Blockchain and Explainable ML-based Secure Network Attack Classification in Smart Homes

Dev Mehta*, Akshat Jingar†, Janam Patel‡, Rajesh Gupta§, Pronaya Bhattacharya¶, Sudeep Tanwar‖, Chinmay Trivedi**, Maniklal Das††
*†‡§‖‖Department of Computer Science and Engineering, Institute of Technology, Nirma University, Ahmedabad, India
¶Department of CSE, Amity School of Engg. & Technology, Amity University, Kolkata, India
††DA-IICT, Gandhinagar, India
Emails: *22bcm015@nirmauni.ac.in, †23bce517@nirmauni.ac.in, ‡21bce197@nirmauni.ac.in, §rajesh.gupta@nirmauni.ac.in, §pbhattacharya@kol.amity.edu, ¶sudeep.tanwar@nirmauni.ac.in, ‖21bce041@nirmauni.ac.in, ‖maniklal_das@daiict.ac.in

***Abstract*—One of the main issues with Internet of Things (IoT) devices in the Smart Home (SH) environment is threats of attacks. A comprehensive Intrusion Detection System (IDS) and a transparent ledger are important for this process. However, another major problem is the lack of a clear explanation of traditional IDS actions. By integrating Explainable AI (XAI) for feature extraction, we add an explainability factor and reduce the complexity of data. A machine learning (ML) model is trained on this data to improve the classification of different types of attacks. In our approach, we experimented using the RT-IoT2022 dataset. This dataset is a real-time indicator of attacks in IoT networks. It provides difficult and real-time scenarios that highlight the complexity of IDS. Furthermore, the benign data the model classifies is stored in the blockchain to make the system secure and transparent. Our XAI-based technique provides better results than previous models. The proposed approach provides a clear and brief view of factors that affect classification actions, which helps users make security decisions. As a result, this study provides an explanation and protection against different types of attacks in SH.**

Index Terms*—*Blockchain, Machine learning, Explainable AI, Intrusion detection, Smart home

I. Introduction

A smart home comprises interconnected devices, sensors, electronic components, and network infrastructure. All integrated within a residential environment make the Smart Home (SH). It is transforming how we live, offering convenience, comfort, and security through interconnected devices. It is like a private home with many devices of home automation which are intelligent [1]. These devices work together using the internet, which makes home devices more efficient, secure and convenient. It provides various applications, including managing thermostats, lights, security cameras, door locks, and kitchen gadgets. Through IoT technology, communication and control of devices occurs via smartphones [2]. Integrating these devices into residential areas offers various benefits to homeowners. These benefits includes advanced security through remote monitoring and access control, improved convenience with automated tasks and alerts increased energy efficiency via self-regulating lighting and temperature control [3].

SH systems generate security challenges in managing their increasing complexity and interconnectivity. The integration of various communication technologies introduces security issues which can be misused by intruders [4]. One of the major threats is unauthorized access, which can result in privacy violations and data theft. There is the risk of device tampering, allowing attackers to gain control and interrupt normal functions. External threats, such as malicious attacks, can sacrifice the security of SH systems. Privacy issues happen due to the collection and misuse of personal data in SH systems. A SH IDS would monitor network traffic and system activities within the home environment to detect any unusual behaviour. SH IDS can be divided into two parts: the first is network-specific (NIDS), and the other is host-specific (HIDS). NIDS would examine traffic between IoT devices and home networks and identify any activity that can be a security threat. HIDS would be installed on individual IoT devices to detect any unauthorized access [5].

ML models achieve high accuracy but lack in transparency, making it difficult to understand how they arrive at their decisions [6]. XAI techniques are being integrated with ML models to introduce explainability in the system. XAI tools like Shapley Additive explanations (SHAP) and Local Interpretable Model-agnostic Explanations (LIME) identify the essential features from the data collected by the IoT devices [7]. Training ML models based on this feature can achieve even higher accuracy in detecting intrusions. This approach utilizes strengths in both disciplines: the detection capabilities of ML and the feature identification power of XAI. Blockchain is an immutable ledger system that records transactions within a specific environment. Furthermore, each block of transaction added to the blockchain is verified by the consensus [8]. This integration results in a more precise and effective IDS for SHs.

Several research efforts have explored ML-based IDS in smart houses. Li et al. [6] designed a double-layered IDS. The first layer effectively identifies malicious traffic, and the other layer identifies types of various attacks. They have proposed a feature selection technique that uses fewer resources and is based on data analysis. Gazdar et al. [9] explored built-in feature selection techniques and then trained decision trees and random forest ML models. Alghayadh et al. [10] implemented hybrid two-layer IDS. The first layer analyzes traffic on the

SH network using XGB, RF, and DT techniques, and the other layer analyses the requests received. RF gives the highest accuracy. Ref. [7] used an XAI to improve trust in IDS with a decision tree model chosen for their clarity. Feature engineering and rule extraction were used to explore the decision tree's effectiveness in IDS.

In our study, we proposed a secure IDS for the SH environment. Instead of using ML methods for IDS, we have added the XAI technique SHAP for feature extraction. It helps us determine which features are most important for detecting intrusions. Using these XAI methods, we found key features, and based on those features, we have retrained different models. After training and testing, the benign data is stored on the blockchain using the smart contract function designed in our system.

A. Research Contributions

The research contributions of the proposed framework are listed below :

- For the SH devices, we have introduced an ML-based intrusion detection approach. This approach used multiple models to assess the system's accuracy.
- Our framework integrates XAI techniques for feature extraction, which helps to find the most important intrusion detection features. Blockchain smart contract is formulated to store benign data.
- Comparison between the traditional ML approach and XAI-based approach is made. The models were assessed using accuracy, f1 score and precision-recall metrics.

B. Paper Organization

The paper is organised as: Section II represents the system model and problem formulation of proposed approach. The Explanation of the proposed approach is provided in Section III. Performance evaluation is represented by Section IV. In Section V, the paper is concluded.

II. System Model

The detailed system model for the proposed framework is presented in this section. Smart home H, utilizes IoT devices I for security of devices $I1$, $I2$, $I3$, and $I4$. There is a network attack scenario on these devices. The attack A comprises 12 different network attacks.

$$I \in H, \tag{1}$$

$$I = (I1, I2, I3, I4), \tag{2}$$

$$I \xleftarrow{\text{attack}} A, \tag{3}$$

$$A \in \{a_1, a_2, \ldots, a_{12}\} \tag{4}$$

The goal is to increase the probability $\psi_{\max}$ of detection of attack given IoT devices and the SH environment. SHAP-based feature extraction increases the probability of predicting the exact attack type.

$$\psi_{\max} = P(A|I, H) \tag{5}$$

The data classified as benign ($D \notin A$) by the ML model is sent to blockchain layer Υ for secure storage.

III. Proposed Approach

Fig. 1 represents the SH scenario, where external sources attack IoT devices. Fig. 2 represents our proposed approach with feature extraction layer, model predictions, and blockchain storage (Images and icons used are referenced from the online website [11].)

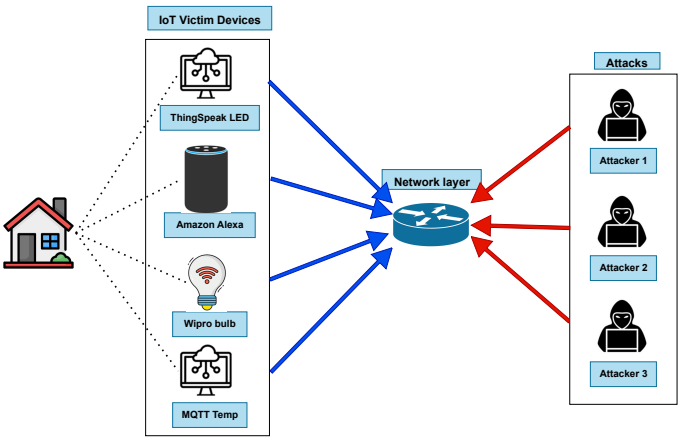


Fig. 1: Smart home environment.

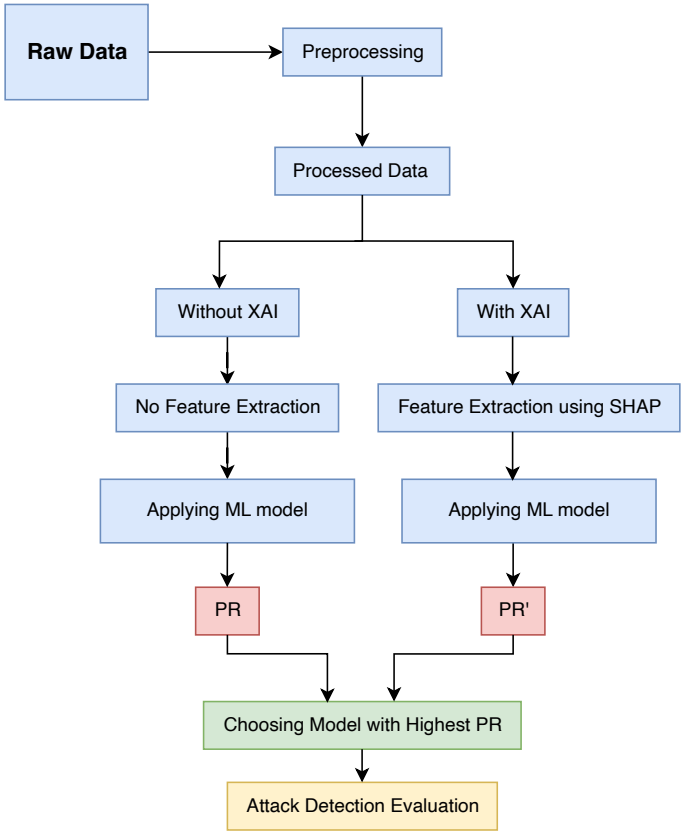


Fig. 2: XAI-based feature extraction and attack classification.

A. Smart Home

In today's technology world, every electronic device depends on smart solutions to increase efficiency and productivity. Smart Homes are rapidly becoming popular. Fig. 1 shows that our model includes devices such as Wipro bulb, MQTT-Temp, ThingSpeak-LED and Amazon Alexa that work together to make our life easier and have their own unique functionalities and capabilities in SH. The Wipro bulb represents an evolution in the lighting system by integrating smart features such as remote control, dimming capabilities and scheduling. Voice commands or smartphones can control these bulbs and are usually connected to the home network via WIFI or Bluetooth. As Fig. 1 shows, there is a risk in the communication protocols of these bulbs that can introduce cyber-attacks, which include unauthorized access or remote control by attackers.

A communication protocol called MQTT is used for IoT applications. Its architecture provides real-time data exchange between devices and servers. It is used for remote monitoring, home automation and industrial control systems. MQTT implementations can be risky to security threats such as unauthorized access and message tempering. An IoT platform called ThingSpeak allows users to collect, examine and visualize data from IoT devices. It is commonly used to control an LED remotely through IoT devices such as microcontrollers. Consider the example: users can send commands to its platform from a mobile application and based on commands, it can control the state of the LED. It can also lead to security threats. If communication between ThingSpeak LED and IoT devices is not encrypted, then attackers can manipulate the commands, leading to unauthorized access and data leakage. Amazon Alexa is a friendly virtual assistant that is always ready to help. You can perform any task by just giving voice commands. It responds only when you say the wake word. There have been cases where it misinterpreted other sounds as commands. If attackers find the risky portion of Alexa's software, there is the possibility of recording the sounds of your home and even taking control of SH devices.

B. XAI Based Feature Extraction and Attack Classification

1) Dataset Description: This section provides information about the dataset used, the preprocessing steps and how the attack classification takes place. The dataset used for training the model is an open dataset RT-IoT2022 [12]. Initially, the dataset had some inconsistencies, which we resolved through preprocessing using exploratory data analysis (EDA) techniques. This dataset is a complete resource that integrates many IoT devices with advanced network attack tactics. It is obtained from a real-time IoT infrastructure. It encompasses both benign and adversarial network behaviors, offering a comprehensive depiction of actual scenarios. The dataset includes data from IoT devices mentioned in the above subsection. It contains information on cyber attacks like SSH, DDoS, etc. Together, this data helps us to understand how network traffic behaves in different situations. The dataset comprises of 123117 rows, 83 features and 12 types of attacks.

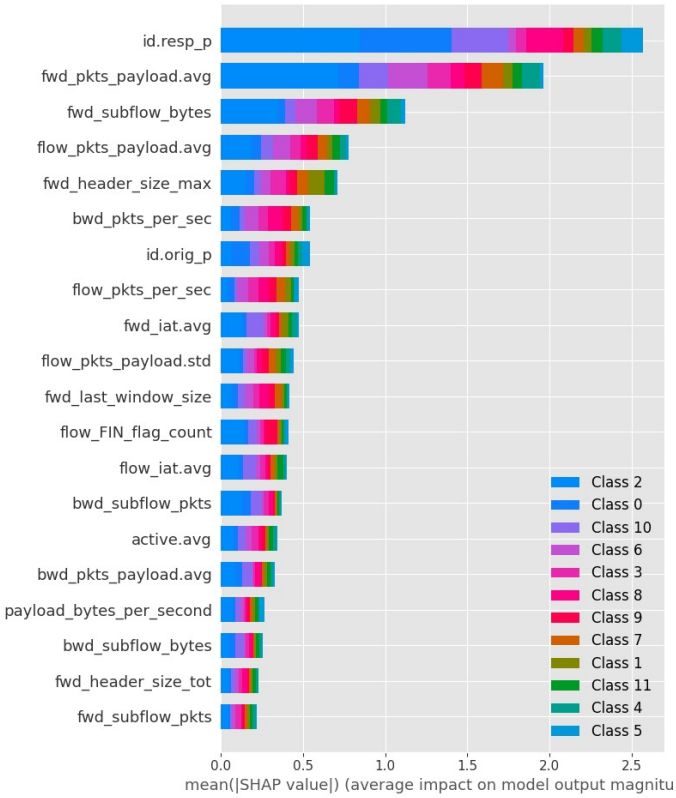


Fig. 3: Feature extraction using SHAP

2) Data Preprocessing: We removed specific attributes like source addresses, destination addresses, etc., which contain only unique IDs. We performed null value analysis and numerical attributes analysis. The categorical features like protocol and services are converted to numerical values. During training, 70% of the dataset is used, and 30% is used for inference time.

3) Comparing Traditional Approaches with XAI: Our methodology involved a two-phase approach to evaluate and improve our machine learning models. Firstly, we fed the raw data directly into an ML and found out its accuracy and the classification report for six different ML models. Then we fed the raw data to the XAI and used SHAP to find out the top 30 features contributing to the classification of attack types. After that, we ran the same six models with those 30 features and found out that after the feature extraction, the accuracy of some models increased, and better classification reports were made. This is because SHAP is a technique used in ML. It is based on ideas from cooperative game theory and helps to explain the predictions made by complicated models. By computing Shapley values, which quantify the marginal contribution of each feature to the model's output, SHAP provides local interpretability for understanding why specific predictions are made. It is model-agnostic, allowing its application to various ML models, and offers insights into the importance of different features in driving model predictions. Computationally, SHAP involves evaluating the

model's predictions for different feature subsets, which can be visualized to aid in understanding the impact of individual features on model behavior. This approach not only improves model performance but also enhances transparency and trust in our ML systems by providing interpretable explanations for model predictions.

In Fig. 3, we illustrate the process of ranking features based

Algorithm 1 Model Training and Feature Extraction

procedure TRAINANDEVALUATEMODELS(δ)
  $i \leftarrow 1$
  while $i \leq 6$ **do**
    $M_i \leftarrow$ TrainModel(δ)
    $Acc_i \leftarrow$ EvaluateModel(M_i, δ)
    $i \leftarrow i + 1$
  end while
  $F \leftarrow$ ExtractFeaturesUsingSHAP(δ)
  $i \leftarrow 1$
  while $i \leq 6$ **do**
    $M'_i \leftarrow$ RetrainModelWithFeatures(M_i, F)
    $Acc'_i \leftarrow$ EvaluateModel(M'_i, F)
    $\Delta Acc_i \leftarrow Acc'_i - Acc_i$
    $i \leftarrow i + 1$
  end while
  return $M_i, Acc_i, F, \Delta Acc_i$
end procedure

on their impact on the output predictor. To quantify this impact, we employ SHAP values. Larger values correspond to higher importance for the associated attributes. The horizontal axis represents the transition of SHAP values from blue to green. SHAP values measure the influence of each feature on the model's predictions. While the feature importance does not offer additional insights beyond the relative importance of attributes. Algorithm 1 explains the training and feature extraction part of our model.

ML Model Prediction:

$$\hat{a}_i = f(\mathbf{d}_i) \tag{6}$$

This equation predicts the probability distribution $\hat{a}_i$ over the 12 attack types for the ith instance using the ML model function $f(\mathbf{d})$, where $\mathbf{d}_i$ represents the feature vector of the ith instance. SHAP Values Computation: Subset Generation:

$$\gamma \subseteq \mu \setminus \{i\} \tag{7}$$

This part generates subsets γ of features, excluding feature i, where γ ranges over all possible subsets of features except the ith feature.

Prediction Difference Calculation:

$$\Delta f_\gamma = f(\mathbf{d}_{S\cup\{i\}}) - f(\mathbf{d}_\gamma) \tag{8}$$

For each subset S, this part calculates the difference in model predictions (Δf_S) between two scenarios: one where feature i is included ($S \cup \{i\}$) and one where it is excluded (S).

Weighted Summation:

$$\Phi_i = \sum_\gamma \frac{|\gamma|!(\mu - |\gamma| - 1)!}{\mu!} \Delta f_\gamma \tag{9}$$

The above equations are inspired from the original paper of SHAP [13]. The SHAP value Φ_i for the ith feature is computed as the weighted sum of all prediction differences (Δf_γ), where the weight is determined by the size of each subset S.

In the feature extraction process using SHAP, D represents the feature matrix of the dataset with dimensions $N \times \mu$, where N signifies the number of instances and μ denotes the number of features. Each feature vector $\mathbf{d}_i$ corresponds to the ith instance in the dataset, with dimensions $1 \times \mu$. a_i denotes the true label of the ith instance, indicating the attack type it belongs to, where a_i ranges from 1 to 12. The predicted probability distribution over the 12 attack types for the ith instance is represented by $\hat{a}_i$, obtained from the ML model function $f(\mathbf{d})$. The function $\Phi(\mathbf{d}_i)$ computes the SHAP values for the ith instance using SHAP. Table I shows the difference between the traditional and XAI based approaches.

C. Blockchain Layer

In Fig. 4 shows the SC function created to record the network communication of the smart home environment. During an attack detection, it records things like timestamp, type of attack, affected device. All the data of the benign class is recorded and stored on the blockchain. This data can be further used for analytics and transparency records. The public function getAttackLog in the SC can be used to find details of a particular transaction ID.

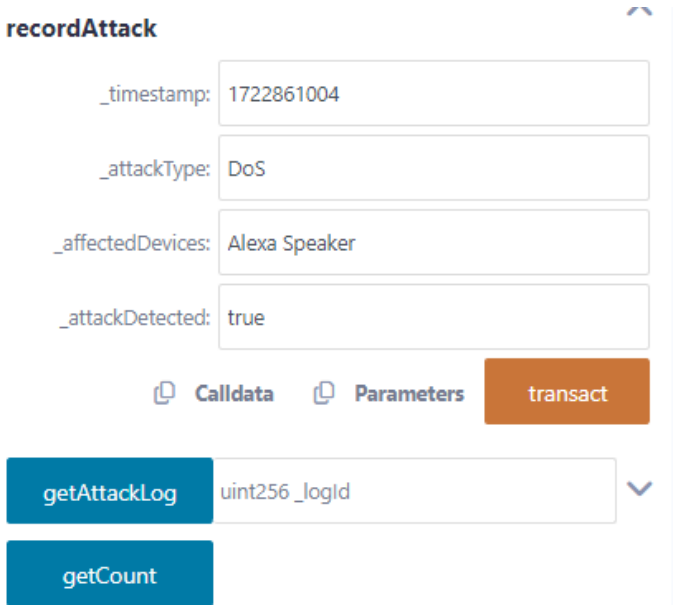


Fig. 4: Feature extraction using SHAP.

IV. PERFORMANCE EVALUATION

This section discusses the performance evaluation of the proposed XAI-based approach for the smart home environment.

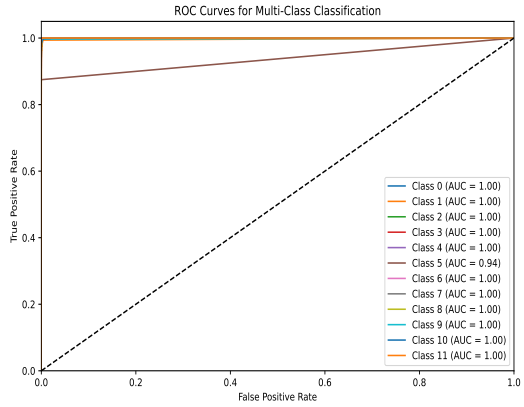
A. Simulation setup and tools

The system model has been simulated on Kaggle Notebook using Python v3.10.13 for our approach. For data preprocessing, Pandas v2.1.1 is used. Numpy v1.26.4 is used to perform operations on arrays. The data analysis and result visualization are carried by Matplotlib v3.7.5. For the purpose of Simulation, the system used is Apple Mac M2 Air which has an 8GB RAM, 8-Core CPU and 8-Core GPU. Keeping the same configurations throughout the testing increases reliability. Using their default parameters, we trained our proposed model and other ML models in TABLE I.

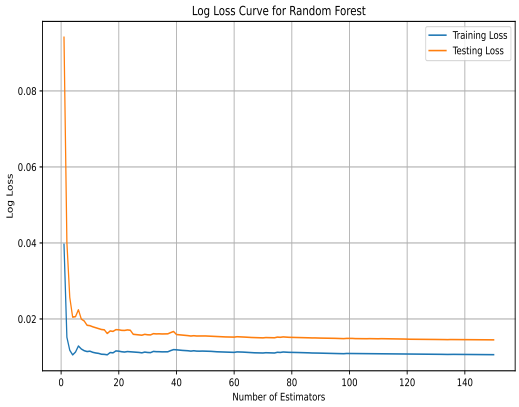
B. Performance analysis

In this section, we have performed the experimental analysis of the system model by comparing the accuracy of different models when features were extracted with the help of XAI using SHAP. TABLE I shows the performance analysis metrics of different models. The performance of the Random forest model achieved 99.69% accuracy, which is higher than before feature extraction was done using XAI, which was 98.03% of Random forest. We investigate various other evaluations metric such as precision, recall, f1 score and ROC- AUC curve also showed consistent results. This shows the effectiveness of our proposed model compared to other models. When we compared Random forest with other model it was clear that it outperformed every other model this is because of its versatility, Robustness of the Outliers and its high predicting capability.

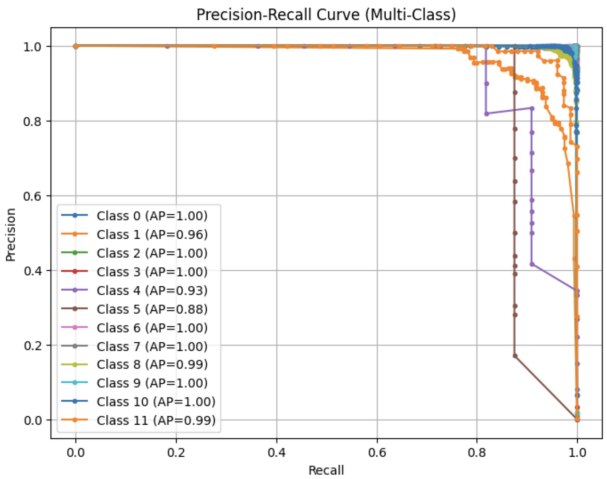
Fig. 5a represents the Receiver Operating characteristics (ROC) curve for our attack detection problem. The ROC curve estimates the classifier's performance when determining which thresholds to use for categorizing instances into classes. It plots the True Positive Rate (TPR) against the False Positive Rate (FPR) for different threshold values. The Area Under the Curve (AUC) measures the model's overall performance. Evaluation metrics like accuracy, precision, etc., can not determine if the model is biased towards some particular class. However, the higher value of the AUC of ROC for all classes determines our unbiased model. This signifies its ability to distinguish attack



(a) ROC-AUC curve



(b) Log loss curve



(c) PR curve

Fig. 5: (a) ROC curve for classes predicted for models, (b) Log loss curve for evaluating loss, and (c) PR curve helps in aging how well your model classifies different attack types.

TABLE I: Performance Metrics of Various Models

Model	Accuracy before feature extraction	Accuracy for XAI features	Precision	Recall	F1-score
Naive Bayes	0.75	0.77	0.88	0.77	0.80
K-Nearest Neighbors	0.97	0.98	0.99	0.99	0.99
MLP Classifier	0.94	0.93	0.93	0.94	0.93
SVM Classifier	0.83	0.83	0.77	0.83	0.78
Logistic Regression Classifier	0.79	0.78	0.80	0.79	0.79
Random Forest	**0.98**	**0.99**	**1.00**	**1.00**	**1.00**

types.

Fig. 5b represents the Log loss curve. This curve dives deeper into how well the model distinguishes between the 12 attack types. Lower log loss values on the y-axis signify the model's confidence in classifying a specific attack type, with higher values indicating difficulty in differentiating that attack from others.

Fig. 5c is Precision-Recall (PR) curve provides insight if the data is imbalanced during performance evaluation. Precision and recall are different performance metrics used for classification model evaluation purposes. We use them particularly significantly when the cost of FPs and FNs differs. The area under the PR curve is computed for each class and then averaged to obtain a single numerical value that summarizes the model's performance. Based on the application's particular needs, it assists in choosing a threshold that achieves an adequate equilibrium between precision and recall.

Fig. 6 shows the concussion matrix to provide a better understanding of how a model is performing on the test data for each attack type with its clarifications.

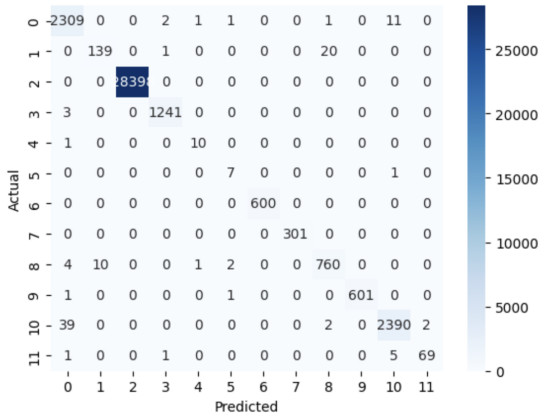


Fig. 6: Confusion matrix for predicted data class using Random Forest

V. Conclusion

This paper focuses on how to use XAI in novel ways to improve attack detection and explanation in Smart home IoT environments. These XAI have significantly increased the efficiency and accuracy of attack detection. The effectiveness of the proposed models has been confirmed through meticulous testing with datasets. These models offer a more reliable method for attack detection and explanation by addressing the drawbacks of previous techniques. This, in turn, has important ramifications, such as enhanced protection of SH networks and devices, quick reaction to attacks, and the development of thorough security plans. In the end, the suggested models produce a more trustworthy and safe ecosystem. To sum up, this paper's XAI feature extraction methods offer a promising way to improve attack detection and explanation in IoT environments.

For future works, we will use a Federated learning-based approach for building a privacy-preserved secure model for varied types of attack classification in the SH environment.Also implementing real-time attack classification in actual smart home environments and enhancing the system's explainability for non-expert users.

References

[1] M. Alaa, A. A. Zaidan, B. B. Zaidan, M. Talal, and M. L. M. Kiah, "A review of smart home applications based on internet of things," *Journal of network and computer applications*, vol. 97, pp. 48–65, 2017.

[2] M. Schiefer, "Smart home definition and security threats," in *2015 ninth international conference on IT security incident management & IT forensics*, pp. 114–118, IEEE, 2015.

[3] Z. Abou El Houda, B. Brik, and L. Khoukhi, ""why should i trust your ids?": An explainable deep learning framework for intrusion detection systems in internet of things networks," *IEEE Open Journal of the Communications Society*, vol. 3, pp. 1164–1176, 2022.

[4] N.-G. Vasilescu, P. Pocatilu, and M. Doinea, "Iot security challenges for smart homes," in *Education, Research and Business Technologies: Proceedings of 21st International Conference on Informatics in Economy (IE 2022)*, pp. 41–49, Springer, 2023.

[5] M. Wang, N. Yang, and N. Weng, "Securing a smart home with a transformer-based iot intrusion detection system," *Electronics*, vol. 12, no. 9, p. 2100, 2023.

[6] T. Li, Z. Hong, and L. Yu, "Machine learning-based intrusion detection for iot devices in smart home," in *2020 IEEE 16th International Conference on Control & Automation (ICCA)*, pp. 277–282, IEEE, 2020.

[7] B. Mahbooba, M. Timilsina, R. Sahal, and M. Serrano, "Explainable artificial intelligence (xai) to enhance trust management in intrusion detection systems using decision tree model," *Complexity*, vol. 2021, pp. 1–11, 2021.

[8] B. Lashkari and P. Musilek, "A comprehensive review of blockchain consensus mechanisms," *IEEE access*, vol. 9, pp. 43620–43652, 2021.

[9] T. Gazdar, "A new ids for smart home based on machine learning," in *2022 14th International Conference on Computational Intelligence and Communication Networks (CICN)*, pp. 393–400, IEEE, 2022.

[10] F. Alghayadh and D. Debnath, "A hybrid intrusion detection system for smart home security," in *2020 IEEE International Conference on Electro Information Technology (EIT)*, pp. 319–323, IEEE, 2020.

[11] "Flaticon." https://www.flaticon.com/, 2024. Accessed on July 22, 2024.

[12] B. S. and R. Nagapadma, "RT-IoT2022 ." UCI Machine Learning Repository, 2024. DOI: https://doi.org/10.24432/C5P338.

[13] S. M. Lundberg and S. Lee, "A unified approach to interpreting model predictions," *CoRR*, vol. abs/1705.07874, 2017.