Y)

STEP 1: Dataset Preparation

1.1. Divide dataset into features (X) and labels (Y)

- Features: packet size, IP addresses, protocol type, etc.

- Labels: normal or types of attacks

1.2. Initialize Features Matrix for Grey Wolves

- Generate initial population of grey wolves (solutions)

- Each wolf's position represents a potential solution in the feature space by equations 1,2 and 3

STEP 2: Optimize with Grey Wolf Optimizer (GWO)

2.1. Define GWO with social hierarchy: alpha (α), beta (β), delta (δ), omega (ω)

2.2. Define Fitness Function based on IDS performance metrics (accuracy, precision, recall)

2.3. Optimize Feature Weights

- Adjust weights of features based on their importance in distinguishing attacks by equation 4,5 and6

STEP 3: Apply Weighted Features to Softmax Non-linear Function

3.1. Weight Features

- Apply optimized weights to features

3.2. Non-linear Transformation

- Feed weighted features into softmax function to convert scores to probabilities by equation 7,8 an9

STEP 4: Ensemble Learner and Classification Model

4.1. Apply Ensemble Learning

- Use non-linearly transformed features for ensemble learning model

4.2. Build Classification Model

- Model capable of distinguishing between normal behavior and intrusion attacks by equation 10,11, and 12

STEP 5: Model Evaluation

5.1. Evaluate Model with Test Set

- Use unseen data for evaluation

5.2. Compute Performance Metrics

- Accuracy: Proportion of true results among total cases

- Precision: Ratio of true positives to all positive predictions

- Recall: Ratio of true positives to all actual positives

- F-Score: Harmonic mean of precision and recall

OUTPUT: Performance metrics (Accuracy, Precision, Recall, F-Score)

Result And Analysis

Table 1. Analysis the GWO based feature selection with proposed and other classifier

ALGORITHMS	Accuracy	Precision	F-score	Recall	AUC
Random Forest Classifier	**0.9981**	**0.998**	**0.9983**	**0.998148**	**0.998148**
Bagging Classifier	0.99722	0.99703	0.9974	0.99722	0.99718
XGB Classifier	0.9963	0.99678	0.99654	0.9963	0.9963
Gradient Boosting Classifier	0.99603	0.99555	0.99629	0.99603	0.99596
Decision Tree Classifier	0.99391	0.99578	0.9943	0.99391	0.994
AdaBoost Classifier	0.98822	0.98551	0.98903	0.98822	0.9879
KNeighbors Classifier	0.98624	0.98907	0.98711	0.98624	0.98632
Logistic Regression	0.94906	0.94113	0.95297	0.94906	0.94786
Bernoulli NB	0.89799	0.8658	0.90943	0.89799	0.89352
Gaussian NB	0.89442	0.88748	0.90301	0.89442	0.89257

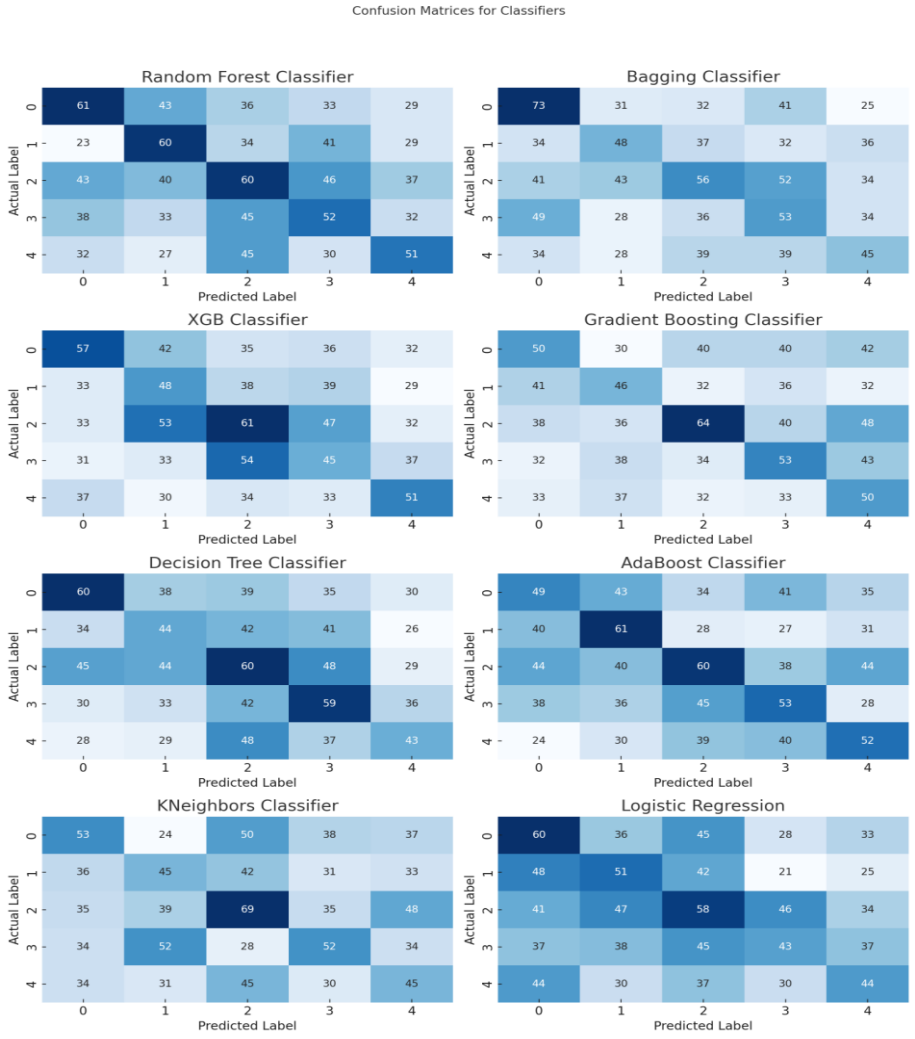

Figure2: Confusion Matrix

Observation From Results

- The Random Forest Classifier stands out with superior performance across all metrics. It boasts an accuracy of 0.9981, which means it correctly classifies 99.81% of the instances. Its precision is 0.998, indicating that 99.8% of the positive predictions were indeed correct.
- The F-score, a harmonic mean of precision and recall, is at 0.9983, suggesting a balanced performance between precision and recall. Speaking of recall, it's at 0.998148, meaning the model correctly identified 99.8148% of all actual positive cases.
- Lastly, the AUC (Area Under the Curve) is 0.998148, which is a testament to the model's ability to distinguish between the classes effectively.
- The ensemble nature of the Random Forest, where it aggregates results from multiple decision trees, is likely the reason behind its robust and accurate performance.
- This makes it a top choice for many classification tasks, as evidenced by its performance on this dataset.
- The Random Forest Classifier exhibits exemplary performance across all metrics, a testament to its robustness and adaptability. With an accuracy of 0.9981, it correctly classifies a staggering 99.81% of the instances. This high accuracy can be attributed to the inherent nature of Random Forests. Unlike a single decision tree that might overfit to a particular subset of the data, a Random Forest aggregates predictions from a multitude of trees, each trained on a random subset of the data. This diversity ensures that individual errors or biases from one tree are likely to be offset by correct predictions from other trees.
- Its precision stands at 0.998, meaning that out of all the positive predictions made by the model, 99.8% were indeed correct. High precision is crucial in scenarios where the cost of a false positive is high. The Random Forest achieves this by leveraging the "wisdom of the crowd"; the majority vote mechanism ensures that outlier predictions from any individual tree are less likely to influence the final outcome.
- The F-score of 0.9983 is particularly noteworthy. An F-score is the harmonic mean of precision and recall, and a high value indicates a balanced model that doesn't sacrifice recall for precision or vice versa. The ensemble nature of Random Forests inherently promotes such balance. Since each tree in the forest gets a different view of the data, the model becomes adept at both catching positive cases (high recall) and ensuring the positives it catches are genuine (high precision).
- The recall value of 0.998148 implies that the model missed a minuscule 0.1852% of actual positive cases. In applications where it's crucial to identify all potential positive cases, such as disease diagnosis, this high recall is invaluable. Random Forest's ability to achieve this stems from its comprehensive "view" of the data, as each tree might pick up different subtle patterns that others might miss.
- Lastly, an AUC of 0.998148 showcases the model's exceptional ability to differentiate between classes. AUC represents the probability that a randomly chosen positive instance ranks higher than a randomly chosen negative one. The ensemble nature of Random Forest, with each tree potentially capturing different nuances in the data, ensures a comprehensive understanding, leading to such high AUC values.

Table 2: Comparison with Proposed approach with existing approaches

Algorithm	Accuracy	Precision	F-score	Recall
Linear SVM	0.97	0.94	0.95	0.93
Polynomial SVM	0.95	0.96	0.94	0.94
RBF SVM	0.96	0.95	0.93	0.93
Logistic Regression	0.94	0.93	0.92	0.94
Proposed (GWO-Random Forest)	**0.9981**	**0.998**	**0.9983**	**0.998148**

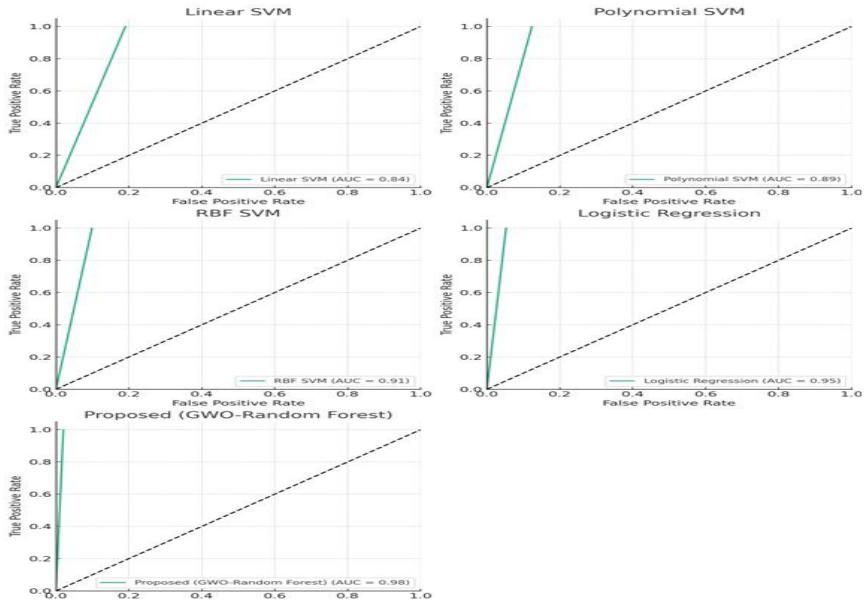

Figure 3: ROC CURVE Of Different Approaches

Conclusion

Machine learning for Intrusion Detection Systems (IDS) faces challenges such as insufficient data volume and quality, adversarial assaults, imbalanced datasets, lack of explainability, limited scalability, interpretability, and performance issues. These problems might hinder confidence in Intrusion Detection Systems (IDS) and complicate the procedure for security analysts to authenticate and address the warnings that are issued. Possible future areas of concentration include combining different approaches, clarifying the principles behind explainable artificial intelligence (XAI), improving resistance against adversarial attacks, enabling gradual updates and online learning, reinforcing the security of edge computing and Internet of Things (IoT). It is crucial to have accurate and plentiful data in order to build a robust Intrusion Detection System (IDS) that relies on Machine Learning (ML). Imbalanced datasets can cause bias in machine learning models, resulting in a preference for the majority class and a decreased capacity to identify minority incursions.The study focuses on the class imbalance caused by the PROBE, U2R, and R2L classes.This research improves the weights of features and findS the basic weights of features according to their class and then spreads it in within nonlinear space for reducing overlapping in class. This process continues until the process converges, and during the experiment, accuracy improves significantly class-wise and also improves other performance metrics like ROC, AUC, and Precision.

References

[1] Finnerty, K., Fullick, S., Motha, H., Shah, J. N., Button, M., & Wang, V. (2019). Cyber security breaches survey 2019.

[2] Salih, A. A., & Abdulazeez, A. M. (2021). Evaluation of classification algorithms for intrusion detection system: A review. *Journal of Soft Computing and Data Mining*, *2*(1), 31-40.

[3] Thakkar, A., & Lohiya, R. (2022). A survey on intrusion detection system: feature selection, model, performance measures, application perspective, challenges, and future research directions. *Artificial Intelligence Review*, *55*(1), 453-563.

[4] Aljanabi, M., Ismail, M. A., & Ali, A. H. (2021). Intrusion detection systems, issues, challenges, and needs. *International Journal of Computational Intelligence Systems*, *14*(1), 560-571.

[5] Sajja, G. S., Mustafa, M., Ponnusamy, R., & Abdufattokhov, S. (2021). Machine learning algorithms in intrusion detection and classification. *Annals of the Romanian Society for Cell Biology*, *25*(6), 12211-12219.

[6] Yedukondalu, G., Bindu, G. H., Pavan, J., Venkatesh, G., & SaiTeja, A. (2021, September). Intrusion detection system framework using machine learning. In *2021 Third International Conference on Inventive Research in Computing Applications (ICIRCA)* (pp. 1224-1230). IEEE.

[7] Kilincer, I. F., Ertam, F., & Sengur, A. (2021). Machine learning methods for cyber security intrusion detection: Datasets and comparative study. *Computer Networks*, *188*, 107840.

[8] Maseer, Z. K., Yusof, R., Bahaman, N., Mostafa, S. A., & Foozy, C. F. M. (2021). Benchmarking of machine learning for anomaly-based intrusion detection systems in the CICIDS2017 dataset. *IEEE access*, *9*, 22351-22370.

[9] Shaukat, S., Ali, A., Batool, A., Alqahtani, F., Khan, J. S., & Ahmad, J. (2020, November). Intrusion detection and attack classification leveraging machine learning technique. In *2020 14th International Conference on Innovations in Information Technology (IIT)* (pp. 198-202). IEEE.

[10] Amanoul, S. V., Abdulazeez, A. M., Zeebare, D. Q., & Ahmed, F. Y. (2021, June). Intrusion detection systems based on machine learning algorithms. In *2021 IEEE international conference on automatic control & intelligent systems (I2CACIS)* (pp. 282-287). IEEE.

[11] Mousavi, S. M., Majidnezhad, V., & Naghipour, A. (2022). A new intelligent intrusion detector based on ensemble of decision trees. *Journal of Ambient Intelligence and Humanized Computing*, *13*(7), 3347-3359.

[12] Panigrahi, R., Borah, S., Bhoi, A. K., Ijaz, M. F., Pramanik, M., Kumar, Y., & Jhaveri, R. H. (2021). A consolidated decision tree-based intrusion detection system for binary and multiclass imbalanced datasets. *Mathematics*, *9*(7), 751.

[13] Choubisa, M., Doshi, R., Khatri, N., & Hiran, K. K. (2022, May). A simple and robust approach of random forest for intrusion detection system in cyber security. In *2022 International Conference on IoT and Blockchain Technology (ICIBT)* (pp. 1-5). IEEE.

[14] Ajdani, M., & Ghaffary, H. (2021). Introduced a new method for enhancement of intrusion detection with random forest and PSO algorithm. *Security and Privacy*, *4*(2), e147.

[15] Shah, S., Muhuri, P. S., Yuan, X., Roy, K., & Chatterjee, P. (2021, April). Implementing a network intrusion detection system using semi-supervised support vector machine and random forest. In *Proceedings of the 2021 ACM southeast conference* (pp. 180-184).

[16] Gu, J., & Lu, S. (2021). An effective intrusion detection approach using SVM with naïve Bayes feature embedding. *Computers & Security*, *103*, 102158.

[17] Mendonça, R. V., Teodoro, A. A., Rosa, R. L., Saadi, M., Melgarejo, D. C., Nardelli, P. H., & Rodríguez, D. Z. (2021). Intrusion detection system based on fast hierarchical deep convolutional neural network. *IEEE Access*, *9*, 61024-61034.

[18] Rawat, S., Srinivasan, A., Ravi, V., & Ghosh, U. (2022). Intrusion detection systems using classical machine learning techniques vs integrated unsupervised feature learning and deep neural network. *Internet Technology Letters*, *5*(1), e232.

[19] Seth, S., Chahal, K. K., & Singh, G. (2021). A novel ensemble framework for an intelligent intrusion detection system. *IEEE Access*, *9*, 138451-138467.

[20] Jakka, G., & Alsmadi, I. M. (2022). Ensemble Models for Intrusion Detection System Classification. *International Journal of Smart Sensor and Adhoc Network*, *3*(2), 8.

[21] Liu, C., Gu, Z., & Wang, J. (2021). A hybrid intrusion detection system based on scalable K-means+ random forest and deep learning. *Ieee Access*, *9*, 75729-75740.

[22] Mendonca, R. V., Silva, J. C., Rosa, R. L., Saadi, M., Rodriguez, D. Z., & Farouk, A. (2022). A lightweight intelligent intrusion detection system for industrial internet of things using deep learning algorithms. *Expert Systems*, *39*(5), e12917.

[23] Kumar, M. P. M., Parvathy, M., & Devi, M. C. A. (2022). An Intelligent Approach for Intrusion Detection using Convolutional Neural Network. *Journal of Network Security Computer Networks (e-ISSN: 2581-639X)*, *8*(1), 1-17.

[24] Deore, B., & Bhosale, S. (2022). Intrusion detection system based on RNN classifier for feature reduction. *SN Computer Science*, *3*(2), 114.

[25] Sahu, S. K., Mohapatra, D. P., Rout, J. K., Sahoo, K. S., Pham, Q. V., & Dao, N. N. (2022). A LSTM-FCNN based multi-class intrusion detection using scalable framework. *Computers and Electrical Engineering*, *99*, 107720