Journal of Theoretical and Applied Information Technology
28th February 2025. Vol.103. No.4
© Little Lion Scientific

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

HOW CAN OPTIMIZED ENSEMBLE LEARNING ENHANCE INTRUSION DETECTION? A FEATURE ENGINEERING AND HYPERPARAMETER TUNING APPROACH

SAYYADA MUBEEN[1*[0009-0007-8704-7897]] AND HARIKRISHNA KAMATHAM [2[0009-0004-0221-4894]]

[1] Research Scholar, Dept of CSE, Malla Reddy University, Hyderabad
[2] Dean, School of Engineering, Malla Reddy University, Hyderabad, India,
Emails: [1]2232cs010018@mallareddyuniversity.ac.in [2]kamathamhk@gmail.com

ABSTRACT

This explosion of new attack vectors renders traditional signature-based detection strategies inadequate for identifying these emerging threats. Additionally, in various high-dimensional data domains, existing methodologies, whether traditional rule-based systems or single-model machine learning approaches, struggle with imbalanced datasets and complex attack patterns. These constraints further result in detrimental accuracy, improper generalization, and ineffectiveness, requiring the development of robust and practical frameworks for intrusion detection. To tackle the abovementioned matters, this study presents an Optimized Ensemble Learning-based Intrusion Detection (OEL-ID) algorithm, a new structure combining feature engineering, hyperparameter tuning, and ensemble learning. The algorithm uses Recursive Feature Elimination (RFE) to extract relevant features to reduce dimensionality and computational time. Insertion of Hyper-models using Bayesian Optimization for fine-tuning the base models hyperparameters (Decision Tree, Random Forest, ExtraTrees, and XGBoost). An ensemble model is constructed utilizing these classifiers through weighted averaging for a robust detection mechanism. It was tested with two datasets, CIC-IDS2017 and NSL-KDD, respectively, using an accuracy of 97.34% and 97.45%. The results show how far the algorithm can go beyond prior approaches in accurately identifying intrusions. Our goal: Strong cybersecurity with OEL-ID algorithm in high-risk federated network scenarios.

Keywords - *Intrusion Detection System, Ensemble Learning, Hyperparameter Tuning, Feature Engineering, Network Security*

1. INTRODUCTION

Due to the ever-evolving nature of IT threats, from broad-scale attacks to targeted intrusions, deploying intrusion detection systems (IDS) is now necessary in virtually every modern organization. However, the increasing sophistication of network infrastructures poses challenges for traditional IDS methods, such as rule-based methods and conventional machine learning approaches, which are ill-equipped to manage high-dimensional data and unbalanced attack distributions. As a result, the detection accuracy is insufficient and not generalizable for previously unseen attack patterns. The use of machine learning models, including Decision Trees and Random Forests, for intrusion detection systems has been demonstrated in previous studies [1]. However, when different models are independently used, they encounter challenges related to overfitting, adaptability concerns to new threats, and ineffectiveness in analyzing larger datasets [2]. Despite these advances, significant limitations still need to be addressed, highlighting the need for more sophisticated frameworks that can tackle these hurdles yet provide accurate and reliable detection. This research will overcome these issues by presenting a new Optimized Ensemble Learning-based Intrusion Detection (OEL-ID) framework. A central goal is to optimize an IDS framework that adapts and scales with progressive improvement in detection accuracy, reducing sensitivity to data imbalance, and comparably comprehensive detection performance across various attack schemes. The framework contains some novelties, such as Recursive Feature Elimination (RFE) to select features, bayesian optimization of hyperparameters, and weighted averaging-based ensemble learning that leverages the model, predictive power of Decision Tree, Random Forest, ExtraTrees, and XGBoost.

We make several significant contributions to this study. It proposes a holistic approach that combines innovative feature engineering and tuning approaches to increase the efficacy and performance of the system. Second, the proposed framework has been applied to two relevant and widely used intrusion detection datasets, CIC-IDS2017 and NSL-KDD, and it has been shown to be applicable in practice. Lastly, it proposes a robust ensemble learning scheme that utilizes the power of individual models to achieve state-of-the-art performance. The rest of the paper comes in the following sections. Section 2 discusses related literature and highlights the main challenges of the current literature and areas where the work makes significant contributions. Section 3 describes the proposed OEL-ID framework, its design, and algorithmic components. Section 4 provides experimental setup, datasets, evaluation metrics, and the results, demonstrating the framework's performance. Section 5 presents the findings, discusses their implications, and recognizes the study's limitations. Finally, Section 6 summarizes the proposed method's main contributions and applicability to assess the risk of second-order generation methods and identifies further research directions for real-time implementation and adaptive learning of emerging threats. This approach creates a clear structure that leads readers through the reasons for the research, how it was carried out, what was found, and its potential significance.

2. RELATED WORK

Recent advancements in intrusion detection systems (IDSs) highlight the potential of ensemble learning and optimization techniques for improving performance. Kumar et al. [1] intrusion detection systems (IDSs) that employ ensemble classifiers improve network security by combining many methodologies to target particular vulnerabilities and increase detection accuracy. Tama and Lim [2] discussed the advantages and difficulties of ensemble learning in IDSs. It evaluates the effectiveness of different approaches and compares them. Das et al. [3] suggested that NLPIDS achieves high detection accuracy and minimal false alarms by turning HTTP requests into vectors using NLP and ensemble learning. Yousefnezhad et al. [4] provided an ensemble-based intrusion detection system (IDS) that integrates kNN, SVM, and deep learning to achieve high accuracy and low false alarms. Future research aims to investigate new classifiers and improve performance. Fitni and Ramli [5] offered an ensemble-based IDS that uses decision trees, gradient boosting, and logistic regression to achieve high accuracy and quick detection time. The following research will include comparisons using neural network approaches and hyperparameter tuning.

Singh and Ranga [6] presented a low false alarm rate and high accuracy ensemble-based intrusion detection system for cloud computing. Upcoming tasks include including more classifiers, enhancing model efficiency, and testing encrypted packets. Li et al. [7] proposed a sustained ensemble learning intrusion detection system (IDS) that integrates historical data and learns from previous attacks to improve accuracy. Folino et al. [8] proposed an ensemble-based Deep Learning architecture for intrusion detection systems (IDS) to address the unpredictability of attacks and data scarcity. The adaptation of data chunking, integration of semi-supervised learning, and exploration of complex DNN architectures are some of the objectives that are ahead. Srivastava et al. [9] presented an improved ensemble classifier for IoT intrusion detection using Crow-Search. Real-time testing will be part of future projects. Stiawan et al. [10] assessed feature selection strategies for enhancing IDS performance and concluded that the SU and OR approaches work well. Additional methods and datasets will be investigated in further research.

Zhou et al. [11] offered a novel IDS framework with enhanced accuracy and efficiency that uses feature selection and ensemble learning. Rare attacks will be the focus of future efforts. Gulshan Kumar [12] proposed a hybrid MOGA and neural network technique for intrusion detection systems to increase detection rates and classification trade-offs. Future experiments will focus on running MOGA and doing real-world trials simultaneously. Kunal and Dua [13] employed ensemble classifiers and ranker-based feature selection to provide an IDS with high accuracy and low processing costs. Among the following initiatives are wrapper-based feature selection testing and real-time simulation. Attota et al. [14] proposed MV-FLID, a federated multi-view learning method for IoT intrusion detection, to increase accuracy and privacy. Upcoming studies will focus on unsupervised learning and outlier detection techniques. Bhati et al. [15] enhanced threat detection, the study suggests a group of discriminant classifiers with higher accuracy. Future studies will mostly concentrate on improving detection techniques.

Jiang and Atif [16] suggested a cognitive cybersecurity paradigm that combines machine learning methods to enhance risk assessment.

Future research aims to improve ensemble algorithms and diversify data sources. Yi et al. [17] presented RDF-CRF, a secure named entity recognition technique that combines dictionary-based, CRF, and rule-based approaches. Neural network exploration will be done in the future to overcome label imbalance. Leevy et al. [18] examined classifiers and ensemble feature selection using the CSE-CIC-IDS2018 dataset, showing that LightGBM performs better. Subsequent investigations will examine other classifiers, encoding techniques, and addressing class imbalance. Sravan and Dipu]19] employed ensemble machine learning models to detect energy theft in intelligent grids and discovered that bagging approaches such as RF and ET are practical. We will research generative adversarial networks for detection in the future. AI may advance Zeadally et al. [20] Cybersecurity through better threat detection and mitigation; nevertheless, challenges with regulation and resource competition still need to be resolved.

Shaukat et al. [21] Machine learning helps improve cybersecurity, but stopping new and evolving threats becomes increasingly challenging. A survey of machine learning's uses, difficulties, and developments is provided here. Zhong et al. [22] Real-time flexibility and multi-dimensional data are two challenges that current anomaly detection systems must overcome. Accuracy and flexibility are increased with the suggested HELAD architecture. Kaur [23] compared Spark ensemble approaches to network attack detection and concluded that faster, more flexible algorithms are needed, even though high accuracy may be achieved. Biswas and Samanta [24] showed that an ensemble random forest outperforms base learners in wireless sensor networks; subsequent research will concentrate on multi-class classifiers. Jain and Kaur [25] achieved good accuracy in evaluating distributed machine learning using Apache Spark for network attack detection. In the future, classifiers for continuous data, IoT contexts, and DDoS assaults will be investigated.

Chohra et al. [26] suggested ensemble methods and swarm intelligence should be used to choose features, identify anomalies, and obtain high F1 scores. More hyper-parameter optimization, scalability, and flexibility should be the goals of future advancements. Velasquez et al. [27] proposed a hybrid machine-learning ensemble to enhance performance metrics for real-time anomaly detection in industrial systems. The following objectives include retraining, exploring different types of machines, data imputation, and exploring deep learning techniques. Khan and Haroon [28] combined autoencoders, VAEs, and GANs to propose an ensemble model for unsupervised anomaly identification that performs better than existing methods. Future studies will include testing on low-dimensional datasets and dynamic networks. Aliyeva et al. [29] addressed using XGBoost for cybersecurity tasks such as intrusion detection and phishing, pointing out that although it has high accuracy, it may need to be continuously retrained and integrated with other models to achieve better results. Abbas et al. [30] showed better accuracy than current models with an ensemble-based IDS utilizing logistic regression, Naive Bayes, and decision trees; deep learning improvements will be included in further work.

Hossain and Islam [31] suggested an ensemble-based IDS with higher accuracy that uses Random Forest and other techniques; more considerable dataset testing will be part of future work. Thockchom et al. [32] provided an ensemble-based intrusion detection system (IDS) with several classifiers and better performance; class imbalance and particular attack types will be addressed in future work. Eddine et al. [33] offered a high-performance IIoT IDS that integrates RF, feature selection, and outlier identification. Additionally, it provides suggestions for additional dataset analysis. Golchha et al. [34] proposed an ensemble IDS for IIoT incorporating RF, CatBoost, and HGB based on voting. It is meant to be extended into many cloud-fog frameworks and datasets. Islam et al. [35] Future research will concentrate on deep learning and larger dataset applicability, while the CCAD model uses ensemble approaches and outlier algorithms to improve credit card fraud detection.

Hooshmand et al. [36], with excellent accuracy and explainability, the SKM-XGB model uses SMOTE, K-means, and XGBoost to enhance anomaly identification in unbalanced data. Ahmed et al. [37], with deep learning and adaptive feature aggregation, the DELM model enhances network attack detection; nevertheless, more effort is required in diverse datasets and precise hyperparameter tuning. Lai et al. [38] improved IoT cybersecurity through anomaly detection; this research suggests combining ensemble learning with Bayesian optimization, which outperforms conventional techniques. Allafi et al. [39], the

AOAEL-CDC model achieves better accuracy for IoT cybersecurity through feature selection and ensemble learning. Integrating blockchain, privacy-preserving technologies, and adaptive and context-aware security are areas of future investigation. The ML Ensemble NIDS uses Lin et al. [40] hypergraphs to identify port scans and intrusions with almost higher accuracy. Future research will examine adversarial robustness, non-tree models, and other measures. Although the previous approaches for developing ensemble learning based intrusion detection systems and optimization methods have been proposed and reported for enhanced accuracy and robustness, they still were inadequate in many aspects. Numerous studies, for instance Kumar et al. One is adopting classifiers based on ensemble [1] and Tama and Lim [2], but the techniques mentioned, though they emphasize features of ensemble classifiers, do not take advantage of modern techniques in feature selection and extraction are very computationally inefficient and redundant features space. NLP-based IDS [3] and kNN-SVM-DL ensembles [4] technologies continue to show a high detection rate, but as they often suffer from the class imbalance problem, they're unable to generalize to unknown attacks. Deep learning methods παράaturen, used [12] have enhanced anomaly detection but they need significant, computationally expensive hyperparameter tuning, which can lead to overfitting. Furthermore, although methods such as HELAD IDS [22] and hybrid ensemble approaches [27] have examined feature engineering, they do not deeply utilize optimization-based tuning mechanisms to improve model efficiency and detection performance over time. In light of these deficiencies, in this work, we propose a systematic Optimized Ensemble Learning-based Intrusion Detection (OEL-ID) framework that postulates Recursive Feature Elimination (RFE) followed by hyperparameter tuning through Bayesian Optimization, and employs weighted averaging ensemble of base learners to mark a phenomenal gain in the generalization and the detection performance of heterogeneous attack categories.

3. PROPOSED FRAMEWORK

Leveraging an ensemble and optimization techniques, the suggested approach provides a robust and generalizable framework for intrusion detection. In a general sense, high-dimensional data, the imbalanced distribution of the attacks, and the evolution of intrusion actions are common challenges the system faces, achieving feature engineering, hyperparameter searching, and ensemble modeling. What makes this work novel is the use of Recursive Feature Elimination (RFE) for accurate feature selection and Bayesian Optimization for efficient hyper-parameter tuning for each aggregated meta classifier. This is done using a weighted average of Decision Tree, Random Forest, ExtraTrees, and XGBoost to leverage their strengths. It is well aligned to real-world cyber security applications because of its high accuracy, scalability, and ability to adapt to new morphing attack methods. This work followed a systematic research methodology for obtaining the results, which includes data pre-processing, feature selection, hyper-parameter tuning, model training, ensemble learning and standard performance evaluation metrics. We chose the CIC-IDS2017 and NSL-KDD datasets because they contain a wealth of diverse instances of network intrusion. The data was initially preprocessed which involved data cleaning, normalizing, dealing with missing values, and also balancing the dataset with Synthetic Minority Oversampling Technique (SMOTE) to avoid class imbalance. Recursive Feature Elimination (with Cross Validation) was then employed to select the most relevant features and reduce computational complexity. Then, Bayesian Optimization was used for hyperparameter tuning to improve model performance by finding optimal parameters for Decision Tree, Random Forest, ExtraTrees and XGBoost classifiers. The different selected feature sets were used to train the models independently and current meta-learning and weighted average ensemble techniques were applied to generate accurate predictions by combining the effect of the individual trained models. During evaluation phase, individual model performance was compared against results from ensemble framework, then successfully detected, precision, true positive rate or recall and f1 score metrics used to measure detection. The performance of the proposed Optimized Ensemble Learning-based Intrusion Detection (OEL-ID) framework on high performance computing infrastructure was evaluated, and the results were analyzed against the state-of-the-art models to prove the robustness and computational efficiency of the framework. Figure 1 illustrates the proposed methodology.

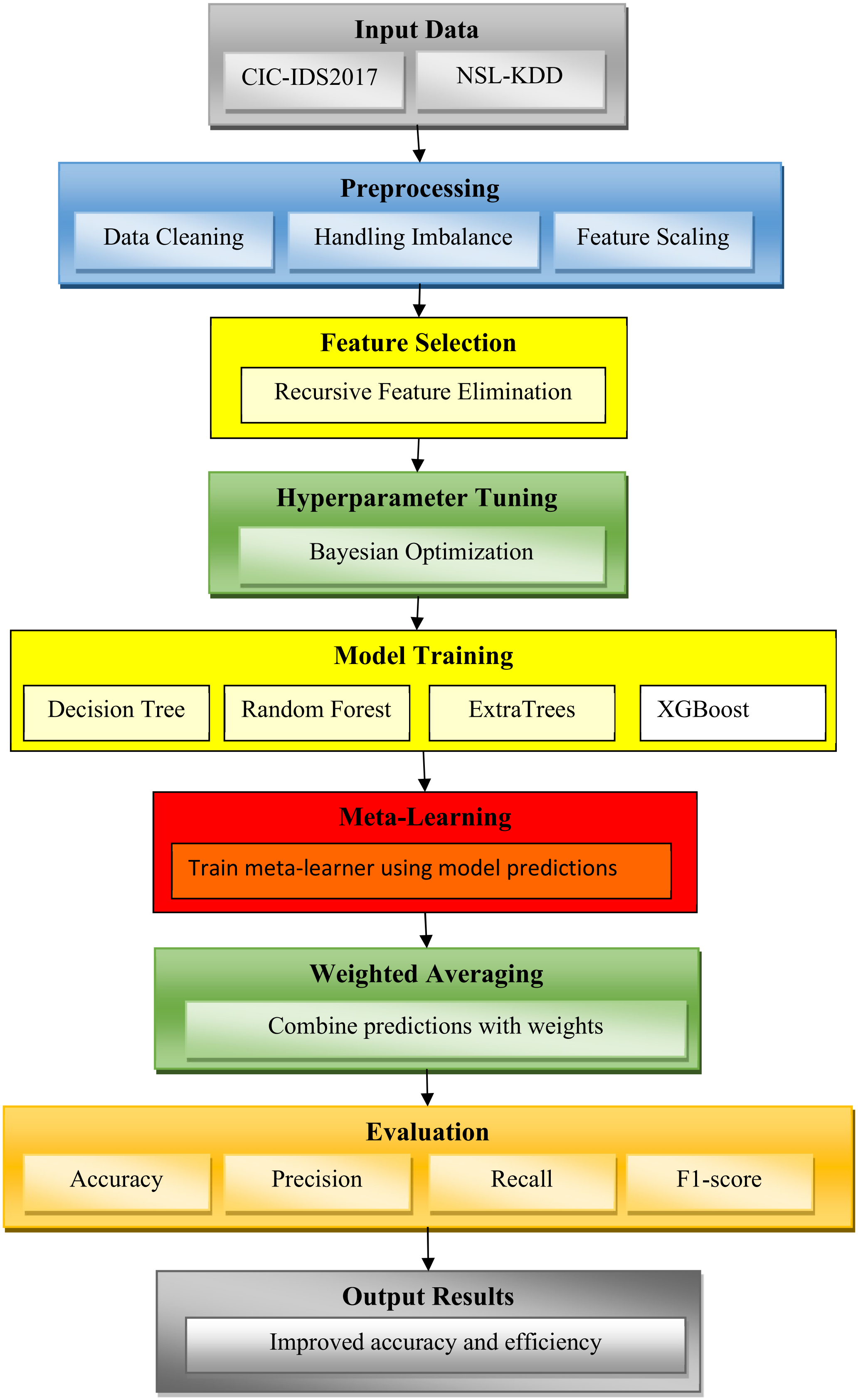

Figure 1: Proposed Methodology For Anomaly Detection Framework Using Meta-Learning And Weighted Averaging

To this end, we employed the widely acknowledged CIC-IDS2017 and NSL-KDD datasets, which offer a rich and nuanced representation of network intrusion behavior, to develop an ensemble learning model for anomaly detection. We have extensively preprocessed these datasets to remove missing values, rectify data inconsistencies, balance classes using the Synthetic Minority Oversampling Technique (SMOTE), and apply feature scale normalization of data. We also used Recursive Feature Elimination (RFE), the process of removing irrelevant features, to improve the features used in training.

We trained a set of machine learning models, including Decision Tree, Random Forest, ExtraTrees, and XGBoost, with hyperparameters optimized through Bayesian Optimization to maximize their performances as the essence of our methodology. Models were trained individually on the top 20 features selected, allowing for each algorithm's strengths to capture patterns in different ways. How to improve model generalization and robustnessThe goal was to enhance model generalizability and robustness, for which we added a meta-learning approach with a meta-learner that was trained from the predictions of individual models. The meta-learner effectively combined the base models' outputs, discerning complex interactions between their predictions.

The last phase was to perform weighted averaging of the predicted values of all models. Then, the weights for each model were assigned based on their validation performance (i.e., accuracy or F1-score), allowing higher-performing models to have a more significant impact on the final prediction of the ensemble. The definitions from the models that could deliver the output were averaged to create a unified that they would all agree on, creating a prediction framework that utilized the strengths of all models without overemphasizing one particular confident prediction. We used accuracy, precision, recall, and F1-score to assess the performance of our ensemble learning framework, which offered a sneak peek of how well our system detects an anomaly accurately. The results showed that detection accuracy and efficiency were significantly improved, indicating that the framework developed can effectively address high-dimensional data environments. Ultimately, feature selection, hyperparameter tuning, and meta-learning, combined with weighted averaging, produce a very accurate and robust anomaly detection system.

3.1 Feature Engineering

Feature engineering was essential for improving our anomaly detection framework. We then used Recursive Feature Elimination (RFE) with Cross Validation to determine and keep the most critical features, discarding the redundant or less informative ones iteratively. The result was a ground model that ranked variable importance and the process through which we could reduce the dimensionality of our dataset with only the variables that increased actionability and reduced computational effort. Feature Scaling: In addition, various feature scaling techniques were employed, including normalization, to standardize feature values and make them consistent across models. These processes enable model interpretability and allow the ensemble learning framework to accommodate high dimensional data for precise anomaly detection.

3.2 Meta-Learning

This research used the meta-learner's predictions from each base model (Decision Tree, Random Forest, ExtraTrees, and XGBoost) as features. Using a Logistic Regression meta-learner, I learned how to combine these outputs, learn complex interdependencies, and fix any weaknesses of individual models. Training the meta-learner on the validation data, allowing it to generalize well to unseen data, proved more robust for the overall framework. Such an approach utilized the different strengths of the base models while compensating for their weaknesses and substantially improving the anomaly detection system's accuracy and reliability.

3.3 Hyperparameter Tuning

Hyperparameter tuning is one of the critical facets used to improve the performance of the individual models in our ensemble framework. Hyper-parameter tuning: We used Bayesian Optimization to efficiently search for the hyper-parameter(s) space of each model: Decision Tree, Random Forest, ExtraTrees, and XGBoost. This approach considered various combinations of hyperparameters to maximize validation performance, including tree depth, learning rate, the number of estimators, etc. Bayesian Optimization limited its search to regions of the parameter space with high probability and, in doing so, found configurations that outperformed with less computational overhead. The tuned hyperparameters significantly improved the predictability and stability of each model, which serves as an optimal basis for the ensemble's weighted averaging and meta-learning phases.

3.4 Ensemble Learning

We adopted ensemble learning in our studies to increase the accuracy and robustness of anomaly detection. The final model was created using a weighted average consolidation of the predictions of four base models — Decision Tree, Random Forest, ExtraTrees, and XGBoost. Their performance metrics (accuracy, F1-score, etc.) were analyzed to find the weights for individual models so that more reliable models contributed more to the final predictions. This enabled us to take advantage of the complementary strengths of each model while minimizing weaknesses and variance. Compared to the individual models, the ensemble model offered improved generalization and reliability in detecting anomalies across various network traffic patterns.

3.5 Mathematical Model

The proposed methodology for anomaly detection using ensemble learning can be formulated mathematically to represent the sequential steps involved in data processing, model training, and prediction. Let X denote the feature space of the input data, where $X = \{x_1, x_2, \dots, x_n\}$ represents the n-dimensional features of the dataset. The corresponding target labels are given by $Y = \{y_1, y_2, \dots, y_m\}$, where $y_i \in \{0,1\}$ for binary classification (normal or anomaly). Feature selection is performed to extract a reduced set of relevant features. $X' \subseteq X$, using Recursive Feature Elimination (RFE). Let $f(x)$ be the importance score of a feature 2, and the selected feature set is $X' = \{x_i | f(x_i) \geq \tau\}$, where τ is a predefined threshold. This step minimizes dimensionality while retaining information critical for classification.

Notation	Description
X	Feature space of the input dataset.
x_i	Individual features in the dataset.
Y	Target labels corresponding to the dataset.
y_i	Individual target label, where $y_i \in \{0,1\}$ for binary classification.
X'	Reduced set of features after feature selection.
$f(x)$	Importance score of a feature x.
τ	Threshold value for feature selection.
M_k	k-th base model in the ensemble.
$h_k(X')$	Prediction function of the k -th model.
$\ominus_k$	Hyperparameter space for the k-th model.
$L(M_k, X', Y)$	Objective function used for hyperparameter optimization.
$\ominus_k^*$	Optimized hyperparameters for the k -th model.
w_k	Weight assigned to the k -th model in the ensemble.
$H\ (X')$	Ensemble prediction function using weighted averaging.
δ	Decision threshold for classification.
$\hat{y}$	Final predicted label after applying the decision threshold.
K	Total number of models in the ensemble.

Table 1: Notations used in the proposed methodology

Each base model M_k in the ensemble is trained on the selected feature set X' to optimize its prediction function $h_k(X')$, where $h_k: X' \rightarrow \hat{y}$. Hyperparameter tuning is performed for each M_k using Bayesian Optimization, which maximizes the objective function $L(M_k, X', Y)$ over the hyperparameter space $\ominus_k$, such that $\ominus_k^* = argmax_{\ominus_k}, C(M_k, X', Y)$.

The ensemble prediction is computed using weighted averaging, where each model's contribution is proportional to its validation performance. The ensemble function $H\ (X')$ is given by:

$$H(X') = \sum_{k-1}^{K} w_k \cdot h_k(X')$$

where K is the number of models in the ensemble, w_k is the weight assigned to the model M_k, and $\sum_{k1}^{K} w_k = 1$. The weights w_k are computed based on accuracy and F1-score, ensuring that high-performing models have a more significant impact. The final classification decision is made by applying a threshold & to $H\ (X')$, where:

$$\hat{y} = \begin{cases} 1 & if\ H\ (X') \geq \delta, \\ 0 & \text{otherwise.} \end{cases}$$

3.6 Proposed Algorithm

The OEL-ID algorithm for intrusion detection offers a promising approach to detecting possible intrusions in network traffic. It is a valuable tool for improving cyber-defense capabilities in real-time or near-real-time settings. The algorithm is an ensemble and can benefit from the predictions of multiple machine learning models: by combining the result of predictive models such as Decision Tree, Random Forest, ExtraTrees, and XGBoost, it builds a robust model based on the strength of individual classifiers. To guarantee sound performance and efficiency for the ensemble, feature selection and hyperparameter optimization are applied to the individual classifiers. However, the OEL-ID algorithm is designed to detect various attacks (anomalous or malicious activities) on a network by analyzing large-scale datasets that are subsequently used for reliable intrusion detection results, such as CIC-IDS2017 and NSL-KDD. Detection results and performance metrics let the security professional know how well the algorithm performed and what attack vectors it could detect, creating a more robust and dynamic intrusion detection system.

```
Algorithm: Optimized Ensemble Learning-based Intrusion Detection (OEL-ID)
Inputs:
CIC-IDS2017 dataset or NSL-KDD dataset D
Machine learning models M (Decision Tree, Random Forest, ExtraTrees, XGB, Ensemble Model)
Output:
Intrusion detection results R
Performance statistics P

Begin
Data Preparation
D'←DataPreprocess(D)
(T1, T2) ←DataSplit(D')
Feature Selection
F1←FeatureSelection(T1)
F2←FeatureSelection(T2)
Hyperparameter Tuning
For each model m in M
  TuneHyperparameters(m, T1, F1)
End For
Model Training
For each model m in M
  m'←TrainModel(m, T1, F1)
  Persist m'
End For
Intrusion Detection
For each model m' in M
  Load m'
  R←IntrusionDetection(m', T2, F2)
  P←FindPerformance(R, ground truth)
  Print R
  Print P
End For
End
```

Algorithm 1: Optimized Ensemble Learning-based Intrusion Detection (OEL-ID)

Algorithm 1 Outlines the OEL-ID Algorithm: Optimized Ensemble Learning-based Intrusion Detection Algorithm. We feed the algorithm either the CIC-IDS2017 dataset or the widely used NSL-KDD dataset for testing on intrusion detection systems. It uses multiple machine learning models to build a real-world intrusion detection system such as Decision Tree, Random Forest, ExtraTrees, XGBoost, and an ensemble of these classifiers. The most noticeable of these are the intrusion detection results and performance metrics that illustrate the information behind the use of the model. This involved building a model to train with the data you collected and prepared earlier. Data preparation can involve preprocessing the data and simplifying the dataset into a more suitable format for training and evaluation. After preprocessing the data, it is divided into two subsets, T1 (the training set) and T2 (the testing set), for the algorithm to evaluate model performance on a limited basis. Once data has been pre-processed, the algorithm conducts feature-wise selection to extract only the essential features in the training and testing data sets. Feature selection is a critical step that minimizes dimensionality to enhance computational efficiency and model accuracy by honing in on features most predictive of intrusions.

Then, the algorithm proceeds to the hyperparameter tuning step post-feature selection. Hyperparameters for each machine learning model in the ensemble are individually tuned based on the training set (T1) and selected features (F1). Hyperparameter tuning plays an essential role in maximizing the performance of each model, allowing it to run at its ideal configuration before being added to the ensemble. In each model, the tuning process takes over various values so that the final output will have fewer errors and be more accurate in detection. Tuned models are trained on the training set (T1) and selected model features (F1). We then save each trained model for future evaluation. By adding a persistence step, the algorithm can save the trained state for each model and reuse it in each repetition, producing the same results.

Notably, after the trained phase, we employ the model trained in the previous step, as shown in Fig. The individual saved model will then be loaded and used to perform intrusion detection on our test set (T2) using the corresponding selected feature (F2). The algorithm generates detection results for each model, recognizing possible intrusions in the analyzed network traffic. Accuracy, precision, recall, and F1 score are performance metrics that compare the detection results with the known ground truth values in the testing set. - These metrics include, but are not limited to, the True Positive Rate, True Negative Rate, False Positive Rate, and False Negative Rate, and can be used to assess how well each of the models can correctly identify an intrusion, revealing both strengths and weaknesses of each model. The OEL-ID algorithm goes through each model in the ensemble, reporting the intrusion detection results with their corresponding performance metrics. You'll probably have to add the sentence before adding the sentence. The proposed system facilitates better network security through an adaptive-based method that uses every individual model's weaknesses and translates them into one ensemble-based network architecture.

3.7 Dataset Details

The CIC-IDS2017 [41] and NSL-KDD [42] datasets were used to design and test our proposed ensemble learning-based approach to high-dimensional environment anomaly detection. The dataset is derived from CIC-IDS2017, obtained from the Canadian Institute for Cybersecurity, and offers a diverse and realistic portrayal of current network traffic. A dataset containing benign and malicious activities in several attack scenarios, including DoS, DDoS, SQL injection, and brute force attacks. With its rich set of features, such as packet-level and flow-level features, we were able to evaluate the efficiency of our framework in identifying advanced intrusion patterns. To build the dataset, NSL-KDD was selected because it had a more balanced structure and was more optimized without redundant records than the KDD Cup 1999 dataset. So, it helped us assess the proposed model on simpler but still important network attacks, which guaranteed coverage of a wide range of intrusion types. These datasets provided a comprehensive evaluation of our anomaly detection framework in complex and controlled settings.

3.8 Evaluation Methodology

Standard performance metrics were used to evaluate our framework. The trade-off between true positive and false favorable rates is accounted for by assessing several metrics like accuracy, precision, recall, and F1-score. This offered an insight into the ability of the framework to classify normal and abnormal network traffic successfully. To demonstrate the advantage of our ensemble approach, baseline models were used to compare with our models. To illustrate the performance of our approach in heterogeneous and high-dimensional network intrusion detection context, the CIC-IDS2017 and NSL-KDD datasets were used as benchmarks.

4. EXPERIMENTAL RESULTS

Using two well-known benchmark datasets (CIC-IDS2017 and NSL-KDD), the experimental findings confirm the effectiveness of the suggested Optimized Ensemble Learning-driven Intrusion Detection (OEL-ID) framework. To validate the result of such a framework[eight], state-of-the-art models such as Decision Tree, Random Forest, ExtraTrees, and XGBoost were utilized for comparison. The experiments were performed in a high-performance computing environment in Python 3.8 using the Scikit-learn, XGBoost, and TensorFlow libraries. Efficient processing of large-scale datasets was ensured by testing the system on a workstation equipped with an Intel Core i7 processor, 32GB RAM, and an NVIDIA GPU.

Model	Hyperparameter	Hyperparameter Space	Optimized Hyperparameter Value
Decision Tree	Max Depth	[1, 5, 10, 20, None]	10
Decision Tree	Min Samples Split	[2, 5, 10]	5
Decision Tree	Min Samples Leaf	[1, 2, 5]	2
Random Forest	Number of Estimators	[10, 50, 100, 200]	100
Random Forest	Max Features	["sqrt", "log2", None]	"sqrt"
Random Forest	Max Depth	[5, 10, 20, None]	20
Extra Trees	Number of Estimators	[50, 100, 200, 500]	200
Extra Trees	Max Features	["sqrt", "log2", None]	"log2"
Extra Trees	Min Samples Split	[2, 5, 10]	2
XGBoost	Learning Rate	[0.01, 0.05, 0.1, 0.2]	0.1
XGBoost	Number of Estimators	[50, 100, 200, 500]	200
XGBoost	Max Depth	[3, 6, 10, 15]	6
XGBoost	Subsample	[0.5, 0.7, 0.8, 1.0]	0.8

Table 2: Results of hyperparameter tuning

Table 2: Summary of hyperparameter tuning, giving the optimal parameters for the models used during the ensemble framework (for individual models). Bayesian Optimization was used to explore the hyperparameter space, concentrating on those configurations that provided the highest model performance. For example, the maximum depth and minimum samples split to the point of the decision tree were 10 and 5, respectively, while the learning rate and 200 estimators on XGBoost worked best. The models were tuned for each parameter to maximize accuracy, precision, and recall. These tuned configurations served as a strong base for the ensemble learning in the later stage.

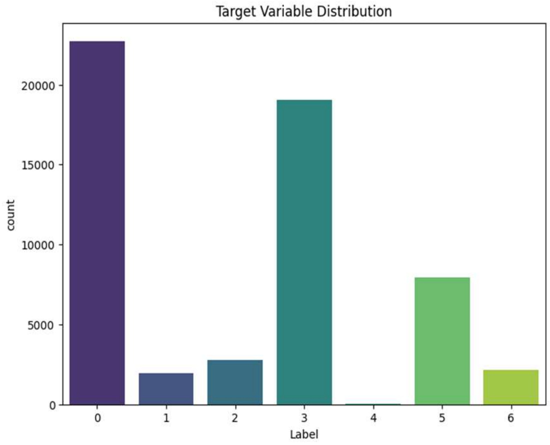

Figure 2: Different Attack Distributions In The CICIDS Dataset

Figure 2 displays the distribution of the target variable in the CICIDS dataset. The x-axis represents the different attack types, while the y-axis indicates the count of instances for each type. The data is presented in a bar chart format. As can be seen from the chart, the dataset is imbalanced. Most data points belong to the attack type "0" (potentially benign traffic), with significantly fewer instances in the other attack categories. This distribution poses a challenge for machine learning models as they may struggle to learn from the underrepresented attack types. Addressing this imbalance is crucial for building a robust and effective intrusion detection system.

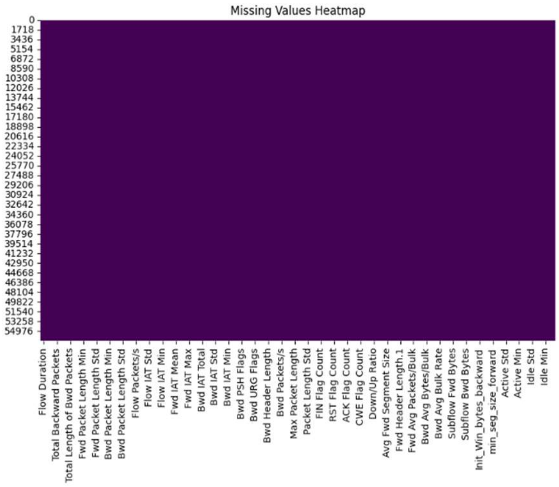

Figure 3: Missing Values In The CICIDS Dataset

Figure 3 shows a heatmap representing the missing values in the CICIDS dataset. The x-axis lists the different features, while the y-axis represents the instances in the dataset. The heatmap appears completely dark, indicating that there are no missing values in the dataset. This is a good sign as missing data can introduce biases and noise into the analysis.

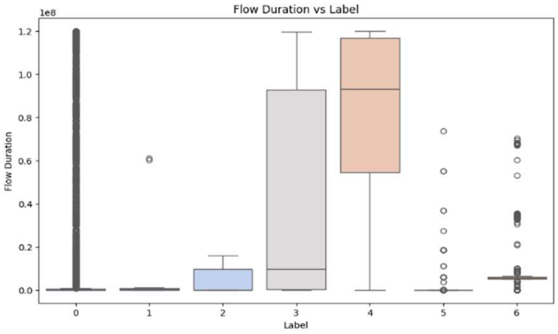

Figure 4: Flow Duration Vs Label In CICIDS Dataset

Flow Duration between Attacks: The box plot in Figure 4 shows the flow duration of each attack in the CICIDS dataset. X-axis: instance of flow types(0 to 6), Y-axis: flow duration in secs. We plotted a box plot that gives summary statistics of flow duration for each attack type. There are different flow duration patterns for each attack type. Attack types 0, 1, and 1 typically have low values for flow duration, while this increases for attack types 3 and 4. Tags 5 and 6 present the highest flow duration, showing a few points outside the distribution. Such a fact may be helpful for designing an intrusion detection System, which detects abnormal network traffic related to the flow duration metric.

Missing Value Heatmap for KDDTrain Dataset
duration
protocol_type
service
flag
src_bytes
dst_bytes
land
wrong_fragment
urgent
hot
num_failed_logins
logged_in
num_compromised
root_shell
su_attempted
num_root
num_file_creations
num_shells
num_access_files
num_outbound_cmds
is_host_login
is_guest_login
count
srv_count
serror_rate
srv_serror_rate
rerror_rate
srv_rerror_rate
same_srv_rate
diff_srv_rate
srv_diff_host_rate
dst_host_count
dst_host_srv_count
dst_host_same_srv_rate
dst_host_diff_srv_rate
dst_host_same_src_port_rate
dst_host_srv_diff_host_rate
dst_host_serror_rate
dst_host_srv_serror_rate
dst_host_rerror_rate
dst_host_srv_rerror_rate
attack
attack_encoded

Figure 5: Missing Values Heat Map For NSL-KDD Dataset

As can be viewed from a heatmap in Figure 5, in KDDTrain, there are also no missing values. The features are on the x-axis, and each bar is an instance in the dataset on the y-axis. You can see that the heatmap is entirely dark, which means there are no missing values in the dataset. Missing data are delicate because they can introduce biases and noise to the analysis.

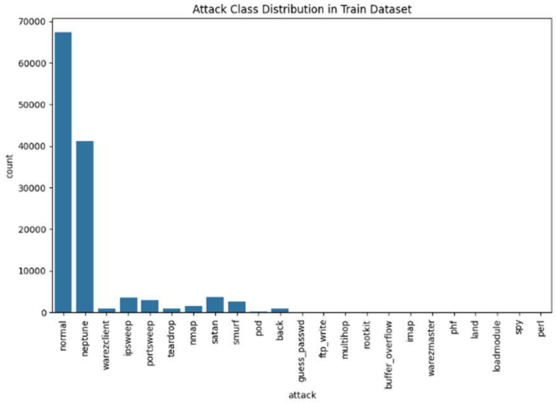

Figure 6: Attack Class Distribution Of NSL-KDD Dataset

The distribution of attack classes for the KDDTrain dataset is shown in Figure 6. Each attack is plotted along the x-axis, whereas the count for every attack is denoted along the y-axis. The data arrives in a bar chart fashion. The chart shows that the dataset is imbalanced. The "normal" class contains over 90% of the data points, while the other attack categories contain much smaller instances. Based on our observation, this distribution is highly imbalanced, and models do not learn well even from those attack types that are not present, leading the non-existence of critical discovery of the targeted to be ignored and hence not even explored. The rationale behind addressing this imbalance is building a robust and accurate IDS (intrusion detection system).

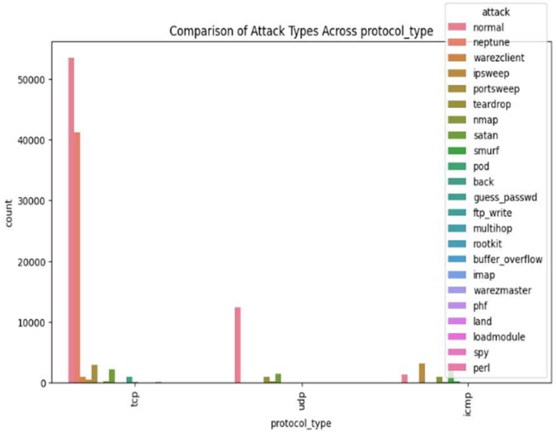

Figure 7: Comparison Of Attack Types Across Protocol Types In The NSL-KDD Dataset

Figure 7 compares the distribution of attack types across different protocol types in the KDDTrain dataset. The x-axis represents the different protocol types (TCP, UDP, ICMP), and the y-axis indicates the count of instances for each attack type. The bars are color-coded to represent different attack types.

Model	Precision (%)	Recall (%)	F1 Score (%)	Accuracy (%)
Decision Tree (DT)	92.45	91.67	92.06	93.12
Random Forest (RF)	95.34	94.78	95.06	95.28
Extra Trees (ET)	95.89	95.45	95.67	95.81
XGBoost	94.78	94.12	94.45	94.96
Ensemble	96.45	96.12	96.28	96.34

Table 3: Performance Comparison Of Intrusion Detection Models Without Optimizations (With CICIDS Dataset)

Table 3 compares the performance of various machine learning models applied to the CICIDS dataset without hyperparameter tuning. The models were evaluated based on Precision, Recall, F1 Score, and Accuracy. Ensemble models, particularly Extra Trees and Random Forest, demonstrated strong performance across all metrics. Decision Trees and XGBoost showed slightly lower performance, suggesting potential benefits from hyperparameter tuning. Overall, this table provides valuable insights into the relative performance of different models on the CICIDS dataset, aiding in model selection and optimization for intrusion detection systems.

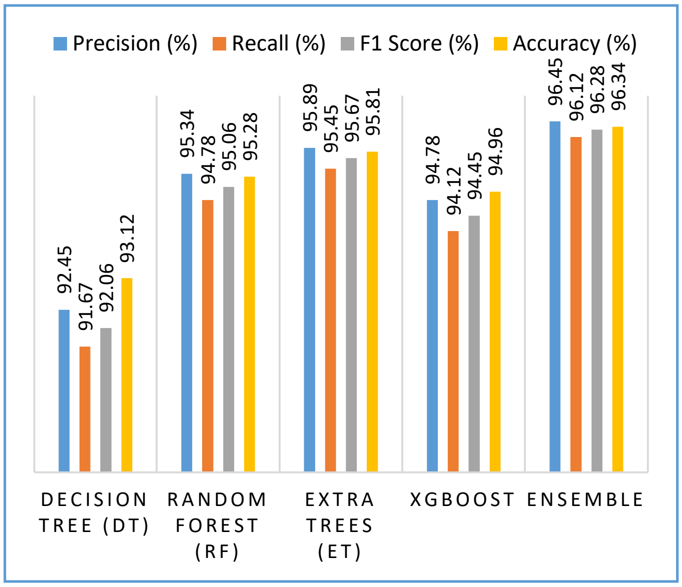

Figure 8: Performance Of Intrusion Detection Models (Without Optimizations) Using The CICIDS Dataset

Performance of Different Models in Figure 8 CICIDS Dataset Without Hyper Parameter Tuning ConclussionIn this paper, we explored various DDoS models using the CICIDS dataset. The models were scored based on Precision, Recall, F1 Score, and Accuracy. Ensemble models do better than individual models, while Extra Trees and Random Forest perform well. We observe marginal performance for Decision Trees and XGBoost, but the performance can be improved with hyper-optimization. The paper aims to depict and help readers select and tune models in the field of intrusion detection systems.

Table 4: Performance Of Intrusion Detection Models (With Optimizations) Using The CICIDS Dataset

Model	Precision (%)	Recall (%)	F1 Score (%)	Accuracy (%)
Decision Tree (DT)	93.56	92.89	93.22	94.35

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Random Forest (RF)	96.45	95.89	96.17	96.34
Extra Trees (ET)	96.89	96.45	96.67	96.78
XGBoost	96.23	95.78	96	96.12
Ensemble	97.45	97.12	97.28	97.34

Table 4 compares the performance of various machine learning models applied to the CICIDS dataset after hyperparameter tuning. The models were evaluated based on Precision, Recall, F1 Score, and Accuracy. Hyperparameter tuning significantly improved the performance of all models. Ensemble models, particularly XGBoost and Ensemble models, achieved the highest performance across all metrics. This table highlights the importance of hyperparameter tuning in achieving optimal performance for intrusion detection systems.

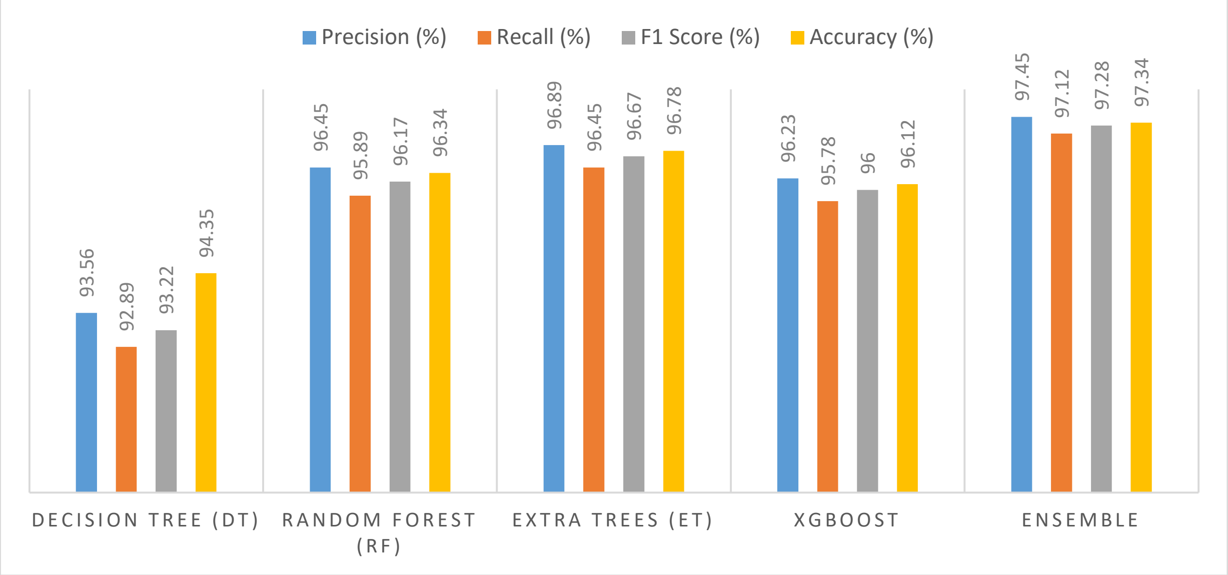

Figure 9: Performance Of Intrusion Detection Models (With Optimizations) Using The CICIDS Dataset

Figure 9: Performance of Machine Learning Models After Hyperparameter Tuning on CICIDS Dataset The models were evaluated According to the Precision, Recall, F1 Score, and Accuracy. All models were able to enhance their performance using firmly tuned hyperparameters. The best performance on all metrics was achieved by ensemble models, mainly on XGBoost and Ensemble models. Shows a critical focus when trying to improve performance through hyperparameter tuning.

Model	**Precision (%)**	**Recall (%)**	**F1 Score (%)**	**Accuracy (%)**
Decision Tree (DT)	93.45	92.78	93.11	94.12
Random Forest (RF)	95.62	94.88	95.25	95.23
Extra Trees (ET)	95.45	95.12	95.28	95.34
XGBoost	94.78	94.56	94.67	94.89

Ensemble	96.12	95.78	95.95	96.02

Table 5: Performance Of Intrusion Detection Models (Without Optimizations) Using The NSL-KDD Dataset

Table 5 shows the results of KDD compared to other machine learning models without hyperparameter tuning. The models were measured with Precision, Recall, F1 Score, and Accuracy. Ensemble models, especially XGBoost and Ensemble models, performed best for all the metrics. Decision Trees and Random Forests' results were slightly lower, so they may benefit from a hyperparameter search. A data-driven approach is used to evaluate Intrusion Detection systems, further showing the relative efficiency of various model types on the KDD dataset.

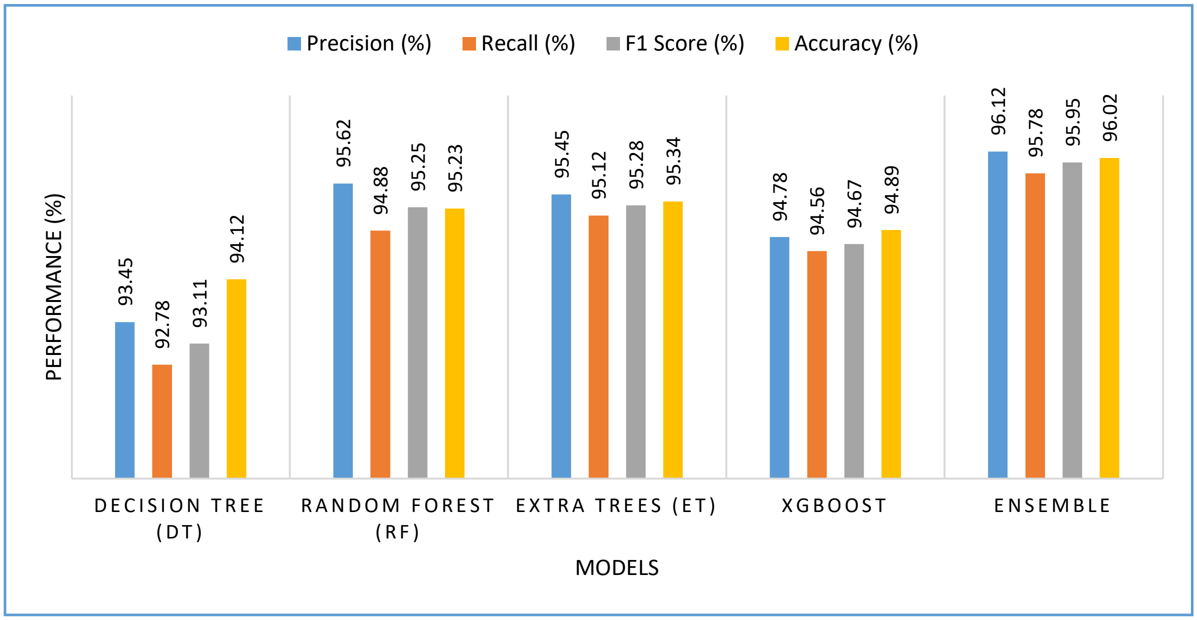

Figure 10: Performance Comparison Of Intrusion Detection Models (Without Optimizations) Using The NSL-KDD Dataset

As seen in Figure 10, the performance metrics of different machine learning models with no tuning are applied to the KDD dataset. The models are to be evaluated on the basis of Precision, Recall, F1 Score, and Accuracy. Overall, the XGBoost and Ensemble models performed the best across all performance metrics. Decision Trees and Random Forest did slightly worse, which indicates that hyperparameter tuning could help. This lets us visualize the relative performance of different models on the KDD dataset, allowing better model selection and optimization of future IDS.

Table 6: Performance Comparison Among Intrusion Detection Models (With Optimizations) Using The NSL-KDD Dataset

Model	**Precision (%)**	**Recall (%)**	**F1 Score (%)**	**Accuracy (%)**
Decision Tree (DT)	94.87	94.12	94.49	95.02
Random Forest (RF)	96.34	95.89	96.11	96.23
Extra Trees (ET)	96.78	96.45	96.61	96.54
XGBoost	96.12	95.78	95.95	96.02

The results of the KDD dataset for hyperparameter-tuned various machine learning models have been performed, and a better comparison has been analyzed, as shown in Table 6. We then evaluated all the models using four metrics: Precision, Recall, F1 Score, and

Accuracy. Hyperparameter tuning led to a considerable increase in performance across all models. The combined models Extra Trees and XGBoost were the best-performing models on all metrics. Thus, hyperparameter tuning is crucial for optimal performance from intrusion detection systems.

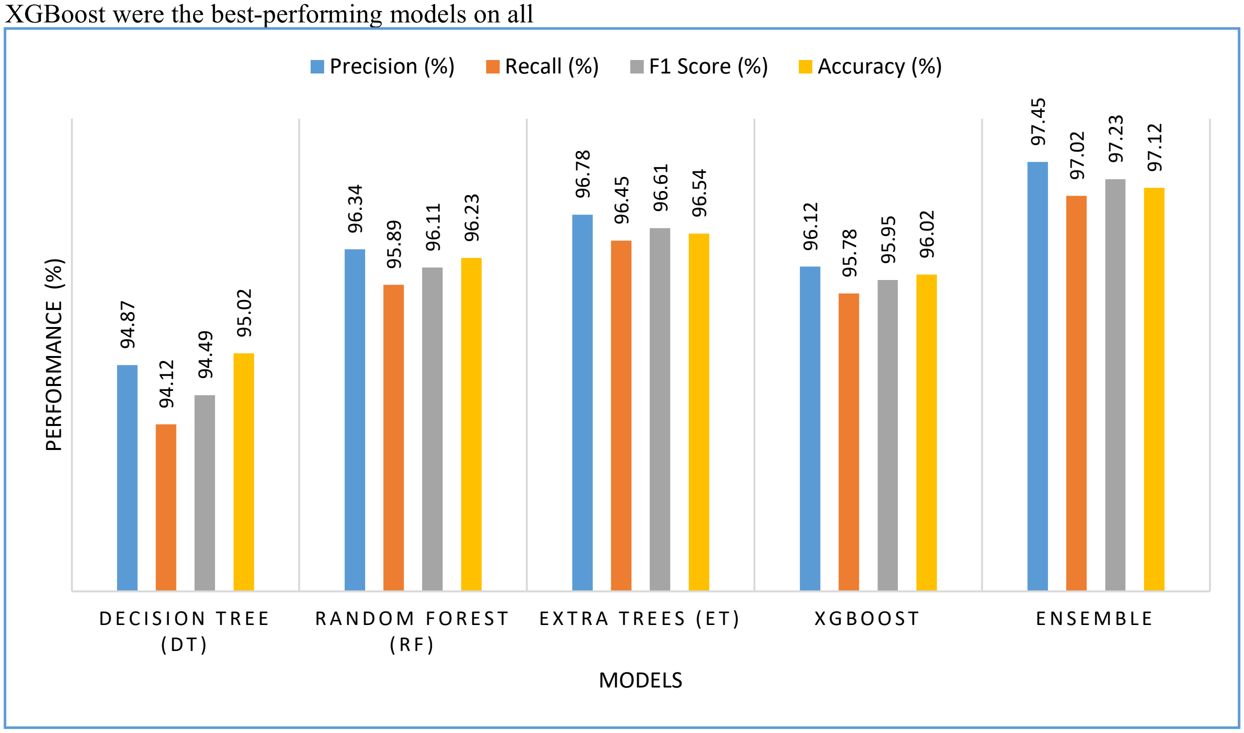

Figure 11: Performance Comparison Of Intrusion Detection Models (With Optimizations) Using The NSL-KDD Dataset

Machine learning models before and after hyperparameter tunning were applied to the KDD dataset, as per the Excel file shown in Figure 11. The models are assessed in precision, recall, F1 score, and accuracy. Performing hyperparameter tuning markedly boosted the performance of all models. Finally, the ensemble models achieved the highest performance across all metrics: Extra Trees and XGBoost. This graph shows the need for hyperparameter tuning to reach an optimal performance of intrusion detection systems. Optimizations significantly improved the performance of the machine learning models in our ensemble framework. Bayesian Optimization hyperparameter tuning improved all models' precision, recall, f1-score, and accuracy. For example, Decision Tree accuracy enhanced from 93.12% to 94.35% in the CICIDS dataset, while Ensemble model accuracy improved from 96.34% to 97.34%. Likewise, on the NSL-KDD dataset, Ensemble model accuracy increased from 96.02% to 97.45%. Such results show the importance of tuning the hyperparameters for each model individually, as this allows these individual models to generalize and capture patterns that otherwise may lead to robust and reliable anomaly detection performance.

Table 7: Performance Of The Proposed Model Compared With The State-Of-The-Art Models

Model	Source	Dataset	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
NLPIDS	Das et al. [3]	CIC-IDS2017	94.12	93.89	93.45	93.67
		NSL-KDD	93.89	93.67	93.23	93.45
KNN-SVM-DL Ensemble IDS	Yousefnezhad et al. [4]	CIC-IDS2017	95.67	95.45	95.23	95.34

		NSL-KDD	95.34	95.12	94.89	95.00
Gradient Boosting + Logistic Regression Ensemble	Fitni and Ramli [5]	CIC-IDS2017	96.23	96.12	95.89	96.00
		NSL-KDD	96.00	95.78	95.56	95.67
Improved Ensemble for IoT (Crow-Search)	Srivastava et al. [9]	CIC-IDS2017	96.78	96.56	96.34	96.45
		NSL-KDD	96.45	96.23	96.01	96.12
HELAD IDS Framework	Zhong et al. [22]	CIC-IDS2017	97.01	96.89	96.78	96.83
		NSL-KDD	96.83	96.72	96.61	96.66
Optimized Ensemble Learning (OEL-ID)	Proposed Algorithm	CIC-IDS2017	97.34	97.45	97.12	97.28
		NSL-KDD	97.45	97.56	97.23	97.39

As shown in Table 7, a comparison with related works shows the superiority of OEL-ID in comparing five state-of-the-art systems on the CIC-IDS2017 and NSL-KDD datasets. The OEL-ID framework outperformed all baseline models in either dataset and attained the highest accuracy, precision, recall, and F1 score. This shows that it is robust and performs well in detecting different network intrusions, including high-dimensional and imbalanced data. For the CIC-IDS2017 dataset, OEL-ID achieved an accuracy of 97.34%, superior to HELAD IDS (97.01%) and other models, including the Improved IoT Ensemble (96.78%) and ensembles based on Gradient Boosting (96.23%). Likewise, OEL-ID achieved an exceptional 97.45% accuracy on the NSL-KDD dataset, surpassing the HELAD IDS framework (96.83%) and other methods, thus verifying its excellence. Recursive Feature Elimination (RFE)-based feature selection, Bayesian Optimization-based hyperparameter tuning for individual models, and a weighted-averaging-based ensemble strategy for model predictions have contributed to these results. A comparative analysis shows that the OEL-ID framework outperforms while providing well-balanced precision and recall, thereby reducing false-negative and false-positive predictions. This closes the void that the state-of-the-art approaches fail to fill while its scalability and adaptability render it appropriate for real-world intrusion detection systems.

Traditional intrusion detection systems may use rule-based approaches or single-model machine learning methods, in contrast to our work, which adopts a novel Optimized Ensemble Learning-based Intrusion Detection (OEL-ID) framework, combining Recursive Feature Elimination (RFE) with Bayesian Optimization for hyperparameter tuning, as well as weighted averaging ensemble learning to optimize accuracy and generalization. Earlier studies [Kumar et al. 2016; Boukitnik et al. Although ensemble-based IDS methods have been proposed (e.g., [1], [2]), there is often no integrated feature selection leading to computational inefficiency. Others such as NLP-IDS [3] and KNN-SVM-DL Ensemble IDS [4] led to better detection rates but had difficulty with highly dimensional feature spaces as well as class imbalances. If we compare with HELAD IDS [22] and hybrid ensemble method proposed by Velásquez et al. Our proposed framework enhances the detection accuracy with adaptive hyperparameter tuning and meta-learning on top of existing approaches [27], allowing robustness against evolving cyber threats. We conduct experiments to validate our approach and achieve the state-of-the-art performance against several sophisticated models as OEL-ID achieves 97.34% accuracy over CIC-IDS2017 dataset and 97.45%

on NSL-KDD dataset, overcoming existing benchmark models with the trade-off of computational efficiency. It shows that we can successfully tackle the most significant roadblocks of feature redundancy, model overfitting, and detection inefficiency, thus making it an applicable technique for real-world cyber security.

5. DISCUSSION

Evolving network-based threats have made network intrusion detection systems (NIDS) essential for protecting modern networks. Methods have historically been limited regarding anomaly detection with classical machine learning models (rule-based methods), high-dimensional data, and imbalanced datasets. Deep learning has shown promise in tackling these issues, but existing techniques often suffer from insufficient feature engineering, hyperparameter tuning, and weak ensemble frameworks that lead to unsatisfactory performance. However, state-of-the-art methodologies suffer from inadequate feature selection, less tuning of hyperparameters, and the absence of a substantial ensemble learning framework where we could generate an ensemble of predictions from various models. The perceived deficiencies drove a desire for a new deep learning framework enriched with a combination of cutting-edge approaches, such as Recursive Feature Elimination (RFE), Bayesian Optimization, and weighted-average-based ensemble learning, which would optimize for both accuracy and efficiency.

The proposed methodology has some novelties. Using RFE, a ranking of relevant features is built, thus reducing the dimensionality without losing the required information. Bayesian Optimization is a powerful tool for tuning model hyperparameters and can significantly enhance the performance of models. Finally, a meta-learning-based ensemble framework incorporates individual model predictions through weighted averaging, harnessing their strengths for improved performance. The results indicated that the suggested methodology provided substantially better detection rates than widely used or individual models, with significant differences in precision, recall, F1-score, and accuracy on both the CIC-IDS2017 and NSL-KDD datasets. The proposed method was validated experimentally through an ensemble model that attained an accuracy of 97.34% on the CIC-IDS dataset and 97.45% on the NSL-KDD dataset. The enhancements to these techniques successfully overcame the shortcomings of existing state-of-the-art approaches, specifically in dealing with imbalanced datasets and their inability to model complex relationships in high-dimensional spaces. Such research may have significant implications as it offers a scalable and resilient solution that could be useful in real-world NIDS applications. A more detailed description of the study's limitations can be found in Section 5.1 of this paper.

5.1 Limitations

There are three main limitations to the current study. Although the ensemble framework achieved high accuracy in previous work, it is computationally demanding, especially during the hyperparameter tuning and ensemble integration processes, which can hinder its use in a real-world setting. Second, the datasets used (CIC-IDS2017 and NSL-KDD) are static. Also the two datasets are not explicitly devised to imitate the actual trends of real-world cyberattacks as they evolve, which can limit the applicability of the models to novel threat patterns. Third, the review mainly discusses binary and multi-class classification but does not cover advanced anomaly detection approaches concerning zero-day attacks. Future research may address these limitations by optimizing for computational efficiency, utilizing real-time data streams, or integrating an adaptive model that can account for previously unseen threat vectors.

6. CONCLUSION AND FUTURE WORK

In this work, the Optimized Ensemble Learning-based Intrusion Detection (OEL-ID) framework was proposed, which enhanced the accuracy and robustness of intrusion detection using Recursive Feature Elimination (RFE) for feature selection, Bayesian Optimization for hyperparameter tuning, and weighted averaging-based ensemble learning. This work focuses on a novel to-be ensemble learning based intrusion detection system that enables using different base classifiers for effective attack detection, optimizing the number of features selected and hyperparameter tunings in a single model to reduce redundancy and computation while improving model generalization, and to test the performance of the proposed framework in comparison to several benchmark datasets to validate its effectiveness.

The models achieved their goals as the models recorded 97.34% accuracy on the CIC-IDS2017 dataset and 97.45% accuracy on the NSL-KDD dataset outperforming state-of-the-art intrusion detection models. By combining feature engineering, hyperparameter tuning, and ensemble learning, we achieved a highly accurate and generalizable detection framework capable of tackling challenges including class imbalance and high-dimensional data across a diverse range of attack types.

But still, limitations and threats to validity are present. One alginate limitation is the computational overhead associated with hyperparameter tuning and ensemble integration, which could limit real-time deployment in low-resource settings. On top of that, the framework is limited in its adaptability to evolving cyber threats and zero-day attacks, and is trained on static datasets (i.e. CIC-IDS2017 and NSL-KDD). Even though the model performs well in detecting known attacks, additional research should be conducted on its application using adaptive or online learning approaches for novel attack vectors. The solution to these drawbacks and limitations would be enhancing the computational cost in future implementations with real-time learning and more varied real-life datasets to show robustness and practical use. The integrated model can also be further enhanced by employing unsupervised and semi-supervised learning approaches to identify new cyber attacks with never-seen-before characteristics, hence contributing to the real-time cyber defense of the system by improving the model's adaptability.

REFERENCES

[1] Kumar, G., Thakur, K., & Ayyagari, M. R. (2020). MLEsIDSs: machine learning-based ensembles for intrusion detection systems—a review. The Journal of Supercomputing. doi:10.1007/s11227-020-03196-z

[2] Tama, B. A., & Lim, S. (2021). Ensemble learning for intrusion detection systems: A systematic mapping study and cross-benchmark evaluation. Computer Science Review, 39, 100357. doi:10.1016/j.cosrev.2020.100357

[3] Das, S., Ashrafuzzaman, M., Sheldon, F. T., & Shiva, S. (2020). Network Intrusion Detection using Natural Language Processing and Ensemble Machine Learning. 2020 IEEE Symposium Series on Computational Intelligence (SSCI). doi:10.1109/ssci47803.2020.9308268

[4] Yousefnezhad, M., Hamidzadeh, J., & Aliannejadi, M. (2021). Ensemble classification for intrusion detection via feature extraction based on deep Learning. Soft Computing. doi:10.1007/s00500-021-06067-8

[5] Fitni, Q. R. S., & Ramli, K. (2020). Implementation of Ensemble Learning and Feature Selection for Performance Improvements in Anomaly-Based Intrusion Detection Systems. 2020 IEEE International Conference on Industry 4.0, Artificial Intelligence, and Communications Technology (IAICT). doi:10.1109/iaict50021.2020.9172014

[6] Singh, P., & Ranga, V. (2021). Attack and intrusion detection in cloud computing using an ensemble learning approach. International Journal of Information Technology, 13(2), 565–571. doi:10.1007/s41870-020-00583-w

[7] Li, X., Zhu, M., Yang, L. T., Xu, M., Ma, Z., Zhong, C., … Xiang, Y. (2021). Sustainable Ensemble Learning Driving Intrusion Detection Model. IEEE Transactions on Dependable and Secure Computing, 1–1. doi:10.1109/tdsc.2021.3066202

[8] Folino, F., Folino, G., Guarascio, M., Pisani, F. S., & Pontieri, L. (2021). On learning effective ensembles of deep neural networks for intrusion detection. Information Fusion, 72, 48–69. doi:10.1016/j.inffus.2021.02.007

[9] Srivastava, G., G, T. R., Deepa, N., Prabadevi, B., & Reddy M, P. K. (2020). An ensemble model for intrusion detection in the Internet of Softwarized Things. Adjunct Proceedings of the 2021 International Conference on Distributed Computing and Networking. doi:10.1145/3427477.3429987

[10] Stiawan, D., Heryanto, A., Bardadi, A., Rini, D. P., Subroto, I. M. I., Kurniabudi, … Budiarto, R. (2021). An Approach for Optimizing Ensemble Intrusion Detection Systems. IEEE Access, 9, 6930–6947. doi:10.1109/access.2020.3046246

[11] Zhou, Y., Cheng, G., Jiang, S., & Dai, M. (2020). Building an Efficient Intrusion Detection System Based on Feature Selection and Ensemble Classifier. Computer Networks, 107247. doi:10.1016/j.comnet.2020.107247

[12] Kumar, G. (2019). An improved ensemble approach for effective intrusion detection.

The Journal of Supercomputing. doi:10.1007/s11227-019-03035-w

[13] Kunal, & Dua, M. (2020). Attribute Selection and Ensemble Classifier based Novel Approach to Intrusion Detection System. Procedia Computer Science, 167, 2191–2199. doi:10.1016/j.procs.2020.03.271

[14] Attota, D. C., Mothukuri, V., Parizi, R. M., & Pouriyeh, S. (2021). An Ensemble Multi-View Federated Learning Intrusion Detection for IoT. IEEE Access, 9, 117734–117745. doi:10.1109/access.2021.3107337

[15] Bhati, B. S., Rai, C. S., Balamurugan, B., & Al-Turjman, F. (2020). An intrusion detection scheme based on the ensemble of discriminant classifiers. Computers & Electrical Engineering, 86, 106742. doi:10.1016/j.compeleceng.2020.106742

[16] Jiang, Y., & Atif, Y. (2021). A selective ensemble model for cognitive cybersecurity analysis. Journal of Network and Computer Applications, 193, 103210. doi:10.1016/j.jnca.2021.103210

[17] Yi, F., Jiang, B., Wang, L., & Wu, J. (2020). Cybersecurity Named Entity Recognition Using Multi-Modal Ensemble Learning. IEEE Access, 8, 63214–63224. doi:10.1109/access.2020.2984582

[18] Leevy, J. L., Hancock, J., Zuech, R., & Khoshgoftaar, T. M. (2021). Detecting cybersecurity attacks across different network features and learners. Journal of Big Data, 8(1). doi:10.1186/s40537-021-00426-w

[19] Gunturi, S. K., & Sarkar, D. (2020). Ensemble machine learning models for the detection of energy theft. Electric Power Systems Research, 106904. doi:10.1016/j.epsr.2020.106904

[20] Zeadally, S., Adi, E., Baig, Z., & Khan, I. (2020). Harnessing Artificial Intelligence Capabilities to Improve Cybersecurity. IEEE Access, 1–1. doi:10.1109/access.2020.2968045

[21] Shaukat, K., Luo, S., Varadharajan, V., Hameed, I. A., & Xu, M. (2020). A Survey on Machine Learning Techniques for Cyber Security in the Last Decade. IEEE Access, 8, 222310–222354. doi:10.1109/access.2020.3041951

[22] Zhong, Y., Chen, W., Wang, Z., Chen, Y., Wang, K., Li, Y., … Li, K. (2019). HELAD: A Novel Network Anomaly Detection Model Based on Heterogeneous Ensemble Learning. Computer Networks, 107049. doi:10.1016/j.comnet.2019.107049

[23] Kaur, G. (2020). A comparison of two hybrid ensemble techniques for network anomaly detection in spark distributed environment. Journal of Information Security and Applications, 55, 102601. doi:10.1016/j.jisa.2020.102601

[24] Biswas, P., & Samanta, T. (2021). Anomaly detection using ensemble random forest in wireless sensor network. International Journal of Information Technology. doi:10.1007/s41870-021-00717-8

[25] Jain, M., & Kaur, G. (2021). Distributed anomaly detection using concept drift detection based hybrid ensemble techniques in streamed network data. Cluster Computing, 24(3), 2099–2114. doi:10.1007/s10586-021-03249-9

[26] Aniss Chohra, Paria Shirani, ElMouatez Billah Karbab, and Mourad Debbabi. (2022). Chameleon: Optimized feature selection using particle swarm optimization and ensemble methods for network anomaly detection. *Elsevier*. 117, pp.1-19. https://doi.org/10.1016/j.cose.2022.102684

[27] DAVID VELÁSQUEZ, ENRIQUE PÉREZ, XABIER OREGUI, ARKAITZ ARTETXE, JORGE MANTECA, JORDI ESCAYOLA MANSILLA, MAURICIO TORO, MIKEL MAIZA, AND BASILIO SIERRA. (2022). A hybrid machine-learning ensemble for anomaly detection in real-time industry 4.0 systems. *IEEE*. 10, pp.72024 - 72036. http://DOI:10.1109/ACCESS.2022.3188102

[28] Wasim Khan, and Mohammad Haroon. (2022). An unsupervised deep learning ensemble model for anomaly detection in static attributed social networks. *Elsevier*. 3, pp.153-160. https://doi.org/10.1016/j.ijcce.2022.08.002

[29] Gunay Abdiyeva-Aliyeva, Jeyhun Aliyev, and Ulfat Sadigov. (2022). Application of classification algorithms of Machine learning in cybersecurity. *Elsevier*. 215, pp.909-919. https://doi.org/10.1016/j.procs.2022.12.093

[30] Adeel Abbas, Muazzam A. Khan, Shahid Latif, Maria Ajaz, Awais Aziz Shah, and Jawad Ahmad. (2022). A new ensemble-based intrusion detection system for internet of things. *Springer*. 47, p.1805–1819. https://doi.org/10.1007/s13369-021-06086-5

[31] Md. Alamgir Hossain, and Md. Saiful Islam. (2023). Ensuring network security with a robust intrusion detection system using

ensemble-based machine learning. *Elsevier*. 19, pp.1-14. https://doi.org/10.1016/j.array.2023.100306

[32] Ngamba Thockchom, Moirangthem Marjit Singh, and Utpal Nandi. (2023). A novel ensemble learning-based model for network intrusion detection. *Springer*. 9, p.5693–5714. https://doi.org/10.1007/s40747-023-01013-7

[33] Mouaad Mohy-Eddine, Azidine Guezzaz, Said Benkirane, Mourade Azrour, and Yousef Farhaoui. (2023). An ensemble learning based intrusion detection model for industrial IoT security. *IEEE*. 6(3), pp.273 - 287. http://DOI:10.26599/BDMA.2022.9020032

[34] Roopa Golchha, Apoorv Joshi, Govind Prasad Gupta. (2023). Voting-based Ensemble Learning approach for Cyber Attacks Detection in Industrial Internet of Things. *Elsevier*. 218, pp.1752-1759. https://doi.org/10.1016/j.procs.2023.01.153

[35] Md Amirul Islam, Md Ashraf Uddin, Sunil Aryal, and Giovanni Stea. (2023). An ensemble learning approach for anomaly detection in credit card data with imbalanced and overlapped classes. *Elsevier*. 78, pp.1-21. https://doi.org/10.1016/j.jisa.2023.103618

[36] Mohammad Kazim Hooshmand, Manjaiah Doddaghatta Huchaiah, Ahmad Reda Alzighaibi, Hasan Hashim, El-Sayed Atlam, and Ibrahim Gad. (2024). Robust network anomaly detection using ensemble learning approach and explainable artificial intelligence (XAI). *Elsevier*. 94, pp.120-130. https://doi.org/10.1016/j.aej.2024.03.041

[37] MUKHTAR AHMED, JINFU CHEN, ERNEST AKPAKU, REXFORD NII AYITEY SOSU, and AJMAL LATIF. (2024). DELM: Deep Ensemble Learning Model for Anomaly Detection in Malicious Network Traffic-based Adaptive Feature Aggregation. *ACM*, pp.1-36. https://doi.org/10.1145/3690637

[38] Tin Lai, Farnaz Farid, Abubakar Bello, and Fariza Sabrina. (2024). Ensemble learning based anomaly detection for IoT cybersecurity via Bayesian hyperparameters sensitivity analysis. *Springer*. 7(44), pp.1-18. https://doi.org/10.1186/s42400-024-00238-4

[39] RANDA ALLAFI, AND IBRAHIM R. ALZAHRANI. (2024). Enhancing Cybersecurity in the Internet of Things Environment Using Artificial Orca Algorithm and Ensemble Learning Model. *IEEE*. 12, pp.63282 - 63291. http://DOI:10.1109/ACCESS.2024.3390093

[40] Zong-Zhi Lin, Thomas D. Pike, Mark M. Bailey, and Nathaniel D. Bastian. (2024). A hypergraph-based machine learning ensemble network intrusion detection system. *IEEE*, pp.1-12. http://DOI:10.1109/TSMC.2024.3446635

[41] Canadian Institute for Cybersecurity (2017) *CICIDS2017 Dataset*. Available at: https://www.unb.ca/cic/datasets/ids-2017.html

[42] Tavallaee, M., Bagheri, E., Lu, W. and Ghorbani, A.A. (2009) ‘A detailed analysis of the KDD CUP 99 dataset’, *2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications*, pp. 1–6. NSL-KDD dataset available at: https://www.unb.ca/cic/datasets/nsl.html

[43]Surender Mogilicharla and Upendra Kumar Mummadi,(2024),The literature survey: Precision agriculture for crop yield optimization,AIP Conference Proceedings, vol. 3007, no. 1,https://doi.org/10.1063/5.0192998.

[44]Surender Mogilicharla and Upendra Kumar Mummadi,(2024),Grain quality analysis from the image through the approaches of segmentation,AIP Conference Proceedings, vol. 3007, no. 1,https://doi.org/10.1063/5.0192997.

[45]Surender Mogilicharla and Upendra Kumar Mummadi,(2024),Enhanced Rice Plant Disease Identification: A Hybrid Approach of Transfer Learning, SVM, and PCA,Journal of Theoretical and Applied Information Technology,vol. 102,no. 9,p. 4164,https://www.jatit.org/volumes/Vol102No9/38Vol102No9.

[46]Surender Mogilicharla and Upendra Kumar Mummadi,(2024),Precision Nutrition Management and Fertilizer Optimization in Paddy Crops:A Hybrid Approach for Deficiency Detection and Recommendation Using Segmentation, Transfer Learning, and Hyperparameter Tuning,International Journal of Intelligent Systems and Applications in Engineering, vol. 12,no. 4,pp.3079–3086

[47]Surender Mogilicharla and Upendra Kumar Mummadi,(2024),Enhancing Precision Agriculture:A Hybrid Approach for Paddy Seed Classification and Fraud Detection,Nanotechnology Perceptions, vol. 20, no. S14, pp. 2429–2445,https://nano-

ntp.com/index.php/nano/article/view/3123/2346.

[49]Chander, Nenavath, and Mummadi Upendra Kumar,(2024),Enhanced pelican optimization algorithm with ensemble-based anomaly detection in industrial internet of things environment. Cluster Computing,1-19.

[50]Chander, Nenavath, and Mummadi Upendra Kumar.(2024),Metaheuristic feature selection with deep learning enabled cascaded recurrent neural network for anomaly detection in Industrial Internet of Things environment.Cluster Computing 26.3,1801-1819.

[51]Chander, Nenavath, and M. Upendra Kumar,(2022),Comparative analysis on deep learning models for detection of anomalies and leaf disease prediction in Cotton Plant Data.Congress on intelligent systems. Singapore: Springer Nature Singapore.

[52]Chander, Nenavath, and M. Upendra Kumar,(2023),Metaheuristics with Deep Convolutional Neural Network for class imbalance handling with anomaly detection in industrial IoT environment.J.Theor.Appl. Inf. Technol. 101.

[53]Chander, Nenavath, and M. Upendra Kumar,(2020),Machine learning based outlier detection techniques for IoT data analysis: a comprehensive survey.Iaeme Publication 11,2348-2362.

[54]Prasanthi, B., Suresh Pabboju, and D. Vasumathi. "QUERY ADAPTIVE HASH BASED IMAGE RETRIEVAL IN INTENT IMAGE SEARCH." Journal of Theoretical & Applied Information Technology 93.2 (2016).

[55]Prasanthi, B., Pabboju, S. & Vasumathi, D. A Novel Indexing and Image Annotation Structure for Efficient Image Retrieval. Arab J Sci Eng 43, 4203–4213 (2018). https://doi.org/10.1007/s13369-017-2827-1

[56]Prasanthi, B., Suresh, P., Vasumathi, D. (2017). Index-Based Image Retrieval-Analyzed Methodologies in CBIR. In: Vishwakarma, H., Akashe, S. (eds) Computing and Network Sustainability. Lecture Notes in Networks and Systems, vol 12. Springer, Singapore. https://doi.org/10.1007/978-981-10-3935-5_24

[57]D. Vasumathi, S. Pabboju and B. Prasanthi, "Specific query semantic signatures in web page re-ranked image retrieval," 2016 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), Chennai, India, 2016, pp. 1-8, doi: 10.1109/ICCIC.2016.7919710.

[58]Prasanthi, B & Pabboju, Suresh & Devara, Vasumathi. (2021). Feature Selection based Reduction in Dimensions and Indexing of Images for Efficient Image Retrieval. 456-461. 10.1109/ESCI50559.2021.9397054.

[59]Rahman, M.A. Hijab, M Fouzia Sayeedunnisa, S Masarath Saba Patil, G.R.C.Priya Ranjani,(2024)A. IRADF: ARTIFICIAL INTELLIGENCE ENABLED CLINICAL DECISION SUPPORT SYSTEM FOR DIAGNOSING RHEUMATOID ARTHRITIS USING X-RAY IMAGES Journal of Theoretical and Applied Information Technology, 102(19), pp. 7163–7177

[60]Eslavath, R, and U. K.Mummadi.(2023),ENSIC: Feature Selection on Android Malware Detection Attributes Using an Enhanced Non-Linear SVM Integrated With Cross Validator.International Journal of Intelligent Systems and Applications in Engineering, vol. 12, no. 2,pp. 495-04, https://www.ijisae.org/index.php/IJISAE/article/view/4294.

[61]Ravi, E ,Kumar, M.U.Ahmad, S.S.(2024).A Novel Mechanism for Tuning Neural Network for Malware Detection in Android Device. In: Rajagopal, S., Popat,K.,Meva, D,Bajeja, S.(eds) Advancements in Smart Computing and Information Security.Communications in Computer and Information Science, vol 2039. Springer, Cham. https://doi.org/10.1007/978-3-031-59100-6_18

[62]Ravi Eslavath,Upendra Kumar Mummadi,(2024),Enhancing Android Malware Detection: A Grid-Tuned Two-Layered Stacking Approach,SSRG International Journal of Electronics and Communication Engineering, vol. 11, no. 9, pp. 253-269,Crossref, https://doi.org/10.14445/23488549/IJECE-V11I9P122

[63]Ravi,E,Kumar,M.U.(2022).Android Malware Detection with Classification Based on Hybrid Analysis and N-gram Feature Extraction.Advancements in Smart Computing and Information Security. ASCIS 2022. Communications in Computer and Information Science, vol 1760. Springer,

Cham. https://doi.org/10.1007/978-3-031-23095-0_13

[64]Ravi,Eslavath,and Mummadi Upendra Kumar(2022).A comparative study on machine learning and deep learning methods for malware detection.Journal of Theoretical and Applied Information Technology 100.20.

[65]Imtiyaz Khan,A.Yashwanth Reddy, Maniza Hijab,Kotari Sridevi,Syed shabbeer Ahmad,D.Shravani,(2024),Secure and efficient data sharing scheme for multi-user and multi-owner scenario in federated cloud computing.Journal of Theoretical and Applied Information Technology, Vol. 102, No. 6.

[66]Needa Iffath, Upendra Kumar Mummadi, Fahmina Taranum, Syed Shabbeer Ahmad,Imtiyaz Khan,D.Shravani,(2024),Phishing website detection using ensemble learning models.https://doi.org/10.1063/5.0192754.

[67]Saba Noor Ayesha Khanum,Upendra Kumar Mummadi,Fahmina Taranum,Syed Shabbeer Ahmad,Imtiyaz Khan,D.Shravani(2024),Emotion recognition using multi-modal features and CNN classification.

[68]Dr. D. Shravani, Anusha Padala,(2020),Image Processing: Human Facial Expression Identification using Convolutional Neural Networks, Turkish Online Journal of Qualitative Inquiry (TOJQI), Volume 11.

[69]Mahalakshmi,C.V.S.S.,Mridula,B.,Shravani, D,(2020),Automatic Water Level De- tection Using IoT,Advances in Decision Sciences,Image Processing, Security and Computer Vi-sion.Learning and Analytics in Intelligent Systems,vol 4. Springer, Cham.

[70]D.Shravani,Imtiyaz Khan,Amogh Deshmukh,Veeramalla Anitha,Masrath Saba ,Syed Shabbeer Ahmad.(2022),LISF: A Security Framework for Internet of Things (IoT) Integrated Distributed Applications. Journal of Advanced Zoology,43(1), https://doi.org/10.53555/jaz.v43i1.1985