Evaluation of XAI algorithms in IoT Traffic Anomaly Detection

Uyen Do
VNUHCM- UIT
Ho Chi Minh, VietNam
uyendtp.15@grad.uit.edu.vn

Laura Lahesoo
TUM
Munich, Germany
laura.lahesoo@tum.de

Rodrigo Matos Carnier
NII
Tokyo, Japan
rodrigo_carnier@nii.ac.jp

Kensuke Fukuda
NII/Sokendai
Tokyo, Japan
kensuke@nii.ac.jp

***Abstract*—Anomaly detection in network traffic, both in general computer networks and specifically in Internet of Things (IoT) networks, plays a crucial role in ensuring computer network security. Over the years, numerous machine learning and deep learning-based anomaly detection tools have been proposed, exhibiting high accuracy in identifying anomalous behavior. However, a significant challenge arises with most machine learning and deep learning algorithms, as they are often considered black-box models that lack interpretability. Consequently, explaining the reasons behind certain network behaviors being labeled as anomalous becomes a difficult task. To overcome this issue, we evaluate the combination of anomaly detectors and eXplainable Artificial Intelligence (XAI) algorithms in IoT traffic anomaly detection. Our research results demonstrate that XAI algorithms can consistently identify the most impactful network features of security anomalies. More specifically, (1) SHAP algorithm is the most robust and reliable in the four tested XAI algorithms for four types of supervised/unsupervised anomaly detection models, independent of two datasets including different anomalies. (2) Image-based XAI algorithms are not suitable for explainability of network anomaly detection.**

***Index Terms*—XAI, IoT, anomaly detection**

I. Introduction

There has been a notable surge in the frequency and severity of attacks directed at IoT networks and devices, causing significant financial loss [17], [19]. Security methods are not keeping up with specific vulnerabilities of IoT networks, which are increasing fast together with the number of Internet-connected IoT devices. Another concern lies in how IoT devices affect people's lives directly, such as sensors in medical equipment and control systems of self-driving car, and cannot afford to fail due to operational anomalies. Therefore, besides security enhancement, anomaly detection (AD) and its mitigation are primary concerns in IoT systems.

Many anomaly detectors are developed and integrated with IoT network devices. These anomaly detectors range from simple algorithms, such as rule-based detection, to complicated algorithms, such as machine-learning behavioral analysis, peer group analysis, and deep learning to improve accuracy in anomaly classification.

Most anomaly detectors do not provide network administrators or end-users with a direct and clear explanation for the decision to classify network traffic as benign or anomalous. There are two exceptions: rule-based anomaly detectors, which allow the users to set up their own rules for classification, and detectors using the decision tree algorithm, which explains the importance of the features used in the prediction through the leaves themselves. However, more insight is needed for advanced detectors that have begun to emerge recently, making use of black-box, more complex deep learning models, such as CNN, DNN, and GNN.

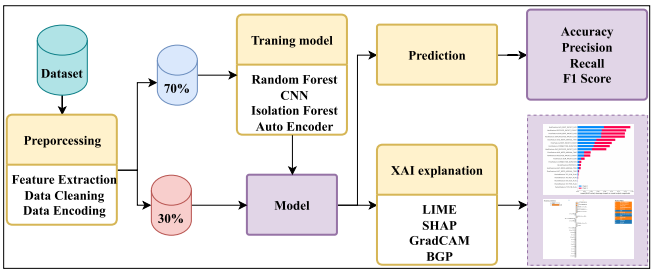

Fig. 1. Pipeline of anomaly detection and XAI-based explanation of results

To overcome this black-box approach, XAI algorithms provide users with interpretable and understandable explanations for predictions made by AI models. However, it is important to evaluate the explanatory power of these algorithms, particularly with regard to their applicability in detecting anomalies in network traffic and IoT traffic.

Our research focuses on evaluating XAI algorithms in AD for IoT traffic, aiming to provide transparent and interpretable detection results. With a focus on explainability, our work bridges the gap between high-performing models and comprehensible decision-making, enhancing the overall reliability and applicability of AD in real-world scenarios. The overview of our processing pipeline is shown in Fig. 1 (The details are discussed in Sec. III).

The contribution of the paper is as follows:

1) We evaluate the explainability of four different XAI algorithms applied to four supervised/unsupervised AD models.
2) We demonstrate that SHAP algorithm is the most robust and reliable among them with two IoT traffic datasets that include different anomalies. In contrast, image-based XAI algorithms output different explanations. We also show that XAI algorithms optimized to anomaly detector models worked in a reasonable time.

II. Related work

With the advancement of AI, deep learning models based on neural networks, which know as black-box models became popular but posed a significant challenge in terms of interpretation. XAI offers as a remedy for addressing the issue of explanation in these black box models.

XAI can be divided into global explanation and local explanation [7]. Global explanation explains the factors that influence the model's results, determining which role a feature plays in the model's predicted outcomes. In contrast, local explanation considers only one input data point and identifies the factors that influence its prediction process. Another classification of XAI algorithms is based on the methods used to provide explanations. One approach is to explain by simplification, turning the original model into interpretable models. The most popular method in this approach is the Local Interpretable Model-Agnostic Explanations (LIME) [12], which uses a regression model to account for a neighborhood dataset generated from the data point under consideration. Another class of methods, such as SHApley Additive exPlanations (SHAP) [10] and Saliency [3], focus on explaining the relevance of features. They use scores to evaluate the contribution of each feature to the prediction outcome. The visualization methods like Gradient-weighted Class Activation Mapping (Grad-CAM) [14], Guided Backpropagation (GBP) [18] are commonly employed to interpret models using image datasets, highlighting influential points affecting predictions and enabling an intuitive understanding of the decision-making process. Table II lists a summary of four XAI algorithms.

XAI algorithms are used for interpretation in many different fields. For AD explanation, two XAI algorithms (LIME and SHAP) are applied to detect malicious domains in DNS queries [1]. These two XAI algorithms are also used to interpret LSTM models in the ML crypto miner detector [6]. SHAP is used to explain the XGBoost model in the network intruder detection [2], AE model in DDOS attack detection [5], sensor behavior [4], and traffic classification of mobile IoT devices model [11], selected importance feature for ANN AD model [13], enhance trust of IDS for IoT network [16].

While these studies have applied XAI algorithms to explain various algorithms in different domains, there has been a notable lack of evaluation regarding their explainability in the context of network anomaly detection. Our research aims to fill this void by conducting a thorough assessment of the explanatory capabilities of four XAI algorithms, representing distinct explanatory methods in four supervised and unsupervised network anomaly detection algorithms. Through this investigation, we seek to shed light on the effectiveness and interpretability of these XAI techniques in the critical task of network anomaly detection.

III. Methodology

Fig. 1 illustrates our AD and interpretation procedure. We begin with two input datasets in the form of pcap network data extraction files. These datasets undergo a preprocessing phase that involves feature extraction, resulting in the collection of 25 features. Additionally, we conduct data cleaning to handle error data points and encode the data appropriately. Subsequently, the processed data is divided into two sets: a training set and a test set, with a ratio of 70/30. The AD models use the training set for training and evaluate the model accuracy by employing the test set. This evaluation yields important parameters related to the performance of AD including accuracy, precision, recall, and F1 score. Simultaneously, we use the XAI algorithms we have chosen, the AD model, and the test set for the interpretation process. This enables us to obtain explanations for prediction results produced by the models. This pipeline processing is implemented and evaluated on SIURU [9], a framework for IoT-traffic AD.

A. Dataset

We conduct an assessment of two distinct datasets:

MQTTset [20]: It specifically focuses on MQTT communications and provides a comprehensive collection of both benign traffic patterns and deliberate attacks aimed at the targeted MQTT network. The dataset encompasses various attack techniques, including Flooding, Denial of Service, MQTT Publish Flood, SlowITe, Malformed Data, and Brute Force Authentication. To conduct our analysis, we extracted a total of 70,983 benign traffic packets and 130,223 MalariaDoS traffic packets from this MQTTset dataset.

CIC IDS 2017 [15]: It consists of benign network traffic as well as many recent and common attacks from replications of real-world scenarios. Its data format is raw pcap files (used in this work) and pre-processed flow feature files generated with CICFlowMeter. The dataset includes benign traffic based on HTTP, HTTPS, FTP, SSH, and email protocols. Additionally, it incorporates various deployed attacks, such as Brute Force FTP, Brute Force SSH, DoS, Heartbleed, Web Attack, Infiltration, Botnet, and DDoS. For this dataset, we extracted 120,302 benign traffic packets and 122,094 DDoS traffic packets captured on Thursday, July 6, 2017.

We extracted raw packet data from the pcap files and obtained 25 distinct empirical features. These features are categorized into three primary groups: Packet Features (denoted P), Host Features (denoted H), and Flow Features (denoted F). The Packet Features encompass characteristics that are specific to each individual packet. The Host Features capture attributes that are associated with each host in the network. Finally, the Flow Features involve parameters that are calculated for individual network flows, characterized by the five-tuple (source IP address, source port, destination IP address, destination port, protocol). Table I provides a detailed breakdown of the 25 extracted features utilized in our analysis.

B. Anomaly Detection Methods

In our study, we conducted a comprehensive evaluation of four model classifiers, encompassing both supervised and unsupervised learning algorithms. Here, we focus on the binary classification model (i.e., the output is benign or anomalous).

Supervised learning: We opted for Random Forest (RF) and Convolutional Neural Network (CNN) as our ML methods for AD. RF is a popular ensemble learning algorithm that combines multiple decision trees to make accurate predictions. CNN is a deep learning architecture well-suited for analyzing structured data such as images or sequential data.

Unsupervised learning: We employed Isolation Forest (IF) and Auto Encoder (AE). IF is a tree-based AD algorithm that isolates anomalies by exploiting their distinctive attributes. AE is a neural network architecture used for AD based on the reconstruction error. We evaluate the effectiveness of these models in AD as well as the applicability of XAI algorithms to different classification algorithms.

For supervised AD, we employed the entire dataset, including both benign and anomalous instances, to train and test the classifiers. In contrast, for unsupervised AD, we exclusively utilized benign traffic data for training the IF and AE models. By focusing exclusively on benign traffic data during training, the unsupervised classifiers can capture the underlying structure of benign instances, enabling them to identify deviations or anomalies during the testing phase.

The evaluation was conducted on a M1 Mac with 16GB of RAM. By evaluating these algorithms on diverse datasets and utilizing a reliable computing environment, we aimed to gain insights into their effectiveness and suitability for AD tasks.

TABLE I
FEATURE LIST

Index	Feature	Description
#1	P[Protocol]	packet protocol (TCP/UDP)
#2	P[IP Header Size]	IP header size (bytes)
#3	P[TCP CWR Flag]	CWR flag in TCP packet
#4	P[TCP ECE Flag]	ECE flag in TCP packet
#5	P[TCP URG Flag]	URG flag in TCP packet
#6	P[TCP ACK Flag]	ACK flag in TCP packet
#7	P[TCP PSH Flag]	PSH flag in TCP packet
#8	P[TCP RST Flag]	RST flag in TCP packet
#9	P[TCP SYN Flag]	SYN flag in TCP packet
#10	P[TCP FIN Flag]	FIN flag in TCP packet
#11	H[Received Packet Count]	# packets from same IP
#12	H[Sum Rcvd Hdr Size]	sum hdr sizes same IP (bytes)
#13	H[Avg Received Hdr Size]	avg hdr size (bytes)
#14	H[Sent Packet Count]	# packets to IP address
#15	H[Sum Sent Hdr Size]	sum hdr sizes (bytes)
#16	H[Avg Sent Hdr Size]	avg hdr size (bytes)
#17	H[Last InterArrival Time]	time since last packet (μs)
#18	H[Avg InterArrival Time]	avg time btw packets(μs)
#19	H[Connection Duration]	time since first packet (μs)
#20	F[Received Packet Count]	# packets in flow
#21	F[Sum Header Size]	sum header sizes (bytes)
#22	F[Avg Header Size]	avg header size (bytes)
#23	F[Last InterArrival Time]	time since last packet (μs)
#24	F[Avg InterArrival Time]	avg time btw packets (μs)
#25	F[Connection Duration]	time since first packet (μs)

C. Explainable AI

We conducted an extensive assessment of the explainability of AD algorithms in network traffic, employing four distinct XAI techniques: LIME, SHAP, Grad-CAM, and GBP. These selected algorithms include both local and global explanations, catering to various data formats such as tabular, image, and time-series data. To facilitate a comprehensive comparison of their functionalities, Table II presents an overview of the operational mechanisms and the types of supported data.

TABLE II
XAI SURVEY

Algorithm	Mechanism	Dataset type	Explanation
LIME [12]	Explanation by Simplification, Perturbation-based	Tabular Image Text	Local
SHAP [10]	Feature relevance explanation, Perturbation-based	Tabular Image	Global Local
Grad-CAM [14]	Visual explanations, Gradient-based, Activation map	Image	Local
GBP [18]	Visual explanations, Gradient-based, Gradident	Image	Local

LIME: It is particularly effective in providing local explanations, explaining the model's decision-making process at the individual instance level.

In our study, we leveraged LIME to explain the output of RF, CNN, IF and AE models operating on a tabular dataset. Because LIME is a local explanatory model, to obtain the complete list of globally important features, we randomly selected 1000 points in the dataset and applied LIME to each point, calculating the results average to produce a complete list of important features. However, LIME faces challenges in interpreting IF and AE results due to its reliance on algorithms with probability scores. To solve this problem, for IF, we convert anomaly scores to probability scores, allowing LIME to be used effectively for interpretation, similar to Ref. [8]. AE makes calculating predicted probabilities difficult. Instead, we can only generate probability scores [1,0] and [0,1] for the labels: benign and anomalous.

SHAP: It delivers comprehensive interpretability for machine learning models. Its foundation lies in cooperative game theory, specifically Shapley values, which effectively quantify the contribution of each feature during the prediction process. SHAP provides a diverse and optimized range of explainers for different algorithms and dataset types. When working with tree-based models such as RFs, IFs, SHAP provides Tree Explainer optimized for tree-based models. The Tree Explainer exploits the characteristics of tree structure, helping to increase the ability to calculate and process complex tree models. We used Tree Explainer to obtain insightful explanations for our RFs and IFs, unveiling the importance of individual features in these models' decision-making processes. In the case of CNN, SHAP offers two types of optimized explainers: Deep Explainer and Gradient Explainer. The Deep Explainer uses an enhanced version of DeepLift to calculate SHAP value while the Gradient Explainer uses expected gradients. However, due to compatibility issues with TensorFlow lib, we focused our evaluation solely on Gradient Explainer. Furthermore, as our dataset was not in image format, we converted the input and output to extract the critical features. For the AE model, which is used in unsupervised learning, we relied on Kernel Explainer

provided by SHAP. Kernel Explainer is the core of SHAP, acting as a model agnostic when used to explain any algorithm output. The limitation of Kernel Explainer is computationally expensive, especially with complex models or data sets.

Grad-CAM and GBP: They are techniques commonly used for interpreting CNN with image datasets. GradCAM highlights important image regions contributing to predictions in CNN, while GBP visualizes input features with significant influence on the model's output. Since Grad-CAM and GBP are XAI algorithms designed for image datasets, we also performed input and output conversions. SHAP Gradient Explainer, SHAP Kernel Explainer, GradCAM, and GBP all function as local explainers, so we evaluated 1000 data points and calculated the mean values to detect feature importance.

In order to assess the explainability of the algorithms, we examined the top important features extracted from each algorithm and compared them. To determine them, we extracted the importance of features 100 times and calculated the mean value for each feature.

To evaluate the explanatory capabilities of the XAI algorithms, we conducted several comparisons. Firstly, we compared the consistency of interpretation results by examining the top important feature provided by various XAI algorithms. Within the same AD model, we used RF feature importance as the baseline and compared it with the top important features extracted from LIME and SHAP algorithms. This analysis allowed us to assess the consistency of interpretation results across different XAI methods.

Secondly, we compared the results of XAI algorithms across different AD models to understand the impact of features. Specifically, we analyzed the results on four prediction models with SHAP: RF, CNN, IF, and AE.

Lastly, we expanded our analysis by comparing the XAI results on two distinct datasets: MQTTset and CIC IDS 2017. This enabled us to evaluate the consistency and effectiveness of XAI methods across different data environments.

In terms of applicability and explanation to users, we considered various factors for comparison, including supported data type and the number of algorithms supported by each XAI method. By considering these factors, we aimed to provide insights into the suitability and practicality of different XAI methods for user-oriented explanations.

IV. Results

A. Anomaly detection performance

Table III summarises the AD performance of the models in the two datasets. It is evident that our supervised learning models exhibit exceptional accuracy (99% accuracy) in both datasets. However, the performance of the unsupervised learning algorithms, IF and AE, is relatively lower, with accuracies of 92% and 94% in the MQTTset dataset, and 85% and 82% in the CIC IDS 2017 dataset, respectively. The discrepancy in performance can be attributed to the higher complexity and diversity of traffic present in the CIC IDS 2017 dataset as compared to the MQTTset dataset. These results show that the learned models have enough performance for the AD and are ready for applying the XAI algorithms to them.

TABLE III
Performance of Anomaly Detection

AD	Dataset	Accurancy	Precision	Recall	F1 score
RF	MQTTset	0.999	0.999	0.999	0.999
CNN	MQTTset	0.996	0.996	0.996	0.996
IF	MQTTset	0.922	0.932	0.898	0.911
AE	MQTTset	0.946	0.961	0.924	0.938
RF	CIC2017	0.999	0.999	0.999	0.999
CNN	CIC2017	0.992	0.992	0.992	0.992
IF	CIC2017	0.855	0.864	0.856	0.854
AE	CIC2017	0.832	0.857	0.831	0.828

B. Explainability

Baseline behavior: We apply XAI algorithms to learned AD models. We denote the result of XAI (Y) for the model X as $R(X/Y)$, e.g. R(RF/LIME) is the result of LIME for the RF model. We use R(RF/RF) as the baseline of the comparison, because the learned RF model outputs the important features without extra XAI algorithms. Table IV lists the top six important features for each XAI algorithm applied to the learned models.

R(RF/RF) outputs connection duration related features (25, 19, 16, 23, 11, 12) as the important features for MQTT dataset. Considering the type of anomalies (MalariaDoS) in the dataset, we conclude that the results of RF are reasonable because this anomalous traffic has shorter connection duration time than that of benign traffic.

Also, the header size related features (11, 12) are important in CIC IDS 2017 data. It includes Slowloris attack in which malicious actors try to open numerous connections to the target web server and keep them open as long as possible, similar to benign traffic. To keep the connection, the attacker sends an incomplete request which does not include the terminating newline sequence. The attacker sends additional header lines periodically to keep the connection alive, but never sends the terminating newline sequence. This leads to the headers for each connection growing larger and larger as the attack progresses. As a result, the sum header size for each connection in a Slowloris attack can become much larger than the header size in a benign HTTP request.

In summary, the explainability of the baseline results is intuitive and reasonable.

Comparison with baseline: We first show that LIME and SHAP are consistent with RF for the RF model. The color in the table represents the top features in R(RF/RF) as the reference. We visually confirm that R(RF/LIME) and R(RF/SHAP) for the two datasets are mostly consistent with the baseline, R(RF/RF). Thus, the explainability of LIME and SHAP is enough for our problem.

Next, we check the explainability of the neural network model. We still observe the consistency of LIME R(CNN/LIME) and SHAP R(CNN/SHAP) in this case. However, R(CNN/GradCAM) and R(CNN/GBP) behave differently. A plausible reason of this is related to the type

TABLE IV
TOP SIX IMPORTANT FEATURES IN DIFFERENT DATASETS. TOP: MQTTSET DATASET. BOTTOM: CIC IDS 2017 DATASET

MQTT

Top feature	Random Forest			CNN				Isolation Forest		Auto Encoder	
	Random Forest	LIME	SHAP	LIME	SHAP	Grad CAM	GBP	LIME	SHAP	LIME	SHAP
1	#25	#19	#19	#9	#19	#10	#19	#24	#11	#14	#19
2	#19	#25	#25	#19	#12	#21	#21	#17	#12	#15	#25
3	#16	#18	#16	#20	#11	#20	#15	#23	#25	#11	#15
4	#23	#12	#12	#12	#20	#25	#14	#18	#15	#12	#14
5	#11	#11	#11	#10	#14	#19	#7	#11	#19	#21	#11
6	#12	#9	#18	#11	#21	#15	#20	#12	#14	#20	#12
Spearman's rho	1.0000	0.8046	0.9931	0.5581	0.6360	0.2212	0.4462	0.6555	0.9064	0.8066	0.8787

CIC IDS 2017

Top feature	Random Forest			CNN				Isolation Forest		Auto Encoder	
	Random Forest	LIME	SHAP	LIME	SHAP	Grad CAM	GBP	LIME	SHAP	LIME	SHAP
1	#12	#15	#12	#19	#19	#25	#12	#9	#11	#12	#11
2	#11	#12	#16	#13	#12	#20	#11	#10	#12	#11	#12
3	#19	#11	#11	#12	#11	#21	#18	#8	#15	#8	#14
4	#18	#16	#15	#20	#14	#23	#6	#7	#14	#1	#15
5	#16	#13	#19	#8	#15	#24	#7	#1	#23	#14	#24
6	#15	#14	#18	#11	#20	#1	#13	#23	#7	#15	#19
Spearman's rho	1.0000	0.8573	0.9888	0.6851	0.7491	0.2679	0.5047	0.1921	0.4553	0.4735	0.7429

of XAI algorithms. GradCAM and GBP are developed for neural network models but their main target is image processing/analysis. Therefore, when we attempted to use these algorithms with tabular data, their interpretability was limited. Furthermore, GradCAM is particularly effective when working with models containing multiple convolutional layers, as it utilizes gradients from the final convolutional layer. In contrast, our CNN model only incorporates one convolutional layer, which makes the interpretation results of GradCAM less efficient for our specific model. By manually checking the output of them, we find simple horizontal bands that correspond to the situation that a few features are highlighted over all the time steps. In other words, the time evolution of traffic features is too simple compared to real-world images.

Furthermore, we check the explainability of the unsupervised models. We see that SHAP outputs the consistent results as in $R(\text{IF}/\text{SHAP})$ and $R(\text{AE}/\text{SHAP})$, though $R(\text{IF}/\text{LIME})$ and $R(\text{AE}/\text{LIME})$ are different from the baseline. LIME's deviation is influenced by the bias in probability scores during the conversion of anomaly scores to probability scores on the IF algorithm and the generation of label-based probability scores on the AE algorithm.

Our visual analysis demonstrates the SHAP is robust against all the types of AD models. Also, we obtain consistent results in two different datasets though the important features themselves are different. LIME offers less algorithmic support, and less stable explanations compared to SHAP. This is due to LIME's local explanation, making it less effective than SHAP's global approach. Besides, forcing models such as IF and AE to use probability to apply LIME causes the interpretation results on these algorithms to have many bias. Thus, our results suggest that XAI algorithms, especially SHAP, are reliable and robust in the context of the network AD.

Quantitative analysis: We further quantitatively examine this similarity with Spearmann's rank correlation coefficients. Spearmann's rank correlation coefficient is a metric to show the correlation of ranks in two datasets. A larger value of the coefficient (close to 1.0) indicates a stronger positive correlation, and a smaller value close to 0.0 represents non-correlation. The bottom row of Table IV lists Spearmann's correlation coefficient between $R(\text{RF}/\text{RF})$ and others. We confirm that SHAP shows the highest correlation coefficient for the learned models, LIME has a lower correlation coefficient and seems to be unstable across all algorithms, and the XAI algorithms for images are less similar to the baseline.

C. Processing costs

We confirm that each XAI algorithm outputs the explanation in a reasonable time. Table V shows that the XAI algorithms have been optimized for each type of algorithm without consuming too much processing time. However, with unoptimized algorithms, their processing time is relatively large. Specifically, with SHAP, to calculate the contribution of features, those features are set to missing and will be re-simulated by taking the features in the background dataset. With SHAP Kernel Explainer, the entire background data set is used, so the computational complexity scales linearly with the number of samples present in the background data. Meanwhile, SHAP Tree Explainer takes advantage of tree architecture to represent the background distribution and calculate the SHAP value. SHAP Gradient Explainer only randomly samples values from the background data set while calculating the expected gradient. Similarly, LIME is unoptimized to explain complex algorithms like AE. Thus, their processing time is up to hours for the MQTTset dataset. Our results clearly suggest that optimized XAI algorithms are one of the criteria for real usage.

TABLE V
PROCESSING TIME FOR ONE ROUND (S) (MQTT DATASET)

RF/RF	RF/LIME	RF/SHAP	CNN/LIME	CNN/SHAP	CNN/GradCAM	CNN/GBP	IF/LIME	IF/SHAP	AE/LIME	AE/SHAP
–	74.31	38.97	4.22	10.29	1.31	0.04	2.73	82.11	11707.71	19933.76

V. DISCUSSION/LIMITATION

Explainability: $R(\mathrm{RF}/\mathrm{RF})$ is reliable for the two datasets, by considering the malicious behaviors. Thus, we can assume that behavior similar to $R(\mathrm{RF}/\mathrm{RF})$ is interpreted as reliable as the explanation. Of course, there is a possibility that each AD model learned different features in more complicated cases. A more detailed analysis could be future work.

Extend to multi-class classifiers: Our analysis is currently based on binary classification of IoT traffic AD with specific anomalies. We will extend our analysis to muli-class classification with more variety of anomalous scenarios.

Suitable XAI for neural network model: Grad-CAM and GBP do not work well in our context, though SHAP and LIME are robust for the neural network based models. We will further investigate or develop a more suitable XAI algorithm for the neural network based anomaly detectors.

More explainability: The important features are useful information to infer what is happening in the network. However, there is still a gap between this explainability and network control. We should further develop more human understandable interpretations, e.g., for security controllers in IoT traffic management.

Optimize XAI algorithm: While the explainability of XAI algorithms on RF, CNN, and IF is reasonable, there remains a significant challenge in explaining performance on complex models such as AE.

VI. CONCLUSION

We demonstrated that the insights obtained through XAI techniques can play a crucial role in understanding the underlying changes in network traffic that distinguish anomalous and potentially malicious connections from benign ones. By employing appropriate XAI algorithms such as SHAP and LIME, it is possible to gain valuable knowledge about the patterns in anomalous network traffic. Our results show that XAI can consistently interpret complex relationships between features and anomalies over different AD methods, allowing us to pinpoint specific indicators that differentiate benign traffic from anomalous behavior. These can lead to refined detection algorithms, additional security measures and overall robustness against different types of attacks.

Acknowledgment: U.D and L.L are supported by the NII internship program. This work is partly supported by JST CREST JPMJCR21M3.

REFERENCES

[1] Nida Aslam, Irfan Ullah Khan, Samiha Mirza, Alanoud AlOwayed, Fatima M. Anis, Reef M. Aljuaid, and Reham Baageel. Interpretable machine learning models for malicious domains detection using explainable artificial intelligence (xai). *Sustainability*, 14(12), 2022.

[2] Pieter Barnard, Nicola Marchetti, and Luiz A. DaSilva. Robust network intrusion detection through explainable artificial intelligence (xai). *IEEE Networking Letters*, 4(3):167–171, 2022.

[3] Ruth C Fong and Andrea Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. In *Proc. ICCV'17*, pages 3429–3437, 2017.

[4] Chanwoong Hwang and Taejin Lee. E-sfd: Explainable sensor fault detection in the ics anomaly detection system. *IEEE Access*, 9:140470–140486, 2021.

[5] Chathuranga Sampath Kalutharage, Xiaodong Liu, Christos Chrysoulas, Nikolaos Pitropakis, and Pavlos Papadopoulos. Explainable ai-based ddos attack identification method for iot networks. *Computers*, 12:32, 02 2023.

[6] Rupesh Raj Karn, Prabhakar Kudva, Hai Huang, Sahil Suneja, and Ibrahim M. Elfadel. Cryptomining detection in container clouds using system calls and explainable machine learning. *IEEE TPDS*, 32(3):674–691, 2021.

[7] Leon Kopitar, Leona Cilar, Primoz Kocbek, and Gregor Stiglic. Local vs. global interpretability of machine learning models in type 2 diabetes mellitus screening. In *Proc. AIME'19*, page 108–119, 2019.

[8] Hans-Peter Kriegel, Peer Kröger, Erich Schubert, and Arthur Zimek. Interpreting and unifying outlier scores. In *Proc. SDM'11*, pages 13–24, 04 2011.

[9] Laura Lahesoo, Uyen Do, Rodrigo Carnier, and Kensuke Fukuda. Siuru: A framework for machine learning based anomaly detection in iot network traffic. In *Proc. AINTEC'23*, page 87–95, 2023.

[10] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Proc. NIPS'17*, pages 4765–4774, 2017.

[11] Alfredo Nascita, Antonio Montieri, Giuseppe Aceto, Domenico Ciuonzo, Valerio Persico, and Antonio Pescapé. Xai meets mobile traffic classification: Understanding and improving multimodal deep learning architectures. *IEEE TNSM*, 18(4):4225–4246, 2021.

[12] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *Proc. KDD'16*, pages 1135–1144, 2016.

[13] Khushnaseeb Roshan and Aasim Zafar. Using kernel shap xai method to optimize the network anomaly detection model. In *2022 9th International Conference on Computing for Sustainable Global Development (INDIACom)*, pages 74–80, 2022.

[14] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proc. ICCV'17*, pages 618–626, 2017.

[15] Iman Sharafaldin, Arash Habibi Lashkari, and Ali A Ghorbani. Toward generating a new intrusion detection dataset and intrusion traffic characterization. *ICISSp*, 1:108–116, 2018.

[16] Marios Siganos, Panagiotis Radoglou-Grammatikis, Igor Kotsiuba, Evangelos Markakis, Ioannis Moscholios, Sotirios Goudos, and Panagiotis Sarigiannidis. Explainable ai-based intrusion detection in the internet of things. In *Proc. ARES'23*, 2023.

[17] Gaurav Somani, Manoj Singh Gaur, Dheeraj Sanghi, Mauro Conti, and Rajkumar Buyya. Ddos attacks in cloud computing: Issues, taxonomy, and future directions. *Computer Communications*, 107:30–48, 2017.

[18] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*, 2014.

[19] Bhagyashri Tushir, Yogesh Dalal, Behnam Dezfouli, and Yuhong Liu. A quantitative study of ddos and e-ddos attacks on wifi smart home devices. *IEEE Internet of Things Journal*, 8(8):6282–6292, 2021.

[20] Ivan Vaccari, Giovanni Chiola, Maurizio Aiello, Maurizio Mongelli, and Enrico Cambiaso. Mqttset, a new dataset for machine learning techniques on mqtt. *Sensors*, 20(22):6578, 2020.