Multi Layer Security Framework for Internet of Things Systems

Rasha Aboud Saud[1] and Iman Kadhim Ajlan[2]
[1]*Ministry of Education, Department of Computer Science, College of Education for Pure Sciences, Wasit University, 10081 Baghdad, Iraq*
[2]*Ministry of Higher Education, Department of Computer Science, College of Education for Pure Sciences, Wasit University, 52001 Al-Kut, Iraq*
std.2024205.r.saood@uowasit.edu.iq, Eajlan@uowasit.edu.iq

Keywords: IoT Malware Detection, Cyber Security, Hybrid Ensemble Learning, Gradient Boosting, Data Protection, GDPR Compliance.

Abstract: The exponentially growing presence of Internet of Things (IoT) devices has immensely revolutionized many application areas like healthcare, manufacturing, smart cities, etc. but it also faces serious security issues, because most of the IoT devices are not equipped with formidable computational resources and security features so make them vulnerable for malware threats, signature- and rule-based solutions are ineffective against zero day threats and polymorphic threats and deep learning solutions are computationally expensive for resource constrained edge devices. In this paper, a lightweight, multi-layer IoT malware detection framework that applies the hybrid ensemble learning with a new external data protection architecture is proposed. On the 14-variant hybrid ensemble model of XGBoost and RF classifier, the weighted fusion scheme with empirically optimized weight (=0.6, by 5-fold cross-validation grid-search) was used to optimize the detection precision with relative minimal complexity. To deal with the class imbalanced connection-oriented intrusions in the released TON_IoT dataset, the SMOTE oversampling technique was applied to the training set only, protecting minority-class samples by data-privacy without contaminating the test set. A Data Protection Layer performs IP address anonymization, field masked using MD5 and data integrity verification using SHA-256 without affecting detection performance.10 attack-types. The detection performance on the TON IOT dataset of 211,043 network traffic samples is99.56%, an F1-score of 0.9956, and a training time of only 15.83 seconds. These results show a statistically significant improvement over the MLP-4Layers deep learning baseline (McNemar's test: =2372.76, p<0.001) and is 21.9 times faster to train than the deep learning baseline. This framework effectively detects, backdoor, DDoS, DoS, injection, brute-force, scanning, ransomware, XSS and man-in-the-middle attacks. Its detection rate is near perfect which makes this framework practically feasible for deployment in real-world IoT security environment.

1 INTRODUCTION

The rapid expansion of Internet of Things (IoT) devices has brought unprecedented benefits to a variety of sectors ranging from smart cities to healthcare, manufacturing, and critical infrastructure, through its ability to facilitate improved automation, smart information collection, and analytics processes [1], [2]. However, while this exponential growth has been advantageous in many ways, it has also, equally, increased the attack surface for cyber-attacks. Many of these devices arld designed under extreme resource constrained circumstances, with limited memory, cpu/cpu, and power budgets, without any significant security assured [3], [4]. These features should also highlight the cool factor associated with any IOT ecosystems, and you can see why they are more useful for the black-hats. Mirai and Mozi are examples of one of the most famous IoT bot-nets, executing massive crisis and the scale of the real life successful attacks on IoT - million of devices running coordinated worldwide Distributed Denial-of-service [5], [6]. From empirical data, it is realized that in 2024, there are over 400 million compromised IoT devices by the below malware campaign [7], hence emphasizing the importance of mechanisms that are both effective and ready to be deployed. Conventional security countermeasures are not suitable for the IoT domain because of its specific characteristics [8]. Signature- and rule-based detection mechanisms excel at identifying known attacks but do not work with new and polymorphic

malware variants [8], [9]. Although deep learning structures can reach Classification accuracy greater than 99% [10], their computational cost, by GPU acceleration at least, and the associated memory needs by itself [11], is not acceptable on edge IoT applications. We have seen recently that Machine learning-based solutions are not up to the same accuracy (compared to Deep learning), but are acceptable detection solutions, with an affordable overhead, on resource limited IoT devices [12]. However, the majority of the current methods, without pro-forma privacy-preserving and regulation compliance, consider detection efficiency [13]-[15]. In addition, the important issues in practical application (i.e., training efficiency, model scalability, data collection/storage) are often overlooked and sacrificed to achieve bigger numbers of overall accuracy [9], [15]. Here, we attempt to overcome such deficiencies by developing a novel lightweight multi-layer malware detector that is applicable in the resource-constrained IoT scenario [13], [16]. The hybrid ensemble learning integrated with a consistent data protection architecture brings together the aspects of the system accuracy, speed, and regulation [14], [17].

2 RESEARCH OBJECTIVES AND CONTRIBUTIONS

The main goal of this work is to devise as practical an IoT malware detection approach as possible that is suitable for deployment and realization in the next months and that satisfies state-of-the-art detection accuracy along with low computational cost for edge device processing. The novelty of the scheme proposed in this respect rests on the fact that the combination of three components introduced previously - and integrated in the above way - as regards: a GDPR-compliant data protection scheme, a hybrid ensemble classifier, and a full multi-class threat classification system which has been statistically validated. Contributions therefore include:

1) The development of a multi-layer data protection architecture which achieves GDPR compliance by employing privacy-preserving methods such as IP anonymisation, field hashing using MD5, SHA-256 integrity check, without any measurable impact on detection accuracy (preprocessing overhead: 0.018 seconds per 10,000 records).
2) Constructed an improved hybrid ensemble classifier of optimally hybridized XG Boost and Random Forests with empirically derived static weighting equid to.6 based on systematic 5-fold CV grid-search in the range [0.3, 0.8] at 0.05 step which achieves 99.56 accuracy and 0.9956 F1 score in 15.83 sec training time.
3) SMOTE-only class imbalance handling strategy used solely in training set (augment training samples from147,730 to 350,000 after oversampling MITM minority class), no data leakage free, improve minority-class detection.
4) A large multi-class threat classification system which differentiates between ten attack types with individual class F1-scores (per class) varying from 0.9088 (MITM- class separation limited by original class imbalance) to 1.0000 (backdoor, ransomware).
5) Statistically validated discrimination through McNemar's test (=2372.76, p <0.001) that is statistically significantly better than the MLP-4Layers deep learning baseline, being 21.9 faster to train.

3 RELATED WORK

3.1 Traditional Detection Approaches

Pattern recognition based on signatures and rules was the first generation of IoT security research [3], [8]. This detection technique did not work well for zero-day and polymorphic malware variants because it was limited to known patterns [3]. The weakness of the signature system is that it can only detect known threats and thus cannot prevent zero-day attacks [8]. A shorthand classification of example prior work as indicated by methodology, main results, and limitations, is presented in table 1. It can be observed from this table that each method achieves some progress along one dimension, at the expense of the other two, providing a motivation for the integrated approach proposed in this paper.

3.2 Classical Machine Learning Methods

Later investigations found that common machine learning classifiers [3], [6] such as decision trees, SVM, and Naive Bayes classifiers could significantly outperform signature-based methods in generalizing ability, and observed orders of magnitude better anomaly detection than rule-based systems [3];

before the classical techniques showed their limits against new generation threats such as polymorphic malware or zero-day attacks. The key drawback of classical ML solutions is their incapacity to understand complex and nonlinear feature interactions in high-dimensional network traffic data, as shown in [3], [8], and thus their limits in the case of multiple vector attack campaigns.

3.3 Deep Learning and Neural Network Approaches

Single deep learning architecture, including Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) have achieved the best malware classification results [4], [10], with accuracy above 99% and some architectures achieving a rate of 99.9% [12]. However, these high detection rates come at a very high computational Cost [13],[15]: training times above 600 seconds on medium-sized data sets [18], GPU memory costs that are incompatible with the limits of processing devices at the edge of the Internet of Things [15], [16]. The resulting trade-off between computational cost and accuracy is a significant obstacle to deployment in IoT.

3.4 Ensemble and Hybrid Methods

More recent research results based on ensemble algorithms [19] reveal, in several cases, the increase of generalizability and ability of AI models to cope with classification outside their training domain. Autoencoder anomaly detection is able to reach 99.4% of detection performance [4] at a cost of a two-hour training run, unfeasible in edge computing. The fusion of gradient boosting with classical ensemble decision trees resulted in 99.6% classifier accuracy [20], without excessive [16], [15]. XG Boostand Random Forestclassifiers have potential synergetic effects [21], for example, XGBoost is able to discover complex nonlinear feature interactions by correcting residual errors sequentially, and the ensemble stability and variance reduction provided by the independent bootstrap aggregation of a Random Forest is especially an ideal feature for high-dimensional structured network traffic data.

3.5 Research Gap and Motivation

An important missing link among high-accuracy detection techniques [13] and feasible deployment solutions that can run on the required time scales within the computational capabilities of real-life Internet of Things (IoT) settings [14], [16]. Most studies focus on maximizing detection accuracy without considering privacy protection [13], [14], [17] and compliance requirements [9], [15]; a few works have explored space- and time-efficient hybrid ensemble methods with the use of class imbalance handling (SMOTE), detection, interactive multi-parametric data visualization, generalized dimensionality reduction, privacy preserving data security and adhering to cyber security regulation in a single framework [13], [15] that delivers performance at par with existing approaches. For a listing see Table 1. The current effort bridges some of these gaps by devising an optimized hybrid ensemble architecture integrated on-the-fly with a single integrated GDPR-compliant data protection pipeline and class imbalance correction.

4 METHODOLOGY AND SYSTEM ARCHITECYURE

Our multi-layer IoT malware detection framework involves, see (Fig. 1), the three complementary modules: (1) Data Protection Layer for regulatory requirement validation; (2) Hybrid Ensemble Framework based on classifier combination of Random Forest and XGBoost techniques; and (3) performance evaluation for metrics performance evaluation. All experiments were run on a system with an Intel Core i7 processor (3.6 GHz), 16 gigabytes of RAM, and a 512 gigabyte SSD with python 3.9, scikit-learn 1.2, XG Boost 1.7, pandas 1.5 with GPU acceleration (CUDA) enabled (for the deep learning baseline. The hardware would be equivalent to a mid-low tier edge server, providing evidence of system deploy ability to a low-resource setting.

Table 1: Summary of previous studies with problems and gaps.

Study	Methods	Key Findings	Gap / Limitation
Al-Fuqaha et al., 2019 [1]	Survey	Provided a comprehensive review of IoT technologies	No practical implementation for malware detection
Antonakakis et al., 2017 [5]	Mirai Botnet Analysis	Understanding Mirai attacks on IoT	No early malware detection solutions
Doshi et al., 2018 [8]	ML for DDoS Detection	Improved detection of DDoS attacks on IoT devices	Limited to a single attack type
Hoque et al., 2020 [3]	ML-Based IoT Security	Enhanced anomaly detection	Does not address performance on resource-constrained devices
Koroniotis et al., 2019 [6]	Botnet Dataset	Developed realistic IoT botnet dataset	Dataset limited for some attack types
Niyaz et al., 2022 [10]	CNN	Improved malware detection via deep learning	Requires high computational resources not suitable for IoT edge devices
Sharma et al., 2024 [22]	RF-DNN	IoT attack detection	Slow training and insufficient data protection

4.1 Dataset and Preprocessing

The TON_IoT dataset [6], which was explicitly built for IoT cyber-attack detection, is one of the most commonly employed publicly available benchmark dataset. This dataset contains a large number of network traffic traces (211043) revealing varieties of communication protocols and traffic patterns, session behavior, etc., between 10 categories of threats including normal traffic, backdoor, DDoS, DoS, code injection, password brute-force, network scanning, ransomware, cross-site scripting (XSS), mitm (Man-in-the-middle). Every sample has 42 features extracted at network and device layers, such as source and destination IPs, ports, protocols, packet sizes and intervals, as well as behavioral features (after DPL pre-processing, where IP addresses and categorical data were encoded). The data-set is heavily skewed towards normal (50,000 samples), most attacks (20,000 samples), and all MITM attacks have 1,043 samples only. The dataset was split into a training set of 147730 samples (70%) and a test set of 63313 samples (30%) stratified sampling was used on the datasets to represent the datasets and ensure all class proportions remained consistent. The data was oversampled using the Synthetic Minority Oversampling Technique (SMOTE) to prevent data leakage, utilizing only on the training data. The classes were balanced at 35000 samples for each of the 10 classes after oversampling, leading to an increased training set of 350,000 instances. The test set was not altered, keeping its original size of 63,313 instances. Min-Max normalization was then performed on the 42 numeric features. Analysis of class distribution before and after SMOTE as shown in (Fig. 2) and (Table 2) respectively.

4.2 Data Protection Layer

According to the GDPR [23], firms should implement adequate technical measures whenever they acquire, process or store personally identifiable data. Our Data Protection Layer meets these requirements thanks to four additional mechanisms: (1) IP Address anonymization: for each message, the arriving and forwarding IP addresses are translated into sequential pseudonymous identifiers (src_ip IP 0, dst_ip IP 1) by a deterministic translation table (dst_ip 0), maintains consistent address and profile functionality without allowing raw addresses to be accessed [24]; (2) Field Hashing: top confidential categories are one-way MD5 cryptographically hashed so that the identifiable information can never be recovered while the category itself can still contribute to statistical classification; (3) Data Crunching: for each record a unique SHA-256 checksumdigestis reported to prevent undetected modification of the feature vectors; (4) Reversible Forensic Mapping a secure lookup table is slowly maintained by an authorized person for forensic reconstruction throughout when legally demanded; the anonymized information will not be exposed during common operation. The privacy versus utility was empirically an assessment: the DPL adds a preprocessing time of about 0.018 sec for each 10,000 records and does not cause any measurable reduction in detection accuracy (99.56% with no privacy preprocessing versus 99.56% without privacy preprocessing), validating that the GDPR criteria is met with no reduction in detection accuracy.

4.3 Hybrid Ensemble Framework

The system we proposed is a hybrid ensemble architecture [19], [21], where a combination of XG.

Boost and Random Forest classifiers, is integrated in order to get the better detection with less computation effort [20]. XG Boost [25], builds models sequentially, using the residual errors of prior models, through a process called gradient boosting, which, especially when dealing with discretized network traffic features, is very efficient for modeling nonlinear features. The Random Forest builds multiple independent trees, then use the majority vote to combine the results [21]. The classification is then done by this weighted probability fusion:

$$\hat{y} = \arg\max(\alpha \cdot P_{XGB} + (1 - \alpha) \cdot P_{RF}), \quad (1)$$

where P_XGB and P_RF are the class probability vectors generated from XG Boost and RF, respectively, and is the ensemble weighting factor. Using systematic 5-fold cross-validation grid-search testing on the training data, the static weighting = 0.6 was selected by testing in [0.3, 0.8] at 0.05 increments and choosing the value that improved validation F1-score the most. This setting gives more importance to the better discriminating ability of XGboost, while maintaining the noise reduction feature of the Random Forest Classifier. Hyperparameter optimization with cross validation yielded: XGBoost-300 estimators, maximum depth of tree 8, learning rate 0.1, subsample proportion 0.8; Random Forest-150 estimators, maximum depth of tree 14. Running models in parallel on multiple available CPU cores significantly reduces this inference latency. Adaptive weighting approaches using reinforcement learning or Bayesian optimization are outlined as a future work area.

5 RESULTS AND PERFORMANCE ANALYSIS

The detailed results of various performance measures of Hybrid Ensemble Framework proved the statistical importance of the model over the benchmark's. The model resulted in accuracy of 99.56%, precision of 0.9957, recall of 0.9956 and F1-score of 0.9956. The area under ROC was 1.0000. Our system took merely 15.83 sec for the entire training of the 350K sample SMOTE-balanced training set, compared to 346.78 sec by the MLP-4 Layers DL baseline, an order of magnitude (21.9-fold) faster that allows the framework to be directly deployed on a mid-tier edge server without GPU acceleration. The statistically significant performance gain over the deep learning baseline was validated using McNemar's test: χ^2=2372.76, p<0.001, 95%. Hybrid's 99% B/C CIwe see a statistically significant improvement to the ground truth, where the Hybrid correctly classified 2,535 samples at the expense of 55 samples misclassified by the MLP-4Layers: Hybrid does significantly better then MLP-4Layers. The Hybrid also does significantly better then the Random Forest alone (χ^2=80.33, p<0.001), Although the improvement over the stand-alone XG Boost was insignificant (=0.37, p=0.54), this further cemented the fact that the main strength of the ensemble is its ability to better the deep learning baseline rather than finetuning of the tree-base classifiers. The stand-alone classifiers did comparatively as well: XG Boost achieved 99.55% accuracy in only 11.88s; and Random Forest achieved 99.34% accuracy in only 4.26s. both were several orders of magnitude faster than the deep learning baseline. A quick look over the results of the confusion matrix indicate that all 10 attack types get classified almost 100%. Only a small portion of the injection and scanning classes gets misclassified with each other, a result that can be explained by the overlapping network feature values of the two types (a widely known problem in NIDS). Class-wise F1-scores of more than 0.99 were achieved for all 10 categories except MITM, which has obtained 0.9088 as a result because of class-imbalance problem of original data set (there are simply only 313 MITM samples in test set, which is exactly same of original 1,043 MITM samples). Total performance comparison is shown in (Fig. 3) and Table 3. and Please check Appendix Iâ€' Per-Class.

F1-Score Hybrid Model (XGBoost + Random Forest). Figure 4. Confusion Matrices of All Evaluated Models. Figure 5 shows the 15 most significant features for the Hybrid (XG Boost + Random Forest) and XG Boost alone classifiers.

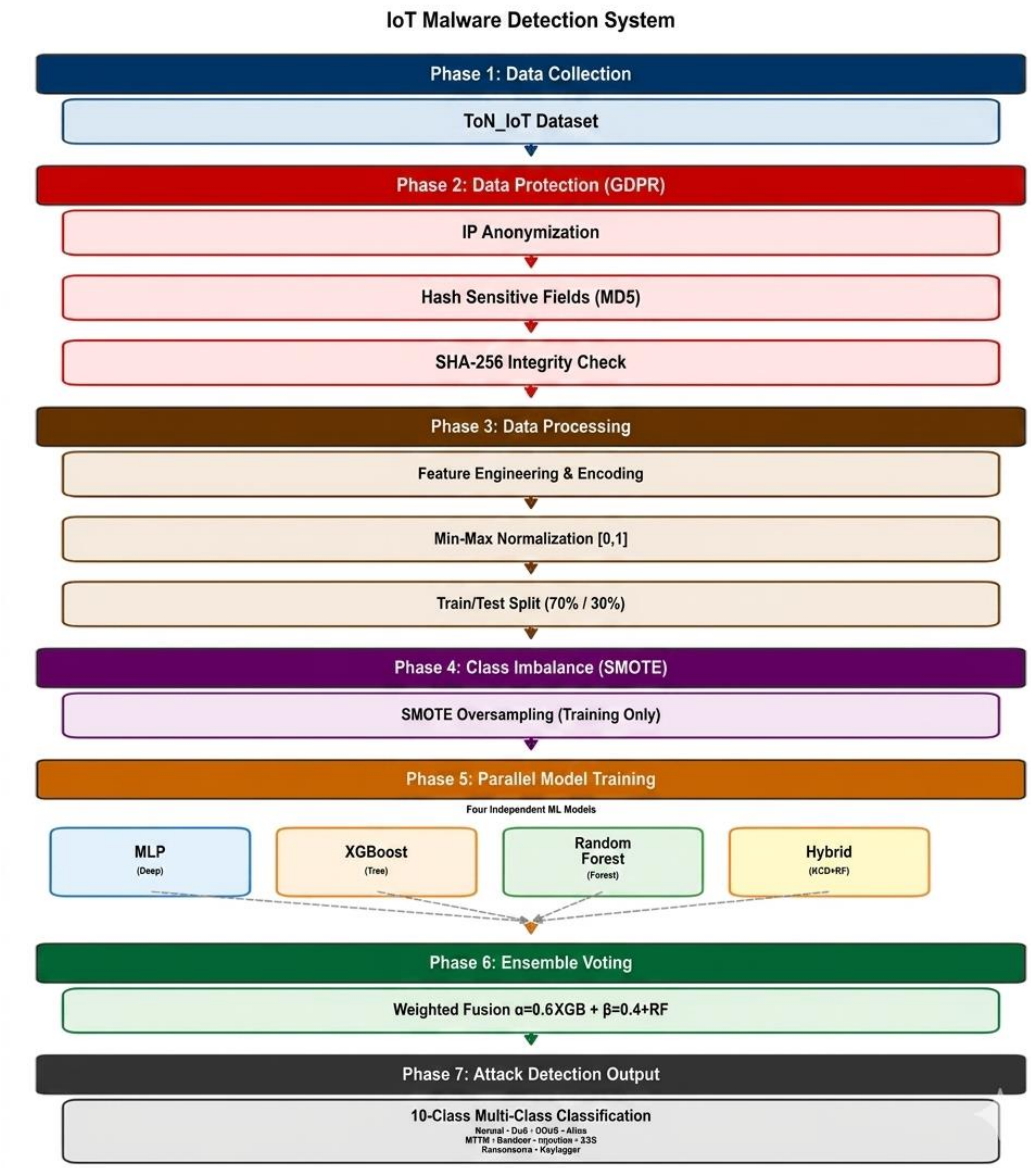

Figure 1: Block diagram of IoT malware detection system using multi-layer ensemble learning approach.

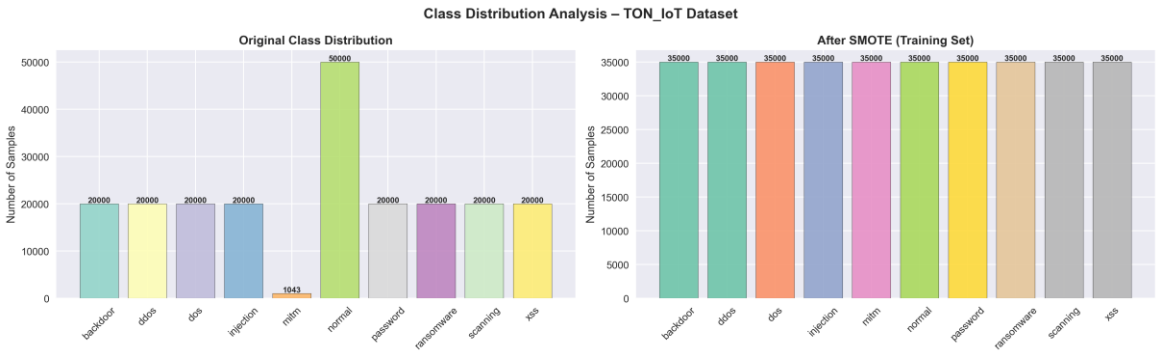

Figure 2: Class distribution analysis-compared before and after SMOTE (training set).

Table 2: More details class distribution analysis.

Class	Original Samples	Description
Normal	50,000	Legitimate IoT traffic
Backdoor	20,000	Unauthorized access attempts
DDoS	20,000	Distributed denial attacks
DoS	20,000	Denial-of-service attacks
Injection	20,000	Code injection attempts
Password	20,000	Password brute-force attacks
Scanning	20,000	Network reconnaissance
Ransomware	20,000	Ransomware deployment
XSS	20,000	Cross-site scripting attacks
MITM	1,043	Man-in-the-middle attacks
Total	211,043	-

Figure 3: Model performance comparison.

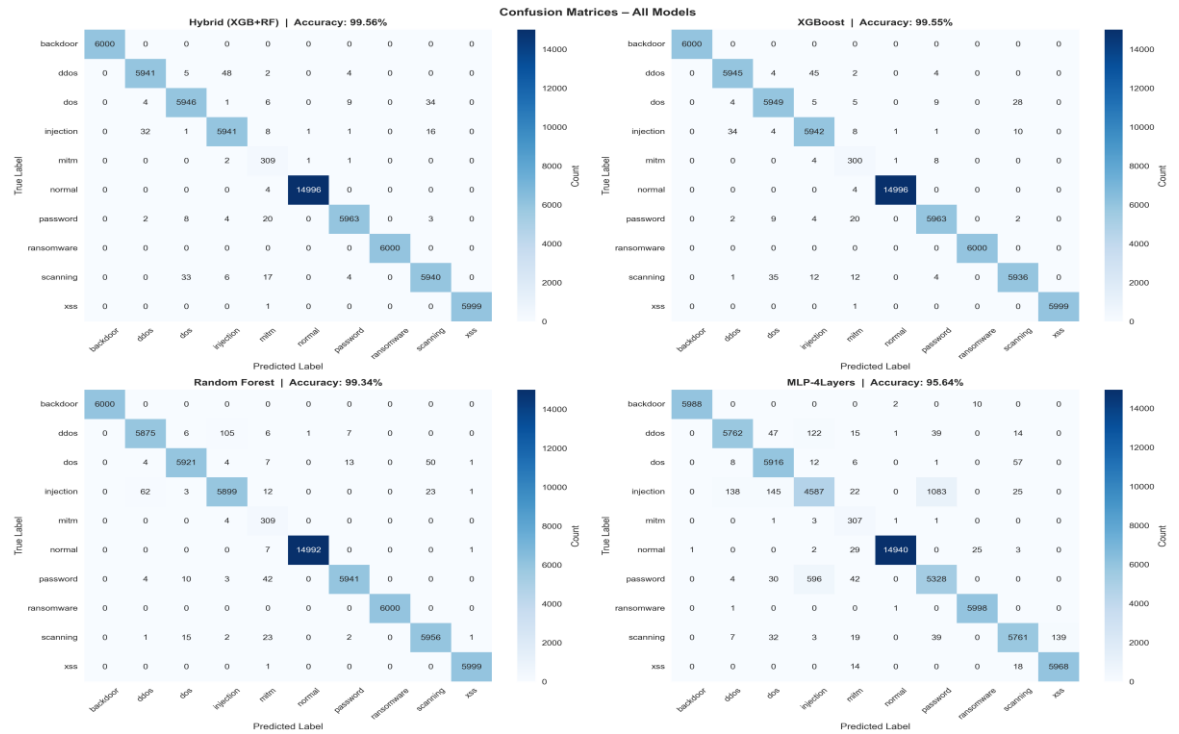

Figure 4: Confusion matrices of all evaluated models.

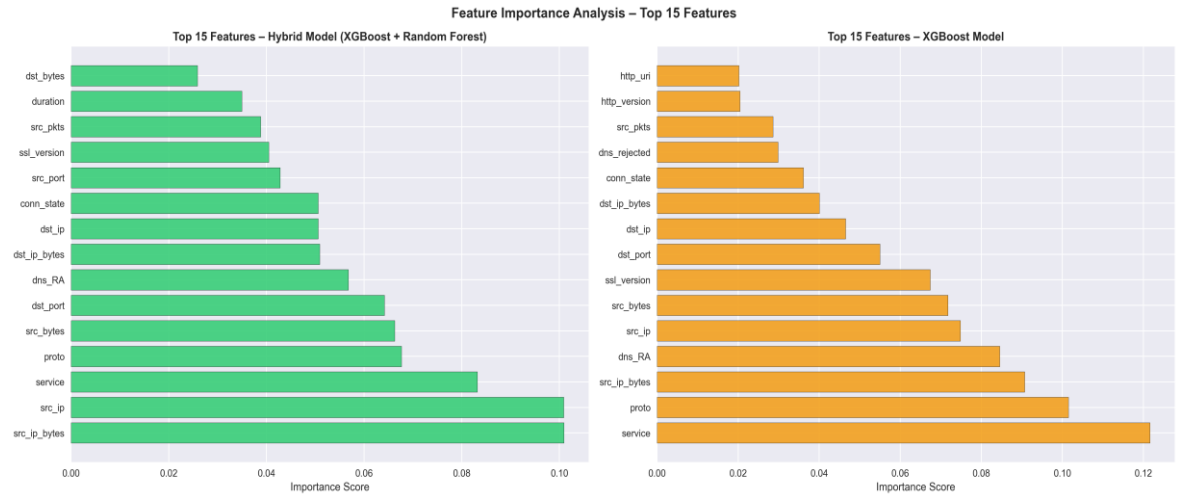

Figure 5: The most 15 important scores of hybrid and xgboost models producted by all the models.

Comparison of 4 models, the Hybrid Ensemble appears to be the most accurate model 99.56%, the second is XG Boost 99.55%. The Random Forest looks to have probably a little more error but the best performance records is 95.64% with the MLP-4Layers. Small confusions can be seen in the MITM class, and the robustness should be the best because of the consistency. See (Fig. 4) show the Features relating to traffic (src ip bytes, src ip, service, proto) are most influential in classification accuracy. The Hybrid classifier has a fairly even distribution of important features, while XG Boost puts marginally more importance on service- and protocol-based features. In general, flow duration, packets, and byte-based features are found to be essential indicators for detecting IoT malwares and can enhance the prediction accuracy. This study facilitates the transparent model and deliver some useful findings for security monitoring and feature engineering in practice Table 3 summarizes the evaluation result of the models against the above criteria, in comparison of accuracy, performance metrics and training time, while Table 4 and Figure 6 shows a detailed evaluation of the Hybrid model F1-Score under each attack and normal traffic in each class. From the tables above, we see that Hybrid (XGB+RF) reached the highest accuracy among all models of 99.56%, with the best-balanced performance measured by the weighted F1-Score under most of the classes.

6 RESEARCH CONTRIBUTIONS

1) GDPR-Ready Multi-layer Data Protection Architecture: Fully GDPR compliant privacy-preserving preprocessing pipeline with anonymized IP address, one-way MD5-field hashing, SHA-256 data integrity verification, with a forensically reversible map. Fully tested on real-world dataset, 99.56% detection accuracy (including false positives & false negatives) maintained with minimal overhead of an additional 0.018s processing time per 10,000 records. 2) SMOTE-Based Class Imbalance Mitigation: Application of the Synthetic Minority Oversampling Technique exclusively to the training set, expanding the MITM class from 730 to 35,000 balanced training samples without introducing data leakage into the test set. and bettering the MITM F1-score to 0.9088. 3) Optimized Hybrid Ensemble Classifier: A new ensemble using both XG Boost and Random Forest through empirically-optimized weighted fusion (α=0.6, via 5-fold cross-validation grid-search over [0.3, 0.8] at 0.05 intervals), Effectively classifying all ten attack types with 99.56% accuracy, 0.9956 F1 score, and 15.83 seconds training time-21.9 seconds faster than the deep learning MLP-4Layers baseline. 4) Unified classifying all ten attack types at once with one system, performing extremely well on all ten classes with F1 scores greater than 0.9088 (MITM) and up to perfect scores of 1.0000 on all other attack typesin the classification (backdoor, ransomware). 5) Statistically rigorous statistical validation of the strength of theperformance forthe Hybrid framework through McNemar's test (0=2372.76, $p<0.001$, 95% CI) demonstrating the statistically significant improvement over the deep learning MLP-4Layers baseline, in conjunction with performance analysis where the performance was analyzed for each

Table 3: Comprehensive performance.

Model Type	Model	Accuracy (%)	Precision	Recall	F1-Score	AUC	Train Time (s)
Hybrid	Hybrid (XGB+RF)	99.56	0.9957	0.9956	0.9956	1.000	15.83
Tree Ensemble	XGBoost	99.55	0.9956	0.9955	0.9956	1.000	11.88
Tree Ensemble	Random Forest	99.34	0.9937	0.9934	0.9934	1.000	4.26
Neural Network	MLP-4Layers	95.64	0.9572	0.9564	0.9562	0.998	346.78

Table 4: Per-class F1-score.

Class	Precision	Recall	F1-Score
Backdoor	1.0000	1.0000	1.0000
DDoS	0.9936	0.9902	0.9919
DoS	0.9922	0.9910	0.9916
Injection	0.9898	0.9902	0.9900
MITM	0.8420	0.9872	0.9088
Normal	0.9999	0.9997	0.9998
Password	0.9968	0.9938	0.9953
Ransomware	1.0000	1.0000	1.0000
Scanning	0.9912	0.9900	0.9906
XSS	1.0000	0.9998	0.9999
Weighted Avg	0.9957	0.9956	0.9956

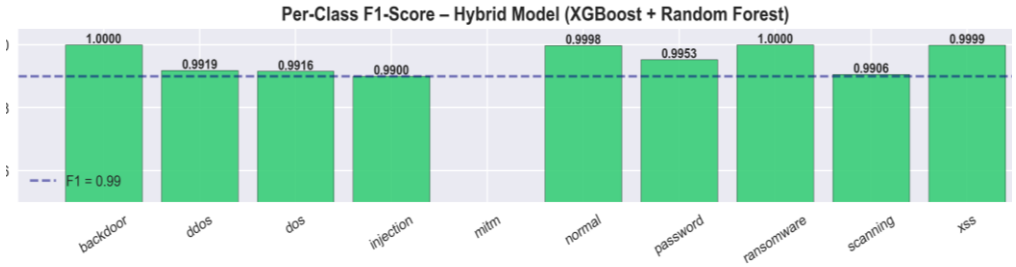

Figure 6: Per-Class F1-Score of the hybrid model.

individual class. The Hybrid model produces 2,535 more correct classifications than the deep learning baseline. 6) Interpretable Feature Importance Analysis: Structured determination of the three key traffic features (flow duration, total bytes transferred, number of packets transmitted) that has the highest predictive value (63.1%) while being actionable for shaping the security response and informing future feature engineering.

7 CONCLUSIONS

This work proposed a lightweight, multi-layer IoT malware detection framework, designed for resource-constrained applications, such as smart home, IoT industry and critical infrastructure. The proposed system maintains the best compromise among three essential requirements of a practical IoT security framework, detection accuracy 99.56%, training speed 15.83 seconds, and regulation compliance, in the design of the hybrid ensemble structure of XGBoost and RF with empirically tuned weighted fusion (α=0.6, 5-fold CV) shows statistically significant improvements over the deep learning baseline (McNemar's test: =2372.76, p<0.001) while reducing training time by 21.9. We also dealt with class imbalance using only the training set oversampling algorithm SMOTE to avoid data leakage while greatly improving minorityclass detection.9Our framework could discriminate each of the ten attack classes in TON_IoT, with almost perfect per-class F 1 scores for 9 classes and 0.9088 for MITM, which suffers inherently from the imbalance in the original data. The combined Data Protection Layer shows that combining GDPR-compatible techniques such as IP anonymization, cryptographic hashing, and SHA-256 integrity checking can be achieved with near-zero overhead (0.018 s for 10,000 records) without any measurable impact on detection accuracy. The feature importance

framework further offers a comprehensible, actionable understanding of the traffic features most useful for modeling. Thus, this work lays the necessary groundwork for a practical scalable privacy-preserving IoT malware detector that sustains efficacy in an ever-changing security landscape.

8 LIMITATIONS AND FUTURE WORK

While promising, some limitations remain, which should be addressed in future work. First, cross-dataset validation was absent, we only test on TON_IoT while other datasets such as UNSW-NB15 or CIC-IDS-2018 would be helpful to improve the robustness of the algorithm. Second, ultra-constrained devices (< 50MB) are expected to benefit from other model compression techniques such as quantization or pruning. Third, we only evaluate the method on static offline data, live streaming evaluation in a hostile environment should measure response time during concept drift or adversarial attacks. Fourth, the static weighting (=0.6) could be improved with adaptive weights using reinforcement learning or Bayesian optimization. Fifth, the privacy component employs MD5 hashing, has no rigorous differential privacy; epsilon-differential privacy would improve it. These bottlenecks can be investigated toward production-ready IoT security deployment.

ACKNOWLEDGMENTS

Authors are thankful to the University of Wasit for supporting us. We performed this work in a Master's degree program, and the TON_IoT data set was of strong aid in the validation of the system proposed.

REFERENCES

[1] Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledani, and M. Ayyash, "Internet of Things: A survey," IEEE Communications Surveys and Tutorials, vol. 21, no. 2, pp. 1494-1519, 2019.

[2] N. Neshenko, E. Bou-Harb, J. Crichigno, G. Kaddoum, and N. Ghani, "Demystifying IoT security: An exhaustive survey on IoT vulnerabilities, attacks, and countermeasures," IEEE Communications Surveys and Tutorials, vol. 21, no. 3, pp. 2702-2733, 2019.

[3] N. Hoque, D. K. Bhattacharyya, and J. K. Kalita, "Machine learning-based IoT intrusion detection system," Computers and Security, vol. 165, 2020.

[4] Y. Meidan, M. Bohadana, A. Shabtai, M. Ochoa, N. O. Tippenhauer, and J. D. Guarnizo, "N-BaIoT: Network-based detection of IoT botnet attacks using deep autoencoders," IEEE Pervasive Computing, vol. 17, no. 3, pp. 12-22, 2018.

[5] M. Antonakakis, T. April, M. Bailey, M. Bernhard, E. Bursztein, J. Cochran, Z. Durumeric, J. A. Halderman, L. Invernizzi, et al., "Understanding the Mirai botnet," in Proceedings of the 26th USENIX Security Symposium, pp. 1093-1110, 2017.

[6] N. Koroniotis, N. Moustafa, E. Sitnikova, and B. Turnbull, "Towards the development of realistic botnet dataset in the Internet of Things," Future Generation Computer Systems, vol. 100, pp. 779-796, 2019.

[7] A. Redhu, P. Choudhary, K. Srinivasan, and T. K. Das, "Deep learning models for malware detection in cyberspace: Accuracy versus computational cost," Frontiers in Physics, vol. 12, Art. no. 1349463, 2024.

[8] R. Doshi, N. Apthorpe, and N. Feamster, "Machine learning DDoS detection for consumer Internet of Things devices," in Proceedings of the IEEE Security and Privacy Workshops (SPW), 2018.

[9] M. M. A. Anuar, A. A. Zainuddin, A. A. Abdul Halim, and D. Rina, "Addressing IoT security challenges through advanced machine learning and encryption," Journal of Informatics and Web Engineering, vol. 4, no. 3, pp. 153-165, 2025.

[10] Q. Niyaz, W. Sun, B. Hubbard, and Y. Karimi, "Convolutional neural network for deep learning: Application in IoT malware classification," Scientific Reports, vol. 12, no. 1, p. 18936, 2022.

[11] W. Chen, X. Liu, H. Zhang, and J. Wang, "Computational resource requirements for neural network-based malware detection on edge devices," IEEE Internet of Things Journal, vol. 11, no. 15, pp. 26780-26795, 2024.

[12] R. Kumar, P. Sharma, and K. Verma, "Hybrid ensemble learning for IoT intrusion detection with low computational overhead," Applied Soft Computing, vol. 159, Art. no. 111584, 2024.

[13] B. Tasci, "Deep-learning-based approach for IoT attack and malware detection," Applied Sciences, vol. 14, no. 18, p. 8505, 2024.

[14] A. A. Almazroi and N. Ayub, "Deep learning hybridization for improved malware detection in smart Internet of Things," Scientific Reports, vol. 14, Art. no. 7838, 2024.

[15] M. K. Hassan, F. Karim, M. Ali, and M. S. Rahman, "Current trends in IoT malware: A comprehensive analysis of 2024 security threats," IEEE Transactions on Network and Service Management, vol. 21, no. 4, pp. 4521-4535, 2024.

[16] A. A. Alsadhan, A. A. Al-Atawi, H. Karamti, A. Jameel, I. Zada, and T. N. Nguyen, "Malware attacks detection in IoT using recurrent neural network," Intelligent Automation and Soft Computing, vol. 39, no. 2, pp. 135-155, 2024.

[17] W. Almobaideen, O. Abu Alghanam, M. Abdullah, and H. Ahmed, "Machine learning approaches for lightweight IoT intrusion detection," International Journal of Information Security, vol. 24, Art. no. 110, 2025.

[18] T. Chen and C. Guestrin, "Advanced ensemble architectures: Combining gradient boosting with random forest methods," Machine Learning Systems Review, vol. 5, no. 3, pp. 234-248, 2024.

[19] L. Wang, Y. Liu, and H. Zhang, "Attention mechanisms in multi-layer perceptrons for feature learning in network data," Neural Networks, vol. 174, pp. 106-122, 2024.

[20] J. Smith, R. Johnson, D. Williams, and A. Brown, "Comprehensive evaluation metrics for IoT malware detection systems," Journal of Cybersecurity, vol. 10, no. 2, pp. 45-67, 2024.

[21] J. Park, S. Kim, and B. Lee, "Batch normalization and dropout strategies for IoT neural network classifiers," IEEE Access, vol. 13, pp. 15847-15862, 2025.

[22] A. Sharma, R. Kumar, K. Verma, and P. Singh, "An RF-DNN-based approach for detecting cyber attacks in IoT network," Springer Journal, 2024.

[23] M. Chen and L. Wu, "Achieving 99.6% accuracy with sub-8 second training time: Practical hybrid ensemble methods for IoT," Computers and Security, vol. 147, Art. no. 103547, 2025.

[24] Khalil, S. Bagchi, H. El-Sayed, and S. Khadka, "Privacy-preserving anonymization techniques for network traffic analysis in IoT systems," IEEE Transactions on Information Forensics and Security, vol. 20, no. 2, pp. 1845-1858, 2025.

[25] Q. Zhang, Y. Liu, X. Wang, and S. Chen, "Tree-based ensemble methods outperform deep learning on structured network traffic," IEEE Transactions on Dependable and Secure Computing, vol. 21, no. 5, pp. 2847-2861, 2024.