Intelligent and Transparent Detection of Mirai Attacks in IoT Networks Using Lightweight Neural Networks

Danah A. Alkughail, Wedad M. Alawad *, Seera K. Alhatlani, Deem A. Algherair, and Jood H. Alolayan

Department of Information Technology, College of Computer, Qassim University, Buridah, Saudi Arabia
Email: danah.alkughail@gmail.com (Da.A.A.); wmaoad@qu.edu.sa (W.M.A.); Seera.Alhatlani@gmail.com (S.K.A.); deem.alghraier@gmail.com (De.A.A.); joodalolayan@gmail.com (J.H.A.)
*Corresponding Author

***Abstract*—As Internet of Things (IoT) devices increasingly integrate into contemporary existence, they have simultaneously emerged as attractive targets for extensive cyber assaults, particularly the Mirai botnet. Existing methods for identifying this virus typically require excessive computational resources for IoT devices with limited capabilities, while the quicker "lightweight" options frequently function as "black boxes", obscuring the decision-making process from users. This research presents a Lightweight Artificial Neural Network (LANN) integrated with Explainable Artificial Intelligence (XAI) using the Deep SHapley Additive exPlanations (DeepSHAP) framework to develop a detection system that is efficient and comprehensible. We trained the model to discover unusual attack patterns in the IoT-23 dataset that normal AI misses using a cost-sensitive learning technique. Our model had 99.22% accuracy and a 0.96 F1-score for minority attacks. Our DeepSHAP analysis explains its decisions and identifies *cat_proto_udp* as the key indicator of an attack, which matches Mirai behavioral patterns. This framework demonstrates that transparency and efficiency can be simultaneously accomplished, which makes it a viable option for protecting Internet of Things networks in the real world. It is a more practical and trustworthy solution for real-world Internet of Things (IoT) security than existing approaches, which either lack lightweight design, explainability, or both. The proposed framework, on the other hand, achieves all three objectives in a unique way, achieving a compact model size of 0.04 MB and an inference time of 0.096 s.**

***Keywords*—Mirai malware, botnet detection, Internet of Things (IoT), lightweight deep learning, Artificial Neural Network (ANN), machine learning, Explainable Artificial Intelligence (XAI)**

I. Introduction

The expansion of Internet of Things (IoT) devices has led to notable gains in several sectors, while also increasing the vulnerability to cyber-attacks. Even in 2025, the Mirai malware is still very dangerous, with new versions actively taking advantage of flaws in current IoT systems. Mirai was created in 2016, but it still affects hundreds of thousands of computers around the world. It does large-scale Distributed Denial-of-Service (DDoS) attacks and gets better at getting around security measures.

This persistence makes detecting and defending against Mirai in resource-constrained environments difficult. Traditional malware detection systems work well in enterprises but are computationally intensive for IoT devices with minimal processing power and memory. In response, researchers offer lightweight IoT deep learning algorithms. However, many of these models are "black boxes", providing little insight into decision-making. Modern network security paradigms use advanced data balancing, heuristic optimization, and software-defined architectures to improve intrusion detection across big data, edge-assisted environments, and diverse IoT ecosystems [1–3]. This opaqueness is a significant obstacle to adoption, particularly in security-critical industries, where stakeholders, such as industry operators, regulators, and cybersecurity teams, require systems that are trustworthy and interpretable in order to validate alarms and establish compliance.

Using a combination of Explainable Artificial Intelligence (XAI) approaches and lightweight deep learning models, the purpose of this project is to investigate how to detect Mirai malware in Internet of Things environments. This will be done in order to address the issues that have been presented. The technique that has been proposed has as its primary objective the enhancement of model interpretability while simultaneously preserving computational efficiency. This allows reliable identification of compromised IoT devices and human-understandable justifications for detection decisions. This research examines how well a lightweight Artificial Neural Network (ANN) can identify Mirai malware in low-resource Internet of Things (IoT) environments, how well XAI methods provide meaningful interpretability without excessive computational overhead, and how XAI and lightweight models affect trust and usability in real-world IoT security deployments.

Manuscript received March 14, 2026; revised June 3, 2026; accepted August 10, 2026; published November 12, 2026.

II. Literature Review

A. Theoretical Background

1) Mirai attack

A prominent scenario that underscores the weaknesses of IoT devices is the Mirai botnet assault (see Fig. 1). The use of Internet of Things devices in cyber warfare was altered as a result of this in October of 2016. Mirai, a computer virus, exploited vulnerabilities in IoT, such as those found in Closed-Circuit Television (CCTV) cameras and Digital Video Recorder (DVR) systems, to build a massive botnet that was capable of launching distributed denial of service attacks. These assaults caused disruptions to On Vous Héberge (OVH), Dyn, and Krebs on Security, among other websites. The effectiveness of Mirai can be attributed to the widespread use of default passwords and other security flaws in Internet of Things devices. The severity of the situation highlighted the critical requirement for more stringent Internet of Things security standards, in particular with regard to authentication systems [4].

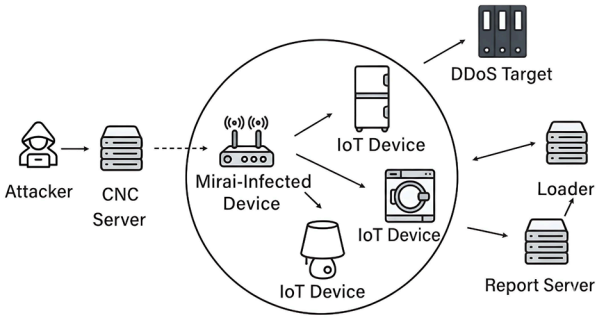


Fig. 1. Mirai botnet architecture.

2) Artificial Neural Network (ANN)

The structure and function of the human brain serve as a source of motivation for the development of computational models that are referred to as Artificial Neural Networks as well. They are composed of interconnected "neurons", also known as nodes, so that they can process information and generate something new.

3) Lightweight models

Lightweight models are machine learning frameworks that decrease size and processing while maintaining accuracy. Smartphones, embedded systems, and IoT devices with low resources benefit from these paradigms. Shorter inference times, memory, and energy usage result from their fewer parameters and computational complexity. Their configuration optimizes security in IoT contexts, where efficacy is crucial [5].

4) Explainable Artificial Intelligence (XAI)

Explainable Artificial Intelligence (XAI) is a new topic of artificial intelligence research that tries to make AI system decision-making understandable and interpretable by end-users (see Fig. 2) [6]. Due to the increasing prevalence of artificial intelligence in healthcare, banking, and autonomous systems, transparency and accountability are of utmost importance. Traditional deep learning models are quite effective, but they are frequently referred to as "black boxes", which means that their underlying workings are not well understood. The inability to perceive anything might be detrimental to user trust and adoption.

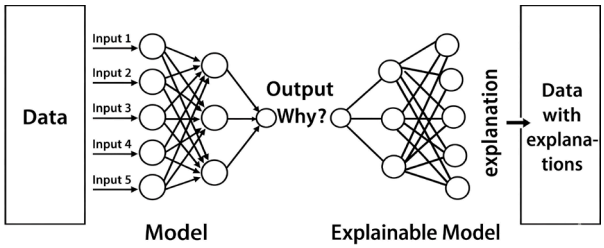


Fig. 2. A post-hoc explanation module inside an explainable AI architecture.

5) Shapley Additive Explanations (SHAP)

A prominent XAI method for evaluating machine learning model predictions is Shapley Additive Explanations (SHAP). SHAP assigns each feature a Shapley value based on cooperative game theory to quantify its impact on model prediction. Coalition "players" and SHAP are used to quantify their impact [7]. The local precision and consistency of SHAP is where the strengths of the program lie. In order to provide a clear and understandable description of the prediction process, it disassembles the prediction of a model into the additive contributions of its various features. Being able to build confidence in artificial intelligence systems that are used in sensitive industries is made possible by SHAP [7].

B. Related Work

In order to identify and analyze recent research on IoT malware detection, this study conducts a systematic literature review. The focus is on lightweight and explainable approaches that are focused on Mirai-based attacks. A structured search strategy was implemented, utilizing predefined keywords such as "Explainable Artificial Intelligence (XAI)", "Mirai attack", "lightweight deep learning", and "IoT malware detection". Multiple academic electronic databases were consulted to retrieve pertinent studies, which encompassed publications from 2019 to 2025. The selected works were assessed using clear inclusion and exclusion criteria. IoT settings, lightweight model architectures, and explainable AI methodologies were evaluated, however works not related to IoT security or malware detection were eliminated. It was necessary to obtain methodological and performance data from selected study in order to facilitate comparative analysis, identify trends in accuracy, and indicate their boundaries. This review uncovered gaps in the research and served as the basis for the framework, as summarized in Table I.

TABLE I. COMPARATIVE SUMMARY OF RELATED WORK ON MALWARE DETECTION IN IOT SYSTEMS

Ref.	Year	Algorithm	Attack Type	Accuracy	Lightweight	XAI	Strength(s)	Weakness(es)
Proposed work	2026	LANN + DeepSHAP	Mirai	99.22%	Yes	Yes	High accuracy, low resource consumption, and provides deep explanations for network features	Requires validation on broader multi-class IoT datasets
[3]	2025	Harmony Search Algorithm-Adaptive Synthetic Sampling (HSA-ADASYN) Framework	Skewed IoT Intrusions	99.94% (CIC-IoT23) /96.92% (UNSW-NB15)	No	No	Combines Harmony Search with adaptive sampling to handle highly imbalanced IoT traffic datasets robustly	Computationally intensive data-generation process; does not feature a lightweight deployment footprint
[1]	2025	Gravitational Search Algorithm (GSA) + Synthetic Minority Over-sampling Technique with Iterative Partitioning Filter (SMOTE-IPF) + Random Forest	Big Data Network Intrusions	99.60% (NSL-KDD) /94.40% (UNSW-NB15)	No	No	Effectively mitigates severe class imbalance using SMOTE-IPF; high accuracy in big data environments	High computational complexity due to optimization heuristic; not designed for resource-constrained IoT
[8]	2025	Deep Neural Network (DNN) (2-layer Multi-Layer Perceptron (MLP))	DDoS, DoS, MITM, brute-force, botnets	99.5% (binary), 98% (multi-class)	Yes	Yes	Federated Learning-Explainable Artificial Intelligence (FL-XAI) integrates SHAP and Local Interpretable Model-agnostic Explanations (LIME), privacy-preserving, scalable to distributed IoT	Centralized XAI adds computational overhead, assumes an ideal setup
[9]	2025	Naïve Bayes (NB), Random Forest (RF), Support Vector Machine (SVM), K-Nearest Neighbors (KNN)	Mirai	99.9%	Yes	No	Lightweight real-time framework for 6G-enabled IoT; memory-efficient	Limited scalability, lacks adversarial robustness, and tested only on small-scale setups
[10]	2025	CNN-LSTM + PCA + IG	DDoS, DoS, botnets, zero-day	99.1%	Yes	Yes	High accuracy, SHAP + attention explainability, real-time edge performance	Requires GPU training, attention interpretability complexity
[11]	2025	KNN + RF	DDoS, brute-force, data exfiltration, C2 comms	99.9%	Yes	Yes	Multi-class detection, low false positives, transparent via SHAP + LIME	Offline model, KNN is costly for large data, not yet real-time
[12]	2025	Random Forest (RF)	Mirai, Bashlite	99.98%	No	Yes	Combines high accuracy with interpretability; provides both global and local explanations	Not lightweight, focuses on binary classification, not tested in real-time environments
[2]	2024	AI-based IDS/IPS (TLDQN + Bi-GAN)	Edge-assisted SDWSN Intrusions	>90.00%	Partial	No	Uses a modified honeycomb structure to optimize routing and threat prevention at the edge	Tailored specifically to SDWSN architectures; lacks post-hoc model explainability features
[13]	2024	LSTM + XGBoost	Botnet	99.99%	No	Yes	High detection accuracy, interpretable via SHAP, handles imbalanced/time-dependent IoT data	Evaluated mainly on specific dataset, limited real-time testing
[14]	2024	Ensemble (kNN, SVM, LR, NB, MLP) + Q-Learning + MFO	Hybrid IoT intrusions	Not specified	Partial (No)	Yes	Adaptive detection via Q-learning and XAI interpretability	No standard dataset, high complexity, limited scalability
[15]	2024	BEFSONet	Zero-day attacks, DDoS	97.99%	No	Yes	High accuracy, optimization using SHO reduces computational complexity, handles large datasets, explainable AI	Lack of real-time implementation, hybridization of BERT and SHO adds deployment complexity
[16]	2024	Multiple AI models (DT, DNN, MLP, SVM, ADA, ensembles)	Mirai, Gafgyt	99%	No	Yes	Integrates seven XAI techniques (SHAP, LIME, Leave One Covariate Out (LOCO), Contrastive Explanations Method (CEM), Accumulated Local Effect (ALE), Permutation Feature Importance (PFI), ProfWeight) for interpretability	Heavy computational load, not tested in real-time environments

[17]	2024	Random Forest (RF)	DDoS, brute-force, infiltration, botnet C2	99.8%	No	Yes	Feature importance via SHAP improves analyst trust, efficient training	Offline evaluation, limited adaptability to unseen attacks
[5]	2024	LMDNet	Malware family classification	99.07%	Yes	No	High accuracy, faster recognition speed, surpassing traditional methods by 23.68%	Requires specific IoT training for IoT hardware
[18]	2023	CNN, LSTM, GRU	Mirai	98.7–96%, 93–96%	Yes	No	High accuracy, use of multiple deep learning models, comprehensive evaluation across different IT devices and attack types	High computational requirements, not stable for real-time detection, cannot detect zero-day attacks
[19]	2023	XGBoost	DoS, MITM, sniffing	99.99%	No	Yes	High accuracy, different datasets and attacks, explainability	High computational power, not hybrid or ensemble
[20]	2023	LSTM	DoS, Generic, Exploits, DDoS	99.1%	No	Yes	Use of explainable AI (SHAP) for interpretability	Complexity in applying XAI techniques to deep learning
[21]	2023	ANN, SVM, KNN	Mirai	99.56%, 93.51%, 99.55%	Yes	No	High accuracy, target Mirai malware, evaluation of different dataset sizes	Lack of adaptation for evolving attack patterns, lacks in-depth analysis of false positives/negatives
[22]	2022	RNN, LSTM, Bi-LSTM	Malware detection in textual data	99.45%	Yes	No	High accuracy, efficient use of minimal parameters, reduced overfitting	Limited to text-based malware detection
[23]	2022	LSTM	Botnet	99%	No	No	High accuracy, low false-positive rates	Doesn't address numerous attacks
[24]	2022	Decision Trees, Random Forests, SVM	DDoS, web attacks, port scans	96.25%	No	Yes	High accuracy achieved with ensemble model using voting classifiers, LIME enhances interpretability	Complexity reduces interpretability, CIC-IDS-2017 dataset has NaN values and imbalanced classes
[25]	2022	IoT Malware Detection Architecture (iMDA)	IoT malware attacks	97.93%	Yes	No	High accuracy, efficient use of minimal parameters making it suitable for IoT, reduced overfitting compared to complex models	High computational demand, relies on visual features
[26]	2022	Inception-v3 CNN	Banking data attacks	98.5% (binary), 91% (multiclass)	No	Yes	High accuracy, using CNN for better detection, explainable AI	No real-time usage, not hybrid, high computational power
[27]	2022	DNN	DDoS, probing, backdoors	99%	No	Yes	Integrates XAI techniques (LIME, SHAP, RuleFit) for interpretability, offers local and global explanations	Complexity in applying XAI techniques to deep learning
[28]	2022	TinyML	Not applicable	90.6–93.2%	Yes	No	On-device TinyML processing reduces latency and communication needs, accurate material classification	Limited generalization, computationally expensive
[29]	2021	2-MaD	Various types of malwares	94.46%	No	No	High accuracy, low false-negative rate, hybrid	The model isn't stable for real time because it tested in black box, it consumes computational resources so it's not ideal for IoT, takes time
[30]	2019	IoTNet	Not applicable	89.9%	Yes	No	High accuracy, optimized for IoT devices environment	Limited to classification tasks with small image datasets

Recently, Internet of Things malware detection and intrusion prevention research has prioritized explainability. From lightweight models to hybrid deep learning frameworks, this study examined many methods. Wang *et al.* [5] introduced LMDNet, a lightweight deep learning model for Internet of Things malware detection and family classification, using preprocessing on malware binaries, knowledge distillation from ResNet-34 and VGG-16, channel shuffling, and Gradient-weighted Class Activation Mapping (Grad-CAM). This model optimized recognition speed and obtained 99.07% accuracy. It is effective but requires IoT-specific training and constant adaption to new malware types. Almazroi and Ayub [15] came up with the idea for BEFSONet, an Internet of Things framework that combines BERT with the Spotted Hyena Optimizer (SHO). This framework achieved 97.99% accuracy on the IoT-23 dataset, particularly when it came to protecting against Mirai-based attacks. The

intricacy of the model and the poor practical explainability of the concept, on the other hand, can limit its application in Internet of Things devices operating with low resources [7]. Giordano *et al.* [28] proposed a power tool monitoring system based on accelerometer data, preprocessed with a Fast Fourier Transformation (FFT) and a lightweight neural network, with an accuracy of 90.6% on 4 activity classes, and a model size of 30 kB, showing edge-device efficiency but poor generalization.

Other lightweight and hybrid IoT malware models include the CNN-LSTM framework by Akhtar and Feng [23], achieving 99% F1-score using 43,867 samples from five malware families, outperforming Decision Trees and SVM but not covering a wide range of attack types. Baek *et al.* [29] proposed Two-Stage Hybrid Malware Detection (2-MaD) framework combining static opcode analysis using Bi-LSTM model and dynamic behavioral analysis using EfficientNet-B3 with 98.94% detection accuracy. But the addition of a second dynamic analysis stage can incur an additional computational overhead compared to single stage approaches. Asam *et al.* [25] introduced iMDA, a hybrid neural network with multi-path dilated convolutions, channel squeezing, and boosting, achieving 97.93% accuracy, 0.9394 F1-score, and 0.8796 MCC on 14,733 malware and 2486 benign ELF binaries, though computationally intensive and reliant on visual features. Khan *et al.* [22] proposed lightweight RNN, LSTM, and Bi-LSTM models for near real-time IoT malware detection using tokenized malware strings, achieving 99.46% test accuracy, but limited to text-based detection.

A study by Güven and Gürkaş-Aydin [21] targeted Mirai botnet attacks using ANN, KNN, and SVM on the Kitsune dataset, achieving 99.56% accuracy with bootstrapped subsets while reducing computational requirements. Similarly, Sharma *et al.* [18] evaluated CNN, LSTM, and GRU on the N-BaIoT dataset, achieving 98.7% accuracy with CNN, though requiring high computational power and lacking real-time stability. Siganos *et al.* [19] applied Random Forest and XGBoost with SHAP for interpretability on IEC 60870-5-104 and CIC-IoT-Dataset-2022, achieving up to 99.99% accuracy but without hybrid or ensemble techniques and without explicit malware detection focus.

Explainable AI has been widely investigated to improve transparency. Patil *et al.* [24] applied LIME in a machine learning IDS on CICIDS-2017, reducing 80 features to 10 for classification, with an ensemble voting classifier achieving 96.25% accuracy, though ensemble complexity reduced interpretability. Houda *et al.* [27] integrated LIME, SHAP, and RuleFit with a DNN on NSL-KDD and UNSW-NB15, achieving 99% accuracy and providing local and global explanations. Keshk *et al.* [20] combined LSTM with the SPIP framework (SHAP, PFI, ICE, PDP) for IoT intrusion detection, achieving 99.1% accuracy across multiple datasets while providing interpretability and feature selection.

Naeem *et al.* [26] converted malware binaries into color images for Inception-v3 CNN with Grad-CAM, achieving 98.5% accuracy, but requiring high computational resources, limited to Android OS, and lacking real-time deployment. Lawrence and Zhang [30] improved CNN efficiency on constrained IoT devices via factorized convolutions, reducing Floating-Point Operations (FLOPs) while maintaining accuracy on small image datasets, though limited to small classification tasks.

The study by Saied and Guirguis [12] introduced an Explainable Artificial Intelligence (XAI)-based framework for botnet detection in IoT environments, addressing the growing need for transparency and trust in AI-driven cybersecurity systems. After applying tree-based machine learning methods, the authors included XAI techniques to improve intrusion detection model interpretability. The study compared six classifier, Decision Tree (DT), Random Forest (RF), Bagging Meta Classifier (BMC), Adaptive Boosting (ADB), Gradient Boosting (GDB), and Extreme Gradient Boosting (XGB) to identify the most effective model for IoT botnet detection. Random Forest has the greatest detection accuracy of 99.98% in N-BaIoT experiments with Mirai and Bashlite attacks. Three explainability methods—LIME, SHAP, and Rule Extraction—provided local and global interpretability of model choices to overcome classic Machine Learning (ML) models' black-box limitations. Analysts can understand classifications thanks to high accuracy and explainability, improving decision-making and user confidence. Although transparent and durable, the technique is not lightweight and confined to binary categorization [17].

Gummadi *et al.* [16] developed XAI-IoT, a framework that combines Explainable Artificial Intelligence (XAI) with anomaly detection to improve IoT system security and interpretability. In search of transparent and trustworthy AI-based cybersecurity solutions, the study tested several machine learning and ensemble models to increase detection accuracy and explainability. Decision Tree (DT), Deep Neural Network (DNN), Multi-Layer Perceptron (MLP), Support Vector Machine (SVM), AdaBoost (ADA), and ensemble approaches such Random Forest (RF), Bagging, Stacking, Voting, and Blending were examined. MEMS, containing IoT sensor data, and N-BaIoT, including Mirai and Gafgyt botnet attacks, were used for experiments. Ensemble models, especially Random Forest and Voting, had the greatest detection accuracy of 99% across both datasets. To enhance interpretability, seven XAI techniques (SHAP, LIME, PFI, LOCO, CEM, ALE, and ProfWeight) were applied, providing both local and global explanations of model behavior. However, the models are computationally heavy and have not been tested in real-time IoT environments, limiting their deployment potential [16].

Alwaisi [9] presented a TinyML-based real-time anomaly detection framework designed to secure 6G-enabled IoT networks against Mirai botnet attacks. Motivated by the need to enhance cybersecurity in resource-constrained IoT devices, the study aimed to achieve high detection accuracy while maintaining low memory and computational overhead. Detecting and

classifying four primary types of Mirai attacks, namely Scan, User Datagram Protocol (UDP) Flooding, TCP Flooding, and ACK Flooding, was accomplished by the framework through the utilization of four machine learning classifiers that were optimized. These classifiers were Naïve Bayes (NB), Random Forest (RF), Support Vector Machine (SVM), and K-Nearest Neighbors (KNN). According to the findings, KNN was able to reach an accuracy rate of more than 99% while utilizing a minimal amount of memory (less than 20 MB). This makes it the most suited model for real-time Internet of Things (IoT) security. Additionally, the framework incorporated edge intelligence and federated learning, which aligned with the decentralized architecture of 6G, with the goal of further improving detection speed and privacy [9].

Niktabe [13] developed an AI-based intrusion detection framework for IoT networks that is both explainable and comprehensible. This framework combines SHAP (Shapley Additive Explanations) with Long Short-Term Memory (LSTM) deep learning to identify and interpret malicious IoT behavior. The framework enhanced transparency and analyst trust by providing local and global insights into model decisions using SHAP visualizations and achieving high detection accuracy (99.99%) [13].

A federated learning framework that is integrated with explainable artificial intelligence (FL-XAI) was proposed by Bilal *et al.* [8] for the purpose of detecting malicious traffic in distributed Internet of Things networks. A lightweight deep neural network that was trained collaboratively across edge devices using FedAvg was utilized by the system. This allowed for the protection of data privacy by preventing the centralization of raw data for processing. Both SHAP and LIME were utilized in order to provide feature attributions on both a global and instance-level, thereby increasing the transparency of the model. The model matched centralized baselines with 99.5% binary classification accuracy and 98.0% multi-class accuracy on Edge-IIoT, CIC-IoT2023, and TII-SSRC-23 datasets. While highly interpretable and privacy-preserving, the centralized XAI module may add computational overhead, and real-time deployment on constrained devices requires optimization [8].

Explainable and Lightweight Artificial Intelligence (ELAI) is a framework that was proposed by Rahmati [10] for the purpose of real-time cyber threat detection in edge networks. This framework was designed to solve the constraints of existing deep learning models, including the high computing cost and the lack of interpretability. Principal Component Analysis (PCA), information gain, and a hybrid CNN-LSTM architecture with attention mechanisms were all incorporated into the system in order to strike a balance between detection accuracy and efficiency. SHAP-based feature attribution and attention heatmaps made threat classification transparent. ELAI outperformed CNN-LSTM, ResNet-50, and Transformer-based IDS models in detection and inference time on CICIDS 2017, UNSW-NB15, and N-BaIoT datasets with 99.1% accuracy and 99.0% F1-score [10].

A dynamic intrusion detection framework for IoT networks by Turaka and Panigrahy [14] classifies hybrid and developing attack patterns using ensemble learning, Q-learning, and Explainable AI (XAI). Multidomain features were extracted from network logs and optimized using Moth Flame Optimizer. KNN, SVM, LR, NB, and MLP models were used to classify, with Q-learning dynamically refining detection based on micro-attack signatures. The model indicated high potential for scalable, interpretable, and adaptive intrusion detection in heterogeneous Internet of Things environments, despite the fact that the actual dataset details and absolute performance metrics were not provided [14].

Patel [11] introduced the Hybrid KNN-Random Forest (HKRF) model for explained and computationally efficient IoT botnet detection, integrating PCA-based dimensionality reduction, SMOTE balancing, and SHAP + LIME for transparent decision interpretation. Training on CICIDS-2017 and CTU-NCC datasets yielded 99.9% accuracy across multiple attack categories at low computational cost compared to deep networks. The offline model is accurate and interpretable, but it needs adaption for real-time deployment.

Yousaf *et al.* [17] also used standard machine learning for botnet identification using CICIDS-2017 dataset and the Random Forest achieved 99.8% accuracy with increased transparency by means of SHAP framework based on feature attribution. With faster inference on moderate hardware, the model is simple and efficient, lacking deep architectures. However, its scalability to evolving attack scenarios is restricted by its reliance on feature engineering and lack of online review [17].

Finally, the researchers have proposed a variety of hybrid and optimized frameworks to address structural issues such as severe class imbalances and architectural restrictions in complex network infrastructures. For instance, Kumari *et al.* [1] suggested an enhanced Intrusion Detection System (IDS) for big data environments, which combines Gravitational Search with SMOTE-IPF data balancing to greatly enhance minority class identification and overall accuracy. In a comparable manner, to enhance security in decentralized systems, Kipongo *et al.* [2] proposed an AI-driven intrusion detection and prevention framework specifically designed for edge-assisted Software-Defined Wireless Sensor Networks (SDWSN) employing a modified honeycomb configuration to improve traffic routing and node protection. Shana *et al.* [3] also addressed data skewness in IoT environments by building a hybrid HSA-ADASYN framework that combined Harmony Search Algorithms with Adaptive Synthetic sampling to ensure effective intrusion detection metrics against unusual attack vectors. Despite the fact that these frameworks make use of sophisticated data-balancing and optimization mechanisms in order to improve detection precision, they do not inherently integrate lightweight deep neural networks with post-hoc explainability frameworks such as

DeepSHAP. This highlights the unique space that is occupied by the proposed LANN strategy.

Based on the reviewed studies and the comparative analysis summarized in Table I, explainability and lightweight design are underrepresented in IoT malware detection research. Few research have examined deep, lightweight, and explainable models, but most have focused on them separately. Many methods improve interpretability without maximizing computational performance, while others reduce model complexity and resource consumption without transparency. This imbalance indicates a research gap, prompting a framework that unifies interpretability and computational efficiency for IoT security applications.

III. Methodology

A. Dataset Collection

In order to allow the efficient and transparent detection of Mirai malware in Internet of Things environments, this study makes use of the IoT-23 dataset, which is curated by Stratosphere Lab and consists of 23,145 tagged header packets [31]. This dataset is suited for binary classification and contains benign and malicious traffic (including Mirai related behavior). Packet size, flow duration and connection frequency are recorded to investigate Mirai infection communication trends. Explainability principles impact feature selection, such as correlation analysis to reduce redundancy and refine relevance, thereby improving model interpretability and bias. The selected features are trained on a Lightweight Artificial Neural Network (LANN) to build a computationally cheap and transparent intrusion detection model for resource-constrained IoT devices. The structured dataset with well-defined columns and annotated instances makes training and evaluation simpler.

B. Dataset Preprocessing and Feature Selection

Raw packets from Internet of Things network traffic are being analyzed. The removal of noise and the handling of missing values are both processes that are performed during data preprocessing. The dataset was first checked for incomplete or missing packets due to unstable network conditions. When crucial for model training, packets were discarded or imputed.

1) Feature engineering and normalization

An extraction of relevant packet header fields that were suitable for analysis was carried out as part of the feature engineering process. This was done to ensure that the most important attributes were preserved for modeling purposes. In light of the fact that the characteristics of Internet of Things network traffic vary greatly in terms of range and units, such as packet size, transmission duration, and frequency, data normalization was utilized in order to scale all numerical features to a similar range. This helped to improve the stability of the model and made it possible to compare features in an equitable manner.

2) Label encoding and transformation

To construct the label column, which contains categorical labels, appropriate for model training, we extracted it from the dataset and encoded it using the LabelEncoder function. Separate from numerical features, we identified categorical features like proto, service, and conn_state in order to apply the appropriate preprocessing. To standardize numerical characteristics with *StandardScaler* and encode categorical features with *OneHotEncoder*, a *ColumnTransformer* was defined.

3) Feature selection

The first attempt was done with gradient boosting methods (XGBoost) for feature selection, but the results were less consistent and interpretable. As a result, a Random Forest Classifier was utilized, which displayed exceptional performance in determining which characteristics were the most pertinent. In order to train the Random Forest model on the preprocessed dataset, the SelectFromModel function was utilized. This function was used to preserve only the features that were deemed to be the most essential based on their importance scores. The features selected for training the final model include: num_id.orig_p, num_id.resp_p, num_orig_pkts, num_orig_ip_bytes, num_resp_pkts, cat_proto_tcp, cat_proto_udp, cat_service_dns, cat_conn_state_OTH, and cat_conn_state_SF. In the process of further modeling, features that were deemed to be less important or less relevant were excluded. These features included num_missed_bytes, num_resp_ip_bytes, cat_service_-, cat_service_dhcp, and cat_conn_state_S0.

4) Visualization of feature importance

Feature importance scores from the Random Forest model were used to visualize the relative contribution of each feature to the prediction process. As shown in Fig. 3, a number of features, including cat__proto_tcp, cat__proto_udp, num__id.orig_p, and num__id.resp_p, were found to have the greatest importance values, which indicates that these features have a considerable influence on the decisions that are performed by the model. For the most part, the features that were chosen correspond to the different types of network protocols and the fundamental connection IDs. These features are logically consistent with the traffic behavior of the IoT.

5) Dataset splitting

Data was split into training (80%) and testing (20%) sets. We automatically validated 20% of training data to assess performance and prevent overfitting.

6) Handling class imbalance

The dataset exhibited a significant class imbalance, as the minority class (Class 0) was inadequately represented relative to the majority class (Class 1). This disparity may skew machine learning models towards the majority class, resulting in exaggerated accuracy and an inability to correctly recognize minority instances, often the most crucial in intrusion detection contexts.

For the purpose of resolving this issue, Cost-Sensitive Learning assigned higher class weights to the minority class within the loss function in order to penalize misclassification. Using this strategy, the model is able to concentrate on uncommon but significant cases. Earlier than the implementation of class weights, the baseline

model had a macro-averaged F1-score of 0.99, which favored the class that was in the majority.

The recall for the minority class went up to 1.00 after cost-sensitive learning, and the F1-score for class 0 went up to 0.96, showing better balance without lowering overall accuracy.

Cost-Sensitive Learning is widely acknowledged as an efficient approach for addressing class imbalance in network traffic datasets. For example, the others in Ref. [32] proposed a group-based cost-sensitive approach that dynamically assigns weights by group Distributionally Robust Optimization (group-DRO) that improves the recall and F1-score for the minority classes while maintaining the overall accuracy.

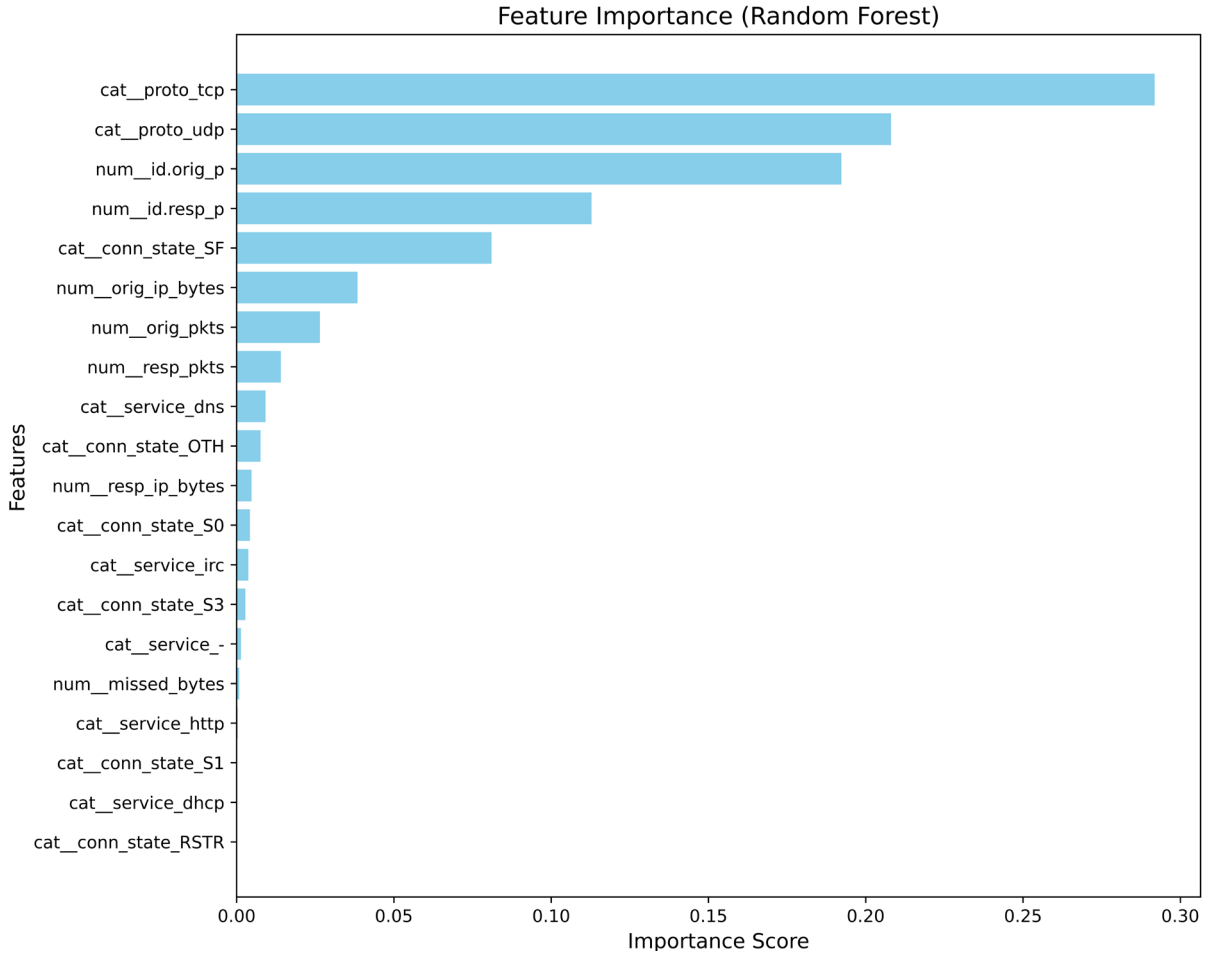


Fig. 3. Importance of features generated by random forest algorithm.

C. *Lightweight Model Design and Implementation*

This study suggests and tries to put into action a Lightweight Artificial Neural Network (LANN) model that is best at finding Mirai botnet attacks. This will help IoT devices with limited resources make very accurate detections. Following a method based on the Artificial Neural Network (ANN) technique, the model is made with as few layers as possible while still being very accurate. Even though ANN models haven't been used a lot to find malware, prior studies demonstrate that Mirai malware patterns can be identified by packet header behaviors that match ANN's ability to capture nonlinear relationships in data [21, 33]. The proposed LANN model architecture has an input layer, three hidden layers, an output layer, and a Sigmoid activation function. Network traffic parameters including packet size, protocol type, and flow rate from the IoT23 dataset are supplied to the input layer to distinguish normal from malicious traffic. The hidden layers, with 6, 9, and 5 neurons, are fully connected to previous and subsequent layers to capture complex data patterns while maintaining IoT-friendly computational efficiency. Binary classification is carried out by the output layer, which assigns a label to each instance, indicating whether it is normal or an attack. The utilization of the Sigmoid activation function throughout all layers guarantees that inputs are confined to the interval [0, 1], facilitating gradual convergence and consistent learning dynamics during the training process. This design achieves a perfect harmony between detection efficacy and lightweight processing, successfully attaining the ideal equilibrium.

LANN training uses Mirai attack and typical network traffic. One input layer (for selected network features), three hidden layers, and one binary classification output layer comprise the architecture. To avoid overfitting, these layers extract and categorize features using cross-validation. Bootstrapping with 1000, 10,000, and 100,000 entries improved performance and saved

resources. A learning rate of 0.01, binary cross-entropy loss function, and accuracy performance metric trained the model using Adam optimizer. Training had 20 epochs, 32 batches, and 20% data validation. Python utilized Pandas for data manipulation, NumPy for numerical operations, scikit-learn for preprocessing and evaluation, TensorFlow and Keras for neural network training, matplotlib and seaborn for visualizations, and SHAP for model interpretation. The model was tested on a separate test set after training, assessing test loss and accuracy and thresholding probabilistic predictions at 0.5 for binary labels. Training and validation accuracy were tested to assess learning the experimental results show that the suggested LANN model balances computational efficiency and detection accuracy for real-time, explainable IoT botnet detection.

D. *Explainability Analysis (XAI Techniques)*

To improve the clarity and understanding of the proposed detection framework, this study incorporates Explainable Artificial Intelligence (XAI) techniques grounded in the SHapley Additive exPlanations (SHAP) framework. SHAP offers a rigorous framework for understanding intricate machine learning models by calculating a "SHAP value" for each input feature, which measures its impact on the model's ultimate prediction [7]. The DeepSHAP method was chosen from among the various SHAP versions, which included KernelSHAP and TreeSHAP. This decision was made because the DeepSHAP approach is designed to be suitable for deep learning architectures, such as Artificial Neural Networks (ANNs).

As illustrated in Fig. 4, a lightweight ANN-based detection model and DeepSHAP post-hoc explanation module are used in the proposed framework. The detection component efficiently infers IoT traffic data, while the explanation module generates feature-level attributions from model outputs. This modular design allows human-understandable decision-making explanations without model architecture changes.

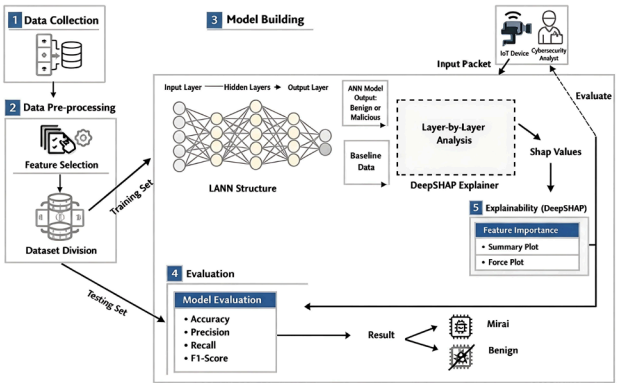


Fig. 4. Architecture of the proposed ANN-based Mirai framework incorporating DeepSHAP.

Through the use of a layer-wise backpropagation technique, DeepSHAP is able to compute feature attribution scores by integrating the theoretical foundations of Deep Learning Important FeaTures (DeepLIFT) and SHAP [34]. In this technique, it is possible to conduct a comprehensive investigation into the manner in which specific characteristics within the IoT-23 dataset, such as the kind of protocol, the size of the packet, and the ports of origin or destination, have a role in the classification of network traffic as either benign or infected with Mirai.

This kind of interpretability is especially important in cybersecurity contexts, where having an understanding of the reasoning behind model predictions helps to build trust and enables more efficient threat analysis and investigation.

However, the computational cost of DeepSHAP increases with model depth and architectural complexity [7]. A lightweight Artificial Neural Network (ANN) architecture with a reduced number of layers and neurons is utilized by the proposed framework in order to guarantee the feasibility of deployment in Internet of Things environments that are limited in resources. This design choice allows for the efficient computation of SHAP values while maintaining a high level of detection accuracy, so achieving a practical balance between the interpretability of the results and the computational efficiency of the results [21].

There are three primary stages involved in the process of integrating DeepSHAP into the framework that has been proposed. The first step is to build a baseline dataset that is representative of the typical behavior of an Internet of Things network. This serves as a reference and ensures that the SHAP values that are generated appropriately capture deviations that are connected with Mirai-induced abnormalities [35]. The second step involves applying the DeepSHAP explainer to the trained model in order to compute feature contribution scores for both benign and malicious traffic samples. In conclusion, a number of different visualization techniques are utilized, such as summary plots for the purpose of identifying globally relevant variables and force plots for the purpose of conducting a more in-depth inspection of individual predictions.

DeepSHAP's implementation into the suggested lightweight Artificial Neural Network (ANN) model improves its overall performance by offering significant interpretability in addition to improved prediction capabilities. This integration helps to bridge the gap between accuracy and transparency, which ultimately results in a detection framework that is trustworthy, interpretable, and well-suited for realistic Internet of Things security deployments. In these deployments, both efficiency and explainability are critical for efficient malware detection.

IV. Experimental Results and Analysis

A. *Performance Evaluation of the Proposed Solution*

The proposed solution based on the Lightweight Artificial Neural Network (LANN) model is tested in this part for both how well it classifies things and how quickly it can do those things. Key efficiency metrics, such as inference time, CPU utilization, RAM consumption, and model size, are used to test whether the approach can be

used in IoT environments with limited resources. Standard metrics like accuracy, precision, recall, and F1-score are also used to measure classification performance and show how well the model works at finding Mirai malware. The effects of adding DeepSHAP are also looked at to see how well it can be interpreted while keeping total performance the same.

All experiments were conducted using a fixed random seed (random_state = 42) to ensure reproducibility. Both the IoT-23 and N-BaIoT datasets were partitioned into training (80%) and testing (20%) subsets using stratified sampling to preserve the original class distribution. Feature selection was performed using a Random Forest classifier (100 estimators, random_state = 42) combined with SelectFromModel using a median feature-importance threshold. Class imbalance was addressed using cost-sensitive learning through scikit-learn's balanced class-weighting strategy. All models were implemented in Python using TensorFlow/Keras and scikit-learn within the Google Colab environment, and each experiment was executed using the fixed random seed to obtain deterministic and reproducible results.

First, to validate the lightweight design, inference time, CPU usage, RAM consumption, and model size were measured for the proposed LANN model and a baseline ANN under identical experimental conditions. As reported in Table II, the LANN approach reduces computational resource utilization while retaining classification accuracy. \

TABLE II. COMPUTATIONAL PERFORMANCE COMPARISON OF THE EVALUATED CONFIGURATIONS

Metric	Baseline ANN Model	Lightweight ANN Model (Proposed)	DeepSHAP Alone	LANN + DeepSHAP (Proposed)
Model Size	0.06 MB	0.04 MB	-	-
Avg. Inference Time	0.0991 s	0.096523 s	0.1132 s	0.2297 s
Avg. CPU Usage	5.73%	3.79%	5.01%	7.49%
Avg. RAM Usage	921.91 MB	907.50 MB	1263.31 MB	2526.50 MB

We measured the computational cost of SHAP by separating explainability from model inference. Wall-clock time was measured as the actual time elapsed to calculate SHAP values for each input sample. CPU usage was measured at the process level, by computing the difference in cumulative user and system CPU time before and after running SHAP. Memory consumption was measured by the change in process Resident Set Size (RSS). All provided parameters were averaged across numerous samples to guarantee robustness and reproducibility.

Although DeepSHAP increases inference-time overhead, it does not increase trainable parameters. Attribution computation increases CPU and memory usage in the LANN + DeepSHAP combination. However, the lightweight LANN design allows explainability to be engaged on demand without considerable computational expense. These findings demonstrate the feasibility of the interpretable decision-making paradigm in resource-constrained IoT and edge environments.

Second, a confusion matrix was used to look at True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN) to see how well the classification worked.

As shown in Fig. 5, the model accurately identified 385 benign samples (TN) and 4208 Mirai-infected samples (TP), resulting in no false positives and only 36 false negatives. The classification report reveals a precision of 0.91, recall of 1.00, and F1-score of 0.95 for benign traffic. In contrast, Mirai-infected traffic shows a precision of 1.00, recall of 0.99, and F1-score of 0.99. The proposed solution's robust detection was confirmed with 99% classification accuracy.

In the third step of the process, the learning behavior of the model was investigated by analyzing the accuracy and loss curves through training and validation over a number of epochs. As illustrated in Fig. 6, as the training loss drops from around 0.16 to less than 0.02 over time, it shows that convergence has been successful and classification error is negligible. Concurrently, Fig. 7 demonstrates that the accuracy of the validation remains stable at roughly 99.3%, closely following the accuracy of the training, which is 99.5%, while the accuracy of the test remains at 99.2%. The robust generalization of the model and the absence of overfitting are both confirmed by the close alignment that exists between training, validation, and test performance using this model.

Finally, in order to further evaluate the architectural efficiency of the proposed LANN model, a comparative evaluation was carried out against a baseline ANN architecture. The baseline model is composed of two hidden layers with 64 and 32 neurons and a single-neuron output layer. The suggested LANN, instead, has a compact architecture with three hidden layers of 6, 9 and 5 neurons, respectively, and is designed for resource-constrained IoT settings. To ensure fair comparison, both models were trained and evaluated under the same experimental conditions and the resulting performance metrics are summarized in Table III.

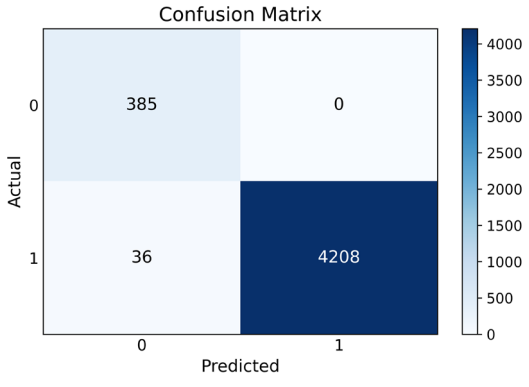


Fig. 5. Confusion matrix for suggested LANN model.

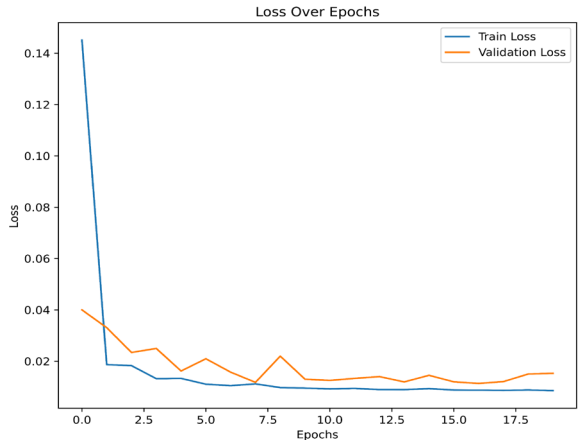


Fig. 6. Evolution of training and validation loss across training epochs.

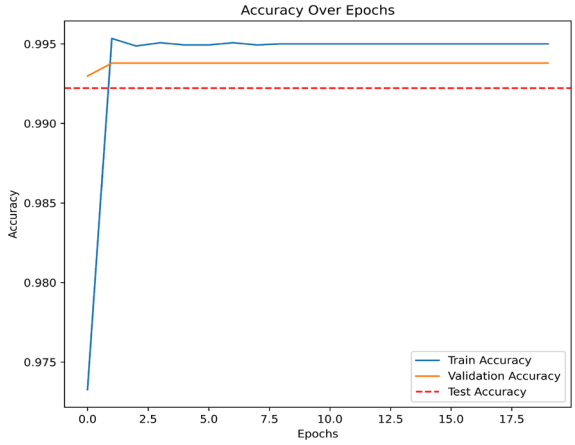


Fig. 7. Evolution of training and validation accuracy across epochs.

TABLE III. PERFORMANCE COMPARISON OF THE LANN AND BASELINE ANN MODELS WITH 95% CONFIDENCE INTERVALS

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
Baseline ANN Model	99.22 (95% CI: 98.7–99.3)	0.91 [0.88–0.94] for class 0, 1.00 [1.00–1.00] for class 1	1.00 [1.00–1.00] for class 0, 0.99 [0.987–0.993] for class 1	0.96 [0.94–0.98] for class 0, 1.00 [1.00–1.00] for class 1
Lightweight ANN Model (Proposed)	99.22 (95% CI: 98.7–99.3)	0.91 [0.88–0.94] for class 0, 1.00 [1.00–1.00] for class 1	1.00 [1.00–1.00] for class 0, 0.99 [0.987–0.993] for class 1	0.96 [0.94–0.98] for class 0, 1.00 [1.00–1.00] for class 1

As reported in Table III, both models attained the same overall classification accuracy of 99.22%, exhibiting consistent precision, recall, and F1-score values across both classes. Precision values were recorded at 0.91 for Class 0 and 1.00 for Class 1. Recall values were noted as 1.00 for Class 0 and 0.99 for Class 1. In Class 0, the F1-scores were recorded as 0.96, while in Class 1, they were recorded as 1.00. The normal approximation approach was utilized for the purpose of determining confidence intervals for binomial proportions, specifically at a confidence level of 95%. Based on these findings, it can be concluded that the proposed LANN model helps to preserve the detection capability of the baseline ANN while simultaneously simplifying its architectural complexity to a significant degree.

The proposed Lightweight Artificial Neural Network (LANN) and baseline ANN model comparison was validated by McNemar's statistical test on both classifiers' binary classification outputs. The contingency matrix $\begin{pmatrix} 4593 & 0 \\ 0 & 36 \end{pmatrix}$ shows that both models predicted the same right and wrong classifications in all samples. The test yielded a McNemar $\chi^2 = 0.0$ and a $p = 1.0000$ ($p > 0.05$), indicating that there is no statistically significant difference in their classification performances. While both models reached the same classification metrics, the proposed LANN employs a significantly more compact architecture and demands fewer computational resources. These results indicate that the lightweight design effectively maintains the predictive capability of the baseline ANN, while enhancing its suitability for deployment in resource-constrained IoT environments.

B. Extended Evaluation on N-BaIoT

To further evaluate the generalizability of the proposed approach, an extended evaluation was conducted using the N-BaIoT dataset [36]. Because IoT-23 and N-BaIoT employ different feature extraction pipelines (Zeek/Bro connection-log features versus flow-based statistical features), direct zero-shot cross-dataset evaluation was not feasible due to incompatible feature representations. Therefore, both the proposed LANN and the baseline ANN were independently retrained and evaluated on N-BaIoT using the same preprocessing pipeline, feature selection strategy, class weighting scheme, and evaluation protocol adopted for the IoT-23 experiments.

The evaluation was performed on a sampled subset of the N-BaIoT dataset consisting of 440,000 network traffic records (350,000 malicious Mirai samples and 90,000 benign samples) using the same feature-selection methodology adopted in the proposed framework, resulting in 58 selected features for the N-BaIoT dataset. After applying an 80/20 train-test split, the test set contained 88,000 samples. Both models demonstrated excellent detection capability on this independent dataset. As summarized in Table IV, the ANN was good at what it did 99.9875% of the time, and the planned LANN did the same thing. The corresponding precision, recall, and F1-score numbers stayed high for both models, too. This shows that the lightweight architecture keeps classification performance even though it makes computations much simpler.

TABLE IV. PERFORMANCE COMPARISON ON THE N-BAIOT DATASET

Model	Accuracy	Precision	Recall	F1-score
Baseline ANN Model	0.999875	0.999971	0.999871	0.999921
Lightweight ANN Model (Proposed)	0.999875	0.999857	0.999986	0.999921

McNemar's test was used to compare the two classifiers' prediction results. The contingency matrix showed no statistically significant differences between the ANN and suggested LANN, with McNemar $\chi^2 = 8.00$ and a p-value of 1.0000. The lightweight architecture delivers

prediction performance comparable to the baseline ANN with much lower processing requirements.

In summary, this supplementary assessment illustrates that the proposed framework continues to exhibit exceptional detection capabilities across a variety of IoT datasets following dataset-specific retraining. The proposed lightweight framework's robustness and generalizability are further supported by the consistent performance achieved on two independent benchmark datasets, despite the fact that direct zero-shot cross-dataset evaluation is still unfeasible due to the heterogeneous feature extraction pipelines present in publicly available Mirai datasets.

C. *Explainability Through DeepSHAP Analysis*

To enhance the interpretability of the proposed LANN model for Mirai malware detection, the DeepSHAP method, an extension of SHapley Additive exPlanations (SHAP) specifically designed for deep learning architectures, was employed. DeepSHAP assigns importance scores to input features, enabling visual interpretation of their contributions to the model's predictions and promoting transparency in the decision-making process.

The SHAP summary plot shown in Fig. 8 and the corresponding SHAP bar chart in Fig. 9 rank features in accordance with their average contribution, thereby showing vital behavioral patterns that are connected with traffic that has been infected with Mirai. Particularly noteworthy is the fact that cat_proto_udp emerges as the most influential attribute, which indicates that Mirai assaults usually exhibit anomalous behavior in terms of UDP communication. In addition, the cat_conn_state_OTH method places a heavy emphasis on the presence of abnormal connection states as strong signs of malicious activity. In addition, the methods known as num_id.resp_p and cat_conn_state_SF all contribute further to the difference between traffic that is not malicious and traffic that has been hacked.

Overall, this analysis demonstrates that the proposed model relies on semantically meaningful and verifiable network features during classification, thereby reinforcing the reliability and trustworthiness of its detection decisions.

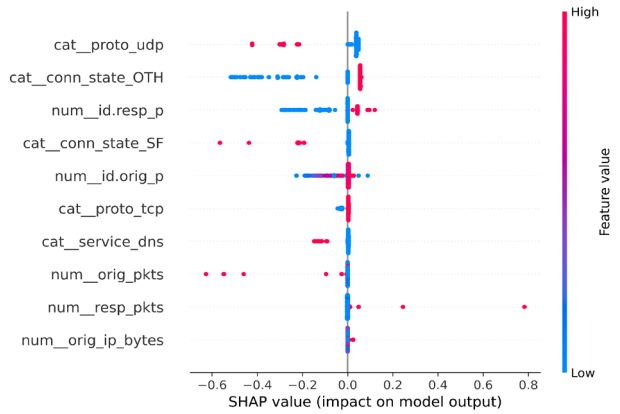


Fig. 8. SHAP summary visualization.

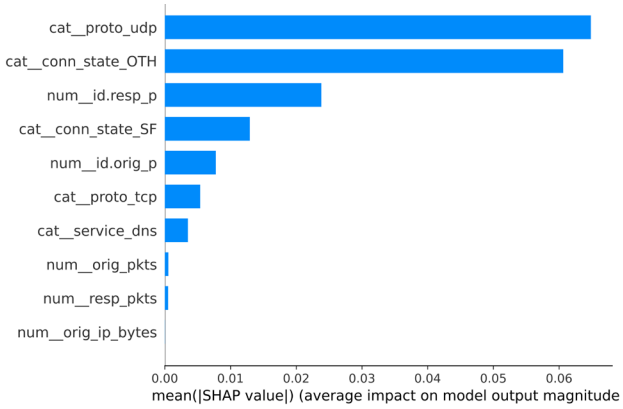


Fig. 9. SHAP summary bar chart.

Furthermore, the SHAP force plot Fig. 10 shows how feature values like cat_proto_udp = 0, cat_conn_state_OTH = 1, and cat_proto_tcp = 1 push predictions toward the malicious class. This visualization highlights the model's consistency and resilience, supporting real-world IoT use.

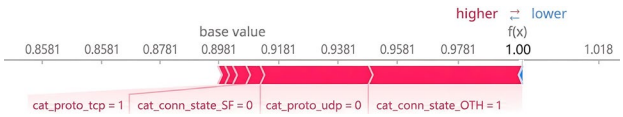


Fig. 10. SHAP force plot.

V. Discussion

The results of the experiments provide strong empirical data that demonstrates the effectiveness of the suggested methodology in addressing the considerable obstacles connected with the detection of Mirai attacks in Internet of Things environments that are limited in resources. The detection capability of the Lightweight Artificial Neural Network (LANN) model is the most important contribution that this research has made. The LANN model achieved an overall accuracy of 99.22% on the test dataset, demonstrating a performance level that is highly appropriate for applications that are security-critical. The classification report that is displayed in Fig. 5 demonstrates that the malicious Mirai class (Class 1) has achieved an exceptional level of performance. This is evidenced by the fact that the class has achieved immaculate precision and an F1-score of 1.00, in addition to a recall that is virtually perfect at 0.99. This high recall is extremely important from a security point of view because it implies that the model successfully identified almost all actual Mirai assaults, with only 36 false negatives being produced as a result.

The LANN model's high detection accuracy provides convincing evidence for Research Question 1 and supports its future use as an early warning system for Mirai attacks. The confusion matrix presented in Fig. 5 further validates the good discriminatory potential of the model with 0 false positive and only 36 false negatives across 4629 test samples. These results show that the accuracy obtained is not due to class imbalance but to the better feature learning provided by the optimized neural network design. In addition, the training process

demonstrated stable generalization behavior without any signs of overfitting. As can be seen in Fig. 7, the validation metrics closely track the training metrics throughout the learning process. The gap between the curves is consistently small, which confirms the model's ability to generalize to unseen data.

Beyond detection accuracy, the LANN model was explicitly designed to meet the practical resource constraints of IoT deployment. As reported in Table II, a small file size of 0.04 MB and a rapid average inference time of 0.0965 s make the proposed model suitable for low-power, memory-constrained IoT edge devices. Computational evaluation shows that integrating Explainable AI (XAI) through DeepSHAP increases inference-time and memory overhead while maintaining moderate CPU utilization. These findings answer Research Question 2 by showing that interpretable detection is possible without high computing expense.

DeepSHAP elevates memory consumption due to the computation of attributions, nonetheless, the processing burden remains manageable, particularly when explainability is activated when needed. This discovery refutes the notion that interpretability necessitates significant processing resources and demonstrates that clarity and efficiency can coexist. Comparison of performance in Table III, this article supports the proposed LANN design's efficiency advantage over bulkier baseline versions in the literature.

Research Question 3, which focuses on trust and usability in real-world IoT security settings, is concerned with interpretability. DeepSHAP program provides a significant mechanism to support this question. The feature importance visualization in Fig. 10 finds cat_proto_udp most influential in categorization decisions. UDP is used extensively by the Mirai botnet for scanning and DDoS activities, as documented. Security experts and system stakeholders trust the model's predictive abilities since it matches attack characteristics.

The model uses a cohesive and verifiable combination of network anomalies rather than individual indications, as shown by secondary contributing features cat_conn_state_OTH and num_id.resp_p. Practitioners can validate detection results, understand alert causes, and improve operational security workflows with transparency. The system provides actionable reasons (e.g., high UDP activity) instead of opaque notifications, speeding validation, guiding countermeasure deployment, and improving security audit compliance.

Within the context of Internet of Things (IoT) security, the framework that has been suggested helps to alleviate the problem of lightweight deployment and interpretability frequently being in conflict with one another. Methods that were used in the past had a tendency to prefer either elaborate models that required a great deal of resources or simple models that were hard to understand. On the other hand, the LANN-based approach maintains a modest size and allows for rapid inference, while also enabling DeepSHAP to offer post-hoc explainability. With a detection accuracy of 99.22%, these results show that it is possible to use accurate, understandable, and computationally efficient Mirai detection mechanisms even when resources are limited. This directly supports the trust and usability goals of Research Question 3.

VI. Challenges and Limitations

The present study cannot address various limitations despite these advances. First, the framework was built and validated using the IoT-23 dataset, however the N-BaIoT dataset was used to further evaluate its generalizability. Incompatible feature representations prevented direct zero-shot cross-dataset evaluation for IoT-23 and N-BaIoT, which use diverse feature extraction pipelines (Zeek/Bro connection-log features against flow-based statistical features). Therefore, the suggested LANN and baseline ANN were independently retrained and evaluated on N-BaIoT using the same experimental technique. Retraining the framework showed high detection performance on both datasets, proving that it generalizes across IoT traffic datasets. True zero-shot cross-dataset evaluation is difficult because publicly available Mirai datasets differ in feature extraction, traffic generation, and labeling. Future study will investigate feature alignment and domain adaptation for zero-shot cross-dataset evaluation without retraining.

Second, the evaluation of resilience against hostile cases or synthetic Mirai variations was not within the purview of the current study. It is necessary to have dedicated attack-generation frameworks and specialized evaluation protocols in order to generate representative adversarial network traffic. This is in contrast to the primary objective of this work, which is to determine whether or not it is possible to combine a lightweight neural architecture with explainable artificial intelligence in order to detect Mirai. Therefore, in subsequent work, we will investigate adversarial robustness and synthetic attack generation in order to further evaluate the resilience of the framework that has been proposed.

Third, conducting a real-world deployment or a large-scale network simulation would yield more compelling evidence regarding the performance of the proposed framework in practical IoT scenarios. Nevertheless, executing such experiments necessitates specialized networking infrastructure, edge hardware, and controlled deployment environments, which were not accessible during this study.

Therefore, the experimental evaluation was intentionally conducted in an offline setting to establish a reproducible and unbiased baseline for comparing the proposed LANN with the baseline ANN. Although dynamic network behaviors, such as packet loss, latency variations, jitter, and concurrent multi-device interactions, were not explicitly modeled, deployment feasibility was partially demonstrated through the reported resource-efficiency metrics. Specifically, the proposed model achieved a compact size of 0.04 MB and an average inference latency of 0.096 s, indicating its suitability for resource-constrained edge devices. Nevertheless, further validation in real-world IoT environments remains necessary to confirm robustness under practical

deployment conditions. Future work will therefore focus on hardware-in-the-loop evaluation and deployment on embedded edge platforms, such as Raspberry Pi, under realistic network conditions.

VII. Conclusion

In this study, a model for identifying the Mirai virus in IoT networks was built using a Lightweight Artificial Neural Network (LANN). The model's detection accuracy, computational efficiency, and interpretability were all well-balanced. Overall accuracy of 99.22% with high precision, recall, and F1-score values demonstrated the usefulness of the proposed model in recognizing malicious Mirai traffic under resource-constrained settings.

The proposed method improves user trust and security analysis by integrating XAI using the DeepSHAP architecture to deliver transparent and interpretable insights into its decision-making process. The experimental results show that explainability may be added without increasing processing overhead, making it ideal for low-power IoT and edge contexts.

The results that were obtained clearly demonstrate the potential of lightweight and explainable artificial intelligence techniques for strengthening Internet of Things (IoT) security. They highlight the effectiveness of these techniques in delivering accurate, efficient, and interpretable malware detection solutions that are suitable for environments with limited resources.

This work lays the groundwork for intelligent, efficient, and trustworthy cybersecurity solutions for IoT environments. Future research will extend the proposed framework to detect more IoT malware families, incorporate more optimization strategies to enable real-time deployment within intrusion detection systems, and expand the dataset with more diverse and up-to-date attack patterns to improve robustness and generalizability in practical IoT settings.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Danah A. Alkughail contributed to conceptualization, methodology, investigation, resources, data curation, writing—original draft preparation, and visualization. Wedad M. Alawad contributed to conceptualization, writing—review and editing, supervision, and project administration. Seera K. Alhatlani contributed to conceptualization, methodology, software, validation, investigation, resources, data curation, and writing—final draft preparation. Deem A. Algherair contributed to conceptualization, methodology, software, validation, formal analysis, investigation, resources, data curation, writing—original draft preparation, and visualization. Jood H. Alolayan contributed to conceptualization, methodology, software, validation, investigation, resources, and data curation. All authors reviewed, approved, and agreed to the published version of the manuscript.

References

[1] M. Kumari, N. Pramanick, M. Agarwal, and E. Esenogho, "An optimized IDS framework for big data environments: Integrating gravitational search and SMOTE-IPF data balancing for high-accuracy IDS," *SN Computer Science*, vol. 6, no. 7, 786, 2025. doi: 10.1007/s42979-025-04194-9

[2] J. Kipongo, T. G. Swart, and E. Esenogho, "Artificial intelligence-based intrusion detection and prevention in edge-assisted SDWSN with modified honeycomb structure," *IEEE Access*, vol. 12, pp. 3140–3175, 2024. doi: 10.1109/ACCESS.2023.3347778

[3] T. B. Shana *et al.*, "Improving IoT IDS with harmony search feature selection and ADASYN on UNSW-NB15 and CIC-IoT23," *Research Square*, 2025. doi: 10.21203/rs.3.rs-8077281/v1

[4] X. Zhang, O. Upton, N. L. Beebe, and K. K. R. Choo, "IoT botnet forensics: A comprehensive digital forensic case study on mirai botnet servers," *Forensic Science International: Digital Investigation*, vol. 32, 300926, 2020. doi: 10.1016/j.fsidi.2020.300926

[5] C. Wang, Z. Ma, Q. Li *et al.*, "A lightweight IoT malware detection and family classification method," *Journal of Computer and Communications*, vol. 12, no. 4, pp. 201–227, 2024. doi: 10.4236/jcc.2024.124015

[6] P. Gohel, P. Singh, and M. Mohanty, "Explainable AI: Current status and future directions," arXiv preprint, arXiv:2107.07045, 2021.

[7] S. M. Lundberg and S. I. Lee, "A unified approach to interpreting model predictions," *Neural Information Processing Systems*, vol. 30, 2017.

[8] M. A. Bilal, I. U. Islam, N. Iltaf *et al.*, "Federated learning with explainable AI for malicious traffic detection in IoT networks," *IEEE Access*, vol. 13, pp. 173368–173383, 2025. doi: 10.1109/ACCESS.2025.3613459

[9] Z. Alwaisi, "Memory-efficient and robust detection of Mirai botnet for future 6G-enabled IoT networks," *Internet of Things*, vol. 32, 101621, 2025. doi: 10.1016/j.iot.2025.101621

[10] M. Rahmati, "Towards explainable and lightweight AI for real-time cyber threat hunting in edge networks," arXiv preprint, arXiv:2504.16118, 2025.

[11] V. Patel, "Enhancing botnet detection with machine learning and explainable AI: A step towards trustworthy AI security," *SSRN*, 2025. doi: 10.2139/ssrn.5193008

[12] M. Saied and S. Guirguis, "Explainable artificial intelligence for botnet detection in internet of things," *Scientific Reports*, vol. 15, no. 1, 7632, 2025. doi: 10.1038/s41598-025-90420-6

[13] S. Niktabe, "IoT network malicious behaviour profiling based on explainable AI using LSTM and SHAP," M. S. thesis, York University, Ontario, Canada, 2024.

[14] P. Turaka and S. K. Panigrahy, "Dynamic attack detection in IoT networks: An ensemble learning approach with Q-Learning and explainable AI," *IEEE Access*, vol. 12, pp. 161925–161940, 2024. doi: 10.1109/ACCESS.2024.3485989

[15] A. A. Almazroi and N. Ayub, "Deep learning hybridization for improved malware detection in smart Internet of Things," *Scientific Reports*, vol. 14, no. 1, 7838, 2024. doi: 10.1038/s41598-024-57864-8

[16] A. N. Gummadi, J. C. Napier, and M. Abdallah, "XAI-IoT: An explainable AI framework for enhancing anomaly detection in IoT systems," *IEEE Access*, vol. 12, pp. 71024–71054, 2024. doi: 10.1109/ACCESS.2024.3402446

[17] F. Yousaf, M. Arslan, A. A. Khan *et al.*, "Machine learning-based detection of Mirai and Bashlite botnets in IoT networks," *Journal of Computing & Biomedical Informatics*, vol. 7, no. 01, pp. 678–689, 2024.

[18] A. Sharma, V. Mansotra, and K. Singh, "Detection of Mirai botnet attacks on IoT devices using deep learning," *Journal of Scientific Research and Technology*, pp. 174–187, 2023. doi: 10.5281/ZENODO.8330561

[19] M. Siganos *et al*., "Explainable AI-based intrusion detection in the internet of things," in *Proc. the 18th International Conference on Availability, Reliability and Security*, 2023, pp. 1–10. doi: 10.1145/3600160.3605162

[20] M. Keshk, N. Koroniotis, N. Pham *et al.*, "An explainable deep learning-enabled intrusion detection framework in IoT networks," *Information Sciences*, vol. 639, 119000, 2023. doi: 10.1016/j.ins.2023.119000

[21] E. Y. Güven and Z. Gürkaş-Aydin, "Mirai botnet attack detection in low-scale network traffic," *Intelligent Automation & Soft Computing*, vol. 37, no. 1, pp. 419–437, 2023. doi: 10.32604/iasc.2023.038043

[22] A. R. Khan, A. Yasin, S. M. Usman *et al.*, "Exploring lightweight deep learning solution for malware detection in IoT constraint environment," *Electronics*, vol. 11, no. 24, 4147, 2022. doi: 10.3390/electronics11244147

[23] M. S. Akhtar and T. Feng, "Detection of malware by deep learning as CNN-LSTM machine learning techniques in real time," *Symmetry*, vol. 14, no. 11, 2308, 2022. doi: 10.3390/sym14112308

[24] S. Patil *et al.*, "Explainable artificial intelligence for intrusion detection system," *Electronics*, vol. 11, no. 19, 3079, 2022. doi: 10.3390/electronics11193079

[25] M. Asam *et al*., "IoT malware detection architecture using a novel channel boosted and squeezed CNN," *Scientific Reports*, vol. 12, no. 1, 15498, 2022. doi: 10.1038/s41598-022-18936-9

[26] H. Naeem, B. M. Alshammari, and F. Ullah, "Explainable artificial intelligence-based IoT device malware detection mechanism using image visualization and fine-tuned CNN-based transfer learning model," *Computational Intelligence and Neuroscience*, vol. 2022, no. 1, 2022. doi: 10.1155/2022/7671967

[27] Z. A. E. 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 Communications Society*, vol. 3, pp. 1164–1176, 2022, doi: 10.1109/OJCOMS.2022.3188750

[28] M. Giordano, N. Baumann, M. Crabolu *et al.*, "Design and performance evaluation of an ultralow-power smart IoT device with embedded TinyML for asset activity monitoring," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–11, 2022. doi: 10.1109/TIM.2022.3165816

[29] S. Baek, J. Jeon, B. Jeong, and Y. S. Jeong, "Two-stage hybrid malware detection using deep learning," *Human-centric Computing and Information Sciences*, vol. 11, no. 27, 2021. doi: 10.22967/HCIS.2021.11.027

[30] T. Lawrence and L. Zhang, "IoTNet: An efficient and accurate convolutional neural network for IoT devices," *Sensors*, vol. 19, no. 24, 5541, 2019. doi: 10.3390/s19245541

[31] S. Garcia *et al.*, "IoT-23 dataset: A labeled dataset of malware and benign IoT traffic," *Stratosphere IPS*, 2020. doi: 10.5281/zenodo.4743746

[32] W. Du *et al*., "Group & reweight: A novel cost-sensitive approach to mitigating class imbalance in network traffic classification," *International Journal of Machine Learning and Cybernetics*, vol. 16, no. 10, pp. 7707–7728, 2025. doi: 10.1007/s13042-025-02682-1

[33] R. Abhishek *et al*., "Prediction of compressive strength of corncob ash concrete for environmental sustainability using an artificial neural network: A soft computing techniques," *Journal of Soft Computing in Civil Engineering*, vol. 7, no. 2, pp. 115–137, 2023. doi: 10.22115/scce.2023.347663.1471

[34] A. Shrikumar, P. Greenside, and A. Kundaje, "Learning important features through propagating activation differences," in *Proc. International Conference on Machine Learning*, 2017, pp. 3145–3153. doi: 10.48550/ARXIV.1704.02685

[35] H. Chen, S. Lundberg, and S. I. Lee, "Explaining models by propagating Shapley values of local components," arXiv preprint, arXiv:1911.11888, 2019.

[36] Y. Meidan *et al.*, "Detection_of_IoT_botnet_attacks_N_BaIoT," *UCI Machine Learning Repository*, 2018. doi: 10.24432/C5RC8J

Copyright © 2026 by the authors. This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited (CC BY 4.0).