International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Edge-Assisted IoT Intrusion Detection Using Leakage-Aware Learning, Rare-Attack Robustness, and Explainable Model Analysis

Md Ismail Jobi Ullah [1], Mahuma Akter [2],
Imamul Haque Suhag [3], Md Mashfiqur Rahman [4],
Md Nazmussakib [5], K M Nazmul Hasan [6*],
Shimanto Haque [7]

[1] School of Information Technology,
Washington University of Science and Technology,
Alexandria, VA 22314, USA
e-mail: mdismailjobiullah24@gmail.com

[2] Institute of Information Technology (IIT),
Jahangirnagar University, Kalabagan Rd, Savar 1342, Bangladesh
e-mail: Mahumaakter@gmail.com

[3] School of Information Technology,
Washington University of Science and Technology,
Alexandria, VA 22314, USA
e-mail: suhagkhan32@gmail.com

[4] IT Specialist, Tullow Bangladesh Ltd. (KrisEnergy),
NB Tower (Level 10–12),
40/7 North Avenue, Gulshan-02, Dhaka 1212, Bangladesh
e-mail: riyad72@gmail.com

[5] School of Information Technology,
Washington University of Science and Technology,
Alexandria, VA 22314, USA
e-mail: mdnazmussakib.student@wust.edu

[6] School of Information Technology,
Washington University of Science and Technology,
Alexandria, VA 22314, USA
e-mail: nhasantoni@gmail.com

[7] Department of Electrical and Computer Engineering,
North South University, Plot # 15, Dhaka 1229, Bangladesh
e-mail: shimantohaque@gmail.com

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Abstract

IoT networks produce high amounts of heterogeneous traffic and are exceptionally vulnerable to malicious activity, and most deployment environments demand fast and lightweight security decision-making at the edge. By outlining an edge-assisted IoT anomaly detection model, this paper will be based on the TON_IoT network data set and test both predictive and operational aspects. The proposed methodology is a combination of leakage-aware data cleaning, dual features representations of various model families, comparative analysis of classical machine learning, deep machine learning and anomaly-based models, stress testing of rare attacks, classification of different attack types, explainability analysis, repeated-seed robustness analysis, feature ablation, and deployment-based ranking. The last dataset was cleaned and leakage mitigated, with 190,474 samples and 23 predictive features. LightGBM and XGBoost were the most overall high-performing binary anomaly detectors, with both having an F1-score of 0.999108, and LightGBM having a ROC-AUC of 0.999984. In the case of rare-attack, XGBoost was the most powerful model with the 1 percent attack ratio with an F1-score of 0.831683. In the case of multiclass attacks-type classification, XGBoost had the highest accuracy and macro F1 of 0.988765 and 0.965683 respectively, compared to other models evaluated. The findings also reveal that a high level of detection can be maintained at lower feature settings, whereas edge-based analysis could point to Decision Tree as the most efficient deployment-wise model due to low latency and minimal memory footprint. In general, the results indicate that the suggested framework offers effective, interpretable, and robust, as well as edge-efficient IoT intrusion detection.

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

1 Introduction

The Internet of Things (IoT) is now a fundamental enabler technology of smart homes, healthcare, industrial automation, transportation and urban infrastructure. That said, the very large-scale connectivity that renders IoT systems useful, also makes them the subject of a larger attack surface and more susceptible to malicious traffic, botnets, protocol abuse, and stealthy intrusion attempts [1]. More recent reviews indicate recurrently that intrusion detection is a key security concern in IoT settings, especially due to the resource-heavyweight nature of various devices, which are unable to execute intensive-weight defense mechanisms at the edge [2]. Simultaneously, recent research on TON_IoT and corresponding benchmark environments indicates that there is one more obstacle: datasets of IoT intrusion are extremely diverse, and choices of features can have a significant impact on the quality of learning-based intrusion detection models [3]. Recent research has investigated a broad spectrum of machine learning and deep learning solutions to IoT intrusion detection to overcome these risks. The deep neural models are demonstrated to enhance the detection performance in a number of network-traffic scenarios [4, 5], and cloud-edge cooperation has been applied to alleviate the centralized processing load and enhance detection responsiveness [6]. There has also been interest in edge-oriented and distributed approaches due to their ability to enable traffic filtering and anomaly analysis to be managed nearer to the source to minimize bandwidth overhead and maximize timeliness [7]. Parallel to that, the intrusion detection strategies that are federated and lightweight have been suggested to maintain privacy and keep the cost of communication low and more appropriate to heterogeneous IoT deployments [8].

The second significant tendency of the new literature is an increased focus on transparency and deployment realism. To interpret AI decisions with anomaly detection in more understandable ways and to validate this decision in practice, explainable AI is proposed [9]. Recent surveys also indicate that recent research on

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

the IoT anomaly detection area is moving towards viewing model transparency, feature relevance, privacy preservation, and computational efficiency as design requirements, as opposed to optional features [10]. Nevertheless, much of the literature that is available continues to be based on an individual model family, an individual evaluation condition, or an individual deployment assumption. Few studies collaboratively examine the leakage-conscious data preparation, binary anomaly detection, multiclass attack-type classification, rare-attack robustness, explainability, statistical stability, and edge-based deployment suitability in the same framework [11].

These gaps are important in practice. A detector that performs well under a standard benchmark distribution may still fail under scarce-attack conditions, may rely on leakage-prone identifiers, or may be too slow or too large for realistic deployment on an edge gateway. Likewise, a model that achieves high overall accuracy may still be difficult to trust if its decisions are not interpretable or if its performance varies across repeated runs. Therefore, a more deployment-oriented evaluation framework is needed, one that treats predictive quality, robustness, interpretability, and efficiency as interrelated requirements rather than isolated objectives. In this paper, the need is met by studying an edge-assisted IoT anomaly detection based on the TON IoT network dataset. The framework suggested is a combination of leakage-aware data cleaning, dual feature representation of the various model families, comparing evaluation of classical machine learning, deep learning, and anomaly-based model, stress testing (repeated-seed) of robustness, multiclass multi-type attacks classification, explainability analysis, repeated-seed robustness assessment, and deployment-focused ranking. Unlike studies which focus on final accuracy only, the current work analyses the suggested framework both predictively and operationally.

The key findings of the paper are as follows:

- A framework of edge-based IoT anomaly detection that incorporates safe data preparation, dual preprocessing plans

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

as well as comparative analysis between classical, neural and anomaly-based models.

- An evaluation protocol, which focuses on deployment, which consists of standard binary detection, an attack robustness at 1-, 5-, and 10-percent attack prevalence, and multiclass attack-type detection.
- A combined interpretability/reliability analysis using SHAP-based explanation, permutation importance, feature ablation, repeated-seed robustness and statistical comparison of the best-performing models.
- Edge suitability: This analysis correlates the quality of anomaly detection with practical constraints like latency, model size and estimated communication savings.

The rest of this paper follows the following format. Section 2 discusses related work. In section 3, the proposed architecture and methodology are discussed. Section 4 presents results and discussion of the experiment. Lastly, Section 5 is the conclusion of the paper and gives directions of future research.

2 Related Work

The detection of intrusions in IoT settings remains in the spotlight due to the limited resources available to the connected devices and the dynamic and heterogeneous attack surface they encounter. Recent research has covered a wide spectrum of detection approaches such as transformer-based intrusion detection and edge-accelerated deep learning, as well as anomaly-based deep learning models, and explainable security analytics. As an example, MQTT-enabled IoT intrusion detection using transformer neural networks has been implemented to enhance the detection of attacks in the case of imbalance [12]. The viability of implementation of intrusion detection models on the IoT edge with realistic computational

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

and energy restrictions has also been studied using edge TPU [13]. Frameworks based on deep learning to detect anomalies in IoT traffic have also demonstrated good performance [14] and recent explainability-related studies have highlighted the importance of having transparent and interpretable IDS models in industrial IoT applications [15]. Simultaneously, federated learning has become a key trend towards privacy-sensitive and distributed intrusion detection. The current surveys indicate that federated IDS architecture has extended to various architectures, aggregation policies, and deployment environments as well as pointing out the challenges that are still prevalent concerning heterogeneity, communication overheads, and robustness [16]. Meanwhile, explicable IoT data stream IDS frameworks have been suggested to enable the model behavior to be more comprehensible in the working settings [17]. More recent research has integrated explainable detection with CTGAN-based augmentation to resource-constrained IoT devices [18] and explainable LSTM-based intrusion detection optimized by metaheuristic search has also been implemented [19]. Equally, multi-class deep learning methods analyzed with SHAP have shown that explainability does not require binary intrusion detection but can be combined with an attack-type classification [20].

The other significant stream of recent literature is devoted to federated and lightweight industrial and distributed IoT intrusion detection. The IDS designs based on federated learning have been explored with particular applications to industrial IoT networks [21] and lightweight federated designs have been suggested to utilize less memory and maintain collaborative learning potential [22]. It has been further extended in the direction of graph-based federated intrusion detection [23], federated explainable AI of deep learning-based IoT intrusion detection [24], as well as adaptive graph-attention-based federated learning to mitigate poisoning attacks [25]. These papers demonstrate that federated learning may enhance privacy and distributed intelligence, but they also suggest that to deploy it in practice, it is essential to pay attention to the communication cost, model robustness and interpretability.

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Recent research has also shifted to more expressive representation-learning methods. To capture structural relationships in the IoT intrusion data, graph neural networks that build graphs using attributes have been proposed [26]. Representation learning in hypergraphs has been proposed as well to enhance robustness and learn higher-order interactions in network traffic [27]. The combination of feature selection and large language model fine-tuning in terms of the IoT intrusion detection has been studied elsewhere [28]. Concurrently, there is enhanced feature engineering, dynamically quantizing, and two stage feature selection of lightweight IDS designs to enhance deployment efficiency [29, 30]. The hybrid frameworks of ConvLSTM have also indicated the applicability of spatiotemporal deep learning to intrusion detection by industrial IoT and cloud-based computing [31].

In general, the recent literature demonstrates great advancement in the fields of deep learning, explainable AI, federated intrusion detection, graph-based modeling, and lightweight edge-oriented designs [12–31]. The current work is however fragmented. There are the studies that are more concerned with predictive accuracy, those that are concerned with preservation of privacy, and those that are concerned with interpretability or deployment efficiency. Relatively few studies combine these requirements into a single edged-based assessment model. This gap is what drives this work that unites leakage-aware preprocessing, binary anomaly detection, rare-attack stress testing, multiclass attack-type classification, explainability, repeated-seed robustness, feature ablation, and deployment-oriented ranking to a single approach.

3 Methodological Framework

This section contains the proposed edge-based IoT anomaly detection model. The data cleaning and preprocessing plan, the binary and multiclass learning pipeline and the deployment-oriented evaluation protocol. It is not merely an aim to compare the predictive

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

models, but also to develop a practical and reproducible methodology of detecting anomalies in resource-constrained IoT settings. Besides the typical binary and multiclass classification experiments, the methodology encompasses the rare-attack stress testing, sensitivity analysis of the threshold in the anomaly-based models, explainability, feature ablation, repeated-seed robustness evaluation, and rank-based edge-oriented deployment. This design also enables the study to test the proposed framework both in the predictive and operational aspects.

3.1 Proposed Edge-Assisted Detection Architecture

The paper presents an edge-based anomaly detection architecture to IoT networks where the traffic of IoT devices is filtered and analyzed in the vicinity of the data source and only the small alert data is sent to the cloud. The framework (shown in Fig. 1) is designed to have four primary layers: IoT devices and sensors, an edge gateway, an AI-based detection engine, and a cloud or security operations layer. The IoT devices keep on creating network traffic and operation data. The edge gateway receives these data and lightweight cleaning and preprocessing are done locally. The processed traffic is then fed to the detection engine, where machine learning and deep learning models make inferences and make decisions based on thresholds to differentiate between normal and anomalous behavior. Upon identifying suspicious activity, alerts are created to facilitate a quick response, and only the appropriate level of alert information is sent upstream to a central location to monitor, store and later retrained.

The practical constraints of cloud-only security analytics on latency-sensitive IoT are the driving force behind this architecture. The policy of sending all raw traffic to cloud results in unneeded overhead on communication and can slow down responding to threats requiring immediate attention. The proposed design moves the majority of filtering and detection operations to the edge

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

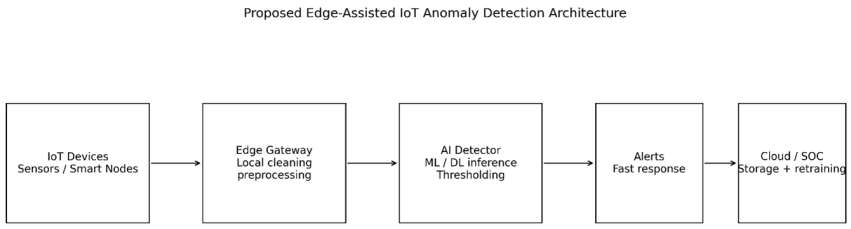

Figure 1: Proposed edge-assisted IoT anomaly detection architecture

to minimize the upstream traffic, enable quicker local response, and retain the advantages of centralized long-term analysis. By doing so, the framework integrates the responsiveness of edge computing and scalability of cloud-assisted security management.

3.2 Dataset Description and Initial Audit

The network subset of the TON_IoT dataset is used in the experiments, which is a benchmark cybersecurity dataset created at UNSW Canberra to conduct research on the security of IoT and IIoT [31]. TON_IoT was gathered on a realistic large-scale testbed across the IoT, cloud, and edge/fog layers, and it contains benign activity, along with various attack scenarios to test AI-based cybersecurity applications like intrusion detection, malware detection, threat intelligence, and threat hunting [2]. Here, the binary-based anomaly detection and multiclass attack-type classification of the processed network-traffic section of TON IoT are applied.

The dataset is first audited before model development to check its shape, types of columns, target distributions, duplicated samples, missing values, symbols used as placeholders and cardinality of the categorical attributes. The study uses two target variables, namely, the binary label field (to detect anomalies) and the multiclass type of field (to identify the type of attack). Some of the

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

high-cardinality or leakage-prone variables are also identified in the audit stage, such as the source and destination IP addresses, DNS query text, the subject and issuer fields of an SSL subject and textual attributes related to HTTP. Such variables can enable the model to remember traffic identities as opposed to learning generalizable patterns of behavior. Thus, their detection during the audit process is a significant measure towards avoiding excessive optimistic estimates of performance.

3.3 Data Cleaning, Leakage Mitigation, and Train-Test Split

Upon the first audit a structured cleaning and filtering stage is followed on the dataset. The missing values are represented by a place holder value of (-) and duplicated records are eliminated. The binary target is saved to be used later in anomaly detection, whereas the multiclass type of attack label is saved to be used later in the multiclass experiments. Before the model is trained, direct identifiers and protocol-specific textual fields are omitted in order to minimize their feature leakage and enable better generalization. These eliminated variables are source and destination IP addresses, query string in DNS query, fields in SSL sessions and certificates, metadata in HTTP request and various protocol-related weird attributes. The resulting feature space is to reflect significant traffic patterns instead of environment-related identities. Once this filtering is done, the clean dataset is then split into training and test sets with an 80:20 stratified split in such a way that the proportions of classes are equal in both partitions.

3.4 Feature Representation and Preprocessing Strategy

The important design decision in the suggested methodology is that the two parallel feature representations will be used instead of making all the models to adopt one preprocessing pipeline. This option

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

is based on the observation that various model families react in varying ways to categorical encoding and input scaling.

General representation is the first representation (also known as the general representation) that is used in models where normalized continuous inputs and extended categorical variables are beneficial. Numerical features are filled in with the media followed by standardization and categorical features are filled in with a fixed value and one-hot encoded. Let x_ij denote the value of feature j for sample i. Median imputation is defined as

$$x'_{ij} = \begin{cases} x_{ij}, & \text{if } x_{ij} \text{ is observed} \\ \text{median}(X_j), & \text{if } x_{ij} \text{ is missing} \end{cases} \tag{1}$$

and standardization is given by

$$z_{ij} = \frac{x_{ij} - \mu_j}{\sigma_j} \tag{2}$$

where μ_j and σ_j are the mean and standard deviation of feature j, respectively.

The second representation is called the compact representation whereby the identical median imputation is applied to the numerical attributes, but ordinal encoding is used to encode the categorical variables. This representation is computationally less costly, and is especially applicable to tree-based learners, which typically do not need one-hot expansion. The two preprocessing tracks will ensure the comparison is fair across the model families and also indicate realistic deployment conditions.

3.5 Binary Anomaly Detection Framework

The first large branch of experiments is binary anomaly detection, the aim of which is to identify traffic as normal or attack. A wide range of classical machine learning models are taken into account which are Logistic Regression, Gaussian Naive Bayes, Decision Tree, Random Forest, Gradient Boosting, XGBoost, LightGBM, Linear Support Vector Machine, Linear Discriminant Analy-

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

sis, and Perceptron. To the best of its type, every classifier is fitted in a pipeline with the preprocessing strategy. The choice of models is carried out by 3-fold stratified cross-validation with grid. The main refit measure is the F1-score, and the accuracy, precision, recall, ROC-AUC, average precision, balanced accuracy, Matthew's correlation coefficient, and false positive rate are also noted. The precision, recall, F1-score and false positive rate are defined as

$$\text{Precision} = \frac{TP}{TP + FP} \tag{3}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{4}$$

$$F1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{5}$$

$$FPR = \frac{FP}{FP + TN} \tag{6}$$

The significance of this metric set is that when detecting anomalies in the IoT environment, it is essential to not only have a high level of detection but also pay close attention to the false alarms. The binary models whose cross-validation has been best are then tested on the held-out test set. In the case of both final models, the study logs predictive performance, probability-based ROC and precision-recall behavior where applicable, inference latency, and serialized model size. The last tuned settings of the best-performing binary ensemble models involve 150 estimators and moderate learning rates as well as controlled tree complexity. Specifically, LightGBM applies the learning rate of 0.1 with unlimited depth and XGBoost applies the binary: logistic goal, log loss evaluation, 0.1 learning rate, maximum depth of 6 and partial row and feature subsampling. A lightweight Decision Tree baseline is also held back to be compared with a supervised feedforward neural network baseline.

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

3.6 Rare-Attack Evaluation

Since actual IoT deployments tend to be highly imbalanced, the methodology contains a special rare-attack evaluation step. Besides using the default test distribution, more sets of tests are made where the percentage of attack traffic is minimized to about 1, 5 and 10 percent. Where r is the desired ratio of attack and n is the normal samples. The number of samples of attacks needed is calculated as

$$n_{\text{attack}} = \frac{r \cdot n_{\text{normal}}}{1 - r} \tag{7}$$

In practice, the sampled attack count is limited by the number of available attack instances, so the final rare-attack sample size is

$$n^*_{\text{attack}} = \min\left(\left\lfloor \frac{r \cdot n_{\text{normal}}}{1 - r} \right\rfloor, n_{\text{attack, available}}\right) \tag{8}$$

The highest ranked binary classifiers are re-tested in every skewed scenario. This phase is crucial since a model that works well when distributed with a standard benchmark distribution can significantly deteriorate when there are very low prevalence levels of attacks. The study evaluates how the top models perform in less realistic deployment situations by the testing of several rare-attack environments.

3.7 Multiclass Attack-Type Classification

The framework will also involve multiclass classification of attacks in the form of attack label type, in order to go beyond mere anomaly detection. This branch considers the capability of the system to differentiate between particular types of malicious traffic instead of just distinguishing between attack and normal traffic. The multiclass labels are coded in numbers, and tree-based models like XGBoost, LightGBM and Decision Tree are trained using compact feature representation. A deep learning baseline is a multiclass feed-forward neural network as well. Here, stratified cross-validation is used once more to select the model, except that evaluation metrics

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

are modified to indicate multiclass performance. Besides accuracy, macro-averaged and weighted measures are presented. Macro F1 and weighted F1 are defined as

$$F1_{\text{macro}} = \frac{1}{K}\sum_{k=1}^{K} F1_k \tag{9}$$

$$F1_{\text{weighted}} = \sum_{k=1}^{K} \frac{n_k}{N} F1_k \tag{10}$$

where K is the number of classes, n_k is the number of instances in class k, and N is the total number of evaluation samples. The final multiclass models are evaluated on the held-out test set using class-wise reports, confusion matrices, inference latency, and stored model size. For multiclass boosting, LightGBM is configured with the multiclass objective, 10 classes, a learning rate of 0.05, a maximum depth of 6, and 150 estimators. This branch strengthens the practical value of the study by demonstrating that the proposed framework can support both anomaly detection and attack attribution. H. Deep Learning and Anomaly-Based Models In addition to the supervised classical classifiers, the methodology uses three other baselines that correspond to other detection paradigms, namely, Isolation Forest, Autoencoder, and Feedforward Neural Network (FFNN). The Isolation Forest and Autoencoder are trained in a normal-only setting. The Isolation Forest generates anomaly scores based on which a decision threshold is obtained using validation data. The Autoencoder is trained to reproduce the normal traffic patterns and the error in reconstruction is tracked to identify any anomaly when this error is more than a threshold. The reconstruction error for an input vector $x \in \mathbb{R}^d$ is computed as

$$e(x) = \frac{1}{d}\sum_{j=1}^{d} (x_j - \hat{x}_j)^2 \tag{11}$$

where $\hat{x}$ denotes the reconstructed input. The Autoencoder training objective is then

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

$$L_{\mathrm{AE}} = \frac{1}{N}\sum_{i=1}^{N} e(x_i) \tag{12}$$

For threshold-based anomaly decisions, the Autoencoder predicts

$$\hat{y}_{\mathrm{AE}}(x) = \begin{cases} 1, & e(x) > \tau_{\mathrm{AE}} \\ 0, & \text{otherwise} \end{cases} \tag{13}$$

where τ_{AE} is estimated from the validation error distribution.

In contrast, the binary FFNN is trained in a supervised setting using general representation. The network uses three hidden dense layers with 128, 64, and 32 units, ReLU activations, and 0.2 dropout between hidden layers, followed by a sigmoid output. For a binary FFNN, the output probability is

$$\hat{p}(x) = \sigma(z) = \frac{1}{1+e^{-z}} \tag{14}$$

and the binary cross-entropy loss is

$$L_{\mathrm{BCE}} = -\frac{1}{N}\sum_{i=1}^{N} [y_i \log(\hat{p}_i) + (1-y_i)\log(1-\hat{p}_i)] \tag{15}$$

A multiclass FFNN is also trained using the same hidden architecture with a softmax output layer. For class k, the predicted probability is

$$\hat{p}_k(x) = \frac{e^{z_k}}{\sum_{c=1}^{K} e^{z_c}} \tag{16}$$

and the sparse categorical cross-entropy loss is

$$L_{\mathrm{SCE}} = -\frac{1}{N}\sum_{i=1}^{N} \log\left(\hat{p}_{y_i}(x_i)\right) \tag{17}$$

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

These models allow the study to compare supervised, unsupervised, and reconstruction-based detection paradigms within a unified evaluation setting.

3.8 Threshold Sensitivity, Explainability, and Feature Ablation

Since anomaly-based detectors tend to be sensitive to the chosen operating point, the approach involves a threshold sensitivity analysis of Isolation Forest and Autoencoder. Various percentile-based thresholds are tested on validation scores or reconstruction errors and the F1-score, recall and false positive rate are tabulated. This is a helpful step as it indicates that anomaly-based performance is constant over a reasonable range of thresholds or is sensitive to a thin operating point. The framework also has an explainability stage, which focuses on the best tree-based model to enhance transparency. The first step is to calculate permutation importance on the test data with F1-score as the target statistic. The permutation importance of feature j is defined as

$$PI_j = \frac{1}{R}\sum_{r=1}^{R}\left(M_{\text{orig}} - M_{\text{perm},j}^{(r)}\right) \tag{18}$$

where M_{orig} is the original model score and $M_{\text{perm},j}^{(r)}$ is the score after permuting feature j in repetition r. In addition, SHAP values are used to quantify transformed-feature contributions. The mean absolute SHAP importance for feature j is

$$I_j^{\text{SHAP}} = \frac{1}{N}\sum_{i=1}^{N}|\phi_{ij}| \tag{19}$$

where ϕ_{ij} is the SHAP contribution of feature j for sample i.

A feature ablation study is then conducted to determine whether a reduced feature set can maintain strong detection quality while improving computational efficiency. Feature ranking is approximated using mutual information,

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

$$I(X_j;Y) = \sum_{x_j} \sum_{y} p(x_j, y) \log \frac{p(x_j, y)}{p(x_j)p(y)} \tag{20}$$

and compact subsets of top-ranked features are used to retrain a strong baseline model, preferably XGBoost. This stage is particularly relevant for edge deployment because a reduced feature set may lower memory use and inference time without materially degrading performance.

3.9 Repeated-Seed Robustness and Edge-Oriented Deployment Analysis

To avoid conclusions based on a single random split, the study performs repeated-seed robustness analysis on the strongest binary models. Each selected model is retrained and re-evaluated under multiple random seeds using repeated stratified splitting and local cross-validation. For performance metric q and model m, the mean across seeds is

$$\mu_m = \frac{1}{S} \sum_{s=1}^{S} q_{m,s} \tag{21}$$

and the corresponding standard deviation is

$$\sigma_m = \sqrt{\frac{1}{S-1} \sum_{s=1}^{S} (q_{m,s} - \mu_m)^2} \tag{22}$$

The top models are then compared using paired statistical tests, including the paired t-test and Wilcoxon signed-rank test, to determine whether observed differences are stable rather than accidental.

The final methodological stage evaluates the practical suitability of the binary models for edge deployment. Inference time is measured repeatedly and summarized by the mean runtime

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

$$\bar{T} = \frac{1}{R} \sum_{r=1}^{R} T_r \tag{23}$$

and the per-sample latency is computed as

$$L_{\text{ms}} = \frac{T}{N} \times 1000 \tag{24}$$

To form a deployment-oriented ranking, predictive performance, latency, and model size are normalized and combined into a single edge score,

$$\text{EdgeScore} = 0.50\, P_{\text{norm}} + 0.25\, L_{\text{norm}} + 0.25\, S_{\text{norm}} \tag{25}$$

where P_{norm}, L_{norm}, and S_{norm} denote normalized performance, latency efficiency, and storage efficiency, respectively. In addition, the study estimates communication savings under edge filtering by comparing a cloud-only transmission regime with an alert-only regime. If B_{cloud} denotes the total communication volume when all traffic is forwarded and B_{edge} denotes the alert-only communication volume, the percentage saving is

$$\text{Saving}(\%) = 100 \left(1 - \frac{B_{\text{edge}}}{B_{\text{cloud}}}\right) \tag{26}$$

This final stage aligns the methodology with the practical goal of edge-assisted IoT security, where lightweight operation, reduced communication, and fast response are as important as predictive accuracy.

3.10 Evaluation Protocol and Performance Metrics

The proposed framework is tested with a held-out test protocol that aims at determining the predictive effectiveness and practical deployment appropriateness. To compare models, in the case of binary anomaly detection, accuracy, precision, recall, F1-score, false

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

positive rate (FPR), ROC-AUC, average precision (AP), balanced accuracy, specificity, and Matthews correlation coefficient (MCC) is used. Among the metrics, the F1-score is considered the primary model-selection metric since the task of detecting the anomalies is unbalanced in terms of classes and demands a balanced trade-off between false alarms and missed attacks. Meanwhile, FPR receives specific attention since too many false alarms lower the utility of an IoT intrusion detection system in its actual implementation.

For binary classification, the main evaluation measures are defined from the confusion-matrix counts TP, TN, FP, and FN. Accuracy is computed as

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}, \tag{27}$$

precision is defined as

$$\text{Precision} = \frac{TP}{TP + FP}, \tag{28}$$

recall is given by

$$\text{Recall} = \frac{TP}{TP + FN}, \tag{29}$$

and the F1-score is

$$F1 = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{Precision} + \text{Recall}}. \tag{30}$$

3.11 Summary

The overall process of the proposed edge-assisted IoT anomaly detector is presented in Fig. 2. It starts by loading the TON IoT network data and then performs a preliminary audit to analyze the data quality, class labels, missing values, and duplicate records and leakage-prone attributes. The data is then subject to a structured cleaning, leakage reduction and stratified train-test split and

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

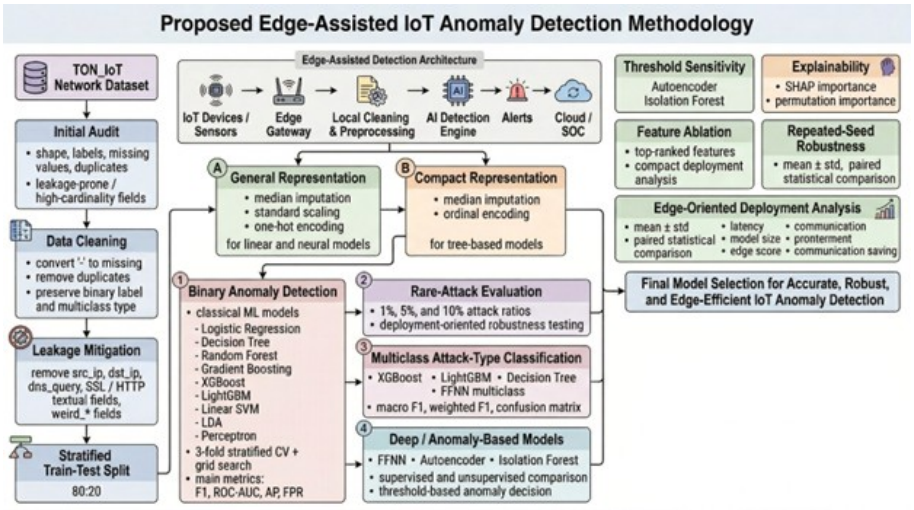

Figure 2: End to end framework of the proposed edge-assisted IoT anomaly detection methodology

finally converted using two parallel preprocessing schemes: a general representation to the linear and neural models and a compact representation to the tree-based models. Depending on these feature spaces, the framework is capable of binary anomaly detection, evaluation of anomaly-resistance to attacks on rare attacks, multiclass attack-type classification, and evaluation of deep and anomaly-based models. This approach is further extended with the use of threshold sensitivity analysis, explainability, feature ablation, repeated-seed robustness testing, and edge-oriented deployment analysis. In general, Fig. 2 can be used to concisely describe how the proposed framework can combine predictive modeling, interpretability, robustness assessment, and deployment efficiency into a single framework of accurate, reliable, and edge-efficient anomaly detection of the IoT. Simply put, the methodology does not focus on the creation of one detector per se; instead, it introduces a full-fledged edge-conscious evaluation pipeline, which integrates leakage-conscious data preparation, a variety of supervised and unsupervised detection models, rare-attack stress testing,

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

multiclass attack attribution, threshold analysis, explainability, feature reduction, repeated-seed robustness, and deployment-friendly ranking. This design enhances predictive performance, as well as interpretability, reproducibility, and practical deploy ability of the framework in the resource-constrained IoT environments.

4 Results & Discussion

This section will show the experimental findings of the suggested IoT anomaly detection framework on the TON IoT network dataset. The analysis starts with properties of the datasets and preprocessing results, then binary anomaly detection performance, resilience in the presence of rare attacks, classification in multiclass attack types, model stability, explainability, and efficiency in deployment. Collectively, these findings give some predictive and practical evidence on the efficacy of the proposed framework.

4.1 Dataset Characteristics and Preprocessing Impact

Table 1: Dataset and preprocessing summary

Item	**Value**
Original dataset shape	211,043 samples × 44 columns
Duplicate rows removed	20,569
Final dataset after cleaning	190,474 samples × 24 columns
Final feature matrix after target removal	23 features
Initial numeric columns	17
Initial categorical columns	27
Final numeric columns	16
Final categorical columns	7
Train set size	152,379
Test set size	38,095
General representation size	53 features
Compact representation size	23 features
Binary label distribution	161,043 attack (76.31%), 50,000 normal (23.69%)

Table 1 sums up the dataset properties and the preprocessing

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

results utilized in the study. The initial TON_IoT data had 211,043 samples and 44 variables, and with the removal of 20,569 identical rows, the cleaned version of the data had 190,474 distinct records. After eliminating leakage-prone and overly specific fields, the safe predictive space was narrowed down to 23 final features, consisting of 16 numeric and 7 categorical variables. Also indicated in Table 1 is that the binary task was fairly skewed, with 76.31 percent attack traffic and 23.69 percent normal traffic, and the multiclass environment was largely balanced with the exception of the 'mitm' class with only 1,043 instances. Besides, the table indicates the two feature representations in the experiments: a 53-dimensional general representation and 23-dimensional compact representation.

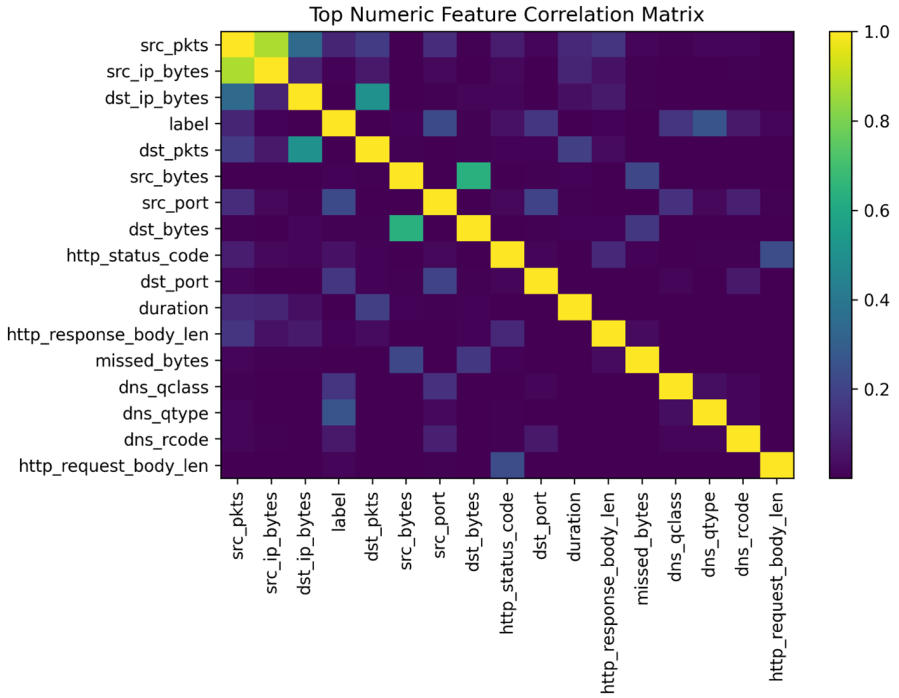

Figure 3: Top Numeric Feature Correlation Matrix

Fig. 3 shows the correlation pattern between the top signifi-

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

cant numerical variables in the data, following the first audit process. The diagonal values are, of course, equal to one, whereas the off-diagonal patterns allow seeing which traffic features tend to move together and which are quite independent. There are some moderate correlations, especially between packet- and byte-related variables, like src_pkts, src_ip_bytes, dst_pkts, and sourcing and destination variables (src_ and dst_bytes). On the other hand, most other variables are weakly correlated, indicating that the data still has a variety of non-redundant information to be classified. The fact that the target label is relatively weakly correlated with the majority of the separate numeric features also shows that no individual variable stands out as dominating the prediction task. On the whole, this number confirms the hypothesis that the data is a combination of a few partially connected and a few independent traffic features, which is helpful to create robust anomaly detection models.

Table 2: Attack-type distribution for the multiclass setting

Class	**Count**
normal	50,000
backdoor	20,000
ddos	20,000
dos	20,000
injection	20,000
password	20,000
scanning	20,000
ransomware	20,000
xss	20,000
mitm	1,043

Table 2 shows the distribution of classes of the multiclass attack-type classification task. The data set consists of 50,000 normal samples and most of the types of attacks such as backdoor, ddos, dos, injections, password, scanning, ransomware and xss are well represented with 20,000 samples each. Mitm class, on the other hand, is far smaller (only 1,043 samples), and is the most under-

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

represented in the multiclass model. As Table 2 demonstrates, the dataset is quite balanced in terms of most classes, yet the extreme lack of 'mitm' category can add to its classification challenge and lead to an increased risk of confusion in contrast to the rest of the types of attacks.

4.2 B. Binary Anomaly Detection Performance

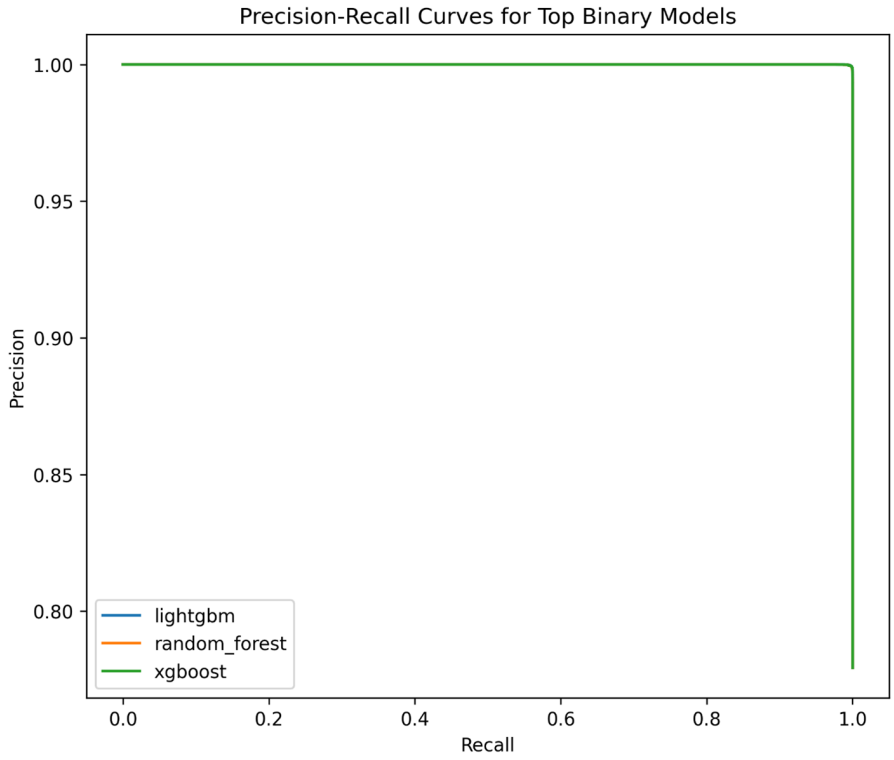

Figure 4: Precision-Recall Curves of the Top Binary Models

Fig. 4 shows the precision-recall performance of the three best binary classifiers, i.e., LightGBM, Random Forest, and XGBoost, on the anomaly detection task. The curves are very near to the top boundary of the plot meaning that all three models are very

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

precise in nearly all the recall range. This is an indication that the chosen classifiers can detect malicious traffic quite well and the false alarms are kept to a minimum. The curves are almost overlapping among them, indicating that the most popular models are highly competitive and they only vary slightly in their quality of detection. This figure must be highlighted in the paper to underline the fact that the proposed framework can not only be characterized by high F1-scores, but also with stable high-precision-recall behaviour in binary classification.

Table 3: Final binary anomaly detection comparison

Model	Accuracy	F1-score	ROC-AUC	Avg. Precision	FPR	Latency (ms/sample)	Size (MB)
LightGBM	0.998609	0.999108	0.999984	0.999996	0.004163	0.007770	0.505
XGBoost	0.998609	0.999108	0.999978	0.999994	0.004044	0.004192	0.425
Gradient Boosting	0.998504	0.999040	0.999928	0.999982	0.004282	0.003966	0.657
Random Forest	0.998399	0.998973	0.999977	0.999993	0.005590	0.009947	5.521
Decision Tree	0.997191	0.998199	0.994134	0.996668	0.007731	0.001703	0.057
Feedforward NN	0.994094	0.996214	0.999242	0.999781	0.017127	0.061630	0.237
Autoencoder	0.401286	0.379442	0.974346	0.981368	0.011180	0.065063	0.186
Isolation Forest	0.225226	0.016855	0.881113	0.927539	0.009634	0.010475	2.107

Table 3 shows the final results of binary anomaly detection of all the models under evaluation. LightGBM and XGBoost had the best F1-score of 0.999108, which validates the fact that gradient-boosted tree models performed the best on the cleaned benchmark. Table 3 also indicates that LightGBM was ranked best overall in the summary that is saved, but XGBoost was almost the same in terms of quality of its detection at lower latency and smaller model size. Gradient Boosting and Random Forest were also very good, albeit with a much larger storage footprint. Comparatively, Decision Tree obtained slightly worse predictive performance, but was the fastest and smallest classical baseline, with the unsupervised Autoencoder and Isolation Forest evidently performing worse on the full binary task.

4.3 Robustness Under Rare-Attack Conditions

Table 4 documents the strength of the models of choice with the increasingly rare attack conditions. XGBoost had the largest F1-

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Table 4: Rare-attack robustness under 1%, 5%, and 10% attack ratios

Scenario	Model	F1-score	Avg. Precision	FPR	Latency (ms/sample)
1% attack	LightGBM	0.827586	1.000000	0.004163	0.006772
1% attack	XGBoost	0.831683	1.000000	0.004044	0.004844
1% attack	Gradient Boosting	0.823529	0.998867	0.004282	0.005087
1% attack	Decision Tree	0.721030	0.530872	0.007731	0.002362
1% attack	Feedforward NN	0.538462	0.961498	0.017127	0.081399
5% attack	LightGBM	0.961915	0.999753	0.004163	0.007003
5% attack	XGBoost	0.961832	0.999606	0.004044	0.006448
5% attack	Gradient Boosting	0.959739	0.999197	0.004282	0.007887
5% attack	Decision Tree	0.930380	0.839292	0.007731	0.003383
5% attack	Feedforward NN	0.858812	0.988416	0.017127	0.066433
10% attack	LightGBM	0.981608	0.999931	0.004163	0.013898
10% attack	XGBoost	0.981589	0.999882	0.004044	0.005572
10% attack	Gradient Boosting	0.980557	0.999710	0.004282	0.005323
10% attack	Decision Tree	0.965839	0.913375	0.007731	0.002165
10% attack	Feedforward NN	0.927897	0.995243	0.017127	0.065996

score of 0.831683, which LightGBM (0.827586) and Gradient Boosting (0.823529) followed closely, with nearly no improvement in the average precision and very low false positives. The 5 and 10 percent attack ratio offered further improvement of performance, and LightGBM and XGBoost are nearly identical, as shown in Table 4. Decision Tree and the feedforward neural network, in contrast, had more pronounced degradation in the rarest attack scenario. All in all, Table 4 substantiates the fact that the best boosted models are highly dependable even when malicious traffic is limited, which is particularly critical in the case of realistic situations of the deployment of IoT.

The reaction of the chosen models to the increase and decrease of the percentage of attacks in the assessment set is shown in Fig. 5. It is particularly helpful since it goes beyond the test-set accuracy and discusses whether the models are also reliable in more realistic and imbalanced circumstances of security. The enhanced classical models, especially LightGBM, XGBoost, and Gradient Boosting, retain almost perfect average precision among all three attack ratios which shows highly good stability. Weaker models, including Decision Tree, Autoencoder, and Isolation Forest, on the contrary,

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

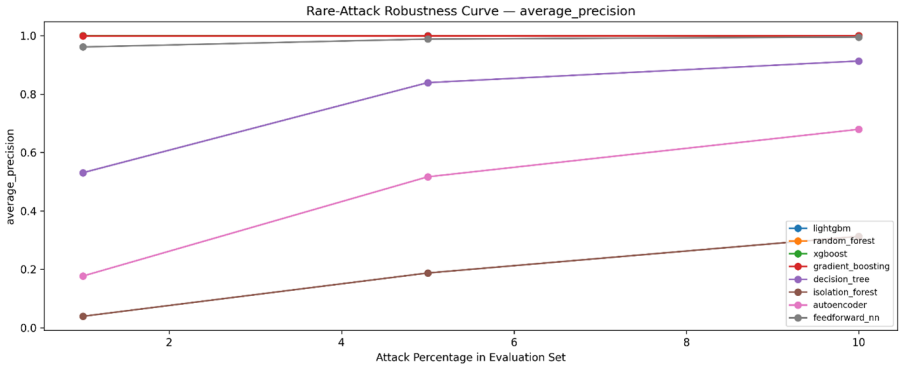

Figure 5: Rare-Attack Robustness Under 1%, 5%, and 10% Attack Ratios

enhance with the increase in the attack ratio, but they are still significantly inferior to the best methods. The feedforward neural network works quite well, but even in the rarest attack case, it is not yet as effective as the best boosted models. In general, the diagram indicates that the best supervised models are not just precise on the standard test set, but also resilient in the case when malicious traffic is sparse, which is a very important condition to apply anomaly detection to the IoT in practice.

4.4 Multiclass Attack-Type Classification

Table 5: Final multiclass attack-type classification comparison

Model	Accuracy	Macro F1	Weighted F1	Macro Precision	Macro Recall	Latency (ms/sample)	Size (MB)
XGBoost Multiclass	0.988765	0.965683	0.988918	0.959053	0.974029	0.027188	3.738
LightGBM Multiclass	0.988056	0.963943	0.988221	0.957207	0.972499	0.089856	4.988
Decision Tree Multiclass	0.986087	0.954872	0.986386	0.947178	0.965322	0.001833	0.251
FFNN Multiclass	0.954876	0.890223	0.954346	0.904581	0.884326	0.057752	0.240

Table 5 gives a summary of the final multiclass attack type classification. The best total model was XGBoost Multiclass that had 0.988765 accuracy and 0.965683 macro F1, and it was marginally better than LightGBM Multiclass. On Table 5, it was seen that Decision Tree Multiclass was significantly faster and smaller, but still

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

competitive with a macro F1-score of 0.954872. Multiclass feedforward neural network was trailing behind the boosted tree models particularly on macro-level. On the whole, Table 5 demonstrates that the gradient-boosted tree models become the best option, as well, in the proposed framework, to classify.

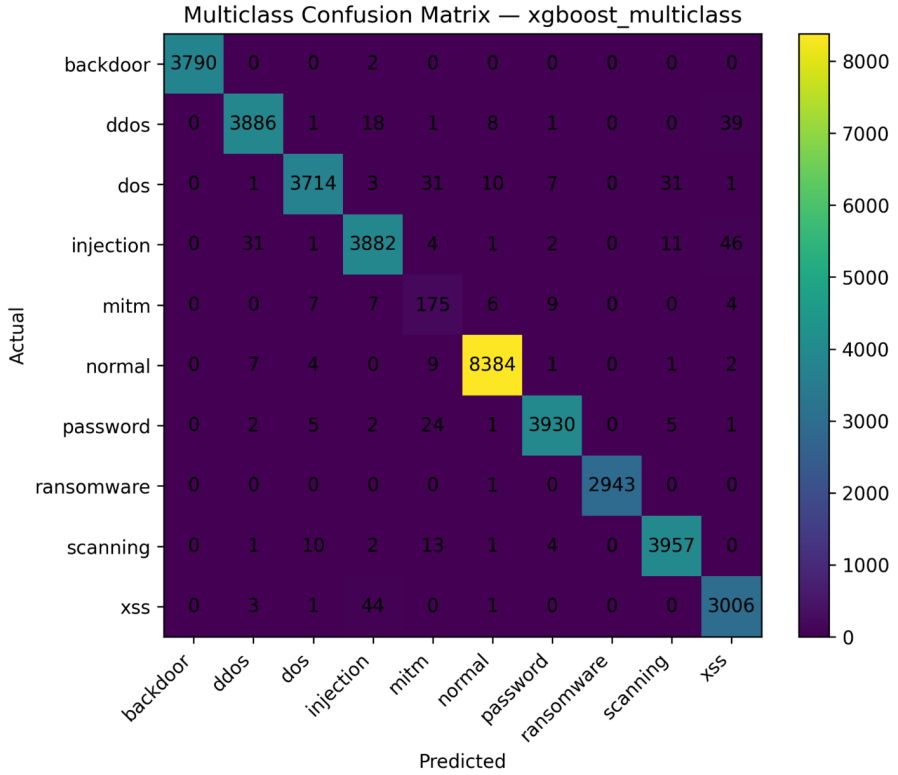

Figure 6: Confusion matrix of XGBoost on multiclass classification

Fig. 6 shows the confusion matrix of the most successful multiclass classifier, i.e., the XGBoost-based attack-type model. The values are mostly concentrated on the main diagonal, and this means that the classifier is properly classifying the majority of the samples in each category of attack. Classes like normal, password, ransomware, scanning and backdoor are identified with a high degree of

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

accuracy with only a few misclassifications. A little confusion is still apparent between some closely related types of attacks, specifically between injection and xss, and to a lesser degree between DDOS, dos, and scanning, which is natural since these attacks may overlap in traffic features. The smaller mitm class is also less easy to classify perfectly, probably due to the much fewer training instances than the other categories. On the whole, the figure demonstrates that the proposed multiclass framework is not only very accurate on a global scale but also can separate most of the attack classes with high accuracy.

4.5 Stability and Comparative Reliability of the Top Models

Table 6: Robust performance comparison of LightGBM, XGBoost, and Gradient Boosting models

Model	F1-score (mean ± std)	ROC-AUC (mean ± std)	Avg. Precision (mean ± std)	FPR (mean ± std)	Latency ms/sample (mean ± std)
LightGBM	0.999202 ± 0.000045	0.999961 ± 0.000029	0.999988 ± 0.000010	0.003663 ± 0.000486	0.007890 ± 0.002225
XGBoost	0.999114 ± 0.000071	0.999969 ± 0.000011	0.999991 ± 0.000003	0.003996 ± 0.000633	0.004166 ± 0.000071
Gradient Boosting	0.998960 ± 0.000120	0.999883 ± 0.000054	0.999965 ± 0.000017	0.004995 ± 0.000757	0.004057 ± 0.000125

Table 6 robust analysis further builds on the confidence that the leading boosted models. LightGBM had the best average F1-score of 0.999202 ± 0.000045, and XGBoost was almost equally good in terms of latency. Notably, the paired significance tests failed to indicate a statistically significant difference between the two models at the 0.05 mark, which leads to the explanation that the quality of the predictive aspect of the two models is practically equal. These findings warrant moving the end-model-selection decision-making process out of pure accuracy to deployment considerations, including latency and model size.

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

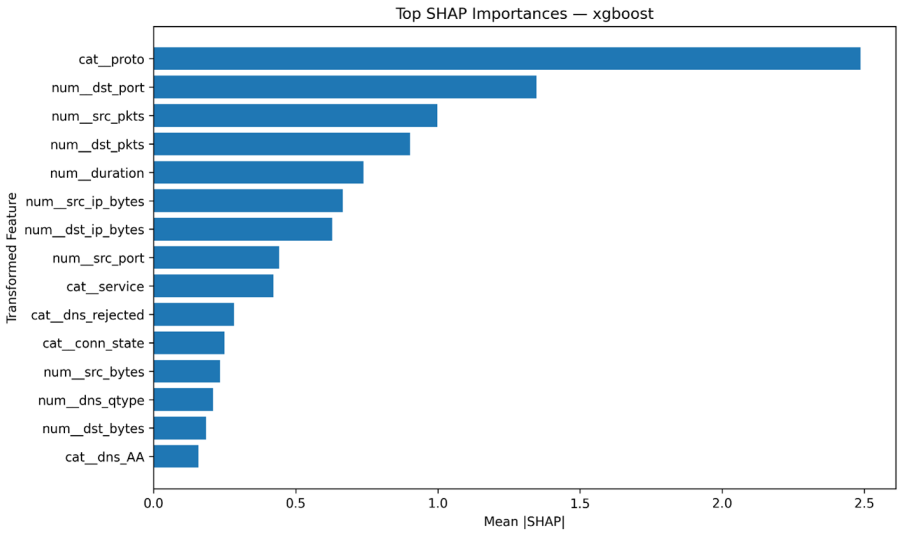

Figure 7: Top SHAP Importances of the XGBoost Model

4.6 Explainability and Feature-Level Interpretation

The importance of features in the XGBoost classifier that produced the best performance is analyzed using SHAP as illustrated in Fig. 7. The number measures the average contribution of each transformed input variable to the final prediction of the model and means SHAP values are higher in magnitude, corresponding to a more significant influence. The most significant factor is the cat_ proto, then num_dst_port, num_src-pkts, num__duration, and num_duration meaning that the protocol type and traffic-flow behavior have the most significant role in differentiating between normal and anomalous activity. Other features such as those related to the bytes and those related to DNS are also important but with less overall significance. Altogether, Fig. 8 reinforces the readability of the suggested structure as it demonstrates that the model is propelled by meaningful communication and traffic features instead of meaningless identifiers or memorized fields.

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

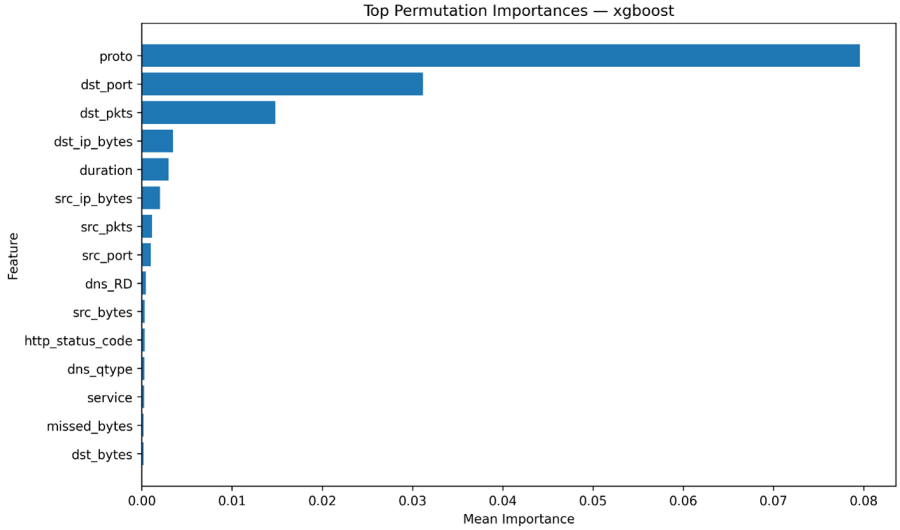

Figure 8: Permutation Importance of the Best Tree-Based Model

Fig. 8 indicates the importance of the original input features of the XGBoost classifier in permutation order. This analysis quantifies the reduction in the performance of the model when each model feature is perturbed randomly, with larger values corresponding to higher practical significance. The most impactful variable is obviously the feature proto, which is then succeeded by dst_port and dst_pkts, which means that the type of protocol and the behavior of the destination side of the traffic are the key elements to the detection of the anomaly. There are other variables like dst_ip_bytes, duration and sour ip bytes which also play a role at a significantly lower level. On the other hand, service, missed_bytes and dst_bytes are the features that make a very small contribution to the final decision process. On balance, Fig. 8 is in favor of the finding that the most successful model is mainly motivated by a small group of significant traffic characteristics.

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

4.7 Feature Ablation and Edge Deployment Suitability

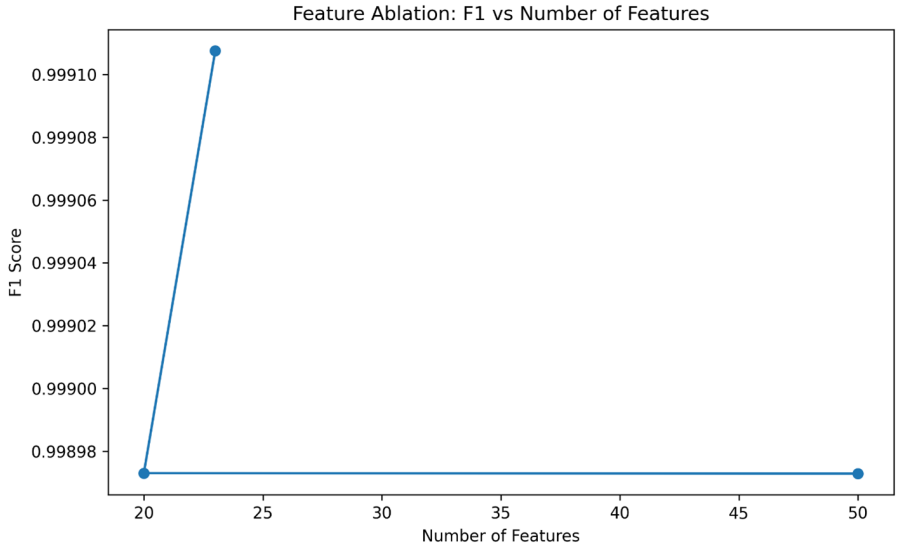

Figure 9: Effect of Feature Reduction on Detection Performance

Fig. 9 illustrates how reduction in features affects the performance of the detected performance of the chosen compact-model baseline. The most significant finding is that the F1-score of the model is very high in all of the tested sets of features, which means that the model is not very reliant on a big number of input variables. The difference between the best performance with the entire compact feature set and reduced subsets is very slight, despite the fact that the former provides the best performance. This finding indicates that with a reduced feature space, almost identical anomaly-detection power can be achieved with the possibility of reduced computational expense. Regarding edge computing, Fig. 9 is a valuable piece of evidence that it is possible to deploy efficiently without a significant reduction in predictive quality.

Fig. 10 indicates practical deployment trade-off between the primary binary detection models by directly comparing F1-score

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

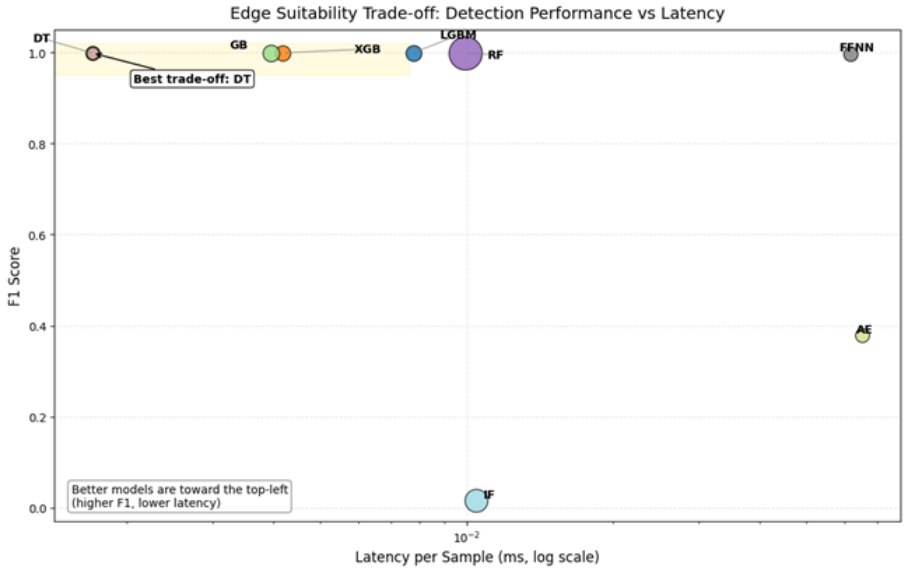

Figure 10: Edge Suitability Trade-Off Between Detection Performance and Latency

and per-sample inference latency. The models that are in the upper-left part of the plot are the best since they have better detection and lesser computational time. The number means that boosted models and, in particular, the XGBoost, Gradient Boosting, and LightGBM have almost perfect F1-scores, but their latency remains greater than that of the Decision Tree model. The overall edge trade-off seems to favor Decision Tree as it maintains extremely high quality of detection, and yet is the fastest and lightest of the top models. Conversely, feedforward neural network is precise, but much slower, whereas Autoencoder and Isolation Forest are not as efficient as alternative paradigms of anomaly-detection. On the whole, it is evident that predictive performance alone cannot be used to select the model, and factors like inference efficiency and practicality in real-time also need to be taken into consideration when deploying edges (Fig. 10).

Table 7 presents the edge-based ranking of the key detection

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Table 7: Edge deployment ranking

Rank	Model	F1-score	Latency (ms/sample)	Size (MB)	Edge Score	Estimated Comm. Saving (%)
1	Decision Tree	0.998199	0.001703	0.057	0.999537	88.927
2	XGBoost	0.999108	0.004192	0.425	0.973378	88.930
3	Gradient Boosting	0.999040	0.003966	0.657	0.963604	88.930
4	LightGBM	0.999108	0.007770	0.505	0.955580	88.929
5	Feedforward NN	0.996214	0.061630	0.237	0.753864	88.912

models by integrating the predictive performance, latency and model size into one deployment outlook. Even though LightGBM and XGBoost showed the best overall binary classification, Table 7 finds Decision Tree as the best edge model due to its ability to combine very high detection quality and the lowest latency (0.001703 ms/sample/sample) with the minimal memory footprint (0.057 MB). XGBoost was second-ranked and represents the most reasonable practical choice when strength and efficiency of prediction are compared. Gradient Boosting and LightGBM also proved to be very competitive, and feedforward neural network, although with high accuracy, was significantly slower. Generally, Table 7 indicates that computational efficiency and predictive accuracy should be taken into consideration when choosing a model to deploy edges.

5 Conclusion

The study introduced an edge-based IoT anomaly detection system based on the TON_IoT network data and analyzed both predictive- and deployment-based. The research was aimed at both comparing models and creating a more extensive evaluation framework that considers the leakage-aware data preparation, dual preprocessing strategies, binary anomaly detection, rare-attack robustness testing, multiclass attack-type classification, explainability, repeated-seed robustness analysis, feature ablation, and edge-oriented deployment ranking. By doing so, the work fulfilled the practical requirement of IoT intrusion detection systems that are both accurate, but also interpretable, stable, and can run on resource-constrained edge environments. The performance of the gradient-

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

boosted tree models on the cleaned TON IoT benchmark revealed that the overall performance of the models was the strongest. LightGBM and XGBoost were the highest scorers in the binary environment with 0.999108 F1-score, and XGBoost also was very competitive in terms of latency and model size. The best boosted models were found to be very reliable even when the prevalence of attacks was lowered to 1 per cent as evidenced by the rare-attack conditions, which has validated their application to more realistic deployment situations. XGBoost performed the best in the multiclass scenario, which proves that the framework proposed can be used to not only identify the anomalies but also sub-type attacks. The explainability findings also indicated that the best models were mostly based on meaningful protocol and traffic-behavior characteristics as opposed to leakage prone identifiers, which reinforces the credibility of the framework. Lastly, the edge-based analysis revealed that despite better predictive capability, boosted models, Decision Tree offered the most efficient trade-off with respect to deployment due to its extremely low latency and low memory footprint. Overall, the results show that effective and realistic IoT anomaly detection cannot be achieved without a combination of high benchmark accuracy and practicality. The ability to survive under infrequent attacks, interpretability, statistical stability, and the ability to deploy a model at scale all affect the appropriateness of a model to real edge-assisted security environments. This combination of these aspects into a single approach gives the present study significance as it results in a more realistic framework of evaluation of IoT intrusion detection research.

This study also possesses several limitations which stimulate further studies. To begin with, the experiments were done with only one benchmark dataset and additional validation of the experiments with other IoT and IIoT datasets would enhance the generalization of the findings. Second, the framework could not quantify the operational cost by real edge-device deployment and power-consumption measures even though latency was considered, the model size, and estimated communication savings. Future efforts should involve

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

real-edged-device deployment and power-consumption measures to further quantify the operational cost. Third, the existing research compared the standard supervised, neural, and anomaly-based models, although future studies may elaborate the framework to federate learning and graph-based intrusion detection, as well as extended lightweight deep architecture. Fourth, concept drift and online adjustment in the face of changing traffic conditions could be incorporated into the rare-attack analysis. Lastly, the stage of explainability can be prolonged with the help of analyst-oriented validation designed to check whether the produced explanations enhance the levels of trust and decision support in the working conditions. Combined, these guidelines can assist in transforming the IoT intrusion detection out of the powerful offline standards into more adaptable, more reliable and implementation-ready edge security frameworks.

References

[1] E. Gyamfi and A. Jurcut, "Intrusion detection in internet of things systems: a review on design approaches leveraging multi-access edge computing, machine learning, and datasets," *Sensors*, vol. 22, no. 10, p. 3744, 2022.

[2] T. M. Booij, I. Chiscop, E. Meeuwissen, N. Moustafa, and F. T. Den Hartog, "Ton_iot: The role of heterogeneity and the need for standardization of features and attack types in iot network intrusion data sets," *IEEE Internet of Things Journal*, vol. 9, no. 1, pp. 485–496, 2021.

[3] X. Yu, X. Yang, Q. Tan, C. Shan, and Z. Lv, "An edge computing based anomaly detection method in iot industrial sustainability," *Applied Soft Computing*, vol. 128, p. 109486, 2022.

[4] Y. Meidan, M. Bohadana, Y. Mathov, Y. Mirsky, A. Shabtai, D. Breitenbacher, and Y. Elovici, "N-baiot—network-based

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

detection of iot botnet attacks using deep autoencoders," *IEEE Pervasive Computing*, vol. 17, no. 3, pp. 12–22, 2018.

[5] A. A. Diro and N. Chilamkurti, "Distributed attack detection scheme using deep learning approach for internet of things," *Future Generation Computer Systems*, vol. 82, pp. 761–768, 2018.

[6] R. Doshi, N. Apthorpe, and N. Feamster, "Machine learning ddos detection for consumer internet of things devices," in *2018 IEEE security and privacy workshops (SPW)*. IEEE, 2018, pp. 29–35.

[7] 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," *Future Generation Computer Systems*, vol. 100, pp. 779–796, 2019.

[8] N. Moustafa, "New generations of internet of things datasets for cybersecurity applications based machine learning: Ton_iot datasets," in *Proceedings of the eResearch Australasia Conference, Brisbane, Australia*, 2019, pp. 21–25.

[9] A. Alsaedi, N. Moustafa, Z. Tari, A. Mahmood, and A. Anwar, "Ton_iot telemetry dataset: A new generation dataset of iot and iiot for data-driven intrusion detection systems," *Ieee Access*, vol. 8, pp. 165 130–165 150, 2020.

[10] N. Moustafa, M. Keshky, E. Debiez, and H. Janicke, "Federated ton_iot windows datasets for evaluating ai-based security applications," in *2020 IEEE 19th international conference on trust, security and privacy in computing and communications (TrustCom)*. IEEE, 2020, pp. 848–855.

[11] N. Moustafa, M. Ahmed, and S. Ahmed, "Data analytics-enabled intrusion detection: Evaluations of ton_iot linux

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

datasets," in *2020 IEEE 19th international conference on trust, security and privacy in computing and communications (TrustCom)*. IEEE, 2020, pp. 727–735.

[12] M. A. Ferrag, L. Maglaras, S. Moschoyiannis, and H. Janicke, "Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study," *Journal of Information Security and Applications*, vol. 50, p. 102419, 2020.

[13] H. Hindy, E. Bayne, M. Bures, R. Atkinson, C. Tachtatzis, and X. Bellekens, "Machine learning based iot intrusion detection system: An mqtt case study (mqtt-iot-ids2020 dataset)," in *International networking conference*. Springer, 2020, pp. 73–84.

[14] P. Nimbalkar and D. Kshirsagar, "Feature selection for intrusion detection system in internet-of-things (iot)," *ICT Express*, vol. 7, no. 2, pp. 177–181, 2021.

[15] M. A. Ferrag, O. Friha, L. Maglaras, H. Janicke, and L. Shu, "Federated deep learning for cyber security in the internet of things: Concepts, applications, and experimental analysis," *IEEe Access*, vol. 9, pp. 138 509–138 542, 2021.

[16] T. D. Nguyen, P. Rieger, M. Miettinen, A.-R. Sadeghi *et al.*, "Poisoning attacks on federated learning-based iot intrusion detection system," in *Proc. workshop decentralized IoT syst. secur.(DISS)*, vol. 79, 2020, pp. 1–7.

[17] E. Özer, M. Iskefiyeli, and J. Azimjonov, "Toward lightweight intrusion detection systems using the optimal and efficient feature pairs of the bot-iot 2018 dataset," *International Journal of Distributed Sensor Networks*, vol. 17, no. 10, p. 15501477211052202, 2021.

[18] W. W. Lo, S. Layeghy, M. Sarhan, M. Gallagher, and M. Portmann, "E-graphsage: A graph neural network based intrusion

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

detection system for iot," in *NOMS 2022-2022 IEEE/iFIP network operations and management symposium*. IEEE, 2022, pp. 1–9.

[19] S. Arisdakessian, O. A. Wahab, A. Mourad, H. Otrok, and M. Guizani, "A survey on iot intrusion detection: Federated learning, game theory, social psychology, and explainable ai as future directions," *IEEE Internet of Things Journal*, vol. 10, no. 5, pp. 4059–4092, 2022.

[20] P. Spadaccino and F. Cuomo, "Intrusion detection systems for iot: opportunities and challenges offered by edge computing and machine learning," *arXiv preprint arXiv:2012.01174*, 2020.

[21] S. Tsimenidis, T. Lagkas, and K. Rantos, "Deep learning in iot intrusion detection," *Journal of network and systems management*, vol. 30, no. 1, p. 8, 2022.

[22] B. I. Farhan and A. D. Jasim, "Survey of intrusion detection using deep learning in the internetof things," *Iraqi Journal For Computer Science and Mathematics*, vol. 3, no. 1, p. 9, 2022.

[23] S. Baniasadi, O. Rostami, D. Martín, and M. Kaveh, "A novel deep supervised learning-based approach for intrusion detection in iot systems," *Sensors*, vol. 22, no. 12, p. 4459, 2022.

[24] O. A. Alzubi, J. A. Alzubi, M. Alazab, A. Alrabea, A. Awajan, and I. Qiqieh, "Optimized machine learning-based intrusion detection system for fog and edge computing environment," *Electronics*, vol. 11, no. 19, p. 3007, 2022.

[25] M. M. Alani and A. Miri, "Towards an explainable universal feature set for iot intrusion detection," *Sensors*, vol. 22, no. 15, p. 5690, 2022.

[26] A. Khudhu and K. Samsudin, "Iot intrusion detection using autoencoder and machine learning techniques," *J. Comp. Sci*, vol. 18, no. 10, pp. 904–912, 2022.

Received: October 27, 2022 Published: December 21, 2022

International Journal of Applied Mathematics
Volume 35 No. 6, 2022
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

[27] H. Bangui and B. Buhnova, "Lightweight intrusion detection for edge computing networks using deep forest and bio-inspired algorithms," *Computers and Electrical Engineering*, vol. 100, p. 107901, 2022.

[28] I. Tareq, B. M. Elbagoury, S. El-Regaily, and E.-S. M. El-Horbaty, "Analysis of ton-iot, unw-nb15, and edge-iiot datasets using dl in cybersecurity for iot," *Applied Sciences*, vol. 12, no. 19, p. 9572, 2022.

[29] S. Neupane, J. Ables, W. Anderson, S. Mittal, S. Rahimi, I. Banicescu, and M. Seale, "Explainable intrusion detection systems (x-ids): A survey of current methods, challenges, and opportunities," *IEEE Access*, vol. 10, pp. 112 392–112 415, 2022.

[30] V. Chang, L. Golightly, P. Modesti, Q. A. Xu, L. M. T. Doan, K. Hall, S. Boddu, and A. Kobusińska, "A survey on intrusion detection systems for fog and cloud computing," *Future Internet*, vol. 14, no. 3, p. 89, 2022.

[31] Y. Mirsky, T. Doitshman, Y. Elovici, and A. Shabtai, "Kitsune: an ensemble of autoencoders for online network intrusion detection," *arXiv preprint arXiv:1802.09089*, 2018.

Received: October 27, 2022 Published: December 21, 2022