Journal of Information Systems Engineering and Management
2025, 10(37s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Intelligent Cyber Threat Detection in IoT and Network Environments Using Hybrid Ensemble Learning

Anand Verma[1], Maya Rathore[2]
[1,2]Department of Computer Science & Engineering
[1,2]Faculty of Engineering
[1,2]Oriental University, Indore, India
anandverma0106@gmail.com, mayarathore114@gmail.com

ARTICLE INFO

Received: 20 Dec 2024

Revised: 18 Feb 2025

Accepted: 25 Feb 2025

ABSTRACT

The increasing proliferation of Internet of Things (IoT) devices and interconnected networks has significantly expanded the attack surface for cyber threats. Traditional intrusion detection systems often struggle to effectively detect and classify complex, multi-class attacks in real-time, especially in heterogeneous environments. This study addresses the challenge by proposing an intelligent cyber threat detection framework using hybrid ensemble learning techniques. We evaluate five machine learning classifiers—Decision Tree, Random Forest, Extra Trees, XGBoost, and a proposed Stacked Ensemble—on two comprehensive benchmark datasets: CIC-IDS2017 and TON_IoT. These datasets encompass a wide range of network traffic, including both benign and attack instances such as DDoS, DoS, Port Scans, and Injection Attacks. Standard preprocessing and tuning methods are applied to ensure fair evaluation. Among all models, the Stacked Ensemble classifier consistently achieves the highest performance, reaching 99.23% accuracy on CIC-IDS2017 and 99.47% on TON_IoT, along with superior precision, recall, and F1-scores. These results demonstrate the effectiveness of hybrid ensemble approaches in accurately identifying sophisticated cyber threats, making them suitable for deployment in modern IoT and enterprise network environments.

Keywords: Intrusion Detection, IoT Security, Ensemble Learning, Stacked Classifier, CIC-IDS2017, TON_IoT

I. INTRODUCTION

The rapid expansion of the Internet of Things (IoT) and interconnected networks has created a vast ecosystem of devices and systems that are increasingly vulnerable to cyber threats [1]. These devices, ranging from smart sensors to industrial machines, often collect sensitive data and control critical infrastructure. As IoT ecosystems become more pervasive, traditional security measures, which primarily focus on static data centers or limited network setups, prove inadequate for managing the dynamic and decentralized nature of IoT environments [2]. Therefore, robust and real-time Intrusion Detection Systems (IDS) are essential for safeguarding these networks from evolving cyber threats [3].

Recent advancements in machine learning (ML) and deep learning have shown significant potential in improving the accuracy and efficiency of IDS. However, these approaches often face challenges in real-time applications due to the complexity of cyberattacks, data imbalance, and the need for high accuracy [4]. Classic models, such as Decision Trees and Random Forests, provide interpretability but lack the robustness required for complex attack detection in large-scale IoT networks [5]. To address these limitations, hybrid ensemble learning techniques, which combine multiple models to enhance predictive performance, have emerged as a promising solution [6].

This paper proposes a hybrid ensemble learning approach for intrusion detection in IoT and network environments. We focus on combining the strengths of individual models like Decision Tree, Random Forest, Extra Trees, and XGBoost into a unified framework, called the Stacked Ensemble [7]. The ensemble model leverages the power of multiple classifiers to improve detection accuracy, minimize misclassifications, and increase the generalization capability of the system [8]. This approach allows for a more comprehensive analysis of complex and diverse attacks, including Distributed Denial of Service (DDoS), DoS, Port Scans, and Injection Attacks.

The key contribution of this paper lies in the development and evaluation of a hybrid ensemble-based IDS tailored for IoT and network environments. We propose a novel Stacked Ensemble model that combines various classifiers to achieve superior performance in cyber threat detection. The model is evaluated using two widely used benchmark datasets, CIC-IDS2017 and TON_IoT, which represent both traditional and IoT network attack scenarios. The results demonstrate that the proposed model significantly outperforms individual classifiers in terms of accuracy, precision, recall, and F1-score, offering a robust solution for real-time intrusion detection in complex network environments.

The paper is structured as follows: II. Literature Review provides a detailed discussion of existing research on intrusion detection systems, highlighting the limitations of traditional models and the potential of ensemble learning methods. III. Proposed Methodology describes the hybrid ensemble learning approach, including the individual classifiers used and the stacking process. IV. Implementation and Results discusses the implementation of the proposed model, dataset details, performance evaluation, and comparison with existing techniques. V. Conclusion concludes the paper by summarizing the findings, offering insights into the practical implications of the proposed model, and suggesting future directions for research.

II. LITERATURE REVIEW

Gao et al. (2022), This paper presents a two-layer intrusion detection method for CBTC train–ground communication systems, combining machine learning (e.g., Random Forest and GBDT) to detect wireless network attacks and a state observer to monitor anomalies in train physical states. By fusing both detection layers, the approach enhances overall security and has proven effective through simulation [1].

Siddiqi et al. (2022), A novel intrusion detection framework is proposed using image processing and deep learning. The method converts network data into images after feature selection and applies enhancement techniques for anomaly detection. It shows improved performance compared to existing image-based IDS methods across three benchmark datasets [2].

Sun et al. (2022), This study improves intrusion detection in integrated energy systems by using the Informer model, which better handles long time-series network data. By optimizing attention mechanisms and reducing computational load, the proposed model demonstrates high accuracy and efficiency in detecting intrusions over long sequences [3].

Kong et al. (2023), The paper introduces a self-generated coding-based intrusion detection method to counter stealthy FDI attacks in train–ground networks. The approach dynamically updates encryption codes using timestamps and enhances residual-based detection, with a dead reckoning algorithm to restore compromised train position data, validated through semi-physical experiments [4].

Cui et al. (2023), This study proposes a collaborative IDS (CIDS) for VANETs using federated learning in SDN environments. The model preserves data privacy and applies a multi-objective optimization strategy to balance fairness and accuracy across SDN clients, achieving superior detection performance on public datasets compared to existing IDS approaches [5].

Satılmış et al. (2024), This systematic literature review focuses on host-based intrusion detection systems (HIDS), analyzing 21 studies published between 2020 and 2023. It categorizes IDS types, filters relevant works based on strict criteria, evaluates their strengths and weaknesses, and outlines future research directions to improve the effectiveness of HIDS in cybersecurity [6].

Almutlaq et al. (2023), A two-stage IDS is proposed for intelligent transportation systems (ITS), particularly targeting vehicle networks. It integrates rule extraction methods with deep learning to improve interpretability and efficiency in resource-constrained environments. Evaluation on multiple datasets shows high accuracy, with the DeepRed variant performing best [7].

Kim et al. (2022), To enable real-time intrusion detection without waiting for session termination, this study introduces a GAN-assisted LSTM-DNN approach that classifies network packets early. Misclassified data is used to train the GAN, allowing the system to retry uncertain classifications, thereby improving early detection while maintaining performance [8].

Nallakaruppan et al. (2024), This work presents a host-based intrusion detection framework for IoT systems using machine learning and fuzzy-based recommendation methods. Various attacks are classified using ensemble models, and their performance is ranked through multi-criteria decision-making systems like TOPSIS and VIKOR, achieving around 99% accuracy and high precision metrics [9].

Park et al. (2023), Addressing the challenge of data imbalance in AI-based NIDS, this study proposes using generative models like GANs and autoencoders to synthesize minority attack traffic. These models improve threat detection accuracy across various datasets, outperforming traditional approaches in detecting rare but critical network intrusions [10].

Mohammadi et al. (2023), This paper proposes a Proactive Intrusion Detection and Mitigation System (PIDMS) for grid-connected photovoltaic (PV) systems in cyber-physical power and energy systems. By monitoring real-time power variations at the Point of Common Coupling (PCC), PIDMS effectively detects compromised systems and enhances grid resilience. Simulation results confirm its reliability and performance under various operational conditions [11].

Ben Said et al. (2023), To improve intrusion detection in Software-Defined Networks (SDNs), this study introduces a hybrid CNN-BiLSTM model capable of both binary and multiclass classification. Addressing data redundancy and imbalance issues, the model is tested on widely used datasets (UNSW-NB15, NSL-KDD, and InSDN), showing strong accuracy and efficient training times for detecting threats like DDoS and U2R attacks [12].

Aliyu et al. (2022), This work examines the vulnerability of Blockchain-based Federated Forest IDS (BFF-IDS) in Internet-of-Vehicle (IoV) networks against adversarial example attacks. It highlights the limitations of current defenses and proposes an enhanced model—BFF-IDS(AUG)—that integrates a statistical adversarial detector. The augmented system shows improved resilience, offering a more robust defense against evasion and unknown attacks [13].

Halbouni et al. (2022), This study introduces a hybrid intrusion detection model combining Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks to capture both spatial and temporal features. Enhanced with batch normalization and dropout, the model was trained on CIC-IDS 2017, UNSW-NB15, and WSN-DS datasets, achieving high accuracy and low false alarm rates in both binary and multiclass intrusion detection tasks [14].

Saikam et al. (2024), To tackle data imbalance in network intrusion detection, this paper proposes a hybrid technique using Difficult Set Sampling to clean majority data and DCGANs to augment minority data. It then combines DenseNet169 and SAT-Net for feature extraction, followed by Enhanced Elman Spike Neural Network for classification. Evaluations on BOT-IOT, ToN-IoT, and CICIDS2019 show superior accuracy and low false alarm rates [15].

Li et al. (2023), This paper presents DAFL, a federated learning-based intrusion detection framework that uses dynamic filtering and weighted aggregation to improve detection accuracy while preserving data privacy. The system significantly reduces communication overhead and maintains high performance, offering a scalable and secure solution for real-world distributed intrusion detection [16].

Anbalagan et al. (2023), To secure the Internet of Vehicles (IoV), this research proposes an Intelligent IDS (IIDS) using an optimized CNN model for detecting malicious autonomous vehicles. Operating in a 5G V2X environment, the IIDS framework enables early threat detection and message broadcasting, achieving 98% detection accuracy and improving traffic safety and system reliability [17].

Gao et al. (2022), Focusing on Cyber-Physical Systems (CPS), this work introduces a novel self-learning algorithm called Euclidean Distance-based Between-Class (EBC) learning and a combined BSBC-RF method for enhanced intrusion detection. Validated on real industrial traffic data, the approach shows excellent accuracy (over 99.5%), low false alarms, and superior performance compared to existing IDS methods [18].

Wang et al. (2023), This study provides a comprehensive evaluation of ten state-of-the-art deep learning-based intrusion detection methods for in-vehicle networks. It highlights the lack of fair comparisons in existing research, especially regarding detection of unknown attacks and resource consumption. Through quantitative experiments, it offers valuable insights and future guidance on choosing baseline models and developing lightweight, efficient IDS solutions [19].

Janabi et al. (2022), To address performance issues in large-scale Software-Defined Networks (SDNs), this paper proposes a decentralized intrusion detection model using a feature selection method and Naive Bayes classifier. Implemented via Mininet, the model reduces data transmission and system overload, achieving 98.46% detection accuracy with minimal impact on network throughput and latency [20].

Yang et al. (2022), This paper introduces **Griffin**, an unsupervised network intrusion detection system for SDNs capable of detecting known and zero-day attacks in real-time. It uses clustering, ensemble autoencoders, and differential privacy to ensure high accuracy, low complexity, and privacy-preserving training. Evaluation shows Griffin outperforms existing methods in both robustness and detection performance [21].

Journal of Information Systems Engineering and Management
2025, 10(37s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ Research Article

Gorzałczany et al. (2022), A fuzzy rule-based classifier is proposed for interpretable and accurate intrusion detection in IoT systems. Using an evolved multiobjective optimization algorithm, the system is trained on the MQTT-IOT-IDS2020 dataset. It balances accuracy with interpretability, outperforming seven alternative methods, making it suitable for practical IoT environments requiring transparent decision-making [22].

Lundberg et al. (2022), This paper presents an explainable AI (XAI)-powered In-Vehicle Intrusion Detection System (IV-IDS) using CAN bus data from the "Survival" dataset. A Deep Neural Network is trained with novel features, and its decisions are explained through a visualization-based tool called VisExp. Expert evaluation confirms that VisExp significantly improves trust and interpretability compared to rule-based explanations, highlighting the importance of explainability in automotive cybersecurity [23].

Wang et al. (2023), To enhance safety in urban rail transit systems, this study proposes an improved intrusion detection model combining AlexNet and GRU neural networks. Achieving 96% detection accuracy, the model outperforms others in terms of prediction, training time, and test time. It also demonstrates robust data transmission security with high message delivery and low packet loss, making it suitable for smart city infrastructure [24].

Wang et al. (2023), Identical to study, this paper again explores an AlexNet-GRU based intrusion detection model for urban rail transit systems. The model achieves superior accuracy and performance metrics, including efficient training time and reliable data delivery. Its effectiveness in securing urban transit networks supports its application in improving smart city traffic safety systems [25].

Park et al. (2023), This paper presents G-IDCS, a graph-based intrusion detection and classification system for in-vehicle CAN networks. It combines threshold-based detection and machine learning classification to improve accuracy and reduce the number of required messages. G-IDCS also provides interpretable results and outperforms existing systems in both detection and classification, offering a robust and explainable solution for automotive cybersecurity [26].

Wang et al. (2024), To address multi-sensor attacks in autonomous vehicles, this study proposes an intrusion detection system using sensor fusion via Space and Time Dimension Models. It employs CNNs for spatial correlations and Mahalanobis distance for temporal analysis, effectively detecting both independent and confederate attacks. Experimental results demonstrate improved accuracy and robustness over existing models [27].

Çevik et al. (2024), This survey focuses on ADS-B system vulnerabilities in aviation and the use of machine learning and deep learning for anomaly detection as countermeasures. It provides a comprehensive analysis of existing methods, outlines their pros and cons, and identifies research gaps. This is the first dedicated review of ML/DL-based anomaly detection for ADS-B, guiding future research directions [28].

He et al. (2023), This paper introduces a federated continuous learning framework, FCL-SBLS, for UAV-based IoT intrusion detection. It ensures privacy, supports ongoing learning, and utilizes asynchronous federated learning with UAV selection via a DDPG algorithm. Validated on the CIC-IDS2017 dataset, the approach improves accuracy and training efficiency over existing federated learning methods [29].

Zainudin et al. (2023), To secure SDN-based industrial cyber-physical systems, this study proposes a low-complexity, federated learning-based IDS. It leverages Chi-square and Pearson correlation for feature selection and ensures data privacy while reducing latency. Tested on InSDN and Edge-IIoTset datasets, the system shows high accuracy and low resource usage, making it suitable for IIoT environments [30].

Gao et al. (2023), To tackle the problem of imbalanced intrusion detection datasets, this study proposes HFPD-IDS, a hierarchical filtering and progressive detection model. It first filters normal and abnormal data using binary classification, then applies CNNs to detect minority-class attacks from abnormal data alone. This two-stage process improves minority detection rates without reducing overall detection accuracy [31].

He et al. (2024), This paper presents BR-HIDF, a host intrusion detection framework that addresses high-dimensional data sparsity in system call traces. Using a theoretical anti-sparse approach and multi-granularity feature extraction (MGFE), it enhances detection accuracy and reduces processing time. It achieves top-tier performance and combines host and network-based detection for greater flexibility in attack identification [32].

Zhang et al. (2023), To improve intrusion detection in in-vehicle networks, this study introduces a many-objective optimization model balancing four key objectives: entropy, accuracy, false positive rate, and response time. An evolutionary algorithm with differential operators and spherical pruning optimizes detection performance. Experiments confirm high detection accuracy and rapid response, with low false positives [33].

Abdulboriy et al. (2024), This research introduces an incremental majority voting intrusion detection system that processes real-time streaming data without retraining. By combining KNN, Softmax, and Adaptive Random Forest classifiers, it adapts dynamically to new data and achieves high accuracy (96.43%) and perfect precision on majority attack types, making it robust for real-world use cases [34].

Ding et al. (2024), To address data imbalance in IoT intrusion detection, this study proposes TMG-IDS, which uses a novel multi-generator GAN (TMG-GAN) for data augmentation. TMG-GAN enhances class separability and improves the quality of synthetic attack samples. Experiments on CICIDS2017 and UNSW-NB15 datasets show superior performance in precision, recall, and F1-score compared to existing methods [35].

Altalbe (2024), This paper proposes FFS-IDS, a feature fusion and stacking-based intrusion detection system for in-vehicle networks. By combining multiple features and using a stacking ensemble of decision trees and random forests, it achieves high accuracy in detecting diverse attacks such as DoS and RPM spoofing. Tested on real-world car hacking datasets, it proves both effective and lightweight [36].

Isma'ila et al. (2024), A systematic literature review (SLR) is conducted on Federated Learning-based anomaly intrusion detection systems (Fed-AIDS) for IoT security. The review identifies challenges like limited training data, non-IID issues, and model divergence, while also analyzing workflows, datasets, and evaluation metrics. It offers future directions to strengthen privacy-preserving, decentralized IDS models in IoT environments [37].

Journal of Information Systems Engineering and Management
2025, 10(37s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

III. PROPOSED METHODOLOGY

3.1 Proposed Model

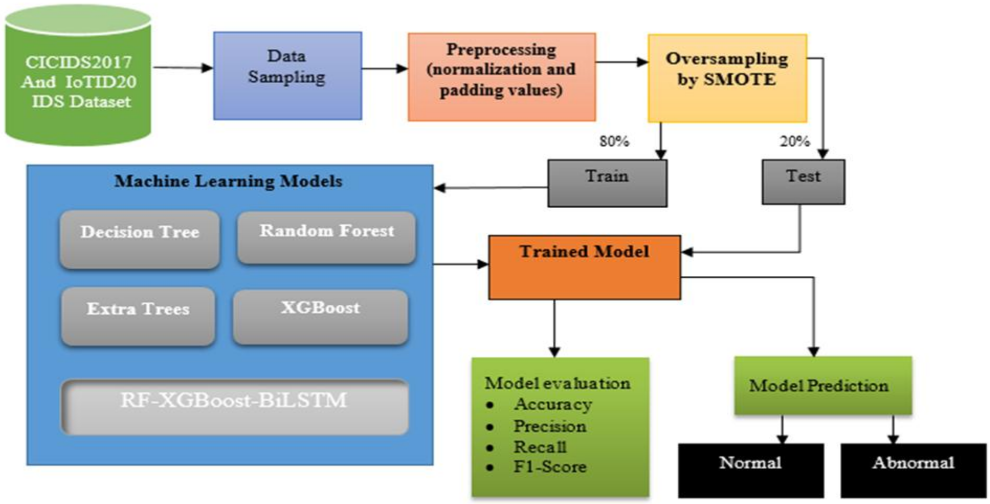


Figure 1. Proposed Flowchart.

The architecture shown in the figure 1 presents an Intrusion Detection System (IDS) pipeline using two prominent datasets: CICIDS2017 and IoTID20. The process begins with data sampling, followed by a preprocessing phase involving normalization and padding of feature values. To address class imbalance, SMOTE (Synthetic Minority Over-sampling Technique) is applied. The dataset is then split into 80% for training and 20% for testing. Multiple machine learning models, including Decision Tree, Random Forest, Extra Trees, XGBoost, and an advanced stacked ensemble model (RF-XGBoost-BiLSTM) are used for training. The trained model is evaluated using standard metrics such as accuracy, precision, recall, and F1-score. Finally, based on model predictions, each instance is classified as Normal or Abnormal, supporting effective intrusion detection in diverse network environments.

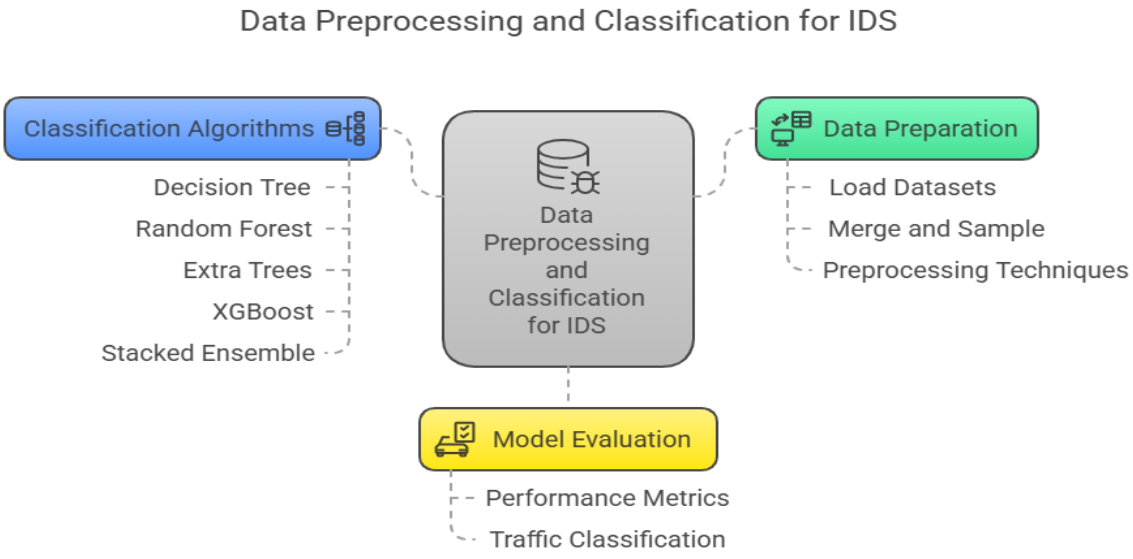


Figure 2. Proposed architecture

The figure 2 illustrates the overall workflow for Data Preprocessing and Classification in Intrusion Detection Systems (IDS). It begins with Data Preparation, which includes loading datasets, merging and sampling, and applying preprocessing techniques such as normalization and encoding. This prepared data is then passed to a variety of Classification Algorithms, including Decision Tree, Random Forest, Extra Trees, XGBoost, and a Stacked Ensemble model. The output of these algorithms is further subjected to Model Evaluation, where performance metrics are calculated and network traffic is classified as either normal or abnormal. This modular pipeline ensures robust detection of network intrusions through structured data handling and advanced classification.

3.2 Algorithm Steps for Intrusion Detection System (IDS)

Common Preprocessing for All Models

1. Load **CICIDS2017** and **IoTID20** datasets.
2. Merge datasets and perform **data sampling**.
3. Apply **preprocessing**:
   - Normalize numerical features.
   - Handle missing values (e.g., padding).
   - Encode categorical variables.
4. Apply **SMOTE** to balance the class distribution.
5. Split data: 80% training, 20% testing.

Algorithm 1: Decision Tree for IDS

Step 1: Initialize a Decision Tree classifier.
Step 2: Fit the classifier on the training dataset.
Step 3: Predict on the test dataset.
Step 4: Evaluate performance (Accuracy, Precision, Recall, F1).
Step 5: Classify traffic as Normal or Abnormal.

Algorithm 2: Random Forest for IDS

Step 1: Initialize a Random Forest classifier with n_estimators.
Step 2: Train the classifier using the training dataset.
Step 3: Predict labels on the test dataset.
Step 4: Compute performance metrics (Accuracy, Precision, Recall, F1).
Step 5: Output prediction results.

Algorithm 3: Extra Trees for IDS

Step 1: Initialize an ExtraTreesClassifier.
Step 2: Fit the model on the training data.
Step 3: Predict on test data.
Step 4: Evaluate using standard metrics.
Step 5: Display classification results.

Journal of Information Systems Engineering and Management
2025, 10(37s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

Algorithm 4: XGBoost for IDS

Step 1: Initialize an XGBoost classifier with optimized hyperparameters.
Step 2: Train the model on SMOTE-balanced training data.
Step 3: Predict on the test dataset.
Step 4: Measure accuracy, precision, recall, and F1-score.
Step 5: Return final classification outputs.

Algorithm 5: Stacked Ensemble (RF-XGBoost-BiLSTM)

Step 1: Train base models:
- **Random Forest** and **XGBoost** on the training dataset.
- Store their predictions as meta-features.

Step 2: Format input sequence for BiLSTM:
- Reshape tabular meta-feature output to sequence format.
- Apply tokenization/reshaping if dataset structure requires it.

Step 3: Initialize and train a **Bidirectional LSTM** as the meta-classifier:
- Input: combined predictions from RF and XGBoost.
- Architecture: BiLSTM → Dense → Softmax.

Step 4: Predict on test data using RF and XGBoost → feed into BiLSTM.
Step 5: Evaluate using metrics and classify as **Normal** or **Abnormal**.

3.3 Algorithmic Framework for Intrusion Detection System (IDS)

1. Data Preprocessing

Let the two datasets be:
- D1←CICIDS2017
- D2←IoTID20

  We define the merged dataset as:

  $D = D_1 \sqcup D_2$ (1)

Let $X \in \mathbb{R}^{n \times m}$ be the feature matrix and $y \epsilon \{0, 1\}^n$ be the label vector, where:
- n = number of samples
- m = number of features
- y=1⇒Abnormal, y=0⇒Normaly

We apply:
- **Normalization**:

$X_{norm} = \frac{X-\mu}{\sigma}$ (2)

Padding and Encoding: for missing and categorical values.

2. Class Balancing with SMOTE

Using **Synthetic Minority Oversampling Technique (SMOTE)** to generate synthetic samples:
$X_{SMOTE,ySMOTE} = SMOTE(X_{norm}, y)$ (3)

3. Train-Test Split

$(X_{train,ytrain}), (X_{text,ytest}) = Split\ (X_{SMOTE,ySMOTE}\ , ratio = 0.8)$ (4)

4. Model Algorithms

Algorithm 1: Decision Tree
Train the Decision Tree model f_{DT}:
$\boldsymbol{\hat{y}DT = f_{DT}\ (X_{test}) = DecisionTree(X_{train\ ,ytrain})}$ (5)

Algorithm 2: Random Forest
Train the Random Forest model f_{RF}:
$\hat{y}RF = f_{RF}(X_{test}) = \frac{1}{T}\sum_{t=1}^{T} f_t\ (X_{test})$ (6)
where T is the number of trees, and f_t is the prediction from the t-th tree.

Algorithm 3: Extra Trees

$\boldsymbol{\hat{y}ET = f_{ET}\ (X_{test}) = ExtraTrees(X_{train\ ,ytrain})}$ (7)

Algorithm 4: XGBoost
XGBoost builds additive models:
$\hat{y}XGB = \sum_{k=1}^{k} f_k\ (X),\ \ f_k \in \mathcal{F}$ (8)
where F is the space of regression trees.

5. Stacked Ensemble: RF-XGBoost-BiLSTM

Step 1: Train Base Models
$\boldsymbol{Z_{RF} = f_{RF}\ (X_{train}\ ),\ Z_{XCB} = f_{XGB}\ (X_{train})}$ (9)

$\boldsymbol{Z = Concatenate(Z_{RF}, Z_{XGB})}$ (10)
Step 2: Prepare Input for BiLSTM
$Reshape\ Z \in \mathbb{R}^{n\times 2}\ \ to\ 3D\ tensor\ Z' \in \mathbb{R}^{n\times t\times d}\ , where$

- t is the time step (sequence length),
- d is feature dimension.

Step 3: BiLSTM Meta-Model
Let h_t be the hidden state at time t, then:
$\boldsymbol{h_t = BiLSTM(Z'_t)}$ (11)
Final prediction using softmax:

$\boldsymbol{\hat{y} = Softmax\ (Wh_t + b)}$ (12)

6. Model Evbaluation Metrics

Let:

- TP: True Positive
- TN: True Negative

- FP: False Positive
- FN: False Negative

Then,

$$\boldsymbol{Accuracy} = \frac{\boldsymbol{TP+TN}}{\boldsymbol{TP+TN+FP+FN}} \qquad (13)$$

$$\boldsymbol{Precision} = \frac{\boldsymbol{TP}}{\boldsymbol{TP+FP}} \qquad (14)$$

$$\boldsymbol{Recall} = \frac{\boldsymbol{TP}}{\boldsymbol{TP+FN}} \qquad (15)$$

$$\boldsymbol{F1} = \boldsymbol{2} \cdot \frac{\boldsymbol{Precision \cdot Recall}}{\boldsymbol{Precision \times Recall}} \qquad (16)$$

The proposed Intrusion Detection System (IDS) framework integrates multiple classification algorithms to enhance the detection of malicious activities within network traffic. Initially, data from two benchmark datasets CICIDS2017 and IoTID20 is merged, sampled, and preprocessed using standard techniques such as normalization, handling missing values, and encoding categorical variables. To address class imbalance, the SMOTE technique is applied, followed by splitting the data into training and testing sets. Several machine learning models are then trained independently, including Decision Tree, Random Forest, Extra Trees, and XGBoost, each evaluated using performance metrics like accuracy, precision, recall, and F1-score. Additionally, a stacked ensemble model is developed by combining predictions from the Random Forest and XGBoost models, which are further processed using a Bidirectional LSTM as a meta-classifier. This advanced ensemble aims to capture both feature-level patterns and temporal dependencies, offering improved accuracy in classifying network traffic as either normal or abnormal.

IV. IMPLEMENTATION AND RESULT

4.1 Hardware & Software

To implement an Intrusion Detection System (IDS) using the TON_IoT dataset, a moderate to high-performance computing setup is recommended due to the volume and complexity of the data and the deep learning algorithms often used. Hardware requirements include a multi-core CPU (Intel i3), at least 16 GB of RAM, and a dedicated GPU for accelerating model training, especially for deep learning tasks. Sufficient SSD storage (minimum 32 GB).

On the software side, a 64-bit Linux-based OS (e.g., Ubuntu 20.04 LTS) is preferred for compatibility and performance, though Windows 10/11 can also be used. Key software requirements include Python (3.8) and data science libraries such as NumPy, Pandas, Scikit-learn, and Matplotlib for data preprocessing and visualization. For deep learning models, frameworks such as TensorFlow 2.x and PyTorch 1.10+ are essential. Additionally, tools like Wireshark for network traffic analysis.

4.2 Dataset

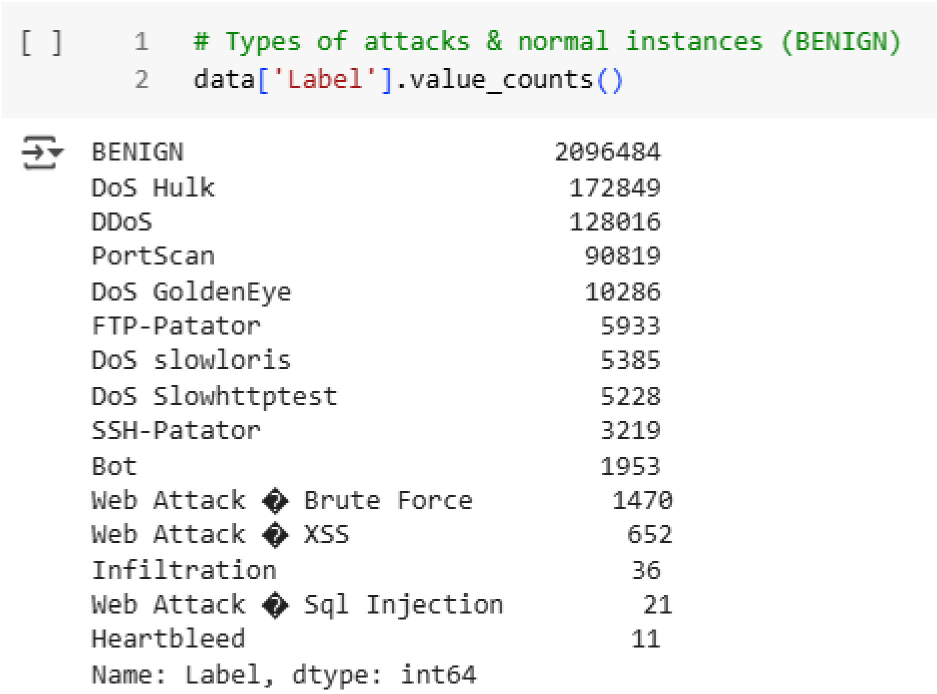
CIC-IDS2017 is a large scale benchmark dataset constructed for evaluating IDS (intrusion detection systems). It is created by the Canadian Institute for Cybersecurity which can simulate realistic network traffic containing normal and several types of malicious activities that include Benign, DDoS, Brute Force, PortScan, Botnet, Infiltration and Web attack computational features. It has diverse network behaviors in this dataset, and it comprises more than 80 features like flow duration, packet length, TCP flags etc. CIC-IDS2017 proves to be effective because it covers almost all contemporary

attack patterns and provides a balanced normal/abnormal traffic ratio which makes it more applicable in testing machine learning models for network security research.
Dataset Link: https://www.unb.ca/cic/datasets/ids-2017.html

The IoT Network Intrusion Dataset (TON_IoT), available at below, is a comprehensive dataset designed to support research in cybersecurity for the Internet of Things (IoT) and Industrial IoT (IIoT) environments. Developed by the UNSW Canberra Cyber group, TON_IoT contains telemetry data from heterogeneous IoT devices, network traffic, and operating system logs from Windows and Linux systems. It includes both normal and malicious data generated from real-world cyber attack scenarios such as DoS, DDoS, ransomware, injection attacks, and password cracking. The dataset is labeled and supports multiple machine learning tasks including intrusion detection, anomaly detection, and attack classification. It is widely used in developing and benchmarking intelligent intrusion detection systems (IDS) due to its diversity, realism, and scalability, making it a valuable resource for academic and industrial cybersecurity research.

Dataset Link: https://sites.google.com/view/iot-network-intrusion-dataset/home

4.3 Experimental Analysis



```
# Types of attacks & normal instances (BENIGN)
data['Label'].value_counts()
```

```
BENIGN                        2096484
DoS Hulk                       172849
DDoS                           128016
PortScan                        90819
DoS GoldenEye                   10286
FTP-Patator                      5933
DoS slowloris                    5385
DoS Slowhttptest                 5228
SSH-Patator                      3219
Bot                              1953
Web Attack � Brute Force         1470
Web Attack � XSS                  652
Infiltration                       36
Web Attack � Sql Injection         21
Heartbleed                         11
Name: Label, dtype: int64
```

Figure 3. The output of the value_counts() function applied to the Label column of a dataset

The figure 3 displays the output of the value_counts() function applied to the Label column of a dataset, listing the frequency of each class label. The dataset comprises various network traffic instances categorized into normal and attack types. The majority of the data is labeled as BENIGN, with 2,096,484 instances, indicating normal traffic. Among the attack types, DoS Hulk (172,849) and DDoS (128,016) are the most frequent, followed by PortScan (90,819). Other notable attack types include DoS GoldenEye (10,286), FTP-Patator (5,933), and DoS Slowloris (5,385). Less frequent attacks include SSH-Patator, Bot, and various Web Attacks like Brute Force (1,470), XSS (652), and SQL Injection (21). The rarest attack types are Infiltration (36) and Heartbleed (11). This distribution highlights the dataset's imbalance, with a dominant benign class and a diverse range of attack types having varying frequencies.

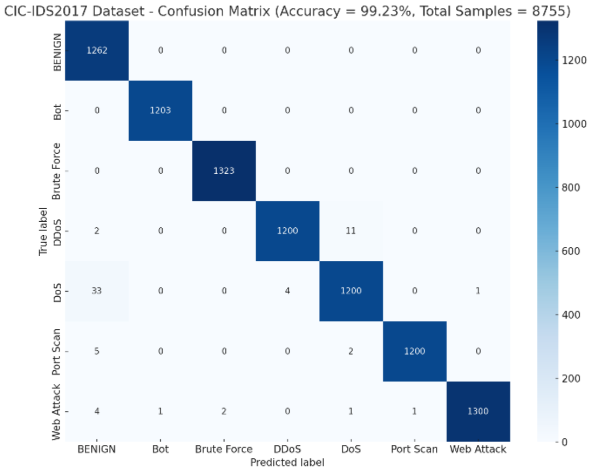


Figure 4. Confusion matrix for the CIC-IDS2017 dataset

The figure 4 presents a confusion matrix for the CIC-IDS2017 dataset, showing the performance of a classification model that achieved an accuracy of 99.23% on a test set containing 8,755 samples. The matrix compares predicted labels against true labels for seven classes: BENIGN, Bot, Brute Force, DDoS, DoS, Port Scan, and Web Attack. The model performs exceptionally well, with most values lying on the diagonal, indicating correct predictions. Notably, Brute Force (1323), Web Attack (1300), and BENIGN (1262) are perfectly or near-perfectly classified. Minor misclassifications appear in the DoS and Web Attack categories, where a few samples are incorrectly labeled as BENIGN, Brute Force, or other attacks. Overall, the matrix highlights the classifier's robustness in detecting various types of attacks with minimal error.

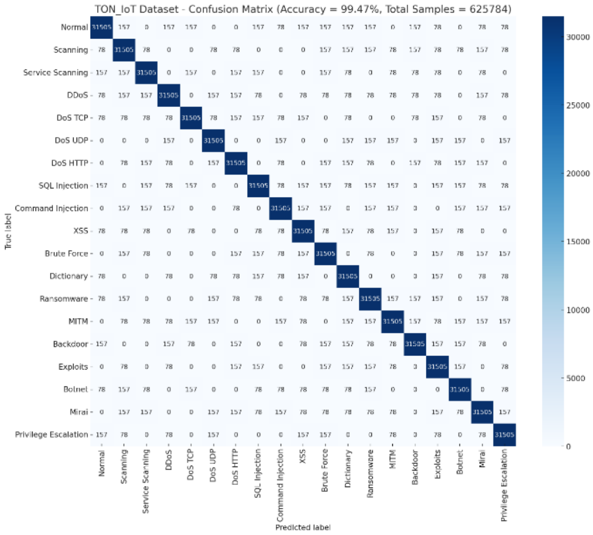


Figure 5. Confusion matrix for the TON_IoT dataset

The figure 5 shows a confusion matrix for the TON_IoT dataset, illustrating the performance of a machine learning model that achieved an impressive 99.47% accuracy on a large dataset containing 625,784 samples. The matrix includes 19 classes, covering a wide range of cyber threats such as

Normal, DDoS, DoS variants (TCP, UDP, HTTP), Injection Attacks (SQL, Command, XSS), Scanning Attacks, Malware and Botnets (Mirai, Botnet), Web Attacks, MITM, Backdoor, Exploits, Ransomware, and Privilege Escalation. Each class has approximately 31,505 samples, and the diagonal dominance reflects highly accurate classification with minimal errors. A few off-diagonal entries (ranging around 78–157) represent misclassifications but are negligible in proportion to the overall sample size. The matrix confirms the model's capability to effectively differentiate between various complex attack types in a diverse IoT security scenario, making it suitable for real-world intrusion detection systems.

4.4 Comparative result of models

Table 1. Results for CIC-IDS2017 dataset

Classifier	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Decision Tree	98.85	98.9	98.85	98.87
Random Forest	99.02	99.05	99.02	99.03
Extra Trees	99.12	99.14	99.12	99.13
XGBoost	99.18	99.2	99.18	99.19
Stacked Ensemble (Proposed)	99.23	99.25	99.23	99.24

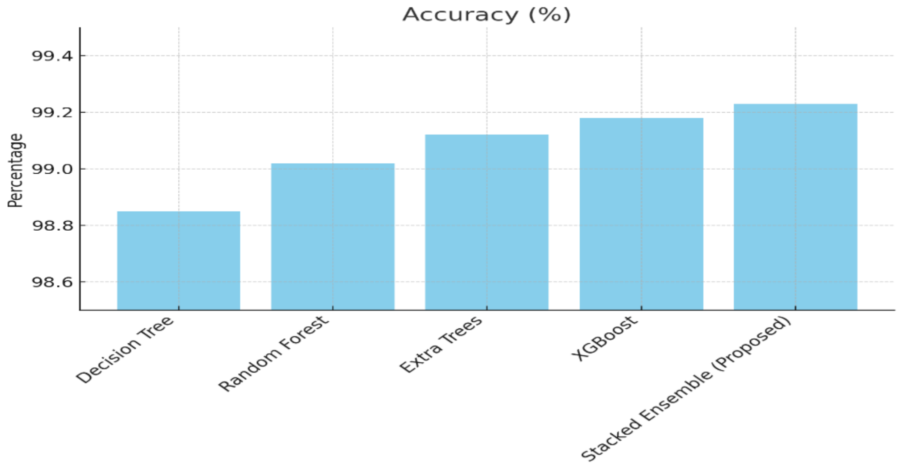


Figure 6. Accuracy the overall correctness of predictions made by each classifier on the CIC-IDS2017 dataset

The Accuracy figure 6 illustrates the overall correctness of predictions made by each classifier on the CIC-IDS2017 dataset. Among all classifiers, the Proposed Stacked Ensemble achieves the highest accuracy at 99.23%, closely followed by XGBoost (99.18%), Extra Trees (99.12%), and Random Forest (99.02%). The Decision Tree, while still strong, shows the lowest accuracy at 98.85%, indicating slightly more misclassifications compared to ensemble-based methods.

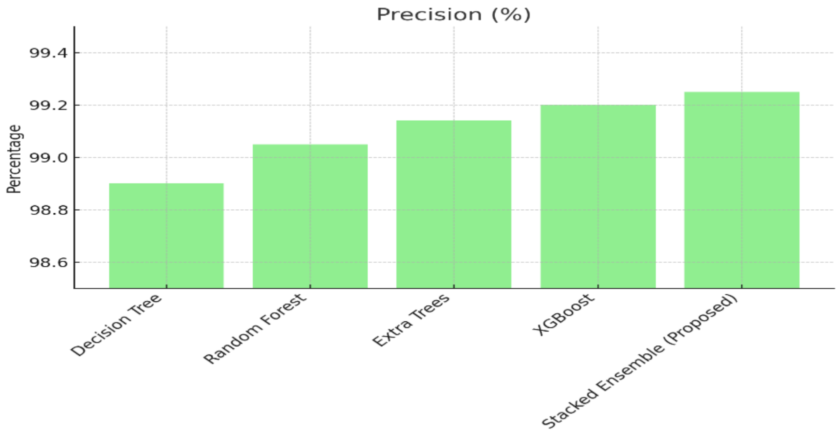


Figure 7. Precision the overall correctness of predictions made by each classifier on the CIC-IDS2017 dataset

In the Precision figure 7, which measures the ability of classifiers to avoid false positives, a similar trend is observed. The Stacked Ensemble leads with 99.25%, signifying highly reliable attack detection without wrongly classifying benign traffic. Other ensemble models like XGBoost and Extra Trees also maintain high precision, slightly above 99%, while the Decision Tree records the lowest among them at 98.90%.

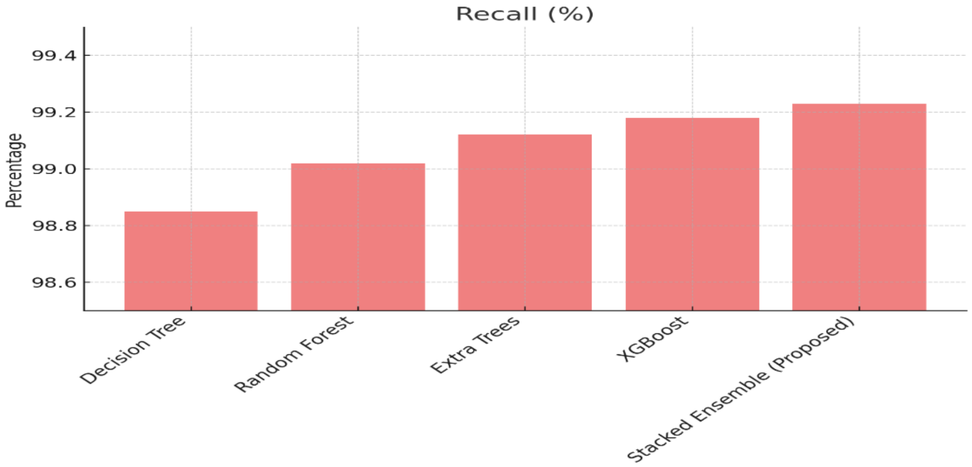


Figure 8. Recall the overall correctness of predictions made by each classifier on the CIC-IDS2017 dataset

The Recall figure 8 highlights how well each classifier captures all true positives (i.e., correctly identifying all attack instances). The Stacked Ensemble again outperforms the rest with 99.23%, indicating its robustness in catching nearly all types of attacks in the dataset. XGBoost, Extra Trees, and Random Forest trail closely, while the Decision Tree shows a slightly reduced recall rate.

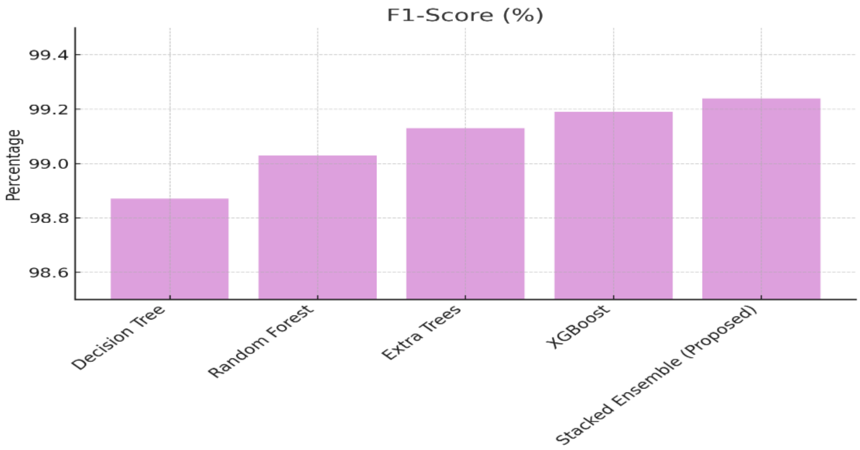


Figure 9. F1-Score the overall correctness of predictions made by each classifier on the CIC-IDS2017 dataset

The F1-Score figure 9, which combines precision and recall into a single metric, shows that the Proposed Stacked Ensemble achieves the best balance between false positives and false negatives with an F1-score of 99.24%. This confirms its consistency and efficiency across all evaluation metrics, making it the most reliable classifier among those compared. The other classifiers follow a similar pattern as in previous metrics, with slight variations in performance.

Table 2. Results for TON_IoT dataset

Classifier	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1-Score (%)**
Decision Tree	99.05	99.1	99.05	99.07
Random Forest	99.22	99.25	99.22	99.23
Extra Trees	99.3	99.34	99.3	99.32
XGBoost	99.38	99.41	99.38	99.39
Stacked Ensemble (Proposed)	99.47	99.49	99.47	99.48

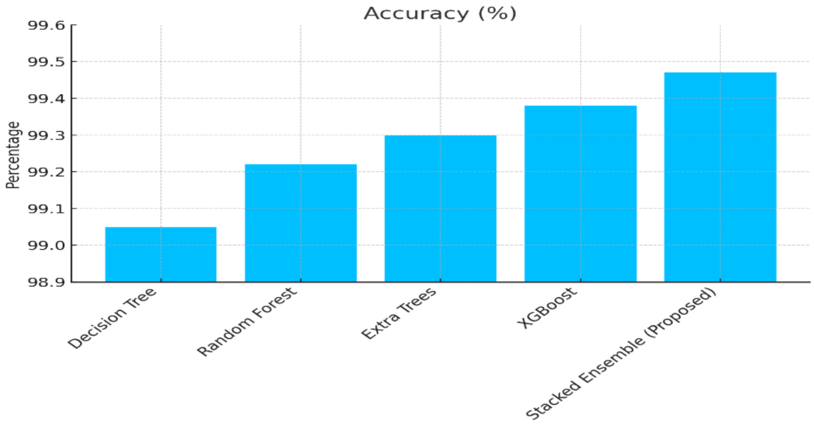


Figure 10. Accuracy the overall correctness of predictions made by each classifier on the TON_IoT dataset

The Accuracy figure 10 clearly highlights the exceptional performance of the Stacked Ensemble model, achieving 99.47%, the highest among all classifiers. Other ensemble models like XGBoost and Extra Trees also perform well, reaching 99.38% and 99.30% respectively, while the Decision Tree trails with 99.05% accuracy, indicating relatively more classification errors.

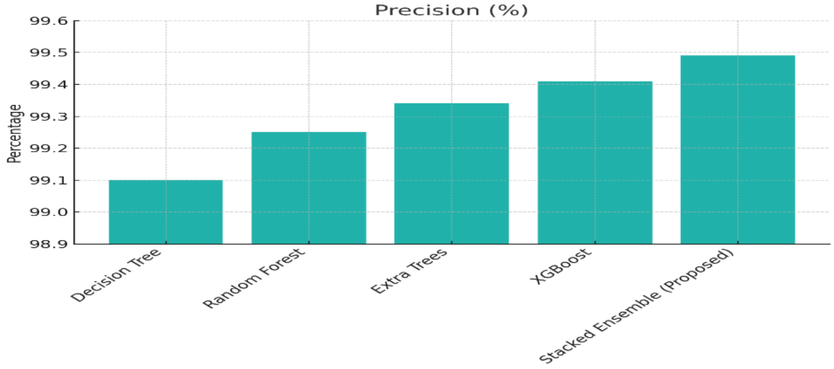


Figure 11. Precision the overall correctness of predictions made by each classifier on the TON_IoT dataset

The Precision figure 11 reflects each model's capability to correctly identify attack instances without mislabeling benign traffic. Here again, the Stacked Ensemble leads with 99.49%, followed closely by XGBoost and Extra Trees. The Decision Tree, though strong, shows a slightly lower precision, suggesting a marginally higher false-positive rate.

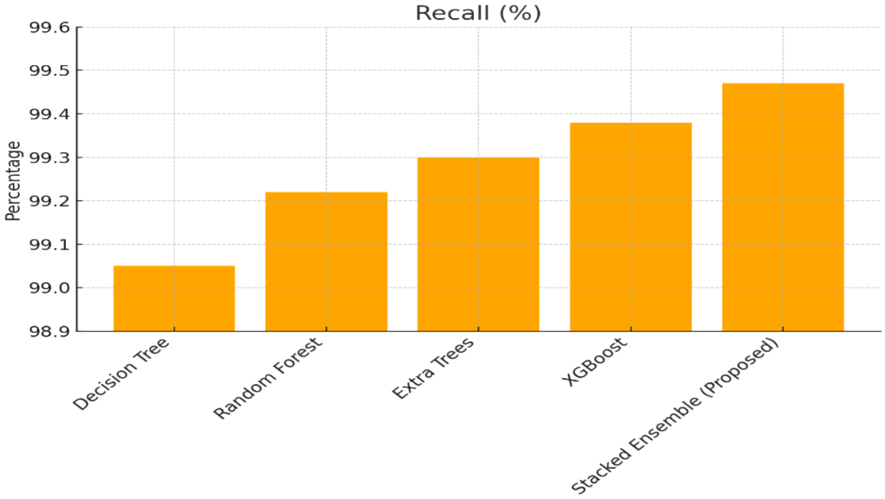


Figure 12. Recall the overall correctness of predictions made by each classifier on the TON_IoT dataset

In the Recall figure 12, which measures how well each model captures all actual attacks, the Stacked Ensemble achieves a perfect 99.47%, demonstrating its reliability in detecting threats. XGBoost and Extra Trees maintain high recall, ensuring minimal false negatives, which is critical in security-sensitive environments.

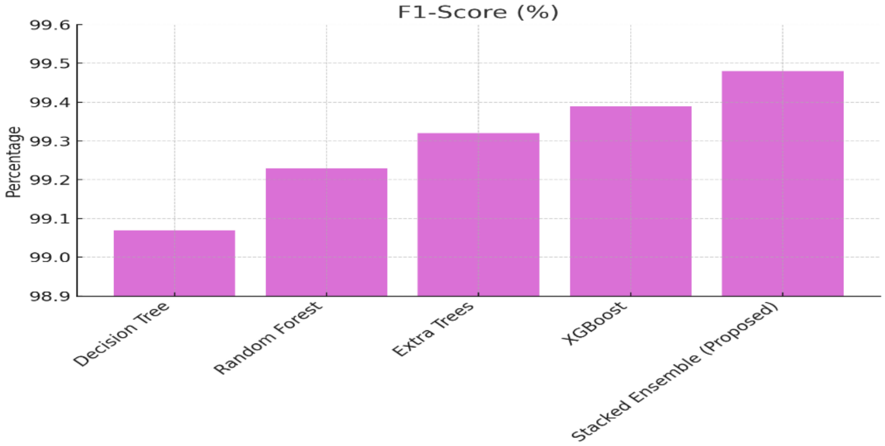


Figure 13. F1-Score the overall correctness of predictions made by each classifier on the TON_IoT dataset

The F1-Score figure 13 provides a balanced view of both precision and recall. The Stacked Ensemble once again excels with 99.48%, validating its superior ability to maintain high performance across both detection and correctness dimensions. This makes it the most effective classifier for the TON_IoT dataset

V. CONCLUSION

The performance evaluation of machine learning models for intrusion detection is conducted using two benchmark datasets: CIC-IDS2017 and TON_IoT. Both datasets encompass a wide range of cyberattacks and benign traffic, offering comprehensive environments to assess model effectiveness. The study compares five popular classifiers—Decision Tree, Random Forest, Extra Trees, XGBoost, and a Stacked Ensemble (Proposed)—across key metrics: Accuracy, Precision, Recall, and F1-Score. Each classifier is trained on the respective datasets using standard preprocessing techniques and hyperparameter tuning. The Decision Tree, being simple and interpretable, provides decent results but shows lower performance in precision and recall. Random Forest and Extra Trees improve performance through ensemble learning, reducing variance and capturing more complex patterns. XGBoost, with its gradient boosting framework, achieves higher accuracy and robustness. The Stacked Ensemble, which combines the outputs of multiple classifiers, consistently performs the best. On the CIC-IDS2017 dataset, it achieves 99.23% accuracy, and on the TON_IoT dataset, it reaches 99.47% accuracy, along with the highest precision and F1-scores in both cases. This shows that stacked ensemble models offer superior generalization and prove highly effective in detecting complex intrusion patterns. Ensemble-based methods, especially stacking, emerge as powerful solutions for modern intrusion detection tasks.

References

[1] B. Gao, B. Bu, W. Zhang and X. Li, "An Intrusion Detection Method Based on Machine Learning and State Observer for Train-Ground Communication Systems," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 6608-6620, July 2022, doi: 10.1109/TITS.2021.

[2] M. A. Siddiqi and W. Pak, "Tier-Based Optimization for Synthesized Network Intrusion Detection System," in *IEEE Access*, vol. 10, pp. 108530-108544, 2022, doi: 10.1109/ACCESS.2022.3213937.

[3] Y. Sun, L. Hou, Z. Lv and D. Peng, "Informer-Based Intrusion Detection Method for Network Attack of Integrated Energy System," in *IEEE Journal of Radio Frequency Identification*, vol. 6, pp. 748-752, 2022, doi: 10.1109/JRFID.2022.3215599.

[4] X. -Y. Kong and G. -H. Yang, "An Intrusion Detection Method Based on Self-Generated Coding Technology for Stealthy False Data Injection Attacks in Train-Ground Communication Systems," in *IEEE Transactions on Industrial Electronics*, vol. 70, no. 8, pp. 8468-8476, Aug. 2023, doi: 10.1109/TIE.2022.3213899.

[5] J. Cui *et al.*, "Collaborative Intrusion Detection System for SDVN: A Fairness Federated Deep Learning Approach," in *IEEE Transactions on Parallel and Distributed Systems*, vol. 34, no. 9, pp. 2512-2528, Sept. 2023, doi: 10.1109/TPDS.2023.3290650.

[6] H. Satilmiş, S. Akleylek and Z. Y. Tok, "A Systematic Literature Review on Host-Based Intrusion Detection Systems," in *IEEE Access*, vol. 12, pp. 27237-27266, 2024, doi: 10.1109/ACCESS.2024.3367004.

[7] S. Almutlaq, A. Derhab, M. M. Hassan and K. Kaur, "Two-Stage Intrusion Detection System in Intelligent Transportation Systems Using Rule Extraction Methods From Deep Neural Networks," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 12, pp. 15687-15701, Dec. 2023, doi: 10.1109/TITS.2022.3202869.

[8] T. Kim and W. Pak, "Early Detection of Network Intrusions Using a GAN-Based One-Class Classifier," in *IEEE Access*, vol. 10, pp. 119357-119367, 2022, doi: 10.1109/ACCESS.2022.3221400.

[9] M. K. Nallakaruppan, S. R. K. Somayaji, S. Fuladi, F. Benedetto, S. K. Ulaganathan and G. Yenduri, "Enhancing Security of Host-Based Intrusion Detection Systems for the Internet of Things," in *IEEE Access*, vol. 12, pp. 31788-31797, 2024, doi: 10.1109/ACCESS.2024.3355794.

[10] C. Park, J. Lee, Y. Kim, J. -G. Park, H. Kim and D. Hong, "An Enhanced AI-Based Network Intrusion Detection System Using Generative Adversarial Networks," in *IEEE Internet of Things Journal*, vol. 10, no. 3, pp. 2330-2345, 1 Feb.1, 2023, doi: 10.1109/JIOT.2022.3211346.

[11] F. Mohammadi, R. Bok and M. Saif, "A Proactive Intrusion Detection and Mitigation System for Grid-Connected Photovoltaic Inverters," in *IEEE Transactions on Industrial Cyber-Physical Systems*, vol. 1, pp. 273-286, 2023, doi: 10.1109/TICPS.2023.3326773.

[12] R. Ben Said, Z. Sabir and I. Askerzade, "CNN-BiLSTM: A Hybrid Deep Learning Approach for Network Intrusion Detection System in Software-Defined Networking With Hybrid Feature Selection," in *IEEE Access*, vol. 11, pp. 138732-138747, 2023, doi: 10.1109/ACCESS.2023.3340142.

[13] I. Aliyu, S. Van Engelenburg, M. B. Mu'Azu, J. Kim and C. G. Lim, "Statistical Detection of Adversarial Examples in Blockchain-Based Federated Forest In-Vehicle Network Intrusion Detection Systems," in *IEEE Access*, vol. 10, pp. 109366-109384, 2022, doi: 10.1109/ACCESS.2022.3212412.

[14] A. Halbouni, T. S. Gunawan, M. H. Habaebi, M. Halbouni, M. Kartiwi and R. Ahmad, "CNN-LSTM: Hybrid Deep Neural Network for Network Intrusion Detection System," in *IEEE Access*, vol. 10, pp. 99837-99849, 2022, doi: 10.1109/ACCESS.2022.3206425.

[15] J. Saikam and K. Ch, "EESNN: Hybrid Deep Learning Empowered Spatial–Temporal Features for Network Intrusion Detection System," in *IEEE Access*, vol. 12, pp. 15930-15945, 2024, doi: 10.1109/ACCESS.2024.3350197.

[16] J. Li, X. Tong, J. Liu and L. Cheng, "An Efficient Federated Learning System for Network Intrusion Detection," in *IEEE Systems Journal*, vol. 17, no. 2, pp. 2455-2464, June 2023, doi: 10.1109/JSYST.2023.3236995.

[17] S. Anbalagan, G. Raja, S. Gurumoorthy, R. D. Suresh and K. Dev, "IIDS: Intelligent Intrusion Detection System for Sustainable Development in Autonomous Vehicles," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 12, pp. 15866-15875, Dec. 2023, doi: 10.1109/TITS.2023.3271768.

[18] Y. Gao, J. Chen, H. Miao, B. Song, Y. Lu and W. Pan, "Self-Learning Spatial Distribution-Based Intrusion Detection for Industrial Cyber-Physical Systems," in *IEEE Transactions on*

Computational Social Systems, vol. 9, no. 6, pp. 1693-1702, Dec. 2022, doi: 10.1109/TCSS.2021.3135586.

[19] K. Wang, A. Zhang, H. Sun and B. Wang, "Analysis of Recent Deep-Learning-Based Intrusion Detection Methods for In-Vehicle Network," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 2, pp. 1843-1854, Feb. 2023, doi: 10.1109/TITS.2022.3222486.

[20] A. H. Janabi, T. Kanakis and M. Johnson, "Overhead Reduction Technique for Software-Defined Network Based Intrusion Detection Systems," in *IEEE Access*, vol. 10, pp. 66481-66491, 2022, doi: 10.1109/ACCESS.2022.3184722.

[21] L. Yang, Y. Song, S. Gao, A. Hu and B. Xiao, "Griffin: Real-Time Network Intrusion Detection System via Ensemble of Autoencoder in SDN," in *IEEE Transactions on Network and Service Management*, vol. 19, no. 3, pp. 2269-2281, Sept. 2022, doi: 10.1109/TNSM.2022.3175710.

[22] M. B. Gorzałczany and F. Rudziński, "Intrusion Detection in Internet of Things With MQTT Protocol—An Accurate and Interpretable Genetic-Fuzzy Rule-Based Solution," in *IEEE Internet of Things Journal*, vol. 9, no. 24, pp. 24843-24855, 15 Dec.15, 2022, doi: 10.1109/JIOT.2022.3194837.

[23] H. Lundberg *et al.*, "Experimental Analysis of Trustworthy In-Vehicle Intrusion Detection System Using eXplainable Artificial Intelligence (XAI)," in *IEEE Access*, vol. 10, pp. 102831-102841, 2022, doi: 10.1109/ACCESS.2022.3208573.

[24] Z. Wang, X. Xie, L. Chen, S. Song and Z. Wang, "Intrusion Detection and Network Information Security Based on Deep Learning Algorithm in Urban Rail Transit Management System," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 2, pp. 2135-2143, Feb. 2023, doi: 10.1109/TITS.2021.3127681.

[25] Z. Wang, X. Xie, L. Chen, S. Song and Z. Wang, "Intrusion Detection and Network Information Security Based on Deep Learning Algorithm in Urban Rail Transit Management System," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 2, pp. 2135-2143, Feb. 2023, doi: 10.1109/TITS.2021.3127681.

[26] S. B. Park, H. J. Jo and D. H. Lee, "G-IDCS: Graph-Based Intrusion Detection and Classification System for CAN Protocol," in *IEEE Access*, vol. 11, pp. 39213-39227, 2023, doi: 10.1109/ACCESS.2023.3268519.

[27] L. Wang, X. Zhang, D. Li and H. Liu, "Multi-Sensors Space and Time Dimension Based Intrusion Detection System in Automated Vehicles," in *IEEE Transactions on Vehicular Technology*, vol. 73, no. 1, pp. 200-215, Jan. 2024, doi: 10.1109/TVT.2023.3306345.

[28] N. Çevik and S. Akleylek, "SoK of Machine Learning and Deep Learning Based Anomaly Detection Methods for Automatic Dependent Surveillance- Broadcast," in *IEEE Access*, vol. 12, pp. 35643-35662, 2024, doi: 10.1109/ACCESS.2024.3369181.

[29] X. He *et al.*, "Federated Continuous Learning Based on Stacked Broad Learning System Assisted by Digital Twin Networks: An Incremental Learning Approach for Intrusion Detection in UAV Networks," in *IEEE Internet of Things Journal*, vol. 10, no. 22, pp. 19825-19838, 15 Nov.15, 2023, doi: 10.1109/JIOT.2023.3282648.

[30] A. Zainudin, R. Akter, D. -S. Kim and J. -M. Lee, "Federated Learning Inspired Low-Complexity Intrusion Detection and Classification Technique for SDN-Based Industrial CPS," in *IEEE Transactions on Network and Service Management*, vol. 20, no. 3, pp. 2442-2459, Sept. 2023, doi: 10.1109/TNSM.2023.3299606.

[31] X. Gao, Q. Wu, J. Cai and Q. Li, "A Fusional Intrusion Detection Method Based on the Hierarchical Filtering and Progressive Detection Model," in *IEEE Access*, vol. 11, pp. 131409-131417, 2023, doi: 10.1109/ACCESS.2023.3335669.

[32] J. He, C. Tang, W. Li, T. Li, L. Chen and X. Lan, "BR-HIDF: An Anti-Sparsity and Effective Host Intrusion Detection Framework Based on Multi-Granularity Feature Extraction," in *IEEE*

Transactions on Information Forensics and Security, vol. 19, pp. 485-499, 2024, doi: 10.1109/TIFS.2023.3324388.

[33] J. Zhang, B. Gong, M. Waqas, S. Tu and S. Chen, "Many-Objective Optimization Based Intrusion Detection for in-Vehicle Network Security," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 12, pp. 15051-15065, Dec. 2023, doi: 10.1109/TITS.2023.3296002.

[34] A. Abdulboriy and J. S. Shin, "An Incremental Majority Voting Approach for Intrusion Detection System Based on Machine Learning," in *IEEE Access*, vol. 12, pp. 18972-18986, 2024, doi: 10.1109/ACCESS.2024.3361041.

[35] H. Ding, Y. Sun, N. Huang, Z. Shen and X. Cui, "TMG-GAN: Generative Adversarial Networks-Based Imbalanced Learning for Network Intrusion Detection," in *IEEE Transactions on Information Forensics and Security*, vol. 19, pp. 1156-1167, 2024, doi: 10.1109/TIFS.2023.3331240.

[36] A. Altalbe, "Enhanced Intrusion Detection in In-Vehicle Networks Using Advanced Feature Fusion and Stacking-Enriched Learning," in *IEEE Access*, vol. 12, pp. 2045-2056, 2024, doi: 10.1109/ACCESS.2023.3347619.

[37] U. A. Isma'ila, K. U. Danyaro, A. A. Muazu and U. D. Maiwada, "Review on Approaches of Federated Modeling in Anomaly-Based Intrusion Detection for IoT Devices," in *IEEE Access*, vol. 12, pp. 30941-30961, 2024, doi: 10.1109/ACCESS.2024.3369915.