Enhancing IoT network defense: advanced intrusion detection via ensemble learning techniques

Salah El Hajla, El Mahfoud Ennaji, Yassine Maleh, Soufyane Mounir
LaSTI Laboratory, National School of Applied Sciences Khouribga, Sultan Moulay Slimane University, Beni Mellal, Morocco

Article Info

Article history:

Received Feb 12, 2024
Revised Apr 8, 2024
Accepted May 12, 2024

Keywords:

Ensemble learning
Intrusion detection system
IoT security
Machine learning
Network anomaly detection

ABSTRACT

The Internet of things (IoT) has evolved significantly, automating daily activities by connecting numerous devices. However, this growth has increased cybersecurity threats, compromising data integrity. To address this, intrusion detection systems (IDSs) have been developed, mainly using predefined attack patterns. With rising cyber-attacks, improving IDS effectiveness is crucial, and machine learning is a key solution. This research enhances IDS capabilities by introducing binary attack identification and multiclass attack categorization for IoT traffic, aiming to improve IDS performance. Our framework uses the 'BoT-IoT' and 'TON-IoT' datasets, which include various IoT network traffic and cyber-attack scenarios, such as DDoS and data infiltration, to train machine learning and ensemble models. Specifically, it combines three machine learning models-decision tree, resilient backpropagation (RProp) multilayer perceptron (MLP), and logistic regression-into ensemble methods like voting and stacking to improve prediction accuracy and reduce detection errors. These ensemble classifiers outperform individual models, demonstrating the benefit of diverse learning techniques. Our framework achieves high accuracy, with 99.99% for binary classification on the BoT-IoT dataset and 97.31% on the ToN-IoT dataset. For multiclass classification, it achieves 99.99% on BoT-IoT and 96.32% on ToN-IoT, significantly enhancing IDS effectiveness against IoT cybersecurity threats.

Corresponding Author:

Salah El Hajla
LaSTI Laboratory, National School of Applied Sciences Khouribga, Sultan Moulay Slimane University
Beni Mellal, Morocco
Email: salah.elhajla@usms.ac.ma

1. INTRODUCTION

The proliferation of the internet of things (IoT) heralds a new era in networking, characterized by the seamless interconnection of myriad devices, ranging from autonomous vehicles and smart home appliances to wearable technology. These devices, equipped with limited computational resources [1], communicate over the Internet, contributing to an anticipated network of 29.7 billion connected devices by 2027 [2]. The economic impact of this technological revolution is estimated to significantly influence the global economy, with projections suggesting a value between $3.9 and $11.1 trillion by 2025 [3]. Despite the remarkable advancements in IoT, the network faces significant cybersecurity vulnerabilities that pose a threat to the integrity and functionality of these interconnected systems.

Addressing these vulnerabilities requires a nuanced understanding of both the technical and security dimensions of IoT. Traditional security measures, while foundational, fall short in addressing the dynamic and sophisticated nature of modern cyber threats. In response, researchers have explored the application of machine learning (ML) techniques in enhancing IoT network defense mechanisms. ML's ability to analyze large datasets and identify patterns offers a promising avenue for the development of advanced intrusion detection systems (IDS).

Churcher *et al*. [4], compared distinct algorithms over the BoT-IoT dataset for binary and multiclass detection. The outcomes of the proposed framework identified random forest (RF) as the top-performing model for binary classification, while K-nearest neighbors (KNN) exhibited the greatest accuracy in multiclass classification. Jaradat *et al.* [5] present a ML-based method for constructing an IDS. Their approach incorporates the use of supervised classification models such as decision tree (DT), support vector machine (SVM), and resilient backpropagation (RProp) on the CICIDS2017 dataset. They utilize the KNIME analytics platform for building classifiers and the MATLAB tool for selecting features. The DT classifier achieved the highest accuracy of 94.72%. Qaddoura *et al.* [6], proposed a methodology that involved three key phases: the reduction phase, the utilization of SVM, the SMOTE technique, and the single hidden layer feed-forward neural network (SLFN) phase. The findings from this study indicated that employing SVM-SMOTE with the SLFN technique with a defined frequency of 0.9 when combined with a k=3 value for the k-means++ clustering approach, produced improved outcomes compared to other detection strategies and parameter settings. Vishwakarma and Jain [7] proposed a botnet detection framework incorporating IoT honeypots and utilizing ML classifiers. Their study involved leveraging traffic data collected from honeypots to train ML classifiers for botnet detection. Pokharel *et al.* [8], introduced a hybrid IDS model, integrating Naive Bayes (NB) and SVM. The dataset of historical logs underwent preprocessing and normalization for this research. Following these enhancements, the suggested model achieved an accuracy and precision rate of 95%. The study revealed an improvement in classifier performance upon the incorporation of session-based features.

However, while these contributions have been pivotal, they often encountered limitations in scalability and adaptability to evolving threat landscapes. A critical gap also remains in the efficacy and efficiency of IDS for IoT. Specifically, the rapid evolution of cyber threats necessitates more dynamic and adaptable defense mechanisms. The literature reveals a particular need for: i) enhancing the detection accuracy of IDS in diverse IoT scenarios; ii) reducing the computational overhead associated with ML models; and iii) improving the scalability of security solutions to accommodate the growing IoT ecosystem.

To this end, ensemble learning methods have emerged as a potent solution, leveraging the collective power of multiple ML models to improve prediction accuracy and stability. Recent studies on anomaly detection [9]-[15] underscore the application of ensemble learning models to improve the efficiency of existing anomaly-based detection strategies. The ensemble classifier, which combines and utilizes various models for predicting performance, demonstrates superior performance compared to a single learning algorithm. Chakraborty *et al.* [12], proposed an ensemble model for outlier detection, overcoming the problems of unbalanced data by obtaining specific characteristics through a stacked autoencoder (SAE) and integrating them in an ensemble probabilistic neural network for both singular and multi-outlier identification. This dependence on the stacked autoencoder contributes to enhanced efficiency and reliability. Al-Haija *et al.* [14], further explored the application of kernel methods, ensemble methods, and neural networks to identify oddities and harmful activities within the IoT environment. Notably, the ensemble learning techniques outperformed other techniques in terms of both detection rates and accuracy.

This study introduces an advanced IDS framework that integrates ensemble learning techniques with ML models, specifically designed for the IoT context. Our approach uniquely combines multiple learning algorithms to improve detection accuracy while addressing the limitations of individual models in scalability and adaptability, this set of classifiers includes DT, Rprop MLP, logistic regression, and RF. By employing ensemble voting and stacking methods, we aim to create a more robust and efficient IDS capable of countering the dynamic threat landscape in IoT networks. Our contributions are twofold: i) we propose a novel IDS framework for classifying binary and multiclass attacks within IoT network traffic, emphasizing the application of feature selection strategies to augment the IDS's effectiveness; and ii) through rigorous evaluation using prominent IoT datasets, we demonstrate the superior performance of our approach compared to existing models, highlighting the potential of our framework in significantly improving IoT network defense.

The present paper is structured as follows: section 2 offers a detailed explanation of the adopted methodology and the proposed framework design. The findings, examinations, discussion, and assessment methodology are covered in section 3. Finally, section 4 delineates and summarizes the study's overall conclusion, along with the planned future work.

2. METHOD

This study adopts a systematic approach to develop an advanced IDS for IoT networks, utilizing ML models and ensemble learning techniques. Our methodology aims to tackle significant cybersecurity challenges in IoT networks by improving anomaly detection accuracy. This section outlines the experimental procedures, offering detailed insights into data preparation, the learning process, and the evaluation metrics used. The suggested system is decomposed into three separate components, each executed through a series of steps. These modules include the data preparation component (DP), the learning process component (LP), and the evaluation process component (EP). They handle inputs from ToN-IoT and BoT-IoT datasets through a sequential series of operations to achieve anomaly-based detection. The system's design is illustrated in Figure 1.

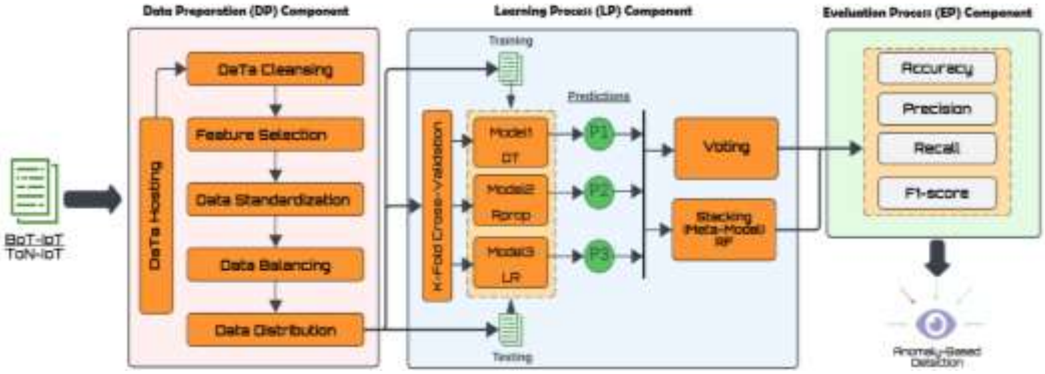


Figure 1. Scheme of our IDS framework

2.1. Datasets description

2.1.1. BoT-IoT dataset

The Bot-IoT dataset has been employed in the experiment to detect simulated attacks within the IoT network [16]. This dataset comprises information gathered from the industrial internet of things (IIoT) devices, sourced from the cyber range lab of UNSW Canberra. It encompasses both regular data traffic and traffic patterns induced by botnets resulting from diverse forms of attacks [17]. The dataset comes in two versions: the complete edition, comprising more than 73.3 million records, and the 5% version, containing around 3.66 million entries. Table 1 and Figure 2 provide information on the training and testing sets for the 5% version utilized in this research. The BoT-IoT dataset encompasses the following target categories: benign category, distributed denial of service (DDoS)/DoS TCP attacks, DDoS/DoS UDP attacks, DDoS/DoS HTTP attacks, keylogging, and sniffing.

2.1.2. ToN-IoT dataset

The second data set utilized in this research is the ToN-IoT, encompassing multiple data sources gathered from the whole IIoT networks. This comprises sensor data from connected nodes, records from Windows and Linux operating systems, and connectivity data. The diverse data were collected through a medium-sized IoT network established by the Canberra cyber range labs in UNSW, this dataset can be accessed through the ToN-IoT repository [18]. The datasets within ToN-IoT are in CSV file type, featuring a designated column showing either attack or benign action, and a subclass labeled "attack-type" specifying a variety of attack types. These cyberattacks were launched and gathered throughout the IIoT network, against various IoT and IIoT sensors. The attacks identified in this network dataset can be categorized into one of nine types: scanning, injection attack, DoS attack, DDoS attack, man-in-the-middle attack, backdoor, cross-site scripting, ransomware, and password cracking. In the ToN-IoT network dataset, each data point consists of 44 features and is labeled with an 'attack-type' categorized as either "attack" or "normal". Table 2 and Figure 3 present the statistics for both normal and attack data records in the train-test ToN-IoT dataset.

Table 1. Description of the 5% Bot-IoT dataset

	BoT-IoT 5%-version dataset	
Class category	Train set	Test set
DDoS	1,541,315	385,309
DoS	1,320,148	330,112
Reconnaissance	72,919	18,163
Normal	370	107
Theft	65	14
Total	2,934,817	733,705

Table 2. Description of ToN-IoT network dataset

ToN-IoT network dataset	
Attack type	No of records
Backdoor	20,000
DDoS	20,000
DoS	20,000
Injection	20,000
Mitm	1,043
Password	20,000
Ransomware	20,000
Scanning	20,000
Xss	20,000
Normal	300,000
Total	461,043

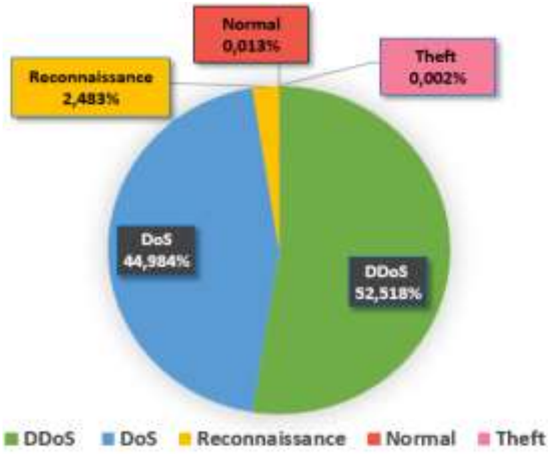


Figure 2. 5% Bot-IoT dataset statistical records

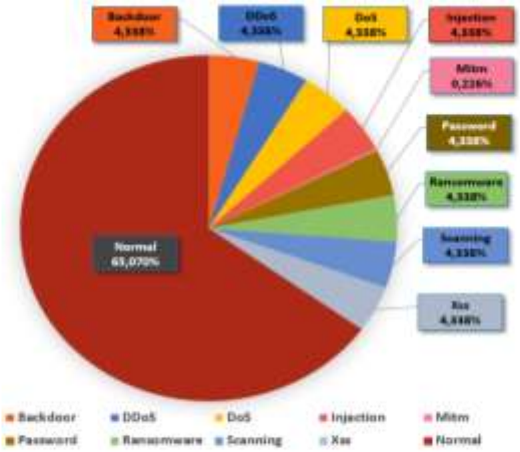


Figure 3. ToN-IoT network dataset statistical records

2.2. Execution of the data preparation component

The DP component focuses on preprocessing tasks for traffic data from the BoT-IoT and ToN-IoT datasets. Forming a table of defined features from the preprocessed traffic data suitable for input into the ML models of the LP component. The module's execution stages encompass the following sequential operations:

2.2.1. Data hosting procedure

The data hosting procedure (DHP) involves maintaining the data on a reliable and accessible surface, ensuring persistence and high reliability. In this study, we employ the KNIME analytics platform as a hosting, training, and evaluating system for both the dataset and the proposed framework. This phase is in charge of receiving the CSV files containing the acquired data entries and importing them into KNIME tables for subsequent preprocessing procedures. Through this hosting procedure, each entry of the IoT traffic is arranged in its raw form in a table, with features displayed as individual columns.

2.2.2. Data cleansing procedure

The data cleansing procedure (DCP) involves examining datasets to gain a more in-depth comprehension and correct misinterpreted data. DCP focuses on identifying and rectifying errors and inconsistencies in the data to enhance its quality. In this study, we performed various DC processes on the imported data, including checks for missing values (locating null-value cells and filling them with the estimated median value using the missing value node). The choice of median is motivated by its resistance to outlier influences compared to mean imputation. We also conducted identification and removal of corrupted values using the missing value column filter node, correction of feature labels (CSV file data often lacks attribute names), preservation of an elementary depiction of the data (ensuring that each attribute is clear and has one specific value for every cell), and checking for duplicate data (ensuring data points are distinct, without duplication of particular records) with the duplicate row filter node. Additionally, label encoding was performed since our datasets include several categorical features requiring conversion into numerical values. One-hot encoding was employed for this purpose using OneHotEncoder (one to many). Features like

Timestamp, ports, and IP addresses were excluded to prevent overfitting to certain machine learning methods during the training stage.

2.2.3. Feature selection procedure

In feature selection, our objective is to choose features that exhibit a high dependence on the target variable. The feature selection procedure (FSP) involves choosing any feature that boosts the ML algorithm's performance while excluding other attributes that could possibly affect the classifier's effectiveness. In this research, we employed the correlation coefficient score (CCS) method and RF for feature importance analysis. The CCS was chosen for its ability to identify linear relationships between features and the target variable [19], essential for reducing dimensionality without sacrificing predictive power. RFs feature importance analysis provided a non-linear perspective, offering insights into complex interactions between features.

2.2.4. Data standardization procedure

The min-max scaler was applied to normalize the data through the normalizer node within a predetermined range (0-1), ensuring all features contribute equally to the model's performance. This step is crucial for models sensitive to feature scale, such as logistic regression. in this approach, the min value of the whole dataset (in each column) is deducted from every single value, and the result is divided by the difference between the min and max values. The formula for the min-max scaling is defined as:

$$y = \frac{(\mathrm{x{-}min})}{(\mathrm{max} - \mathrm{min})} \tag{1}$$

2.2.5. Data balancing procedure

Given the class imbalance evident in our IoT datasets, as depicted in Figures 2 and 3, we employed a hybrid approach that utilizes the SMOTE technique [20], [21]. This involved oversampling minority classes and selectively eliminating specific samples from the majority class, thereby enhancing the model's sensitivity to less frequent attack types. We chose this method for its capability to generate synthetic samples, enriching the dataset without causing overfitting.

2.2.6. Data distribution procedure

This particular phase is crucial in ML projects, involving the division of the dataset into training and testing sets. This study implemented a random split strategy with an 80:20 ratio for training and testing, following common literature recommendations. Additionally, a 5-fold cross-validation process was incorporated to ensure optimal validation and testing. This method involves dividing the dataset into five folds, with each fold used once for testing and validation while the rest form the training set. Overall evaluation metrics are then calculated using the stored metrics from all 5-folds.

2.3. Execution of the learning process component

Every ML algorithm comes with specific constraints, such as maintaining a balance between low bias and slight variance. Ensemble learning emerges as a solution to overcome the limitations inherent in individual ML techniques. The idea of ensemble learning was first introduced by [22] in 1979 with the aim of enhancing the performance of standalone ML algorithms. In this paradigm, a diverse set of models, often referred to as weak learners, are combined to create a single optimized predictive model that outperforms a solitary model in terms of accuracy. The modular structure of ensemble learning is designed to mitigate high variance's overfitting problems. The selection of algorithms to be integrated into the ensemble learning technique should be based on considerations such as computational expense to attain improved overall performance.

In the process of ensemble learning, the initial step involves splitting the dataset into training and testing data. The training data is then further divided into multiple subsets using different techniques, such as with/without replacement. These subsets are assigned to chosen models for training in the subsequent stage. Following the training, the testing sets (unseen) are introduced as a source for these prediction-trained models. Ultimately, the predictions generated by all weak-models are aggregated using either the majority voting or an average technique. The mathematical equation for this process is represented as:

$$E_{n,m} = \frac{1}{N} \sum_{n=1}^{N} P^{n} \; n = 1,2,3 \ldots . N \tag{2}$$

$E_{n,m}$ signifies the result of the ensemble classifier comprising (N) ML models, while P denotes the prediction made by every individual model.

In this work, the suggested method assesses the efficiency of four ML models (DT, Rprop, logistic regression, and RF as a meta-model) to conduct more comprehensive investigations and gain deeper insights into the suggested approach. Moreover, ensemble learning models utilizing the four ML learners are designed to detect unusual network traffic and identify the behavioral characteristics of IoT network traffic emanating from compromised IoT nodes. Voting and stacking were chosen as methods for ensemble learning, given that their anticipations are weighted based on the significance of weak-models. The weighted probabilities are after that aggregated to obtain the overall probability.

2.4. Execution of the evaluation process component

The EP is a crucial task aimed at measuring and overseeing the evaluation metrics to assess the system's adherence to its objectives and specifications. To confirm the effectiveness of the proposed framework, we employed four generally accepted metrics for evaluation: accuracy, false positive rate (FPR), precision, recall, and F-measure. These metrics will be calculated among the confusion matrix, which provides pertinent formulations for each measure based on true positive rate (TPR), true negative rate (TNR), FPR, and false negative rate (FNR) results.

- Precision: the system's ability to correctly identify the existence of a security breach or an attack. It represents the association between correctly anticipated attacks and the actual events.

$$Precision = (TruePositives/(TruePositives + FalsePositives)) * 100\%$$

- Recall: the capability of the system to accurately identify an attack transpiring on a network.

$$Recall = (TruePositives/(TruePositives + FalseNegatives)) * 100\%$$

- Accuracy: the capability of the system to correctly distinguish between typical and malicious packets. It signifies the proportion of correct predictions in relation to the overall number of instances.

$$Accuracy = ((TrueNegatives + TruePositives)/(TrueNegatives + TruePositives + FalsePositives + FalseNegatives)) * 100\%$$

- F-measure: the average of precision and recall, indicating the proportion of attacking and normal flow instances that were correctly predicted in the testing set.

$$F - measure = (2 * (Precision * Recall)/(Recall + Precision)) * 100\%$$

2.5. Experimental setup

The experiments were conducted on a high-performance computing setup. This setup featured an 8th generation Intel Core i7 processor, which provided substantial processing power, and 32 GB of RAM to ensure smooth and efficient operation even with large datasets. Additionally, an NVIDIA Quadro M2000M GPU was used to accelerate the computational tasks, particularly during the training of machine learning models. The KNIME analytics platform was chosen as the central tool for model building and testing. KNIME was selected for its versatility and robust support for the machine learning algorithms utilized in this study, making it an ideal choice for the research.

3. RESULTS AND DISCUSSION

In this section, we present the outcomes derived from our experiments conducted on the ToN_IoT and BoT-IoT datasets, employing the proposed framework. To assess the effectiveness of our system, three commonly used machine learning models are employed for comparative analysis. Binary and multiclass classification are conducted on both datasets. In binary detection, all attack classes are consolidated into a unified collection, sharing identical label ID. Additionally, the outcomes obtained from experiments on our datasets undergo further validation through a comparison with recent studies. The outcomes are showcased in a tabulated format and assessed using the evaluation metrics outlined in the preceding section.

3.1. Binary classification

3.1.1. Binary classification of the BoT-IoT dataset

Table 3 showcases the results achieved by each single classifier in the binary classification on the BoT-IoT. The outcomes indicate that all models exhibit excellent performance in identifying anomalies, characterized by very high detection rates of 99.989–99.999% and low FPR. Notably, the proposed stacking framework demonstrates a notable improvement compared to other algorithms. The stacking framework outperforms all other classifiers in accuracy, recall, and f-measure, with DT and logistic regression being the only ones matching the stacking framework results in precision. Similarly, RProp and voting ensemble are the only ones matching the stacking framework results in recall.

Table 3. Results of binary classification on the BoT-IoT

Model	Accuracy	Precision	Recall	F-measure	FPR
DT	99.991%	99.999%	99.999%	99.999%	0.011
MLP RProp	99.989%	99.998%	100%	99.999%	0.018
Logistic regression	99.996%	99.999%	99.999%	99.999%	0.0
Voting	99.998%	99.998%	100%	99.999%	0.007
Stacking	99.999%	99.999%	100%	99.999%	0.0

3.1.2. Binary classification of the ToN-IoT dataset

Comparable patterns in the outcomes are observed for binary classification for this dataset too. The models exhibit a bit lower efficiency compared to the previous dataset, scoring an accuracy rate of 70.3%–97.313%. However, this is anticipated due to the higher diversity present in the ToN-IoT dataset. Notably, the proposed stacking framework consistently enhances performance across all metrics in comparison to the other models. In terms of accuracy, DT is the only model matching the results achieved by the stacking framework, as illustrated in Table 4.

Table 4. Results of binary classification on the ToN-IoT

Model	Accuracy	Precision	Recall	F-measure	FPR
DT	97.312%	95.817%	96.959%	96.385%	0.025
MLP RProp	70.307%	56.210%	88.8%	68.831%	0.405
Logistic regression	93.689%	88.650%	95.086%	91.755%	0.071
Voting	94.488%	88.960%	97.131%	92.866%	0.070
Stacking	97.313%	95.804%	96.988%	96.393%	0.024

3.2. Multiclass classification

3.2.1. Multiclass classification of the BoT_IoT dataset

The BoT_IoT dataset comprises five distinct classes, with four of them representing various types of threats, while the remaining one signifies regular traffic. Table 1 provides a description of these classes, indicating the types of attacks present in the data along with their number of records used in classification. The classification results obtained from our ensemble learning frameworks are depicted in Table 5.

The outcomes indicate that the suggested stacking ensemble framework excels in multiclass classification too, attaining exceptionally high scores across all evaluation metrics. Additionally, it is evident that the other algorithms also exhibit strong performance in this classification scenario. The stacking framework consistently achieves rates of 0.0 for both false positives (FP) and false negatives (FN), resulting in highly accurate label predictions.

Table 5. Results of multiclass classification on the BoT-IoT

Model	Accuracy	Precision	Recall	F-measure	FPR
DT	99.998%	99.996%	99.989%	99.992%	0.0
MLP RProp	99.997%	99.488%	91.311%	95.224%	0.0
Logistic regression	99.998%	99.983%	99.919%	99.951%	0.0
Voting	99.996%	99.999%	99.996%	99.997%	0.0
Stacking	99.999%	99.999%	99.999%	99.999%	0.0

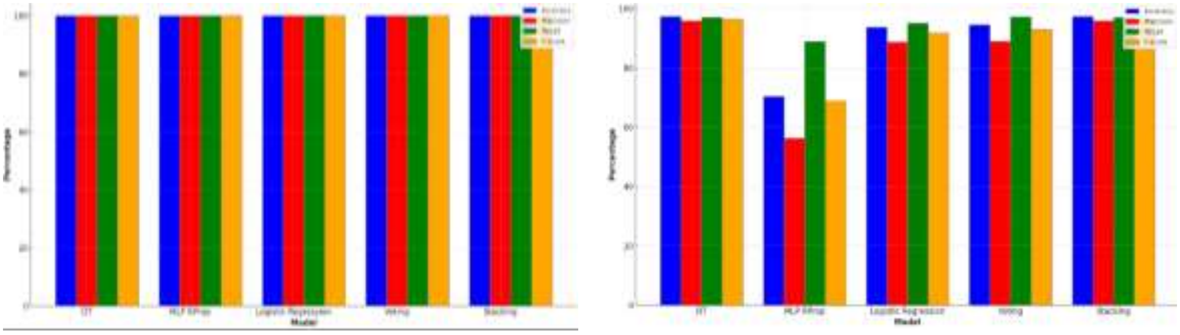
3.2.2. Multiclass classification of the ToN_IoT dataset

As depicted in Table 2, the ToN-IoT encompasses ten data classes, with nine representing various attacks and the remaining class denoting normal traffic. This introduces a more intricate scenario compared to the BoT-IoT dataset. Table 6 provides metrics from the multiclass classification over the ToN-IoT dataset. Due to the heightened complexity of the experiments, our proposed frameworks achieve scores lower than before. Nevertheless, they still exhibit excellent performance, surpassing all other models, particularly the voting framework.

Table 6. Results of multiclass classification on the ToN-IoT

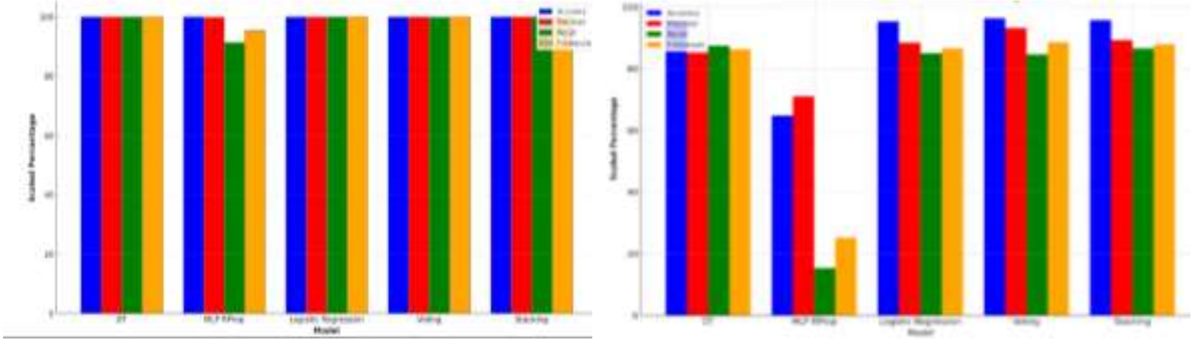
Model	Accuracy	Precision	recall	F-measure	FPR
DT	95.551%	85.263%	87.344%	86.290%	0.005
MLP RProp	64.714%	71.005%	15.314%	25.194%	0.04
Logistic regression	95.312%	88.410%	84.980%	86.661%	0.005
Voting	96.321%	93.119%	84.555%	88.631%	0.004
Stacking	95.757%	89.199%	86.515%	87.836%	0.004

Figures 4 and 5 illustrate the comparative analysis among the ensemble learning classifiers and the supervised ML algorithms. Specifically, Figures 4(a) and 4(b) provide detailed insights into the binary classification results, while Figures 5(a) and 5(b) focus on the multiclass classification outcomes. The analysis highlights that the suggested stacking classifier surpasses the ensemble voting model on both datasets for binary (99.999%, 97.313% respectively) and multiclass classification (99.999% on BoT-IoT dataset), with the exception being the multiclass classification on the ToN-IoT dataset (96.321% for voting compared to 95.757% for stacking).



(a) (b)

Figure 4. EL models performance for binary classification on (a) BoT-IoT datasets and (b) ToN-IoT datasets



(a) (b)

Figure 5. EL models performance for multiclass classification on (a) BoT-IoT datasets and (b) ToN-IoT datasets

3.3. Comparison with other studies

To conclude the assessment of the previously presented results, Table 7 provide a comparative examination of our ensemble framework with other contemporary cutting edge IoT-based attack detection techniques employing ML or deep learning methods within a similar research domain. The tables juxtapose our finest empirical findings obtained from the proposed framework with the corresponding metrics mentioned in previous researches. The comparative measures encompass the detection method, total of target classes for each detection method, and the percentage of validation accuracy. The outcomes derived from our datasets illustrate that our framework yields favorable results compared to other suggested approaches.

Table 7. Comparative analysis with recent studies

Research	Publication year	Detection method	Total of target classes	Validation accuracy
[23]	2019	SVM	5-Classes	81.00%
[24]	2019	Ensemble learning	5-Classes	85.21%
[25]	2020	DEEP-CNN	4-Classes	90.00%
[9]	2021	Ensemble learning	Binary	96.35%
[26]	2022	Ensemble learning	Binary	86.00%
[26]	2022	Ensemble learning	10-Classes	77.00%
[27]	2023	Ensemble learning	Binary	78.80%
[28]	2023	PCC-CNN	Binary	99.00%
				98.00%
				99.00%
[28]	2023	PCC-CNN	23-Classes	94.00%
			15-Classes	97.00%
			5-Classes	91.00%
[29]	2023	DNN	5-Classes	93.47%
[30]	2023	Ensemble learning	5-Classes	88.41%
			8-Classes	98.52%
			5-Classes	91.03%
Proposed framework on ToN-IoT	-	Ensemble learning	Binary	97.313%
Proposed framework on BoT-IoT	-	Ensemble learning	Binary	99.99%
Proposed framework on ToN-IoT	-	Ensemble learning	10-Classes	96.321%
Proposed framework on BoT-IoT	-	Ensemble learning	5-Classes	99.99%

The superior performance of the stacking ensemble model, especially evident in the BoT-IoT dataset results, supports our hypothesis that ensemble learning can significantly enhance intrusion detection in IoT networks. This technique, by combining the strengths of various learning algorithms, offers a promising approach to dealing with the complexity and evolving nature of cyber threats facing IoT environments. Compared to recent studies, such as those presented in Table 7, our framework demonstrates superior accuracy in both binary and multiclass classifications. This improvement can be attributed to our framework’s ability to integrate diverse detection methodologies, thereby capturing a broader spectrum of anomaly patterns. Unlike the predominantly single-model approaches explored in previous works.

While our results are promising, they are not without limitations. The dependence on extensive and diverse datasets for training and validation points to the need for ongoing data collection efforts to ensure the model’s relevance against new and emerging threats. Additionally, the disparity in model performance between the BoT-IoT and ToN-IoT datasets highlights the influence of dataset characteristics on detection capabilities. This variance is a critical consideration for deploying IDSs in real-world IoT networks, where the nature of threats and network configurations can greatly differ. These factors must be considered when extrapolating our findings to broader applications.

This study lays the groundwork for future research aimed at refining and enhancing the stacking framework’s capabilities. Exploring hybrid models and incorporating emerging machine learning techniques may further elevate its performance, particularly in environments with an evolving threat landscape. In summary, our exploration into advanced intrusion detection via ensemble learning techniques presents a significant stride toward enhancing IoT network defense. The remarkable performance of the Stacking model across various metrics and datasets not only demonstrates the potential of ensemble learning in this domain but also sets the stage for future research aimed at developing more resilient and adaptable IoT security solutions.

4. CONCLUSION

The escalating instances of anomalies and attacks within the IoT networks underscore the urgent need for more robust threat identification strategies. Our study has elucidated the pivotal role of ML, with a specific focus on ensemble methods, in advancing the detection capabilities of IDS for IoT ecosystems. Through comprehensive analysis and experimentation, our proposed IDS framework not only exhibits superior classification accuracy over existing models but also showcases the profound impact of ensemble techniques, such as stacking, on enhancing the efficiency of attack and intrusion detection. This breakthrough marks a significant enhancement in IoT security methodologies, setting a new benchmark for future endeavors in the domain. Looking forward, we aim to delve into the challenges of adversarial attacks and defenses in IoT security, a critical area given the evolving sophistication of threats. This future direction is not just about advancing our technical knowledge but also about providing actionable solutions to protect IoT networks against emerging adversarial tactics. The implications of our work extend to both the research domain and the practical realm of IoT security, offering a pathway to more resilient network environments. This balance between immediate applicability and future research potential highlights the broader significance of our study for enhancing the security infrastructure of IoT networks.

REFERENCES

[1] K. Albulayhi, A. A. Smadi, F. T. Sheldon, and R. K. Abercrombie, "IoT intrusion detection taxonomy, reference architecture, and analyses," *Sensors*, vol. 21, no. 19, p. 6432, Sep. 2021, doi: 10.3390/s21196432.

[2] S. Sinha, "State of IoT 2023: number of connected IoT devices growing 16% to 16.7 billion globally," *IOT Analytics*, 2023. https://iot-analytics.com/number-connected-iot-devices/ (accessed Nov. 06, 2023).

[3] K. Rose, S. Eldridge, and L. Chapin, "The internet of things: an overview. Understanding the issues and challenges of a more connected world.," *The Internet Society*, no. October, p. 80, 2015.

[4] A. Churcher *et al.*, "An experimental analysis of attack classification using machine learning in IoT networks," *Sensors*, vol. 21, no. 2, p. 446, Jan. 2021, doi: 10.3390/s21020446.

[5] A. S. Jaradat, M. M. Barhoush, and R. S. B. Easa, "Network intrusion detection system: machine learning approach," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 25, no. 2, pp. 1151–1158, Feb. 2022, doi: 10.11591/ijeecs.v25.i2.pp1151-1158.

[6] R. Qaddoura, A. M. Al-Zoubi, I. Almomani, and H. Faris, "A multi-stage classification approach for IoT intrusion detection based on clustering with oversampling," *Applied Sciences*, vol. 11, no. 7, p. 3022, Mar. 2021, doi: 10.3390/app11073022.

[7] R. Vishwakarma and A. K. Jain, "A honeypot with machine learning based detection framework for defending IoT based Botnet DDoS attacks," in *2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI)*, Apr. 2019, pp. 1019–1024, doi: 10.1109/ICOEI.2019.8862720.

[8] P. Pokharel, R. Pokhrel, and S. Sigdel, "Intrusion detection system based on hybrid classifier and user profile enhancement techniques," in *2020 International Workshop on Big Data and Information Security (IWBIS)*, Oct. 2020, pp. 137–144, doi: 10.1109/IWBIS50925.2020.9255578.

[9] P. Kumar, G. P. Gupta, and R. Tripathi, "An ensemble learning and fog-cloud architecture-driven cyber-attack detection framework for IoMT networks," *Computer Communications*, vol. 166, pp. 110–124, Jan. 2021, doi: 10.1016/j.comcom.2020.12.003.

[10] M. Sarhan, S. Layeghy, N. Moustafa, and M. Portmann, "NetFlow datasets for machine learning-based network intrusion detection systems," in *Big Data Technologies and Applications: 10th EAI International Conference, BDTA 2020, and 13th EAI International Conference on Wireless Internet, WiCON 2020*, 2021, pp. 117–135, doi: 10.1007/978-3-030-72802-1_9.

[11] E. Tsogbaatar *et al.*, "DeL-IoT: a deep ensemble learning approach to uncover anomalies in IoT," *Internet of Things*, vol. 14, p. 100391, Jun. 2021, doi: 10.1016/j.iot.2021.100391.

[12] D. Chakraborty, V. Narayanan, and A. Ghosh, "Integration of deep feature extraction and ensemble learning for outlier detection," *Pattern Recognition*, vol. 89, pp. 161–171, May 2019, doi: 10.1016/j.patcog.2019.01.002.

[13] N. An, H. Ding, J. Yang, R. Au, and T. F. A. Ang, "Deep ensemble learning for Alzheimer's disease classification," *Journal of Biomedical Informatics*, vol. 105, p. 103411, May 2020, doi: 10.1016/j.jbi.2020.103411.

[14] Q. A. Al-Haija and A. Al-Badawi, "Attack-aware IoT network traffic routing leveraging ensemble learning," *Sensors*, vol. 22, no. 1, p. 241, Dec. 2021, doi: 10.3390/s22010241.

[15] Q. A. Al-Haija, "Top-down machine learning-based architecture for cyberattacks identification and classification in IoT communication networks," *Frontiers in Big Data*, vol. 4, p. 782902, Jan. 2022, doi: 10.3389/fdata.2021.782902.

[16] N. Koroniotis, N. Moustafa, E. Sitnikova, and B. Turnbull, "Towards the Development of realistic botnet dataset in the internet of things for network forensic analytics: Bot-IoT dataset," *arXiv*, Nov. 2018, doi: 10.48550/arXiv.1811.00701.

[17] N. Koroniotis, N. Moustafa, E. Sitnikova, and J. Slay, "Towards developing network forensic mechanism for botnet activities in the IoT based on machine learning techniques," in *Mobile Networks and Management: 9th International Conference, MONAMI 2017*, 2018, pp. 30–44, doi: 10.1007/978-3-319-90775-8_3.

[18] N. Moustafa, "A new distributed architecture for evaluating AI-based security systems at the edge: network TON_IoT datasets," *Sustainable Cities and Society*, vol. 72, p. 102994, Sep. 2021, doi: 10.1016/j.scs.2021.102994.

[19] E. C. Blessie and E. Karthikeyan, "Sigmis: a feature selection algorithm using correlation based method," *Journal of Algorithms & Computational Technology*, vol. 6, no. 3, pp. 385–394, Sep. 2012, doi: 10.1260/1748-3018.6.3.385.

[20] J. Wang, M. Xu, H. Wang, and J. Zhang, "Classification of imbalanced data by using the SMOTE algorithm and locally linear embedding," 2006, doi: 10.1109/ICOSP.2006.345752.

[21] S. Bagui and K. Li, "Resampling imbalanced data for network intrusion detection datasets," *Journal of Big Data*, vol. 8, no. 1, p. 6, Dec. 2021, doi: 10.1186/s40537-020-00390-x.

[22] B. V. Dasarathy and B. V. Sheela, "A composite classifier system design: concepts and methodology," *Proceedings of the IEEE*, vol. 67, no. 5, pp. 708–713, 1979, doi: 10.1109/PROC.1979.11321.

[23] C. Ioannou and V. Vassiliou, "Classifying security attacks in iot networks using supervised learning," in *2019 15th International Conference on Distributed Computing in Sensor Systems (DCOSS)*, May 2019, pp. 652–658, doi: 10.1109/DCOSS.2019.00118.

[24] X. Gao, C. Shan, C. Hu, Z. Niu, and Z. Liu, "An adaptive ensemble machine learning model for intrusion detection," *IEEE Access*, vol. 7, pp. 82512–82521, 2019, doi: 10.1109/ACCESS.2019.2923640.

[25] W. Jung, H. Zhao, M. Sun, and G. Zhou, "IoT botnet detection via power consumption modeling," *Smart Health*, vol. 15, p. 100103, Mar. 2020, doi: 10.1016/j.smhl.2019.100103.

[26] N. Naz *et al.*, "Ensemble learning-based IDS for sensors telemetry data in IoT networks," *Mathematical Biosciences and Engineering*, vol. 19, no. 10, pp. 10550–10580, 2022, doi: 10.3934/mbe.2022493.

[27] R. N. Bin Rais, O. Khalid, J. Nazar, and M. U. S. Khan, "Analysis of intrusion detection using ensemble stacking-based machine learning techniques in IoT networks," in *International Conference on Advances in Computing Research*, 2023, pp. 329–344, doi: 10.1007/978-3-031-33743-7_27.

[28] M. Bhavsar, K. Roy, J. Kelly, and O. Olusola, "Anomaly-based intrusion detection system for IoT application," *Discover Internet of Things*, vol. 3, no. 1, 2023, doi: 10.1007/s43926-023-00034-5.

[29] A. Awajan, "A novel deep learning-based intrusion detection system for IoT networks," *Computers*, vol. 12, no. 2, p. 34, Feb. 2023, doi: 10.3390/computers12020034.

[30] B. A. K. Hammood and A. T. Sadiq, "Ensemble machine learning approach for iot intrusion detection systems," *Iraqi Journal for Computers and Informatics*, vol. 49, no. 2, pp. 93–99, Dec. 2023, doi: 10.25195/ijci.v49i2.458.

BIOGRAPHIES OF AUTHORS

Salah El Hajla is currently a Ph.D. student at the Laboratory of Sciences and Techniques for Engineering at Sultan Moulay Slimane University, Morocco. With a Master's degree in Distributed Computing Systems and Big Data from the Faculty of Science ibn Zohr, Agadir (2019). His research focuses on developing innovative solutions for IoT network security and applying advanced machine learning techniques to enhance cyber defenses. He can be contacted at email: salah.elhajla@usms.ac.ma.

El Mahfoud Ennaji Ph.D. student at the University of Sultan Moulay Slimane Beni Mellal, Laboratory of Science and Engineering Technologies at ENSA Khouribga. His research interests are cybersecurity, IoT, and machine learning. He can be contacted at email: elmahfoud.ennaji@usms.ac.ma.

Prof. Dr. Yassine Maleh is an associate professor of cybersecurity and IT governance at Sultan Moulay Slimane University, Morocco, since 2019. He is a double Ph.D. in computer sciences and IT Management. He is the founding chair of IEEE Consultant Network Morocco and founding president of the African Research Center of Information Technology and Cybersecurity. He is a senior member of IEEE He has published over than 140 papers (international journals, book chapters and conferences/workshops), 27 edited books, and 5 authored books. He is the editor-in-chief of the International Journal of Information Security and Privacy. He can be contacted at email: y.maleh@usms.ma.

Prof. Dr. Soufyane Mounir is an associate professor at the National School of Applied Sciences of Sultan Moulay Slimane University, Beni Mellal, Morocco, since 2014. He got his Ph.D. in Electronics and Telecommunication, from University Hassan 1st, Morocco. His research is multidisciplinary that focuses on telecommunications, VoIP, signal processing, embedded systems and cyber security. He is an active member of LaSTI Laboratory, ENSA Khouribga. He can be contacted at email: s.mounir@usms.ma.