

Explainable AI for Real-Time Adversarial Attack Detection in Edge Machine Learning Pipelines

Dinesh Kumar Garg¹, A. Chandra Sekaran², Govinda RaviRaj Mulasa³

Senior IT Manager, SIEEE, Honeywell Intelligrated, USA.

Managing Director, Techpower Solutions, Nagercoil, Tamil Nadu, India.

Product Architect, USA.

dineshkumargarg878@outlook.com

Abstract—The rapid deployment of machine learning models on edge devices has enabled real-time intelligent applications in domains such as healthcare, smart transportation, industrial automation, and Internet of Things (IoT) systems. The use of machine learning models on edge devices has opened up the door to intelligent applications in real time across several areas, including healthcare, smart transportation, the Internet of Things (IoT), and industrial automation. The distributed and resource-constrained nature of edge environments, however, makes machine learning pipelines susceptible to adversarial attacks that can alter input data and lead to wrong predictions. In this paper, a framework for real-time adversarial attack detection in edge machine learning pipelines using Explainable Artificial Intelligence (XAI) is proposed. The proposed approach extends the monitoring of incoming data and the behavior of the model, looking for unusual patterns linked to adversarial perturbations, and, additionally, adds an explainability layer, providing interpretable information about what features and decision factors are behind the detection outcome. It combines lightweight machine learning, adversarial sample analysis, and real-time detection and explanation with SHAP to provide low-latency security monitoring on edge devices with limited resources. The results from experimental testing with the Edge-IIoT-set dataset are shown to be effective in distinguishing between normal and malicious network traffic with an overall classification accuracy of 97.71%. The proposed solution offers an interpretable and reliable security mechanism to secure edge machine learning pipelines against adversarial attacks while preserving the requirements of real-time edge intelligence.

Keywords—*Explainable Artificial Intelligence, Adversarial Attack Detection, Edge Machine Learning, Real-Time Detection, Edge Computing, Adversarial Machine Learning, Model Security, Interpretable AI, IoT Security, Machine Learning Pipelines.*

I. INTRODUCTION

Intelligent applications are becoming more prevalent, driven by the rapid growth of edge computing and machine learning (ML) and the ability to process data near its source,

which significantly decreases communication latency, bandwidth usage, and reliance on centralized cloud structures [1]. The applications for Edge ML are becoming more widespread, including the Internet of Things (IoT), autonomous systems, healthcare monitoring, industrial automation, smart surveillance, and intelligent transportation [2]. Although there are many benefits to deploying ML models at the edge, it also poses a host of security concerns [3]. The machine learning pipelines of edge devices are vulnerable to different types of security attacks, especially adversarial attacks, because they are usually deployed in distributed, heterogeneous, and resource-constrained environments [4].

In adversarial attacks, the attacker crafts legitimate inputs that are carefully modified but typically not distinguishable to humans and, when fed into an ML model, can yield incorrect results [5]. Weaknesses in the decision boundaries of models can be exploited by attacks like the Fast Gradient Sign Method (FGSM), Projected Gradient Descent (PGD), etc. [6]. The effectiveness of an adversarial attack could result in misguided decisions and affect the trustworthiness and safety of the entire intelligent system in real-time edge applications. So, it is critical to have effective and efficient mechanisms to detect adversarial inputs [7].

Traditional approaches to security typically try to detect unusual inputs or enhance the security of the model but have poor understanding of why certain inputs are identified as suspicious [8]. To overcome this limitation, Explainable Artificial Intelligence (XAI) offers opportunities to create explainable explanations for model decisions [9]. Feature attribution and local explanation methods can be used to find features that lead to adversarial detection decisions, which can enhance the transparency of the adversarial detection process and help security analysis [10].

In this study, we concentrate on creating an XAI-based framework to detect adversarial attacks in real-time edge ML pipelines. The aim of the proposed approach is to combine the lightness of the attack detection mechanism with an explainability layer that is able to detect the suspicious inputs and give understandable evidence for each detection decision made. The objective of the framework is to optimize the detection effectiveness, explanation quality, efficiency of computation, and low latency to enhance the security, transparency, and trustworthiness of machine learning systems in the edge environment with resource constraints.

II. RELATED WORKS

The concepts of explainable artificial intelligence (XAI), adversarial machine learning, and secure edge intelligence have been explored from various angles in recent research [11]. Prior work has shown that the use of XAI can be a key avenue to enhance transparency and trust in machine learning systems, especially when delivering interpretable insights into models' decisions. The use of XAI in communication and intelligent networks has also been explored, focusing on its benefits in enhancing human understanding of automated decisions [12]. However, the techniques of XAI can be targeted in a black box attack, revealing the need for secure and reliable XAI mechanisms.

Adversarial machine learning studies have focused on attack strategies, vulnerabilities, and defenses in wireless, mobile, and intelligent systems [13]. Adversarial examples have been shown to impact computer vision applications such as edge detection, and other research has been conducted on adversarial attacks on deep reinforcement learning and federated learning environments [14]. There are several detection-oriented approaches that have been proposed to identify adversarial examples based on information-theoretic characteristics and learned representations. The connection between adversarial examples and model decision boundaries has also been studied.

Nevertheless, current methods primarily focus on adversarial detection, explanation, and edge security as isolated issues. Edge and IoT devices are considered more resource-constrained, so lightweight XAI research is crucial to reduce computational requirements in these devices [15]. Yet, there have been few studies on combining real-time adversarial attack detection with explainable decision mechanisms in edge machine learning pipelines. This discrepancy is the reason for the need to develop a unified framework that can detect attacks efficiently and also give a meaningful explanation to security decisions.

III. Methodology

The proposed framework is designed to enable real-time identification of adversarial attacks in edge-based machine learning systems and to preserve the interpretability of the security decisions. The methodology combines lightweight machine learning, adversarial attack analysis, real-time detection, and Explainable Artificial Intelligence (XAI). This framework takes network traffic data as input and does some preprocessing to get a standardized representation of features. The processed data is then fed into a light machine learning model that provides the prediction and confidence to the output. The model behavior is further explored from a model adversarial manipulation perspective. The XAI component gives feature-level explanations of the generated security decisions, allowing the identification of security decision characteristics leading to adversarial behavior. The proposed architecture is made up of the functional components shown in Fig. 1, as they relate to data preparation, edge classification, adversarial sample analysis, attack detection, explainability, and security response.

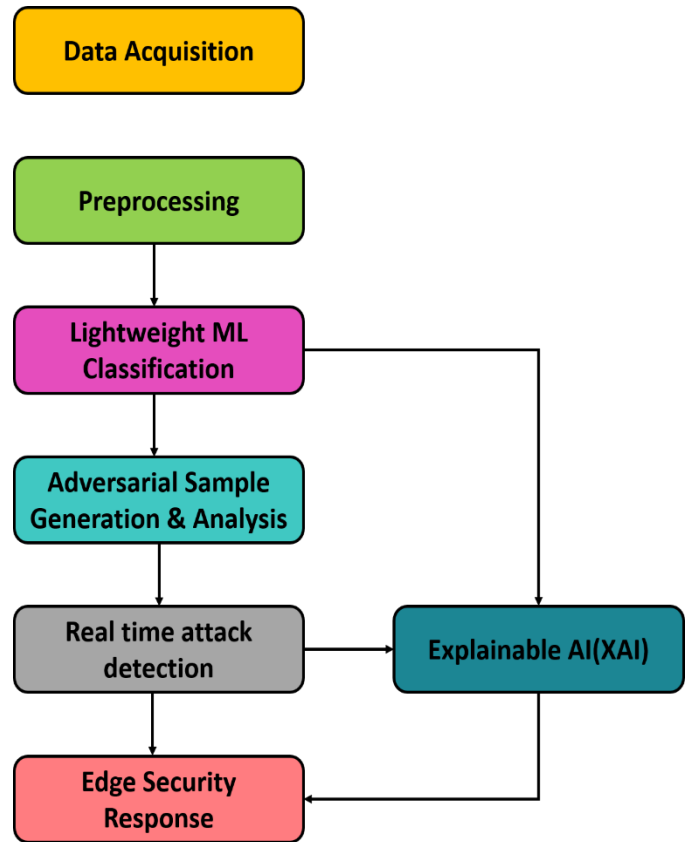


Fig. 1. Overall architecture of the proposed XAI-based real-time adversarial attack detection framework.

The data acquisition component is responsible for getting the network-flow records from data sources, and the preprocessing component cleans the data, transforms it, and normalizes it. The lightweight machine learning model is used

to produce the first classification result. The adversarial samples created via representative attack techniques are used to test the robustness of the model and train the detection mechanism. The part of the detection module that detects malicious behavior in the real-time incoming sample by checking with the learned behavioral patterns. The XAI module then examines the decision made about the detection and determines which features had the biggest impact on the classification. Finally, the edge security component generates the necessary response and logs pertinent data, including the prediction status, adversarial probability, explanation features, and processing latency. The operation of each component is described in the following subsections.

A. Data Acquisition and Preprocessing

The proposed framework starts with collecting network traffic data from the Edge-IIoTset dataset, which is composed of the benign and malicious records of network flows for different attack scenarios of IoT and edge computing in internet environments. The aggregated records are then processed at the edge layer to be ready for machine learning analysis. Records are duplicated, and missing values or inconsistent data are eliminated with preprocessing. Categorical attributes are transformed into numeric representations, and continuous features are normalized so that features with widely varying numerical ranges do not overwhelm the learning process. The processed data set is split into training, validation, and testing subsets in a 70:15:15 ratio. The objective is to use the training set to learn the classification patterns, the validation set to select hyperparameters and determine a threshold, and the independent test set to finally check for the performance of the classification.

B. Lightweight Edge Machine Learning Model

The feature vectors are passed into a lightweight edge-based inference neural network after the preprocessing. The scheme proposed takes into account binary attack detection and multiclass traffic classification. The model is used to distinguish between normal and attack traffic in the binary setting and between the different traffic categories in the multiclass setting, where the traffic categories are represented by the fifteen classes in the Edge-IIoTset dataset. The network is made up of fully connected layers of nonlinear activation functions that are followed by an appropriate output layer for the specific classification problem. In binary classification, a sigmoid activation function outputs the probability that the input is from the attack class, while the multiclass model has an output layer that outputs the most likely traffic class. The model minimizes the loss of classification that corresponds to each of these classes during training. The lightweight

architecture decreases the computing and memory needs of the model, enabling deployment on resource-constrained edge devices.

C. Adversarial Sample Generation

Adversarial samples are created from a subset of the test set using the Fast Gradient Sign Method (FGSM), Projected Gradient Descent (PGD), and Carlini and Wagner (C&W) attacks to assess the robustness of the trained edge model. These attacks inject small perturbations into the input in an engineered way to change the output of the model. Instead, the adversarial sample is created as follows for FGSM:

$$x_{adv} = x + \epsilon \text{sign}(\nabla_x J(\theta, x, y)) \quad (1)$$

where x represents the original input, x_{adv} represents the adversarial input, ϵ controls the perturbation magnitude, J is the classification loss, θ represents the model parameters, and y denotes the true class. The gradient points to the direction in which the input needs to be changed in order to maximize the classification loss. PGD generalizes the FGSM idea by applying multiple iterations of perturbations while keeping the perturbed sample in a specified perturbation region. The C&W attack reframes adversarial sample generation as an optimization problem where it seeks an adversarial sample whose input is close to the original while still changing the output of the model while preserving the constraints on the perturbation. The samples generated based on these adversarial methods, namely FGSM, PGD, and C&W, produce representative adversarial samples to test the robustness of the proposed detection framework under different perturbation strengths.

D. Real-Time Adversarial Attack Detection

The adversarial detection module receives incoming edge data and the prediction approach of the main ML model. The input representation to the detector is the normalized input features, prediction probability, and selected behavioral characteristics. It then calculates the likelihood of the current sample being adversarial. A threshold is chosen from the validation dataset, and the detection decision is made using this threshold. Sensitivity can be expressed as

$$P_{adv} = D(X, P, C) \quad (2)$$

where D represents the adversarial detector, X represents the processed input features, P represents the ML prediction, and C represents the model confidence. If P_{adv} exceeds the predefined threshold, the sample is classified as adversarial; otherwise, it is considered benign. This mechanism enables suspicious samples to be identified before they can influence downstream edge decisions.

E. Explainable AI-Based Decision Analysis

Once a detection decision is made, the XAI layer generates an interpretable explanation of the prediction. SHAP is used to measure the contribution that each of the input features makes to the adversarial or benign decision. The model output can be shown as

$$f(x) = \phi_0 + \sum_{i=1}^n \phi_i \quad (3)$$

where ϕ_0 represents the baseline model output and ϕ_i represents the contribution of the i^{th} feature. Positive feature contributions indicate characteristics that increase the likelihood of an adversarial classification, whereas negative contributions support the benign classification. The explanation layer therefore transforms the detector's numerical decision into feature-level evidence that can be interpreted by security analysts. This improves transparency and helps identify the characteristics responsible for suspicious behavior.

F. Edge Deployment and Real-Time Response

The optimized trained classifier, adversarial detector, and explanation are executed in an edge environment. The data is processed locally without the need for a continuous transmission to a centralized cloud server. The system first preprocesses, then classifies, then analyzes adversarial examples, and finally generates explanations for every sample received. In the event of detecting an adversarial sample, the system will trigger a security alert and be able to reject or isolate the suspicious input. Processing times are captured for each stage to assess the real-time performance. Additionally, lightweight architecture design and numerical quantization methods can be applied to optimize the models and reduce memory usage and inference latency.

G. Performance Evaluation

The proposed framework is assessed in terms of accuracy, precision, recall, F1-score, false-positive rate, ROC-AUC, and detection latency for binary attack detection and 15-class traffic classification. "Precision" refers to the accuracy of attack classification, and "high accuracy" refers to the accuracy of the detected attacks matching the actual adversarial samples. Recall is how well the detector can detect malicious inputs, which is crucial for security purposes. The F1-score is a balanced measure between precision and recall. Besides, edge-specific metrics like inference latency, memory consumption, model size, and throughput are measured to assess the framework's ability to run under real-time resource constraints. The proposed approach is compared with conventional machine learning approaches and adversarial detection

approaches to show the effectiveness and computational efficiency of the proposed approach.

H. Experimental Setup and Reproducibility

The proposed framework is deployed and tested on the Edge-IIoT-set dataset in a controlled machine learning environment. The dataset has been split into training, validation, and test sets with a ratio of 70:15:15, following stratification. The data has been stratified and split into training, validation, and test subsets in the ratio 70:15:15. The weights of the lightweight neural network are optimized with the set of parameters in the optimization procedure and the learning procedure. The hyperparameters are selected based on the validation subset, and the threshold for detecting adversarial examples is determined using the set of hyperparameters that yield the best performance on the validation subset. For the adversarial evaluation, three attacks (FGSM, PGD, and C&W) are applied with the given perturbation settings. This explainability module uses SHAP to create feature-level explanations and explanations globally. The results of classification, adversarial detection behavior, explanation stability, and processing efficiency are documented during the evaluation. The implementation environment, model architecture, training parameters, adversarial attack parameters, and evaluation parameters should be kept consistent across experimental runs to ensure reproducibility.

I. Complete Processing Flow

The entire framework starts with the collection of network traffic data from the chosen dataset, which is then preprocessed in order to remove noise, transform, and normalize the input features. The processed data are then used for light-weight machine learning classification to get the initial prediction and confidence score. The adversarial detection module then checks the characteristics of the input and the model behavior to decide if the sample has adversarial properties. Upon the discovery of suspicious behavior, the XAI module processes the decision on detection and then decides on the features that are most influential in the classification process. The resulting explanation gives interpretable evidence of the factors that led to the identification of the sample as benign or adversarial. In the end, the edge security part responds to the threat detected and also keeps relevant data, such as the prediction outcome, adversarial probability, and features that contribute to the prediction, as well as processing latency. This integrated process allows the proposed framework to accomplish adversarial attack detection and give understandable security choices while guaranteeing the low latency needs of edge machine learning frameworks.

IV. RESULT AND DISCUSSION

The proposed explainable AI-based adversarial attack detection framework was tested on the Edge-IIoTset dataset, and its performance was examined with respect to its classification, adversarial robustness, and interpretability. The experimental analysis takes into account the evaluation of binary attack detection and the evaluation of detailed classification of 15 classes of traffic in order to have a full evaluation of the model. Furthermore, global and local explanations of the model's decisions are explored to uncover the network features that impact the model. The robustness of the detection and explanation mechanisms is then studied further under various levels of adversarial perturbations. In addition, training convergence and class-wise performance are also computed to confirm the stability and generalizability of the proposed framework. The results obtained are presented and discussed in the next subsections.

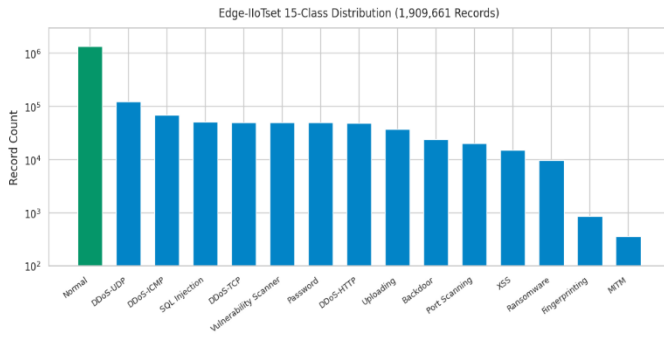


Fig. 2. Edge-IIoTset 15-Class Distribution

As shown in Fig. 2, the number of samples collected for each traffic class varies significantly for the Edge-IIoT set. The largest sample size is associated with normal traffic, while DDoS-UDP and other attack classes have much larger sample sizes compared to minority classes like fingerprinting and MITM. To ensure reliable multiclass attack classification, overall accuracy should be combined with class-wise and macro-averaged evaluation measures, as the class imbalance shows.

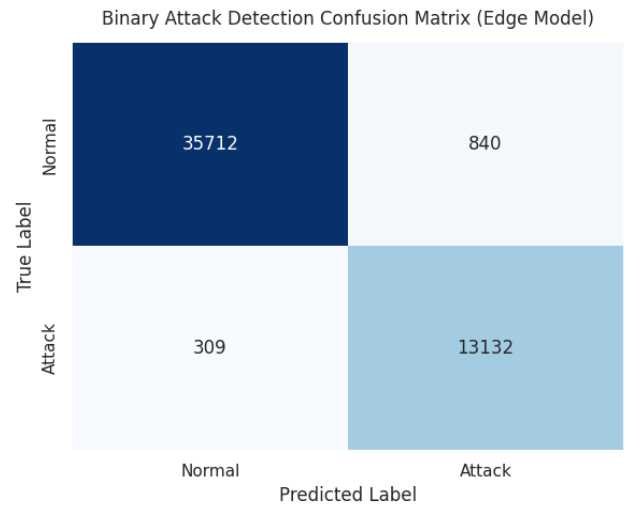


Fig. 3. Binary Attack Detection Confusion Matrix

As shown in Fig. 3, the binary attack detection model accurately detects 35,712 normal samples and 13,132 attack samples. 840 "normal" samples are misclassified as attacks, and 309 "attack" samples are misclassified as normal. There are 840 "normal" samples misclassified as "attack" and 309 "attack" samples misclassified as "normal." The low rate of false negatives shows the model's effectiveness in detecting malicious traffic. The confusion matrix thus offers verification of the binary classification potential of the proposed framework for distinguishing between the normal traffic and attack traffic.

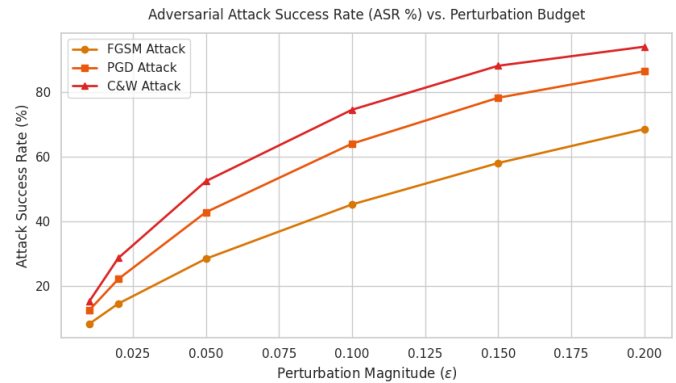


Fig. 4. Adversarial Attack Success Rate Under Different Perturbation Budgets

As revealed in results given in Fig. 4, the success rate of an adversarial attack grows steadily for FGSM, PGD, and C&W attacks as the size of the perturbation grows. The C&W attack has the highest attack success rate for the perturbation levels considered, followed by PGD and FGSM attacks. The higher success rate as the perturbation budget got larger suggests that the larger the perturbation, the more it affects the model's predictions. The results serve as a representative adversarial setting to test the proposed real-time detection

framework and its susceptibility to detecting suspicious inputs for different perturbation levels.

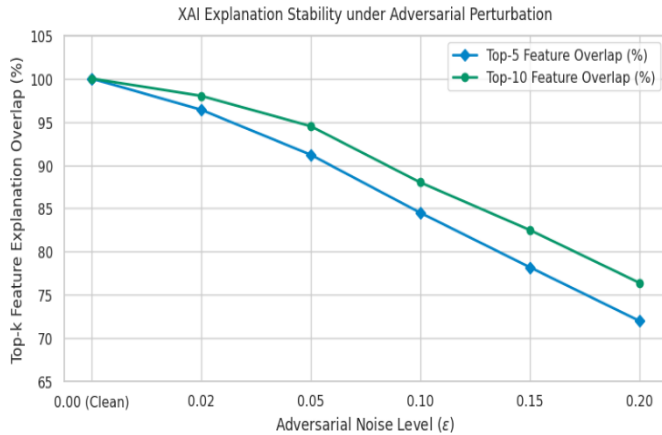


Fig. 5. XAI Explanation Stability Under Adversarial Perturbation

As demonstrated in Fig. 5, the stability of the SHAP explanations gradually decreases with increasing adversarial perturbation level. For the top-10 feature overlap, the value is consistently higher than the top-5 feature overlap over the range of evaluated perturbations, suggesting that a wider set of influential features is more robust in the presence of perturbations. The overlaps of the top-10 and top-5 remain around 76% and 72%, respectively, at the point of (ϵ = 0.20). These findings suggest that the SHAP explanations maintain a significant level of consistency across the features even with larger adversarial perturbations.

TABLE I. CLASSIFICATION PERFORMANCE REPORT FOR EDGE-IIOTSETS

Class	Precision	Recall	F1-Score	Support
Normal	0.9993	0.9770	0.9880	35712
DDoS-UDP	0.9768	0.9771	0.9769	3182
DDoS-ICMP	0.9549	0.9775	0.9661	1778
SQL Injection	0.9427	0.9774	0.9598	1330
DDoS-TCP	0.9545	0.9771	0.9657	1310
Vulnerability Scanner	0.9538	0.9771	0.9653	1309
Password	0.9321	0.9770	0.9541	1307
DDoS-HTTP	0.9317	0.9772	0.9539	1271
Uploading	0.9180	0.9772	0.9467	963
Backdoor	0.8674	0.9777	0.9193	629
Port Scanning	0.8750	0.9771	0.9232	523
XSS	0.8244	0.9772	0.8943	394
Ransomware	0.7470	0.9802	0.8479	253
Fingerprinting	0.2157	1.0000	0.3548	22

MITM	0.1087	1.0000	0.1961	10
Accuracy			0.9771	49993
Macro Avg	0.8135	0.9805	0.8541	49993
Weighted Avg	0.9812	0.9771	0.9786	49993

As shown in Table I, the classification result on the Edge-IIoT-set dataset shows that the proposed model can achieve 97.71% overall accuracy on the 15-class dataset. The weighted F1-score is 97.86%, which shows good overall accuracy on the test set. The F1 scores for most of the high-frequency traffic categories are above 0.90, such as Normal, DDoS-UDP, DDoS-ICMP, SQL Injection, DDoS-TCP, Vulnerability Scanner, Password, DDoS-HTTP, Uploading, Backdoor, and Port Scanning. The macro-average F1-score, however, is only 85.41%, likely because there are only 22 and 10 samples of fingerprinting and MITM, respectively, that account for the significantly lower performance of the minority classes. The results show good overall classification performance and the impact of having severely imbalanced classes on minority class performance.

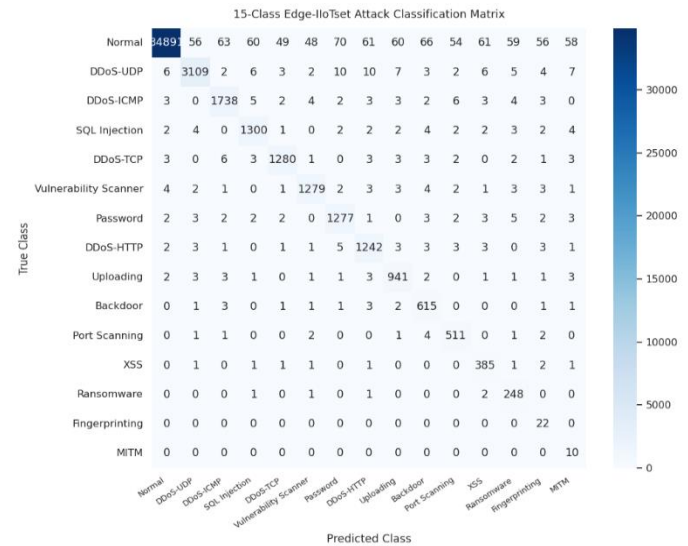


Fig. 6. 15-Class Edge-IIoTset Attack Classification Matrix

The classification results of the proposed model on the Edge-IIoTset dataset are presented in 15 classes, as shown in Fig. 6, which proves the capability of the proposed model to differentiate between the normal traffic and the individual attack classes. A high value along the diagonal means that many of the samples are classified in the same class as the actual class. As can be seen from the results of the classification of the classes in Table I, the results of the classification of the high-frequency categories such as Normal, DDoS-UDP, DDoS-ICMP, SQL Injection, and DDoS-TCP are satisfactory.

Minority classes like fingerprinting and MITM have significantly fewer instances and lower precision and F1 scores, however. The confusion matrix thus gives detailed evidence of the model's behavior on the level of classes and the impact of class imbalance on minority class recognition.

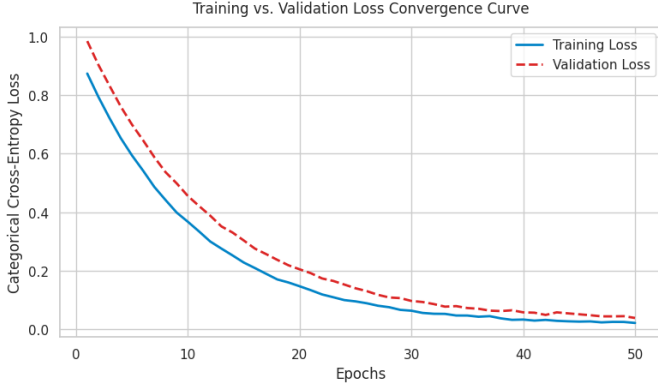


Fig. 7. Training and Validation Loss Convergence

As shown in the convergence behavior of both the training loss and the validation loss in Fig. 7, the loss decreases significantly in the initial few epochs and gradually settles down to lower values with subsequent training. The overall convergence of the validation loss curve remains quite consistent with the training loss curve with only minor signs of overfitting within the period studied. Decreased loss values in both data sets show that the model gradually decreases the classification error, and it shows stable learning behavior from the training set and the validation set.

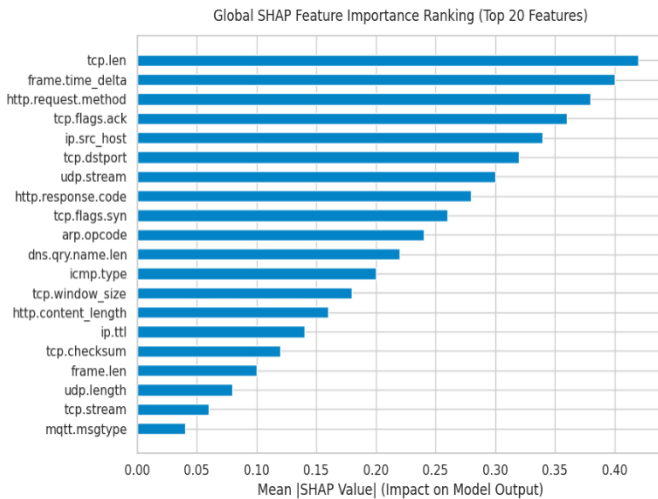


Fig. 8. Global SHAP Feature Importance Ranking

The global SHAP analysis, as shown in Fig. 8, helps to identify the features in the network that have the biggest influence on the model prediction: tcp.len, frame.time_delta, and http.request.methods have the highest mean absolute SHAP values, followed by TCP flags, IP-related attributes,

UDP stream characteristics, and HTTP response information. A larger mean absolute SHAP value represents a larger mean absolute effect that the feature has on the model's predictions over the set of samples used. The ranking thus gives a meaningful depiction of the main network-flow attributes that impact the attack classification decision made by the model.

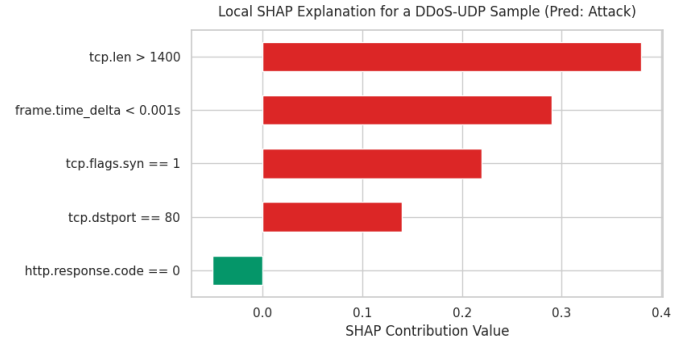


Fig. 9. Local SHAP Explanation for a DDoS-UDP Sample

Local SHAP explanation for a single DDoS-UDP sample labeled as an attack, as shown in Fig. 9, highlights the contributions of each feature to predicting the sample as an attack. The items with the most positive contribution to the attack prediction are tcp.len > 1400, frame.time_delta < 0.001 s, tcp.flags.syn = 1, and tcp.dstport = 80. On the other hand, if http.response.code = 0, it is a negative contributor to the explained attack output. These are feature-level contributions, each of which can be interpreted as evidence of the impact of the individual features of the network on the model's prediction for the selected DDoS-UDP sample.

V. CONCLUSION

The proposed EAI-supported framework offers an integrated solution for detecting adversarial attacks in real-time in the edge machine learning pipelines. The overall traffic classification accuracy achieved by the proposed model on the Edge-IIoTset dataset is 97.71% for 15 traffic classes, which proves the effectiveness of the proposed model in distinguishing between normal traffic and various traffic attack categories. The binary classification results further show the ability of the framework to classify normal and attack traffic, and adversarial experiments are conducted with FGSM, PGD, and C&W to investigate the model behavior as perturbations increase. Global and local analysis of SHAP values delivers interpretable information on features of the network that are relevant to the prediction of attacks, and explanation-stability analysis tests the robustness of the feature-level explanation in the presence of adversarial perturbations. The lightweight architecture is designed to help deploy in resource-constrained edge environments, ensuring low computation and memory consumption. The proposed framework will combine these

elements of adversarial analysis, explainability, and edge-oriented processing to provide real-time cybersecurity monitoring. Future work can involve aspects such as deploying to the edge layer and hardware, online learning with adaptation, more scenarios for adversarial attacks, and further optimizations of explanation latency.

REFERENCES

- [1] Tjoa, Erico, and Cuntai Guan. "A survey on explainable artificial intelligence (xai): Toward medical xai." *IEEE transactions on neural networks and learning systems* 32, no. 11 (2020): 4793-4813.
- [2] Guo, Weisi. "Explainable artificial intelligence for 6G: Improving trust between human and machine." *IEEE Communications Magazine* 58, no. 6 (2020): 39-45.
- [3] Kuppa, Aditya, and Nhien-An Le-Khac. "Black box attacks on explainable artificial intelligence (XAI) methods in cyber security." In *2020 International Joint Conference on neural networks (IJCNN)*, pp. 1-8. IEEE, 2020.
- [4] Cosgrove, Christian, and Alan Yuille. "Adversarial examples for edge detection: They exist, and they transfer." In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 1070-1079. 2020.
- [5] Asim, Muhammad, Yong Wang, Kezhi Wang, and Pei-Qiu Huang. "A review on computational intelligence techniques in cloud and edge computing." *IEEE Transactions on Emerging Topics in Computational Intelligence* 4, no. 6 (2020): 742-763.
- [6] Liu, Jinxin, Michele Nogueira, Johan Fernandes, and Burak Kantarci. "Adversarial machine learning: A multilayer review of the state-of-the-art and challenges for wireless and mobile systems." *IEEE Communications Surveys & Tutorials* 24, no. 1 (2021): 123-159.
- [7] Ilahi, Inaam, Muhammad Usama, Junaid Qadir, Muhammad Umar Janjua, Ala Al-Fuqaha, Dinh Thai Hoang, and Dusit Niyato. "Challenges and countermeasures for adversarial attacks on deep reinforcement learning." *IEEE Transactions on Artificial Intelligence* 3, no. 2 (2021): 90-109.
- [8] Wen, Jing, Siu-Ming Yiu, and Lucas CK Hui. "Defending against model inversion attack by adversarial examples." In *2021 IEEE International Conference on Cyber Security and Resilience (CSR)*, pp. 551-556. IEEE, 2021.
- [9] T. Dinesh, S. Ranushika, and C. sneka, "Residual Feature Fusion with Transformer-Inspired Decoding for Accurate and Efficient Semantic Segmentation," *Journal of Progressive Engineering Studies (JPES)*, vol. 1, no. 1, pp. 44-53, 2026.
- [10] Vignesh, S., Bhawna Bhawna, Shubham Anand, and Shailender Kumar. "A comprehensive review of adversarial learning and impact of unsharing weights across classes." In *2021 Fourth International Conference on Computational Intelligence and Communication Technologies (CCICT)*, pp. 1-8. IEEE, 2021.
- [11] Islam, Mir Riyanul, Mobyen Uddin Ahmed, and Shahina Begum. "Local and global interpretability using mutual information in explainable artificial intelligence." In *2021 8th International Conference on Soft Computing & Machine Intelligence (ISCM)*, pp. 191-195. IEEE, 2021.
- [12] Gao, Song, Ruxin Wang, Xiaoxuan Wang, Shui Yu, Yunyun Dong, Shaowen Yao, and Wei Zhou. "Detecting adversarial examples on deep neural networks with mutual information neural estimation." *IEEE Transactions on Dependable and Secure Computing* 20, no. 6 (2023): 5168-5181.
- [13] Tian, Zehao, Zixiong Wang, Ahmed M. Abdelmoniem, Gaoyang Liu, and Chen Wang. "Knowledge representation of training data with adversarial examples supporting decision boundary." *IEEE Transactions on Information Forensics and Security* 18 (2023): 4116-4127.
- [14] Crook, Barnaby, Maximilian Schlüter, and Timo Speith. "Revisiting the performance-explainability trade-off in explainable artificial intelligence (XAI)." In *2023 IEEE 31st International Requirements Engineering Conference Workshops (REW)*, pp. 316-324. IEEE, 2023.
- [15] Kumar, Kummari Naveen, Chalavadi Krishna Mohan, and Linga Reddy Cenkeramaddi. "The impact of adversarial attacks on federated learning: A survey." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 46, no. 5 (2023): 2672-2691.
- [16] Murugan Lakshmanan, Govinda RaviRaj Mulasa, and Rajaguru Ganesan, "Efficient Facial Emotion Recognition using Adaptive Feature Selection and Lightweight Transformer," *Journal of Progressive Engineering Studies (JPES)*, vol. 1, no. 1, pp. 15-32, 2026.
- [17] Basheer, Fathima. "Explainable Ai in Edge and Iot Devices: A Review of Lightweight Techniques." In *2025 IEEE 4th International Conference on Technology, Engineering, Management for Societal impact using Marketing, Entrepreneurship and Talent (TEMSMET)*, pp. 1-6. IEEE, 2025.