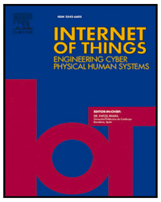
Internet of Things 33 (2025) 101714

Contents lists available at ScienceDirect

Internet of Things

journal homepage: www.elsevier.com/locate/iot



Enhancing AI transparency in IoT intrusion detection using explainable AI techniques

Yifan Wang [a,1], Muhammad Ajmal Azad [b,*], Maham Zafar [c], Ammara Gul [b]

[a] *Warwick Manufacturing Group, United Kingdom*
[b] *Birmingham City University, United Kingdom*
[c] *National Centre for Cyber Security, Pakistan*

ARTICLE INFO

Keywords:
Interpretable artificial intelligence
Intrusion detection system
Internet of Things
Machine learning
Cyber security
Explainable AI

ABSTRACT

Internet of Things (IoT) networks continue to grow and have been integrated into critical applications such as healthcare, industrial control, and national infrastructure. The interconnected nature and resource-constrained devices can create numerous entry points for malicious actors who can bring about data breaches, unauthorised access, service disruptions, and even compromise critical infrastructure. Ensuring the security of these networks is essential to maintain the integrity and availability of services that could have serious social, economic, or operational consequences. Automated Intrusion Detection Systems (IDSs) have been widely used to identify threats with high accuracy and reduced detection time. However, the complexity of machine learning and deep learning models poses a serious challenge to the transparency and interpretability of the produced detection results. The lack of explainability in AI-driven IDS undermines user confidence and limits their practical deployment, especially among non-expert stakeholders. To address these challenges, this paper investigates the use of Explainable AI (XAI) techniques to enhance the interpretability of AI-based IDSs within IoT ecosystems. Specifically, it applies SHapley Additive exPlanations (SHAP) and Local Interpretable Model-Agnostic Explanations (LIME) to different Machine learning models. The models' performance is evaluated using standard metrics such as accuracy, precision, and recall. The results show that incorporating XAI techniques significantly improves the transparency of IDS results, allowing users to understand and trust the reasoning behind AI decisions. This enhanced interpretability not only supports more informed cybersecurity practices but also makes AI systems more accessible to non-specialist users.

1. Introduction

At its core, IoT (Internet of Things) consist of a vast network of interconnected physical devices embedded with sensors, software, and other technologies. Recent forecasts indicate that there will be a total of 40 billion IoT devices at the end of 2030 [1]. This expansion is driven by advancements in connectivity (5G, 6G, high-speed wifi), computing power (edge and cloud computing paradigms), and the decreasing cost of sensors. These devices continuously generate, collect, and exchange massive amounts of real-time data, which can be analysed for meaningful decisions and process automation. This deluge of information is far too immense and complex for human analysis and requires the application of Artificial Intelligence (AI), Data Science and machine

[*] Corresponding author.
E-mail address: majmalazad@gmail.com (M.A. Azad).
[1] Yifan Wang performed this work while a student at the University of Warwick.

https://doi.org/10.1016/j.iot.2025.101714
Received 30 May 2025; Received in revised form 14 July 2025; Accepted 18 July 2025

Available online 29 July 2025

Learning for a meaningful decision-making process. Machine learning and AI algorithms are deployed to analyse patterns, predict failures, optimise operations, and automate decision-making across IoT networks. For instance, AI-powered systems can monitor energy consumption in smart buildings, anticipate maintenance needs in factories, or even coordinate traffic flows in urban centres. This symbiotic relationship between IoT and AI unlocks unprecedented capabilities, making these connected systems a smart system and proactive rather than merely reactive.

AI plays an important role in the IoT ecosystems for autonomous process controls and decisions; however, the lack of transparency in its decision-making processes raises significant concerns [2]. Traditional black box AI models often provide effective outcomes without transparent explanations for how those decisions were reached. In the context of IoT, AI may control vital infrastructure, manage sensitive personal data, or trigger emergency responses. A lack of understanding regarding the AI's rationale in these scenarios can lead to distrust and make troubleshooting a complex task for human users. The human users might also feel reluctant to trust the decisions and recommendations made by the AI or ML systems. Explainable Artificial Intelligence (XAI) can be seen as a platform to provide some meaningful information about the decisions made by AI systems. XAI focuses on developing AI models whose results can be understood by humans. It aims to make AI decisions transparent, interpretable, and accountable, moving beyond mere prediction to provide insights into why a particular decision was made. For example, when an AI system within an IoT network flags an anomaly or identifies a potential security threat, XAI can provide the context and reasoning behind that decision. This transparency empowers human operators and administrators to validate AI judgments, identify potential biases or errors, and build trust in automated systems.

As IoT networks expand, they become prime targets for cyberattacks due to their distributed nature and often limited security capabilities. AI-based IDSs are essential for detecting malicious activities, unusual data patterns, or unauthorised access attempts; however, these non-XAI systems do not provide detailed insight about the origin of the anomaly, its impact and why a particular traffic pattern has been classified as the anomaly. With XAI, the IDS could explain, "Anomaly detected due to unusual data packet size (2x normal) originating from IP address X on device Y, consistent with a DDoS attack signature". Such detailed explanations could help security teams to quickly understand, prioritise, and respond to threats. This would significantly enhance the overall security posture and resilience of complex IoT deployments.

In this research, classical machine learning (ML) models and explainable AI (XAI) methods are employed to create an interpretable intrusion detection system for IoT networks. Two popular ML models are chosen: Decision Tree (DT) and Logistic Regression (LR), which are recognised for their interpretability and simplicity. Furthermore to increase model explainability, we implement two model-agnostic XAI techniques: SHAP (SHapley Additive exPlanations) and LIME (Local Interpretable Model-Agnostic Explanations). SHAP gives each feature a contribution value in terms of cooperative game theory, whereas LIME constructs a straightforward local model for explaining individual predictions. We introduce the LIAM (LIME and SHAP-based Interpretable AI Model) framework, which integrates DT or LR with SHAP and LIME to produce precise and explainable intrusion predictions. The "shape" refers to distribution and direction of feature contributions in SHAP explanations, facilitating visualisation of how each feature affects a prediction. This enables better comprehension of prevalent attack patterns in IoT traffic. Let $\mathbf{x} = [x_1, x_2, \ldots, x_n]$ be an input feature vector, and $f(\mathbf{x})$ be the output of the model. The prediction can be formulated as per SHAP as:

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

where ϕ_0 is the expected model output (base value), and ϕ_i is the contribution of feature x_i. The two IoT dataset are used to train and evaluate the models, providing labelled IoT traffic data for both normal and attack scenarios.

Techniques like SHAP and LIME can be used to provide the explanation to security analysts to understand why a particular network activity is flagged as malicious [3]. This integration facilitates faster incident response and timely root cause analysis [4,5]. A significant concern is the computational overhead associated with generating explanations, particularly with complex models and high-volume data from heterogeneous devices. Furthermore, achieving high accuracy often involves complex models that are inherently harder to explain. XAI has been researched in other domains; however, its implementation and efficacy in IDS with IoT networks are less explored. This research fills that gap by comparing SHAP and LIME in visualising and explaining IDS results across a number of datasets.

This research paper addresses a critical gap by enhancing the transparency and user trust of IoT-specific Intrusion Detection Systems through the rigorous application and analysis of XAI. To this extent, we not only analyse the performance of various machine learning models but also attempt to explain why particular traffic has been classified as malicious or non-malicious. We also empirically validate the applicability and effectiveness of XAI techniques such as SHAP and LIME within the unique context of IoT network security. Firstly, it evaluates the XAI's feasibility by conducting a rigorous empirical comparison of different XAI techniques (SHAP for global and local explanations, LIME for local insights) across a diverse range of machine learning algorithms (Decision Trees, Linear Regression). By developing intuitive methods like bar charts, 3D scatter plots, and heatmaps, coupled with dimensionality reduction (PCA, t-SNE), the study aims to translate complex security insights into easily understandable formats. This paper makes following major contributions:

- We present a novel and rigorous empirical comparison of leading XAI techniques, SHAP and LIME, when applied to inherently interpretable machine learning models (specifically Decision Trees and Logistic Regression) within the critical domain of IoT Intrusion Detection Systems. Unlike much existing XAI research that focuses solely on explaining complex black-box models, our work uniquely quantifies how XAI can deepen the understanding of even transparent models, providing nuanced insights into their decision-making processes. This includes detailed performance metrics (e.g., Decision Tree accuracy of 0.9968 on CICIDS2017) and an analysis of feature contributions. This is important for establishing a baseline for the tradeoff between interpretability and accuracy in resource-constrained networks.

- This paper offers a significant practical contribution by demonstrating how the strategic integration of SHAP and LIME effectively mitigates the long-standing interpretability-accuracy trade-off in IoT IDS. By applying these XAI methods to models suitable for resource-constrained IoT devices, we show that it is possible to achieve high detection accuracy while simultaneously providing clear, actionable explanations.
- We contribute to bridging the communication gap between complex AI outputs and human understanding by developing intuitive visualisation methods. This includes leveraging techniques like bar charts, 3D scatter plots, heatmaps, and dimensionality reduction (PCA, t-SNE) to translate intricate security insights derived from IDS into easily digestible formats for non-expert users. The SHAP summary plots, for instance, visually represent feature importance and impact.

The paper is organised as follows. Section 2 defines the problem addressed in this paper. Section 3 explores existing research on XAI, IDS, and machine learning. Section 4 details the proposed framework and provides details about local and global explanations. Section 5 explains experimental Setup, data set used, and the evaluation metrics used in this paper. The Results and Analysis is presented in Section 6. Section 7 provides a detailed discussion about the integration of XAI and ML models. Section 9 concludes the paper.

2. Problem definition

The adoption of AI in critical domains such as healthcare, finance, cybersecurity, and Internet of Things ecosystems requires that they not only achieve higher performance accuracy but also be explainable to non-technical people. AI models often operate as black boxes and do not offer detailed insight into the internal reasoning of the models and their decision-making process. The challenge is designing an explainable mapping $\mathbb{X} \to \mathbb{E}$, where $\mathbb{E}$ denotes an interpretable explanation space, such that $g \approx f$ locally or globally. This explanation should be understandable to human stakeholders and must not compromise the detection accuracy or scalability of the original model. The problem, therefore, is to develop XAI frameworks that balance the trade-off between interpretability and performance. It should minimise the joint loss function $\mathcal{L}(f, g) = \alpha \cdot \mathcal{L}_{\text{pred}}(f) + \beta \cdot \mathcal{L}_{\text{explain}}(g)$, where $\mathcal{L}_{\text{pred}}$ measures detection performance, $\mathcal{L}_{\text{explain}}$ quantifies interpretability or explanation, and $\alpha, \beta \in \mathbb{R}^+$ are weights reflecting the trade-off priorities. Addressing this problem is essential for deploying AI systems that are both effective and trustworthy.

3. Literature review

This section explores the current landscape of XAI in the context of IDS, particularly within the Internet of Things. The integration of XAI directly into IoT security solutions is rapidly increasing, especially within IoT networks [6]. Arisdakessian et al. [7] designed a novel cybersecurity model that holistically integrates XAI, federated learning, and game theory to identify IoT threats through a two-tier classification strategy. The practical complexities of deploying such systems across diverse IoT ecosystems remain a significant consideration. Chakir et al. [8] proposed a SHAP-based explainable ML system for Industrial IoT (IIoT) web applications. Their work demonstrated effectiveness in both binary and multi-class classification; however, it has scalability issues within real industrial settings. Further emphasising the practical application of XAI, Kumar et al. [9] introduced a lightweight SD-IoT IDS for feature selection to enhance efficiency and accuracy. However, its adaptability and interpretability under varied and evolving attack patterns in heterogeneous networks are not scalable. Kalakoti et al. [10] proposed an active learning-based IDS incorporating SHAP and LIME for the Security Operations Centre. Similarly, Vakula Rani et al. [11] integrated SHAP and LIME into an XAI-based IDS. They reported high accuracy and interpretability but might not perform well under sophisticated and novel attacks. Ahakonye et al. [12] also designed an IIoT IDS using SHAP and LIME to study decision trust and interpretability. The results highlight that the computation is high for the SHAP model, and LIME has occasional instability. This highlights an important challenge regarding Xai: achieving a desirable balance between transparency and computational efficiency in resource-constrained IoT environments.

Beyond XAI-specific applications, extensive research has been performed to address the performance of ML models for IDS. Sambandam et al. [13] compared several ML classifiers (e.g., Random Forest (RF), K-Nearest Neighbours (KNN), Logistic Regression (LR)) on the UNSW-NB15 dataset. However, they noted limitations in detecting minority attack types (e.g., Worms, backdoors). This might happen due to the class imbalance problem. Pansare et al. [14] similarly highlighted dataset quality and feature selection for ML-based IDS. Their work emphasises algorithm adaptation to specific data and attack types but does not consider real-world deployment challenges like dynamic threats or class imbalance. Rani et al. [15] also analysed several machine learning models for IDS. Their study highlights the accuracy-efficiency trade-offs but overlooks practical deployment aspects such as processing time or system resource consumption in IoT contexts. Sayadi et al. [16] tackled hardware-level IDS by evaluating ML models against low-separability attacks. The results show that XGBoost performed well on clean data but was unable to identify stealthier threats. In the context of IoT edge computing, Jiang et al. [17] used a J48 decision tree on a home IoT edge gateway. This lightweight approach effectively balanced real-time detection with resource constraints. Vigneswaran et al. [18] used shallow and deep neural networks on the KDDCup-99 dataset. Malibari et al. [19] proposed MDLIDS-SSE, combining deep learning with advanced optimisation algorithms for smart spaces. Xue et al. [20] proposed a deep transfer learning model (autoencoders and CNNs) for generalisability across IoT domains. This system can effectively handle unlabelled data and domain adaptation. Mehedi et al. [21] also proposed a residual neural network-based IDS for IoT via deep transfer learning. Saif et al. [22] introduced the Hybrid Intelligent IDS (HIIDS) for IoT healthcare. It provides high accuracy in detection and has a lower computational cost, which makes it most suitable for resource-restricted environments such as healthcare systems.

Explainable Artificial Intelligence (XAI) has been extensively studied to enhance the transparency, interpretability, and trustworthiness of AI systems [23–27]. This research is particularly important in critical domains such as healthcare, energy, finance, cybersecurity, and IoT, where opaque decision-making can have significant consequences [28–30]. Siganos et al. [31] proposed an XAI-integrated IDS for both IoT and IIoT systems. The system uses SHAP values to explain ML and DL models. Their system showcased high detection performance alongside increased trust and transparency. For IoT device classification, Lavrenovs and Graf [32] utilised Naive Bayes and LIME to improve the classification performance. Abou El Houda et al. [33] proposed an XAI-based IDS using deep learning over UNSW-NB15 while integrating RuleFit and SHAP for interpretability. Their system enhanced trust and transparency in AI security systems by making decisions explainable to human experts. Djenouri et al. [34] proposed a visual-oriented XAI framework using time-series-to-image conversion and VGG16 genetic algorithm optimisation. This model proved effective in anomaly detection, along with visual and interpretable results. The complexity of the system limits it application to IoT implementation. Anjomshoae et al. [35] suggested a model-agnostic explanation technique based on Contextual Importance and Contextual Utility. This system uses visual and natural language explanations to make black-box predictions understandable to both experts and non-experts. Marwa et al. [36] proposed a novel explainable intrusion detection framework for Internet of IoT networks. It uses the Long Short-Term Memory model for cyberattack detection, and a custom SPIP for feature selection and explanation framework. It integrated SHAP, Permutation Feature Importance and Partial Dependence Plots (PDP) to generate meaningful input features and interpret model behaviour.

Zolanvari et al. [37] propose a novel, statistical, model-agnostic XAI framework. The framework explains the outputs of a primary black-box AI by modelling its statistical behaviour. The framework emphasises speed for real-time numerical applications and claims superiority over LIME in terms of explainability. However, in this paper, we explore enhancing IoT IDS transparency by applying SHAP and LIME to inherently interpretable models like Decision Trees and Logistic Regression. Its novelty lies in analysing XAI's deeper insights for these models and developing user-centric visualisations. Ali et al. [38] proposed HDRL-IDS, a Hybrid Deep Reinforcement Learning system for adaptive threat detection in 5G/IoMT. A novel 5G testbed dataset has been created for the evaluation. HDRL-IDS operates in a black-box setting and does not explicitly integrate XAI or focus on explaining its adaptive DRL decisions to humans. Ali et al. [39] propose a novel feature engineering method (LEMDA) for supervised ML-based IDS in IoT systems. Its primary objective is to reduce data dimensionality and improve model performance (F1-score, accuracy, safety score). LEMDA aims to simplify models and improve efficiency; however, it does not directly focus on XAI or the interpretability of the final model's decisions, nor does it explicitly utilise SHAP or LIME. Instead, it focuses on preprocessing data to make any ML model perform better and faster. However, this paper focuses on enhancing transparency by applying XAI (SHAP, LIME) to inherently interpretable supervised learning models (Decision Trees, Logistic Regression).

4. Proposed methodology: An explainable AI-based IDS framework for IoT environments

XAI provides a clear, interpretable explanation for why a particular network activity is flagged as malicious. This explanation not only enhances trust and accountability but also enables more effective mitigation strategies and informed decision-making for future threat prevention. This insight helps administrators to pinpoint compromised network segments, identify critical features exploited by attackers, and refine security policies. By providing this contextual information through XAI, the network analysts can make informed decisions, whether it is debugging the IDS model, updating threat intelligence, or implementing new security measures to prevent similar attacks. The proposed framework aims to develop a robust and effective explainable AI-based IDS, leveraging various XAI methods to generate both local and global explanations. The local approach focuses on providing detailed explanations for individual network traffic instances classified as intrusions. The global approach offers overarching insights into the complex functioning of the IDS model. The local and global models increase the security expert's and users' trust in the IDS's specific decisions. The innovation of this framework lies in its integrated combination of local and global explanations, which enables a comprehensive description of the inner workings of Decision Tree models.

4.1. Underlying model selection: Decision trees and linear regression for interpretable IoT data analysis

The selection of Decision Trees as one of the underlying machine learning models for this IDS framework is strategically driven by its inherent interpretability and suitability for classification problems within IoT network environments. Unlike complex "black-box" models, Decision Trees are "white-box" models, meaning their decision-making process is transparent and easily understood by humans. Each path from the root to a leaf in a Decision Tree represents a clear set of rules, directly explaining why a particular network traffic instance is classified as normal or an intrusion. This inherent explainability aligns perfectly with the core objective of XAI, reducing the need for post-hoc explanation techniques in some aspects and immediately fostering trust in the IDS's judgments.

Decision Trees are well-suited for handling both numerical and categorical data commonly found in network traffic datasets. Their ability to model non-linear relationships and identify important features through a hierarchical structure makes them effective for detecting various types of cyberattacks. Furthermore, for IoT environments, Decision Trees can be relatively lightweight in terms of computational resources compared to deep learning models, making them potentially more suitable for deployment on resource-constrained edge devices. While complex attack patterns might benefit from more sophisticated models, the clear, rule-based nature of Decision Trees provides a strong foundation for an interpretable ID. This would allow security analysts to quickly grasp the logic behind an intrusion alert and take precise actions.

In addition to Decision Trees, Logistic Regression will also be employed as an underlying machine learning model. Logistic Regression is a powerful and widely used linear model for binary classification, offering a high degree of inherent interpretability. Its coefficients directly indicate the direction and magnitude of a feature's influence on the log-odds of the target variable. This simplicity and transparency make it an excellent choice for an explainable IDS, as its decision-making process is relatively easy to comprehend without extensive post-hoc analysis. Furthermore, Logistic Regression provides probabilistic outputs, which can be valuable information for security analysts in assessing the confidence of an intrusion detection and prioritising alerts based on risk. Its computational efficiency makes it highly suitable for resource-constrained IoT environments, allowing for rapid training and inference. By including Logistic Regression, this study can also establish a robust baseline for comparison against the more complex Decision Tree model.

4.2. Local explanations for individual IoT intrusion events

The proposed framework primarily utilises SHAP and LIME to generate local explanations for individual predictions, providing granular insights into specific intrusion alerts. These two methods are chosen for their complementary strengths in providing both theoretically sound and locally faithful explanations, which are crucial for building trust and understanding in IoT IDS.

- SHAP values quantify the contribution of each input feature to an individual prediction. For a specific IoT network packet flagged as malicious, SHAP will visually explain which features (e.g., unusual source port, abnormally high packet size, or specific flag combinations) were most influential in that particular classification. This intuitive visualisation allows XAI recipients, such as security analysts, to easily understand how the values of specific features affected the predicted probabilities made by the IDS model for that unique instance. SHAP is highly motivated by its strong theoretical foundation in cooperative game theory, ensuring that feature contributions are fairly distributed and sum up to the model's output. This property provides a robust and consistent measure of importance for individual predictions. For Example, If an IoT IDS flags a specific data flow from a smart sensor as "malicious", SHAP can reveal that the "unusually high frequency of connection attempts" from that sensor's IP address, combined with a "non-standard destination port", were the primary drivers for the intrusion prediction.
- LIME complements SHAP by focusing on local fidelity. It explains individual predictions by approximating the black-box model locally with a simpler, interpretable model (e.g., a linear model). For a specific instance, LIME generates perturbed versions of that instance, obtains predictions from the black-box model for these perturbed points, and then trains a weighted, interpretable model on this locally sampled data. This approach is highly effective for understanding the immediate vicinity of a prediction and is model-agnostic, making it versatile for any underlying IDS model. LIME's strength lies in its ability to provide quick, intuitive explanations for single instances, which is vital for real-time analysis in dynamic IoT environments where a security expert needs immediate clarity on a specific alert. For Example, for a specific IoT device's network traffic, LIME could highlight that for a particular packet, the "flag sequence" and "payload size" were the most critical features in its classification as an anomaly, even if other features were within normal bounds.

4.3. Global explanations for overall IoT IDS behaviour

To provide a holistic understanding of the IDS's functioning and the general characteristics of cyberattacks within IoT networks, the proposed framework primarily leverages SHAP for generating global explanations. SHAP (Global Aggregation): Beyond individual predictions, SHAP also provides a powerful global view of feature importance by aggregating individual Shapley values. This is achieved by computing the average of the absolute Shapley values for each feature across all instances in the dataset. After obtaining these average absolute SHAP values, features are typically sorted based on their importance score, and the top features are presented. This provides a unified and theoretically sound measure of global feature importance, which is crucial for understanding the overall behaviour of the IDS. The ability of SHAP to seamlessly transition from local to global explanations makes it an invaluable tool for comprehensive IDS analysis. For example, a global SHAP summary plot could visually represent the overall impact and distribution of SHAP values for features like "packet rate" or "source port" across all IoT traffic, showing their general contribution to intrusion detection and highlighting features that consistently push predictions toward "intrusion" or "normal".

5. Experimental setup and evaluation metrics

The experiment consists of four main processes, which are data processing, model training, model interpretation, and interpretation and analysis. The data processing process mainly uses CICIDS2017 and UNSW-NB15 as the dataset to perform the necessary processing, including data clarity, feature selection, and data division. The processed data are then applied to the decision tree and logistic regression models for training. In this experiment, we will use the multi-classification method to keep the original dataset features classified and the binary classification method to classify the data into two categories: 'normal' and 'attack' for training. The results of the binary classification are interpreted using the SHAP and LIME models. The corresponding confusion matrices are plotted for the multi-classification results. Finally, after the XAI model interpretation, the results are analysed and evaluated accordingly.

Two publicly available datasets are used in this study which are: the UNSW-NB15 and the CICIDS2017. These datasets were chosen because of their broad coverage of the diversity and complexity of modern cybersecurity threats and are well-suited for evaluating the performance of machine learning models and XAI tools. To ensure the reliability of the statistical results, the experiments will be repeated at least twice for each configuration. This study uses a multi-subject comparative analysis to apply different XAI models to multiple machine learning models for the purpose of assessing performance differences.

The UNSW-NB15 dataset was created by the University of New South Wales (UNSW) in Australia to replace the outdated KDD99 dataset. The dataset contains a total of nine types of attacks, namely Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic,

Table 1
Confusion matrix schematic.

		Predicted		
		Positive	Negative	Total
Actual	Positive	TP	FP	$TP + FP$
	Negative	FN	TN	$FN + TN$
	Total	$TP + FN$	$FP + TN$	N

Table 2
Performance Metrics for CICIDS2017 trained by Decision Tree with multi-classification.

Accuracy	Precision	Recall	F1-score
0.9968	0.7751	0.8227	0.7862

Reconnaissance, Shellcode and Worms. These types of attacks are widely found in modern network environments, making UNSW-NB15 a highly diverse and representative dataset. The dataset was generated from 100 GB of raw network traffic, processed by Bro-IDS and extracted 49 features with high realistic adaptability. The CICIDS2017 dataset, created by the Canadian Institute for Cyber Security (CIC), is a comprehensive dataset for evaluating intrusion detection systems. It simulates real-world network environments and contains seven days of network traffic data covering a wide range of attack types, including Brute Force, Heartbleed, DoS, DDoS, Botnet, Web Attack, Infiltration, PortScan, and FTP-Patator. The dataset's complexity and diversity make it ideal for studying the application of machine learning and XAI techniques to intrusion detection.

5.1. *Predictive performance evaluation indicators*

In this experiment, four metrics, accuracy, precision, recall and F1-score, are introduced to evaluate the prediction performance of different machine learning models. The prediction performance of the models is further evaluated by combining the confusion matrix. Table 1 illustrates the specific definition of the confusion matrix, which includes specific metrics such as True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN). True positives refer to samples whose true labelling label is positive and which are predicted to be positive by the model. True negatives denote samples whose true labelling label is negative and which are predicted to be negative by the model. False positives refer to samples whose true labels are negative but are predicted to be positive by the model. False negatives denote samples whose true labels are positive but predicted to be negative by the model. Based on the above metrics, the specific performance of the model can be further evaluated, mainly through the following four performance evaluation metrics:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

$$F1\text{-}score = \frac{2 * Recall * Precision}{Recall + Prec} \tag{4}$$

6. Results and analysis

In this section, we evaluate the performance of decision trees and logistic regression in conjunction with XAI techniques to assess both detection effectiveness and interpretability.

6.1. *Analysing XAI for CICIDS2017*

In this section, we analyse the performance of machine learning models and XAI methods on the CICIDS2017 dataset.

6.1.1. *Multiclass analysis for decision tree*

We initially evaluated the performance of decision tree models on the CICIDS2017 dataset using a multiclass classification approach, where each network flow instance is assigned to one of several attack or benign categories. The results of this evaluation are illustrated in Fig. 1, which presents the normalised confusion matrix. The model demonstrates high predictive accuracy for several major categories, including BENIGN, DDoS, and DoS Hulk, as indicated by the strong values along the matrix's diagonal. This reflects the decision tree's capacity to accurately learn dominant attack patterns in the dataset.

However, certain categories with similar behaviour profiles show signs of misclassification. For instance, DoS GoldenEye and FTP-Patator are classified correctly in most cases but still experience some overlap, with normalised accuracy values of 0.93 and

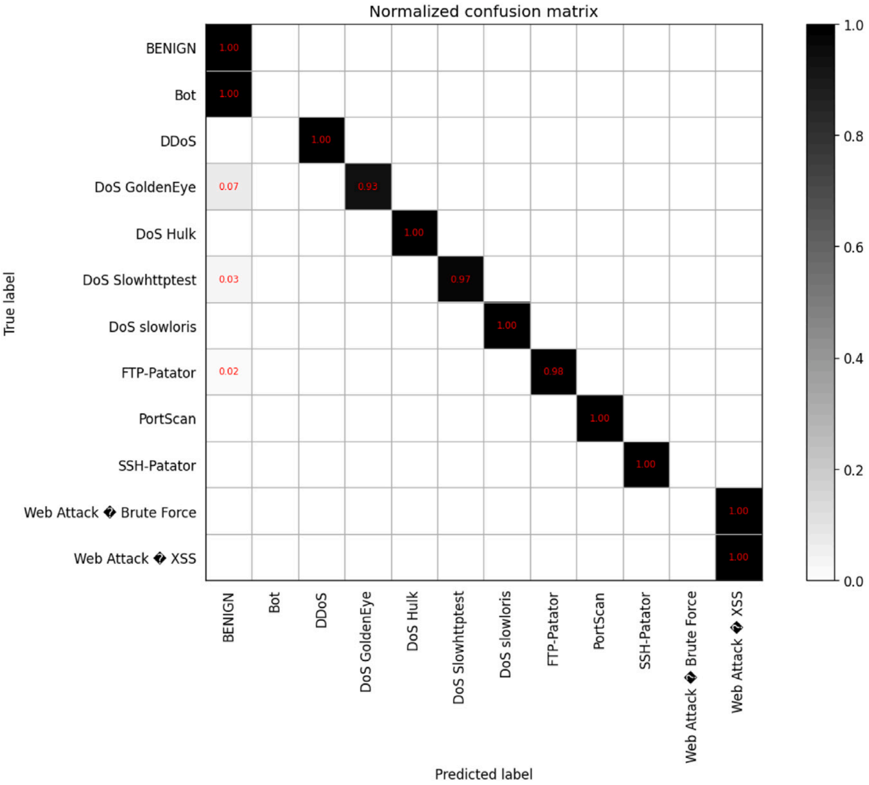


Fig. 1. Confusion Matrix for CICIDS2017 trained by decision tree with multi-classification.

Table 3
Performance metrics for CICIDS2017 trained by decision tree with binary classification.

Accuracy	Precision	Recall	F1-score
0.9980	0.9937	0.9961	0.9949

0.98, respectively. These results suggest that while decision trees are effective for capturing hierarchical and rule-based structures in the data, their performance can be challenged when attack types exhibit subtle behavioural similarities.

Quantitative evaluation metrics are provided in Table 2. The decision tree achieves a high overall accuracy of 0.9968, confirming its robust general performance. However, class-wise metrics reveal nuances in predictive capability. The precision of 0.7751 indicates that while most predicted attacks are correct, the model occasionally confuses similar attacks, reducing its ability to perfectly distinguish between them. The recall score of 0.8227 reflects a strong ability to detect actual attacks, although a fraction are still missed. The F1-score of 0.7862 captures the balance between precision and recall, showing that the model maintains reliable but improvable classification performance across classes.

To understand the underlying decision logic of the model, we applied SHAP, a widely used XAI method for interpreting tree-based models. Fig. 2 shows SHAP value plots for the BENIGN class in the multiclass classification task. Each feature's contribution to the prediction is visualised, with features ordered by their importance. The SHAP values on the x-axis represent the impact of a feature on the model's output: positive values push the prediction toward the BENIGN class, while negative values push it away.

The analysis reveals that Average Packet Size and Bwd Packet Length Std are the most influential features in the decision tree's classification for BENIGN flows, exhibiting consistent and strong contributions across multiple samples. Other features, such as Fwd Header Length and Fwd IAT Min, also play a role, though with less dominant impact. The colour distribution (red = high feature value, blue = low) further contextualises how feature magnitude correlates with classification direction.

6.1.2. Binary class analysis for decision tree

To further evaluate the effectiveness of decision tree models in intrusion detection, we conducted a binary classification task on the CICIDS2017 dataset. In this context, all attack categories were aggregated into a single class labelled ATTACK, while normal traffic was retained as the BENIGN class. This binary categorisation reflects a practical intrusion detection system (IDS) scenario where the primary objective is to distinguish between normal and malicious behaviour, regardless of the specific attack type.

Fig. 3 presents the confusion matrix for the decision tree model trained on the binary-labelled dataset. The distribution indicates that the model exhibits excellent discriminatory power, with only a marginal number of misclassifications. In particular, the very

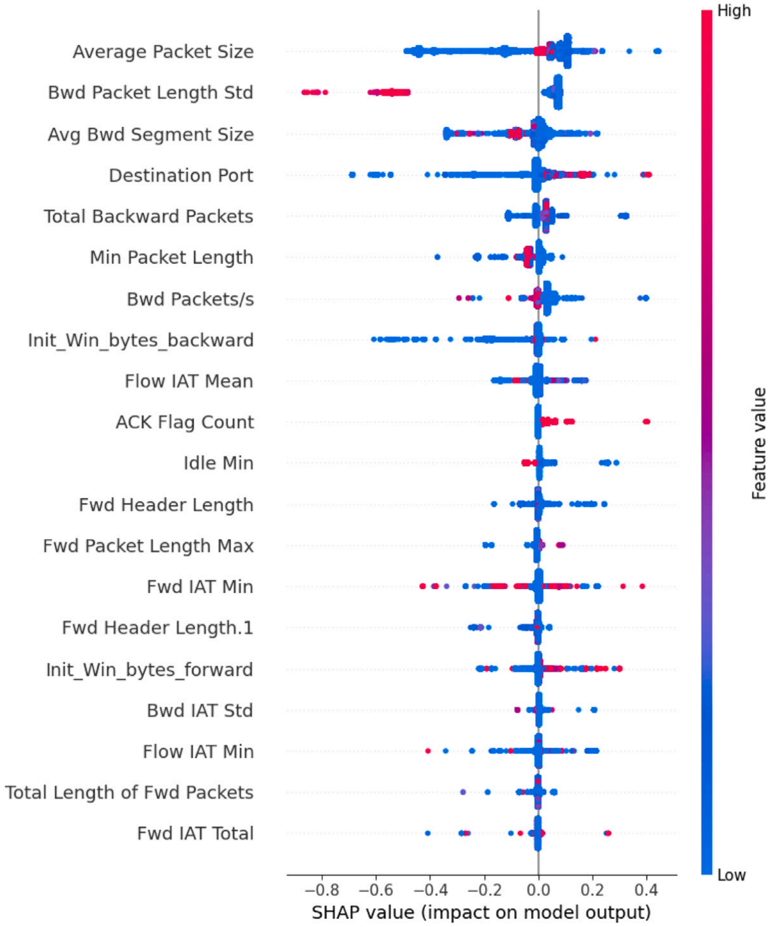


Fig. 2. SHAP summary plot for BENIGN class on CICIDS2017 dataset.

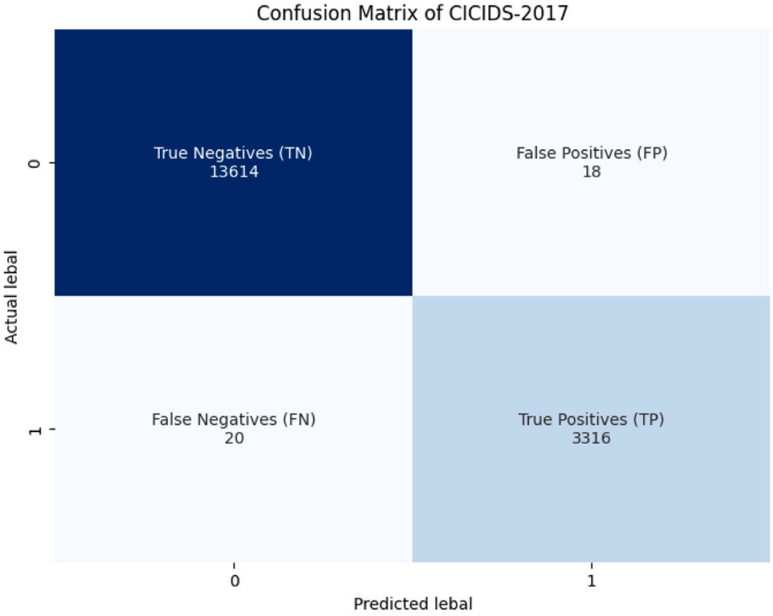


Fig. 3. Confusion matrix of CICIDS2017 trained by decision tree with binary classification.

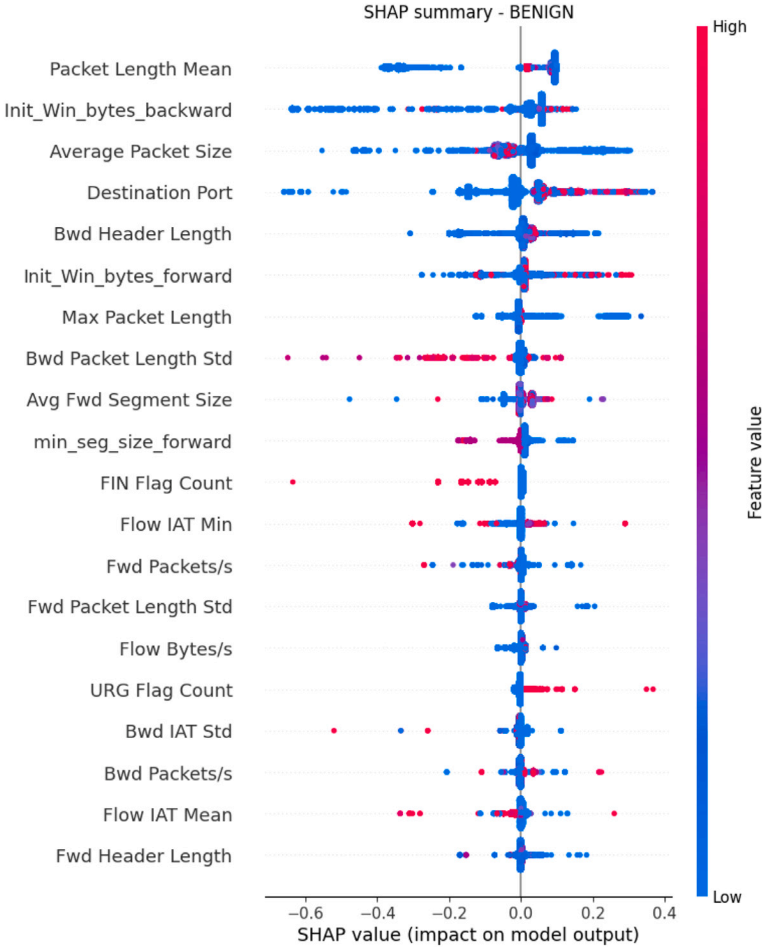


Fig. 4. SHAP summary plot on BENIGN class using decision tree on CICIDS2017 dataset with binary classification.

low false positive rate suggests that the decision tree is highly reliable in identifying legitimate traffic, which is crucial in operational IDS deployments to reduce alert fatigue and unnecessary investigations. Quantitative performance metrics, shown in Table 3, further support the model's effectiveness in binary classification: the accuracy of 0.9980 indicates that nearly all predictions (both BENIGN and ATTACK) are correct while achieving the precision of 0.9937. The high precision suggests that instances predicted as attacks are indeed malicious, with a minimal rate of false alarms. The system achieves an F1-score of 0.9949, which indicates a strong balance between precision and recall, making the model both reliable and robust for real-world IDS use.

Compared to the multiclass scenario, the binary classification task shows slightly improved overall performance, due in part to the simplified classification boundaries. The decision tree benefits from reduced complexity and is able to generalise more effectively when only required to separate normal from abnormal traffic. These results affirm that decision trees, when applied to binary intrusion detection problems, can offer high accuracy, low false positive rates, and maintain interpretability. Such qualities make them attractive candidates for deployment in network security environments where real-time, explainable, and resource-efficient detection is essential.

6.1.3. SHAP interpretation and analysis

Figs. 4 and 5 present SHAP summary plots for the two binary classes: BENIGN (normal traffic) and ATTACKS (malicious traffic). These visualisations offer critical insights into how specific network features contribute to the classification decisions. Fig. 4 illustrates the SHAP values for samples predicted as BENIGN. This highlights a set of features that drive the decision tree to classify network flows as normal. The most influential features in this class include the following important features. Packet Length Mean: The higher values of average packet length are strongly associated with normal traffic. This suggests that legitimate traffic tends to maintain consistent and stable packet sizes. The second features which contribute more is the TCP window size in the backwards direction. A larger initial window size is typically indicative of healthy, well-established connections commonly found in benign communication. Together, these features signal traffic stability and protocol compliance, which are reliable indicators

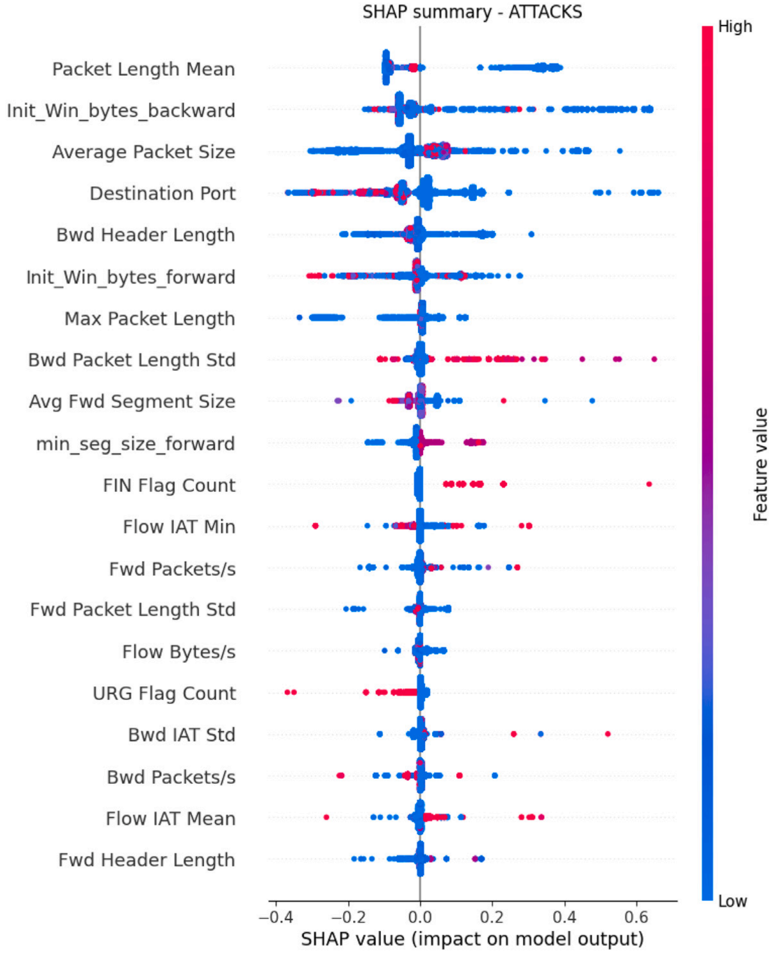
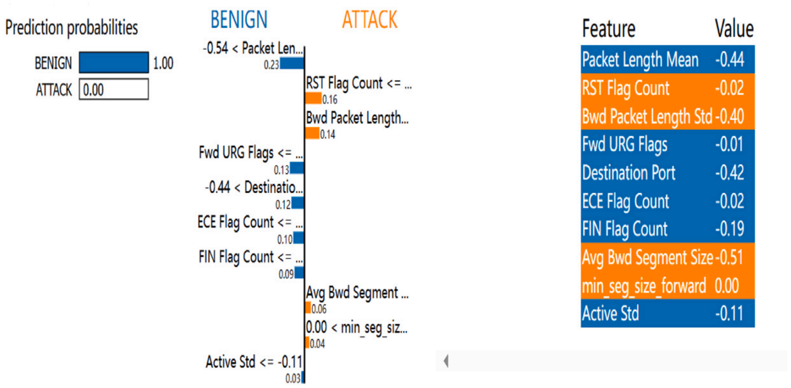


Fig. 5. SHAP summary plot on ATTACKS class using decision tree on CICIDS2017 dataset with binary classification.



Feature	Value
Packet Length Mean	-0.44
RST Flag Count	-0.02
Bwd Packet Length Std	-0.40
Fwd URG Flags	-0.01
Destination Port	-0.42
ECE Flag Count	-0.02
FIN Flag Count	-0.19
Avg Bwd Segment Size	-0.51
min_seg_size_forward	0.00
Active Std	-0.11

Fig. 6. LIME explain for a BENIGN sample using decision tree on CICIDS2017 dataset with binary classification.

of non-malicious behaviour. Their positive SHAP values for the BENIGN class reflect their strong contribution toward identifying normal flow patterns.

Fig. 5 displays the SHAP summary for instances classified as ATTACKS, identifying features that influence the model toward predicting malicious activity. The most impactful features for this class include: Packet Length Mean and Average Packet Size. These features play a prominent role, but in the context of attack traffic, both very high and very low values contribute to deviations from

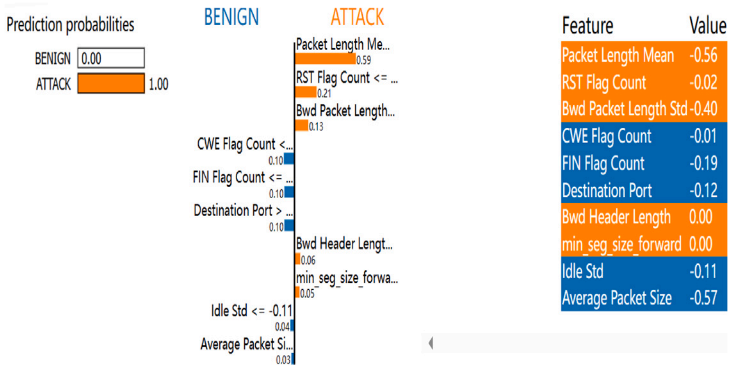


Fig. 7. LIME explain for an ATTACK sample using decision tree on CICIDS2017 dataset with binary classification.

normal patterns. Attack flows often exhibit bursts or abnormally small packets, depending on the attack type (e.g., DoS vs. brute-force). The lower values associated with TCP backwards window size are associated with attack traffic, possibly due to disrupted or spoofed TCP sessions that fail to establish a standard initial window size. The SHAP values show clear variance depending on port usage. Unusual or non-standard destination ports – often targeted during attacks – heavily influence classification toward ATTACKS. This aligns with known behaviours of many intrusion attempts that exploit specific services or vulnerabilities on uncommon ports. The Bwd Header Length and Fwd IAT Min also contribute significantly, which indicates that packet header sizes and inter-arrival timings can differ in coordinated or automated attack behaviours. These patterns suggest that the decision tree leverages both statistical regularities and protocol-level anomalies to distinguish between benign and attack traffic. SHAP values help illuminate how variations in these features – especially when deviating from established norms – drive the model's classification decisions.

The SHAP-based interpretability analysis reinforces the decision tree model's capability to learn meaningful distinctions between normal and malicious traffic in a binary IDS setting. The plots demonstrate that the model's predictions are grounded in explainable feature contributions rather than arbitrary thresholds. Moreover, SHAP provides feature-level granularity, allowing network analysts to understand not just which features matter, but how they matter—i.e., whether high or low values increase the likelihood of an attack.

6.1.4. *LIME interpretation results and analysis*

To gain local interpretability and explain individual classification decisions made by the decision tree model, we employed LIME. LIME provides instance-level insights by approximating the model's behaviour in the region of a specific prediction with an interpretable, linear surrogate model. Figs. 6 and 7 present LIME explanations for two representative samples, one classified as BENIGN and the other as ATTACK. Each figure shows the top ten contributing features, along with their individual weights in influencing the model's prediction. The feature contributions are colour-coded: green bars push the prediction toward the predicted class, while red bars indicate the opposite, which slightly favours the other class.

Fig. 6 visualises a sample that the model classified as BENIGN with 100% confidence. The most influential feature is Packet Length Mean (value: −0.44). This feature contributes strongly toward the BENIGN class. The low average packet length suggests a stable, unfragmented communication pattern, which is a typical attribute for normal network behaviour. Other key features that contribute toward the BENIGN prediction include Fwd URG Flags and the destination port. The low value of FWD URG is indicative of non-urgent traffic, aligning with benign activity. On the other hand, features like RST Flag Count (used to reset TCP connections) show a weak inclination toward the ATTACK class, but their contribution is minimal and insufficient to influence the overall classification. This confirms that the model relies on strong, coherent signals rather than being swayed by isolated anomalies.

Fig. 7 shows the explanation for a sample predicted as ATTACK with 100% confidence. Packet length can contribute as significantly high or irregular average packet length indicates anomalous packet structures often seen in attack vectors such as DDoS or flooding. The High variability in the size of packets in the reverse direction also suggests abnormal behaviour in the communication flow. While a few features – such as Destination Port and Flow IAT Mean – offer weak support for the BENIGN class, they are outweighed by stronger attack indicators. The decision tree model, therefore, maintains high certainty due to a consistent alignment of high-impact features with known attack patterns.

These LIME visualisations are transparent, understandable explanations of the particular feature contributions to the model's binary outputs.

6.1.5. *Multi-class analysis for logistic regression*

Fig. 8 presents the normalised confusion matrix of the logistic regression model trained on the CICIDS2017 dataset for multiclass classification. The matrix visually represents the model's ability to correctly classify various categories of network traffic, including both benign and multiple attack types. Correct predictions are indicated along the main diagonal, where a higher value signifies a greater proportion of correctly classified samples for that class. The logistic regression model demonstrates strong performance across several prominent classes. For example, BENIGN traffic is correctly identified with a 97% accuracy rate. The high accuracy

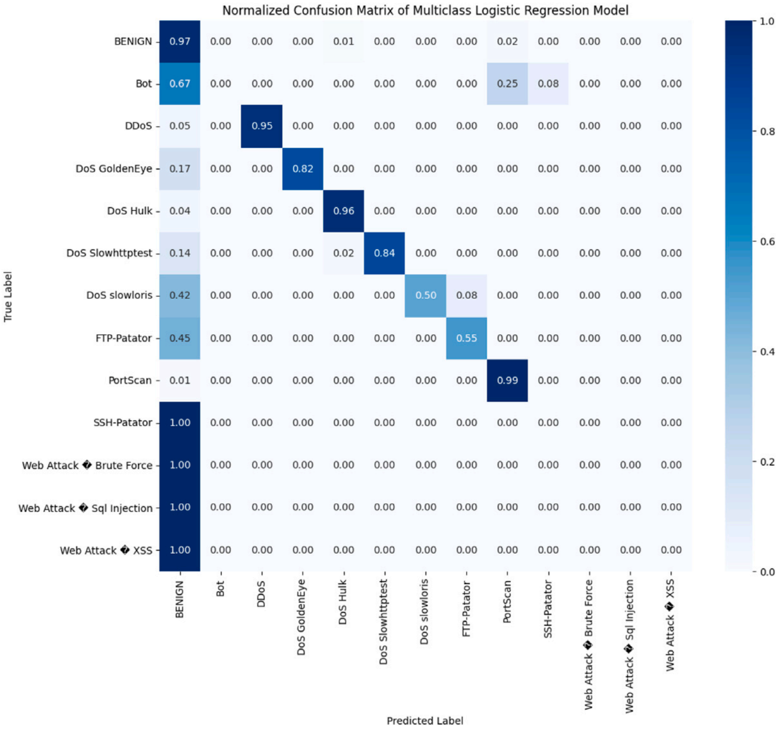


Fig. 8. Confusion matrix for CICIDS2017 trained by logistic regression with multi-classification.

Table 4
Performance metrics for CICIDS2017 trained by logistic regression with multi classification.

Accuracy	Precision	Recall	F1-score
0.9370	0.9364	0.9370	0.9366

is also observed for DDoS (95%) and DoS Hulk (96%), which indicates a reliable detection of large-scale denial-of-service attacks with distinguishable traffic patterns. However, the model struggles with some categories. For example, it detects DoS GoldenEye, with an accuracy of 82%. This affected prediction might be due to DoS Goldeneye similarity with other DoS variants in packet size and flow features. FTP-Patator performs poorly with an accuracy of 55%. This highlights a significant challenge in detecting brute-force attempts over FTP, possibly due to subtle differences between attack and benign login patterns or their low occurrence in the dataset. These limitations suggest that feature overlap among certain attack categories reduces the separability required by logistic regression, which relies on linear decision boundaries.

Table 4 provides the overall performance metrics of the logistic regression model on the same multiclass task. The model achieves a high accuracy of 0.9370 and a precision of 0.9364. The low precision indicates a low rate of false positives. The recall of 0.9370 shows that the model is equally effective in identifying actual attacks, minimising false negatives. The F1-score of 0.9366 confirms a strong balance between precision and recall.

Notably, some attack types are detected with near-perfect accuracy, such as SSH-Patator and PortScan, both with classification rates approaching 1.00. This suggests that the model is particularly effective for attacks that exhibit distinctive behavioural signatures, such as repeated failed login attempts (SSH-Patator) or abnormal scanning patterns (PortScan).

6.1.6. Binary classification for logistic regression

Fig. 9 shows the confusion matrix for the logistic regression model trained on the CICIDS2017 dataset for binary classification, where traffic is classified as either BENIGN (normal) or ATTACK. The model achieved 19,682 true negatives and 4166 true positives, which clearly indicates its strength in accurately identifying both normal and malicious traffic. These results highlight that the logistic regression model can successfully distinguish between benign and attack flows in the majority of cases. However, the model also produced 735 false positives – instances where benign traffic was incorrectly classified as malicious – and 868 false negatives, where actual attack traffic was classified as normal. These misclassifications are notable because they may impact the practical utility of the model in a real-world IDS. False positives can lead to unnecessary alerts and increased operational overhead, while false negatives are more critical, as they represent undetected attacks that could lead to security breaches. The presence of these errors suggests that, although the model is effective overall, there is room for improvement in refining decision boundaries and enhancing feature discrimination.

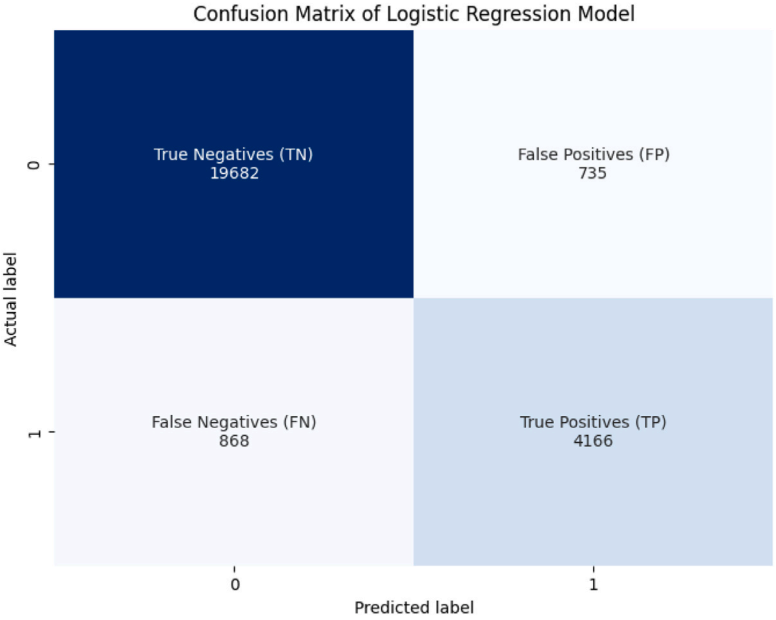


Fig. 9. Confusion matrix for CICIDS2017 trained by logistic regression with binary classification.

Table 5
Performance Metrics for CICIDS2017 trained by Logistic Regression with binary classification.

Accuracy	Precision	Recall	F1-score
0.9370	0.9364	0.9370	0.9366

The model's performance metrics, presented in Table 5, further support these observations. It attained an accuracy of 0.9370, demonstrating strong overall classification capability. The precision of 0.9364 indicates that when the model predicts an attack, it is correct most of the time, with a relatively low false alarm rate. Its recall value of 0.9370 confirms that it is able to correctly identify a large portion of actual attacks, minimising missed detections. Finally, the F1-score of 0.9366 shows a well-balanced trade-off between precision and recall, validating the model's robustness and stability in the binary classification setting. Logistic regression performs reliably in distinguishing between normal and attack traffic in the CICIDS2017 dataset under binary classification.

6.1.7. SHAP interpretation results and analysis for logistic regression

Fig. 10 presents the SHAP summary plot for the logistic regression model trained for binary classification on the CICIDS2017 dataset. As a linear model, logistic regression produces a global interpretation of feature importance, where the SHAP values are uniform across all samples. The global SHAP summary helps to identify the most influential features that consistently drive the model's decisions across the entire dataset. The plot reveals that the features with the highest impact on the model's outputs are Packet Length Variance, Flow IAT Max, Average Packet Size, Max Packet Length, and Idle Max. These features encapsulate critical aspects of network traffic behaviour. For instance, high packet length variance and unusually large inter-arrival times (IATs) may signal anomalous or irregular flows, often seen in attacks. Similarly, features like average and maximum packet sizes help distinguish between typical user activity and traffic patterns generated by malicious entities such as DDoS bots or scanning tools. The SHAP values show that these variables have both positive and negative contributions depending on their values, and collectively they guide the model's predictions toward classifying traffic as either BENIGN or ATTACK.

Fig. 11 complements this analysis by providing a SHAP Force Plot for a single sample from the dataset. This provides a detailed, local explanation of the model's decision-making. The force plot visualises the cumulative effect of individual features on the model's output. In this instance, the base value (log-odds) and the final output prediction are both approximately −7.43. This indicates that the positive and negative feature contributions largely cancel each other out. This suggests that the sample sits near the model's decision boundary, where it does not strongly lean toward either class. The logistic regression model finds itself in a state of indecision, as certain features push the prediction toward the BENIGN class while others pull it toward the ATTACK class. This highlights the importance of understanding the interplay of features in borderline predictions. The SHAP force plot, in this case, provides a transparent and interpretable visualisation that supports model auditing, debugging, and trust in AI-driven IDS.

6.1.8. LIME interpretation analysis for logistic regression

Figs. 12 and 13 illustrate LIME visualisations that explain the predictions made by the logistic regression model. LIME provides a localised, sample-specific interpretation by approximating the model's behaviour in the neighbourhood of a particular instance.

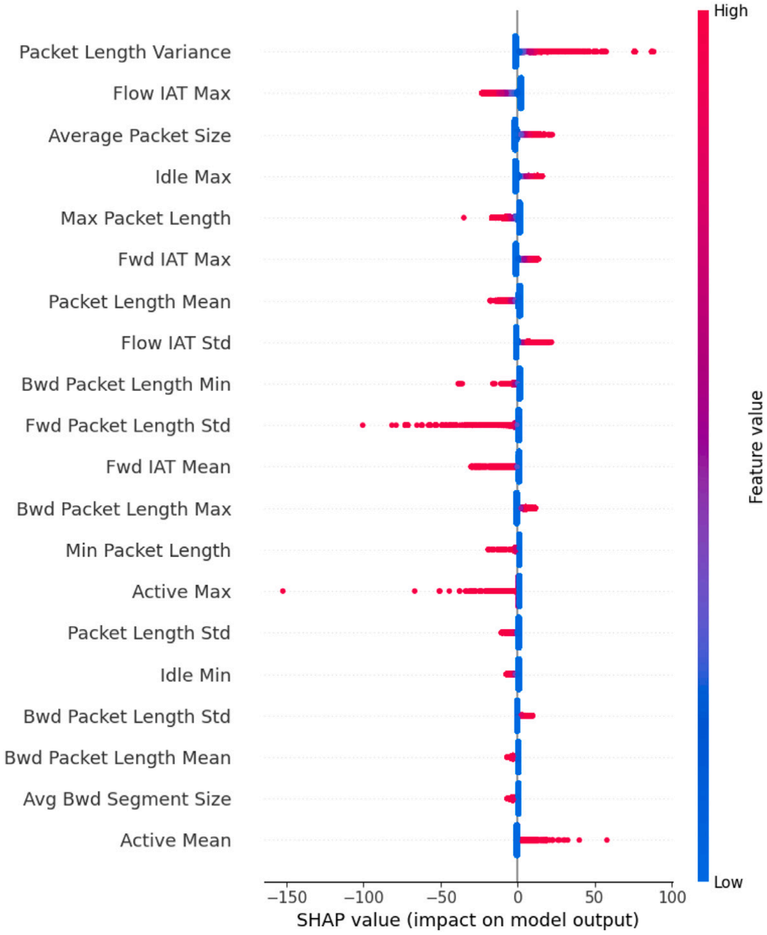


Fig. 10. SHAP summary plot for feature impact in binary classification using logistic regression on CICIDS2017 dataset.

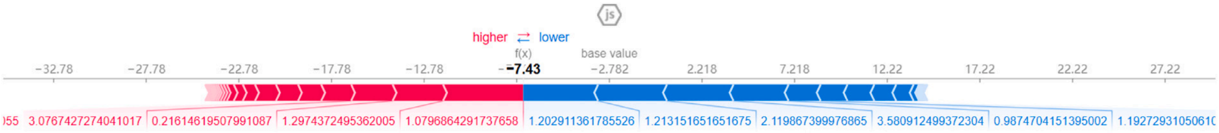


Fig. 11. SHAP force plot in binary classification using logistic regression on CICIDS2017 dataset.

In Fig. 12, the model classifies a given traffic sample as BENIGN with 100% confidence. This high level of certainty is driven by the influence of key features such as Fwd URG Flags and Packet Length Variance, both of which are aligned with typical characteristics of normal network traffic. These features contribute positively to the BENIGN classification by indicating stable and expected traffic behaviour patterns. Although some features—like RST Flag Count and CWE Flag Count—may suggest traits associated with attack traffic but their negative contributions are minimal. As a result, the model confidently and correctly identifies the sample as normal. In contrast, Fig. 13 presents a sample that is classified as ATTACK with 73% confidence. In this case, a different set of features, such as Flow IAT Max, Average Packet Size, Packet Length Mean, and Idle Min, play a central role in steering the prediction toward the attack class. These variables reflect irregularities or anomalies in the timing and structure of the packets, common signs of malicious behaviour like scanning or denial-of-service attacks. While some features, including Idle Max, Packet Length Variance, and Active Mean, do show mild support for the BENIGN class, their influence is not sufficiently strong to counterbalance the dominant attack-related indicators. Consequently, the model leans toward classifying the sample as an attack, though with a lower degree of confidence than in the previous benign case.

6.2. Result analysis for UNSW-NB15

In this section, we analyse the performance of machine learning models and XAI methods on the UNSW-NB15 dataset.

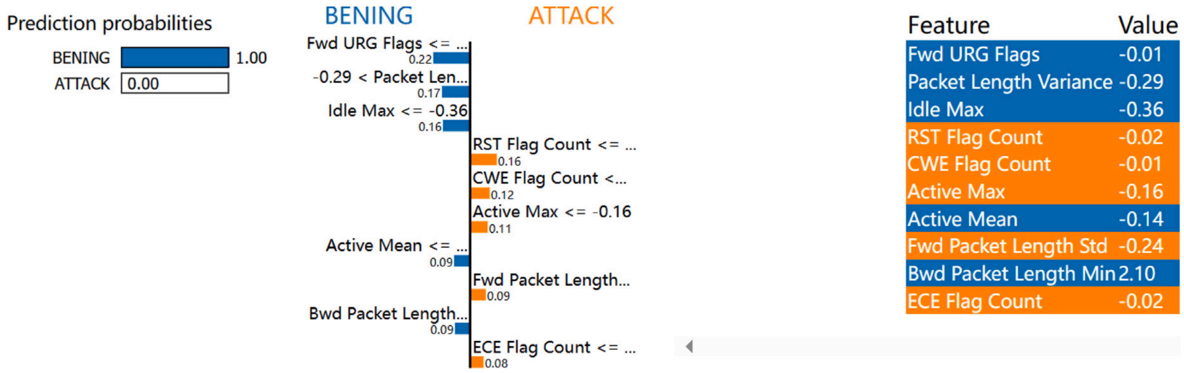


Fig. 12. LIME explain for a BENIGN sample using logistic regression on CICIDS2017 dataset.

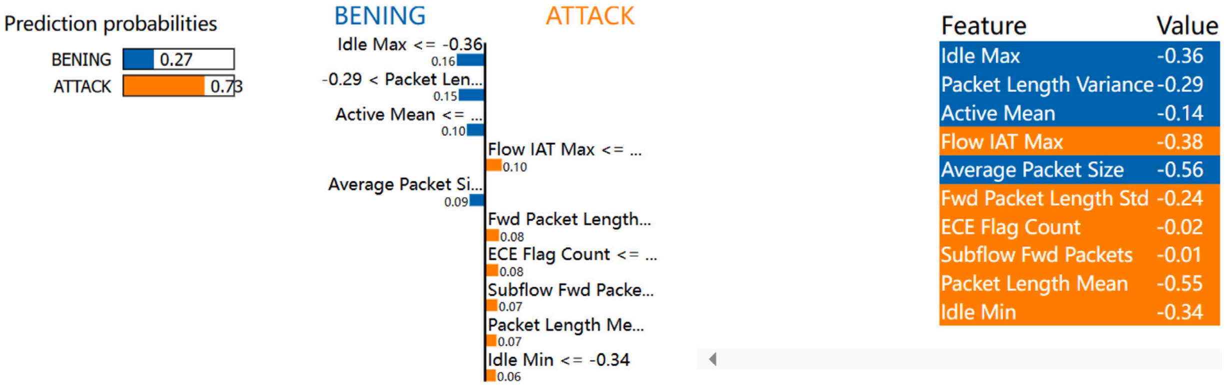


Fig. 13. LIME explain for an ATTACK sample using logistic regression on CICIDS2017 dataset.

Table 6
Performance metrics for UNSW-NB15 trained by logistic regression with multi classification.

Accuracy	Precision	Recall	F1-score
0.6786	0.7342	0.6786	0.6927

6.2.1. *Multiclass analysis for decision tree*

The normalised confusion matrix in Fig. 14 presents the performance of the decision tree model on the UNSW-NB15 dataset for multi-class classification. The model exhibits strong classification ability for certain attack types, achieving notably high accuracy for the Exploits class (0.96) and moderate accuracy for Generic (0.67) and Backdoors (0.71), suggesting its effectiveness in identifying well-represented or distinct attack patterns. However, its performance significantly declines for classes such as Worms (0.05), where most instances are misclassified as DoS or Shellcode. This is likely due to overlapping or highly similar feature distributions among these classes, which impedes the model's capacity to distinguish them accurately. A similar issue is observed in the confusion between DoS and Traffic Analysis attacks. These misclassifications underscore the challenge of detecting attack types with subtle or shared characteristics, indicating the need for enhanced feature engineering or the adoption of more sophisticated models capable of capturing nuanced distinctions.

Table 6 summarises the overall performance of the logistic regression model on the same dataset. It achieves an accuracy of 0.6786, a precision of 0.7342, a recall of 0.6786, and an F1-score of 0.6927. These metrics reflect a reasonably balanced performance, with acceptable precision and a fair compromise between recall and precision across multiple attack categories. While the results demonstrate the model's competence in handling a range of attacks, they also highlight limitations in recognising classes with overlapping features, reinforcing the need for advanced modelling strategies or more informative features to enhance detection accuracy in complex multi-class scenarios.

6.2.2. *Binary classification for decision trees*

Fig. 15 shows the confusion matrix for the decision tree model applied to binary classification on the UNSW-NB15 dataset. The model correctly identified 24,815 true negatives (normal traffic) and 41,797 true positives (attack traffic), demonstrating strong capability in distinguishing between benign and malicious activity. However, it also produced 12,185 false positives—normal traffic mistakenly classified as attacks—and 3535 false negatives—attacks incorrectly labelled as normal. Table 7 details the model's

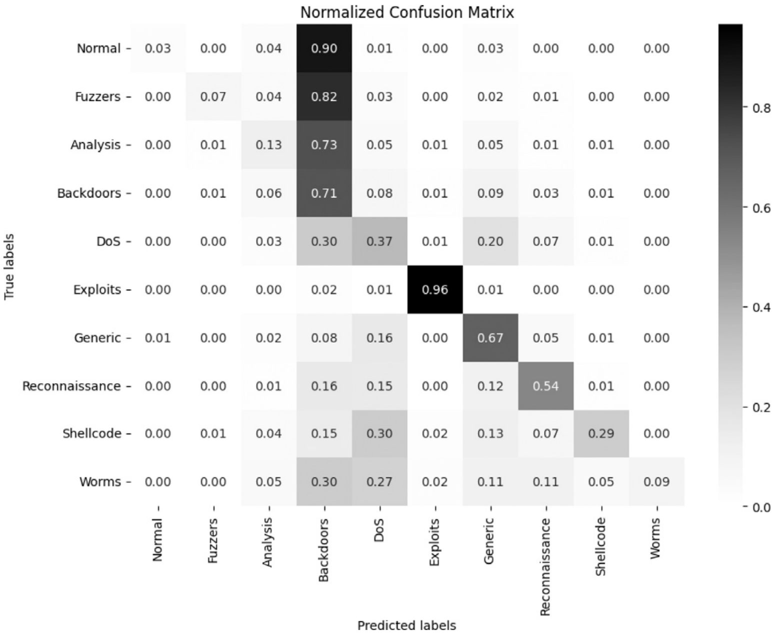


Fig. 14. Confusion matrix for UNSW-NB15 trained by logistic regression with multi classification.

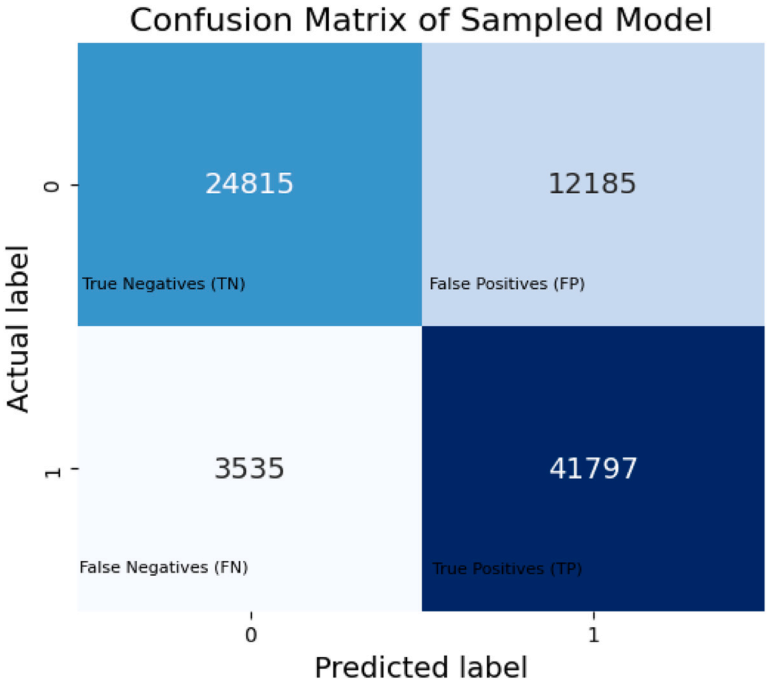


Fig. 15. Confusion matrix for UNSW-NB15 trained by decision tree with binary classification.

performance metrics on the UNSW-NB15 dataset. The decision tree achieves an overall accuracy of 80.90%, reflecting a high rate of correct classifications. Its precision of 0.7743 indicates strong control over false positives, meaning fewer unnecessary alerts. Meanwhile, a recall of 0.9220 highlights the model's robust ability to detect actual attacks, minimising missed detections. The F1-score of 0.8417 reflects a solid balance between precision and recall, indicating stable performance across these metrics. Overall, the model demonstrates dependable effectiveness in identifying attack traffic, with relatively low compromise on false alarms, making it a promising candidate for real-world intrusion detection applications.

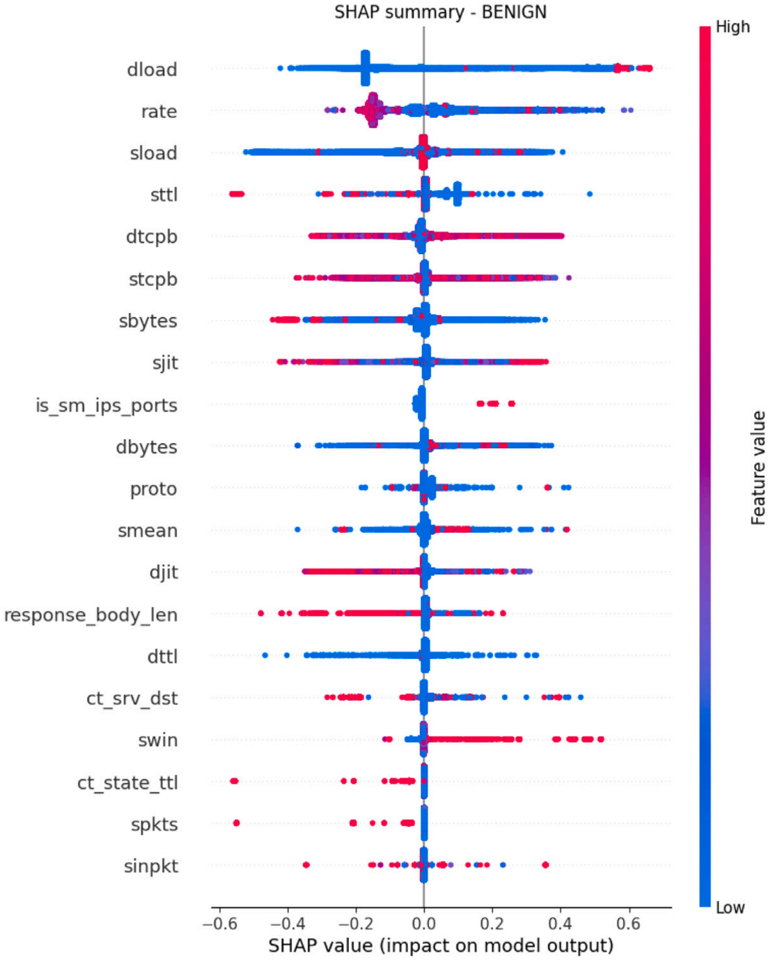


Fig. 16. SHAP summary plot on BENIGN class using decision tree on UNSW-NB15 dataset.

Table 7
Performance metrics for UNSW-NB15 trained by decision tree with binary classification.

Accuracy	Precision	Recall	F1-score
0.8090	0.7743	0.9220	0.8417

6.2.3. SHAP interpretation for decision trees

Figs. 16 and 17 present SHAP-based interpretations of the decision tree model trained on the UNSW-NB15 dataset for binary classification, illustrating the impact of key features on predictions for the BENIGN and ATTACK classes, respectively. Fig. 16 focuses on the features that most strongly influence the model's prediction of normal (BENIGN) traffic. Among these, dload (download load) and rate (data transfer rate) stand out as the most influential, with higher values of these features contributing positively toward predicting BENIGN traffic. This suggests that typical benign network behaviour is characterised by certain levels of download activity and data transfer rates, which the model effectively leverages to differentiate normal traffic from attacks. Conversely, Fig. 17 highlights the same features – dload and rate – but in the context of predicting attack traffic (ATTACK). Interestingly, while these features remain critical, their influence reverses: higher values of dload and rate now push the prediction toward the ATTACK class. This inversion indicates that the model captures nuanced patterns where the magnitude and distribution of these features differ between normal and malicious activity. The contrasting SHAP value directions reveal that it is not the presence of these features, but their relative scale and context that are key to classification. The SHAP analysis not only validates the model's reliance on these features but also provides transparent insight into how their variations guide the classification. This highlights which features should be prioritised for feature engineering or monitoring in practical intrusion detection systems.

6.2.4. LIME interpretation and analysis for decision trees

Fig. 18 illustrates a LIME explanation for a sample confidently predicted as BENIGN with 100% certainty. The feature is_sm_ips_ports representing the identical source and destination IPs and ports, and ct_state_ttl, which represents the Time-to-Live

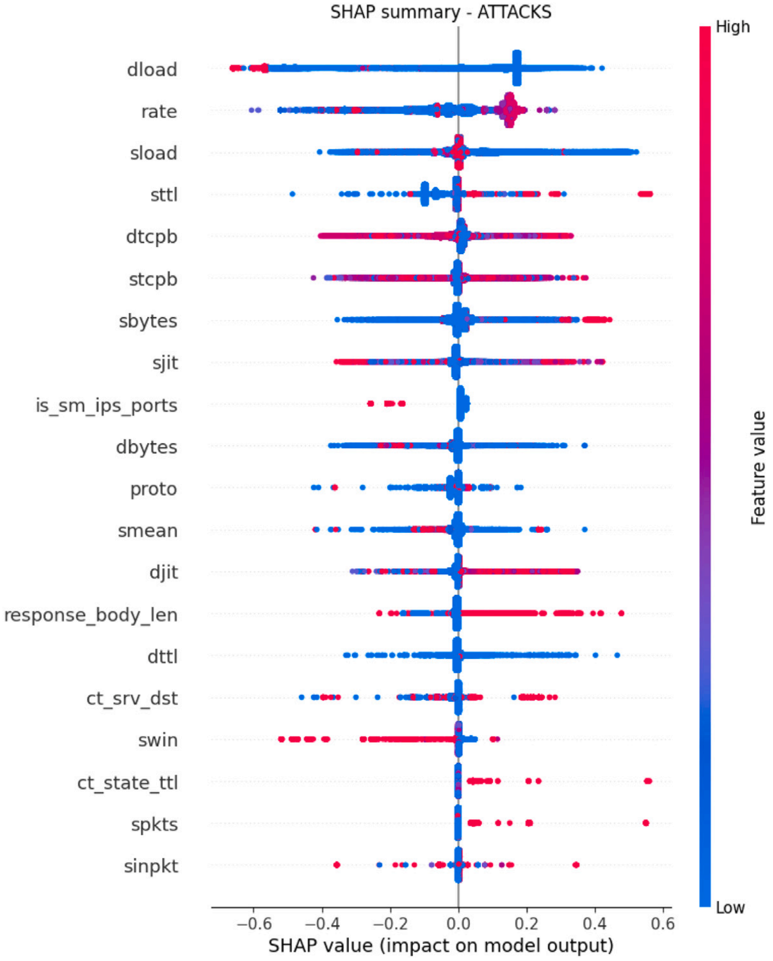


Fig. 17. SHAP summary plot on ATTACKS class using decision tree on UNSW-NB15 dataset.

(TTL) value of the connection state, plays an important role in pushing the model toward this benign classification. The strong positive influence of is_sm_ips_ports suggests that the connection exhibited typical, legitimate network behaviour, as identical IP/port pairs often reflect stable, expected communication patterns rather than spoofing or other malicious tactics. Similarly, ct_state_ttl likely reflected TTL values consistent with normal, long-lived connections, further representing the benign nature of the traffic. Although features like smean (average server packet length) and dload (download traffic load) showed a slight inclination toward an ATTACK prediction, their influence was marginal and insufficient to override the dominant benign signals. This case demonstrates the model's robustness and its capacity to weigh features effectively, prioritising strong, benign indicators while mitigating minor contradictory signals.

In contrast, Fig. 19 provides a LIME explanation for a sample classified as ATTACK with 73% confidence, with the remaining 27% leaning toward BENIGN. Here, is_sm_ips_ports and sttl (server TTL) were the most influential features steering the prediction toward an attack. The values of these features align closely with known patterns of malicious traffic, where identical source and destination IPs/ports might be exploited for reconnaissance or denial-of-service attacks, and abnormal TTL values are often manipulated by attackers to evade detection or fingerprint network devices. Additionally, ct_srv_dst, which counts the number of connections to the server from various sources, and swin (server's receiving window size) further reinforced the attack classification. Elevated ct_srv_dst can signal distributed attack attempts or traffic floods, while unusual swin values may reflect abnormal network behaviour typical of certain exploits or attack techniques. Despite weak benign support from smean and dload, these were insufficient to counterbalance the strong attack indicators. This example highlights the model's ability to integrate multiple features and effectively identify malicious behaviour even in the presence of some conflicting benign evidence.

6.2.5. Multi-classification for logistic regression

Fig. 20 presents the normalised confusion matrix for a logistic regression using a multi-class classification approach. This matrix provides detailed insight into the model's ability to distinguish between various attack types and normal traffic. The model performs

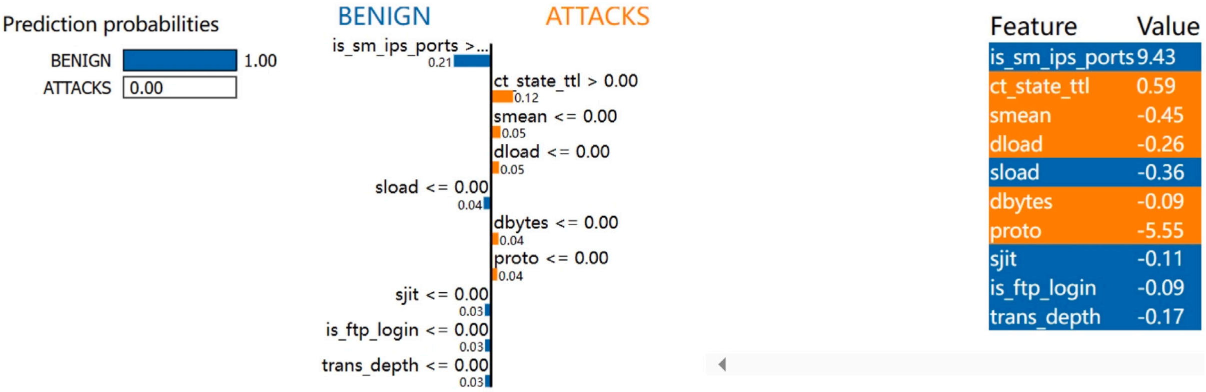


Fig. 18. LIME explain for a BENIGN sample using decision tree on UNSW-NB15 dataset.

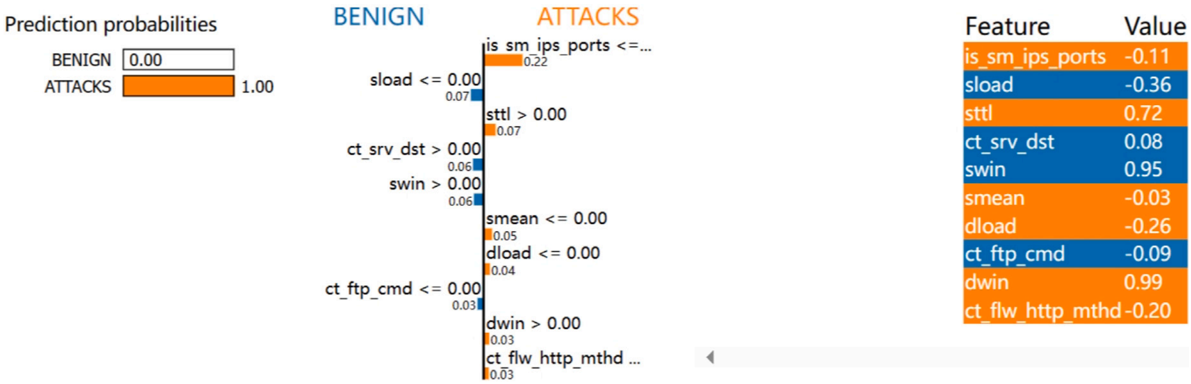


Fig. 19. LIME explain for an ATTACK sample using decision tree on UNSW-NB15 dataset.

Table 8
Performance metrics for UNSW-NB15 trained by logistic regression with multi classification.

Accuracy	Precision	Recall	F1-score
0.6179	0.4738	0.6179	0.5077

reasonably well in identifying certain categories of attacks. For instance, it classifies Exploits with a high degree of accuracy (97%), followed by Generic attacks (87%) and Backdoors (71%). These results suggest that the logistic regression model is effective in recognising attack types that exhibit strong, distinctive feature patterns. This is due to their unique behavioural signatures in network traffic, such as consistent payload patterns or distinct protocol usage. However, the model struggles significantly with other classes, particularly Worms, which it correctly identifies only 14% of the time. Most worm instances are misclassified as DoS or Shellcode, likely due to feature overlap between these categories. Worms and shellcode-based attacks often share payload execution traits and port-scanning behaviours, which can confuse linear models like logistic regression that lack the capacity to model complex, non-linear decision boundaries. Additionally, the model also performs poorly on DoS and Traffic Analysis attacks, where it exhibits confusion with multiple other classes. These misclassifications reflect the model's limited ability to differentiate between attack types that may share similar timing patterns, packet sizes, or protocol characteristics.

The overall performance metrics are summarised in Table 8, which further highlights the model's moderate effectiveness in multi-class IDS. The logistic regression model achieves an accuracy and recall of 61.79%, indicating that it correctly identifies roughly two-thirds of the samples. However, a precision score of only 47.38% reveals that the model frequently raises false alarms, predicting attacks where none exist. This imbalance leads to a relatively low F1-score of 0.5077. These performance outcomes collectively suggest that while logistic regression can detect broader attack patterns, such as exploits or generics but its linear decision-making nature limits its ability to handle classes with overlapping differences.

6.3. *Binary classification for logistic regression*

Fig. 21 illustrates the confusion matrix for the decision tree model applied to the binary classification task. The result provides valuable insight into how well the model distinguishes between normal and attack traffic. The confusion matrix reveals the following

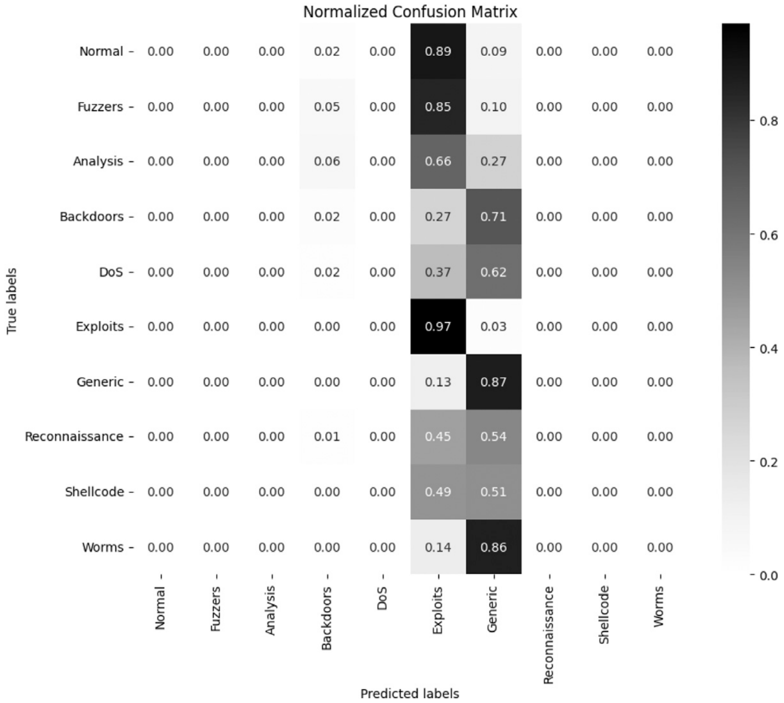


Fig. 20. Confusion matrix for UNSW-NB15 trained by logistic regression with multi classification.

Table 9
Performance metrics for UNSW-NB15 trained by logistic regression with binary classification.

Accuracy	Precision	Recall	F1-score
0.7512	0.7712	0.7512	0.7404

key outcomes: 20,329 true negatives (TN), 16,671 false positives (FP), 3817 false negatives (FN), and 41,515 true positives (TP). These results indicate that the model performs reasonably well in classifying attack traffic, as evidenced by the large number of true positives. Moreover, the model is particularly effective in identifying normal flows, given the relatively high number of true negatives. However, the model also generates a notable number of false positives, which indicates that some benign traffic is incorrectly flagged as malicious. This may lead to unnecessary interventions in a real-world deployment. Similarly, while the false negative rate is comparatively lower, it still indicates that some attack instances are being missed, which could pose a security risk in critical infrastructure scenarios. Table 9 presents the detailed performance metrics for the binary logistic regression model on the same UNSW-NB15 dataset. The model achieves an accuracy of 75.12%, indicating that approximately three out of four predictions are correct across the entire dataset. This level of accuracy is reasonable given the dataset's complexity and class imbalance. The recall value of 91.57% suggests that the model is highly effective in detecting actual attack traffic while successfully identifying a large proportion of true positives. However, the precision value of 71.32% reveals that a significant portion of the samples predicted as attacks are actually benign. This means that while the model detects many attacks, it also produces a moderate number of false alarms, which can undermine the model's practical usability in real-time security monitoring systems. The F1-score of 0.7404 reflects a balanced trade-off between precision and recall, representing the model's overall classification effectiveness. A high F1-score close to 1.0 is desirable in intrusion detection tasks, but 0.7404 is not ideal.

6.3.1. *SHAP interpretation and analysis for logistic regression*

Fig. 22 presents the SHAP summary plot for the logistic regression model in a binary classification setting. The SHAP values in logistic regression reflect the uniform contribution of each feature to all predictions. In this summary plot, the most influential features are identified as djit (destination-to-source jitter), dload (download load), and sload (upload load). These features represent various aspects of network flow behaviour that are commonly indicative of abnormal activity. Notably, djit exhibits a wide distribution of SHAP values, meaning that it significantly affects the model's decision, sometimes pushing the prediction toward the ATTACK class and other times toward the BENIGN class, depending on its value. This broad range suggests that djit is a context-sensitive feature that behaves differently depending on the specific network flow it is associated with. Similarly, high values of dload and sload are typically associated with unusual traffic patterns and are therefore strongly aligned with attack predictions.

To complement the global view, Fig. 23 displays a SHAP force plot for a single instance from the UNSW-NB15 dataset. This visualisation breaks down the model's prediction by showing how each individual feature contributes either positively or negatively

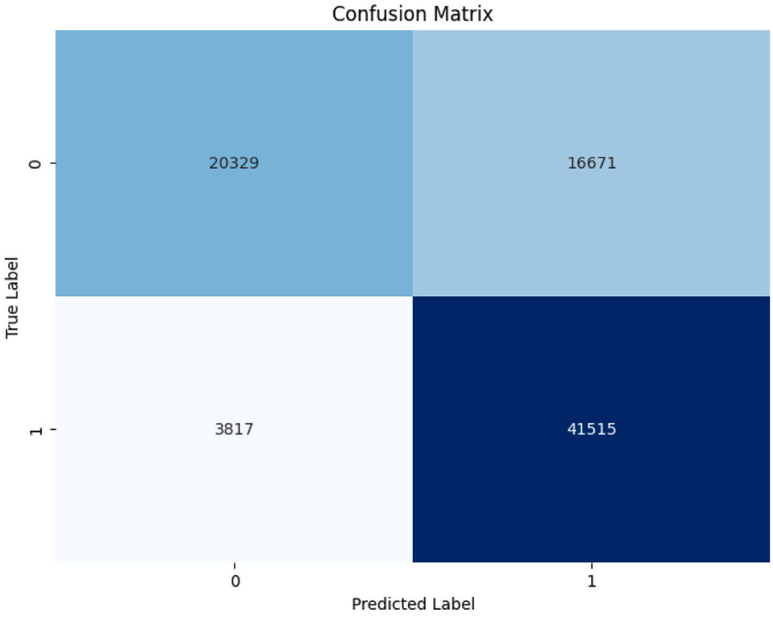


Fig. 21. Confusion matrix for UNSW-NB15 trained by logistic regression with binary classification.

to the final output. In the force plot, the base value represents the model's expected output before seeing the input features, while the output value reflects the final prediction after accounting for all feature effects. In this example, the positive and negative contributions nearly cancel each other out, resulting in a prediction close to the model's baseline. This suggests that the model perceives the sample as being near the decision threshold. The force plot provides interpretability at the individual prediction level, helping analysts and domain experts to identify which features are tipping the model's judgment.

6.3.2. LIME interpretation and analysis for logistic regression

Fig. 24 presents a LIME visualisation for a sample from the BENIGN class. The model classifies the sample as normal with 100% confidence, reflecting its strong certainty in identifying benign traffic. This high level of confidence stems from several key features that exert a strong positive influence on the model's decision. Specifically, features such as dwin (destination window size), dbytes (bytes sent from the destination), trans_depth (depth of the transaction), and djit (destination-to-source jitter) contribute significantly to the classification. These attributes are typically indicative of stable, low-risk network behaviour and are consistent with normal traffic patterns. Few features, such as dpkts (destination packet count), ct_state_ttl (connection state and TTL combination), and dload (download load), offer slight support for an ATTACK classification; their impact on the overall decision is marginal. This suggests that the model is not only highly confident but also resilient to conflicting signals when clear indicators of normal behaviour are present. The LIME explanation in this context demonstrates the model's interpretability and reliability, giving insight into the features most influential in real-time traffic assessment.

Fig. 25 shows the LIME explanation for a sample from the ATTACK class. This particular instance is classified as malicious with a confidence level of 96%, indicating strong model certainty in its classification decision. The primary contributors to this decision include dwin (destination window size), sload (upload load), ackdat (acknowledgement data packets), and is_sm_ips_ports (a boolean feature indicating whether source and destination IP addresses and ports are the same). These features are typically associated with abnormal or suspicious network behaviour—for example, high uploads and acknowledgement patterns are often linked to scanning or flooding activities, and identical source–destination ports may suggest spoofing or reflection-based attacks. Few features—such as dbytes, dttl (destination time-to-live), and trans_depth—lend weak support toward a BENIGN classification; however, they are not influential enough to offset the dominant attack-signalling features. The model's high confidence in labelling the sample as an attack demonstrates its ability to prioritise the most discriminative features, even when minor indicators of benign traffic are present.

6.4. Trade-off between interpretability and accuracy

The development and deployment of AI, ML models while detecting anomalous patterns from network traffic involves navigating a fundamental challenge of trade-off between model interpretability and Detection accuracy. The highly accurate models, like deep learning architectures, operate as black boxes, offering superior performance but little insight into their decision-making processes. Conversely, a simpler model may sacrifice a degree of detection accuracy but offer a detailed explanation of the model's working. The interpretability-accuracy trade-off comes from the inherent complexity of modern machine learning algorithms. Simpler models, like Decision Trees and Logistic Regression, are referred to as "white-box" models because their internal logic is directly observable and

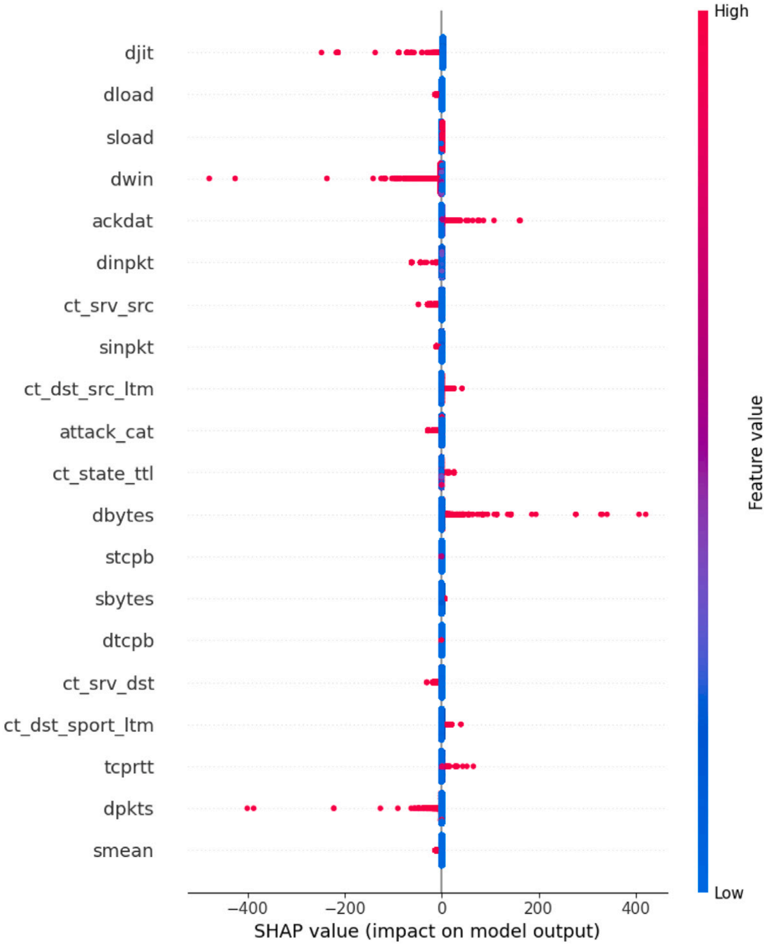


Fig. 22. SHAP summary plot for feature impact in binary classification using logistic regression on UNSW-NB15 dataset.

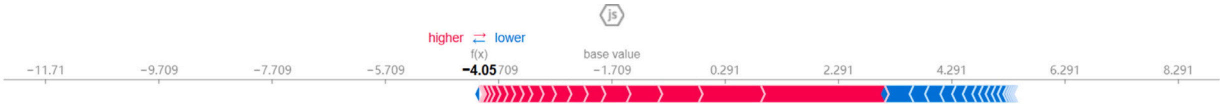


Fig. 23. SHAP force plot in binary classification using logistic regression on UNSW-NB15 dataset.

understandable. Their decision paths or feature weights provide a clear rationale for their predictions. However, their simplicity can limit their ability to capture highly intricate, non-linear relationships and subtle patterns present in complex, high-dimensional data. This may result in lower accuracy compared to complex models. On the other hand, the "black-box" models, such as deep neural networks (e.g., Convolutional Neural Networks, Recurrent Neural Networks) or ensemble methods (e.g., Random Forests, Gradient Boosting Machines) offer acceptable detection accuracy. These models perform well at learning complex representations and achieving high accuracy by identifying highly nuanced patterns. However, their internal working mechanism is not understandable, which makes it challenging to trace a prediction back to specific input features or understand the underlying reasoning. In the context of IoT IDS, where rapid and accurate threat detection is vital, but human trust and intervention are equally critical, this trade-off becomes a central concern.

Decision Trees are a prime example of models that prioritise interpretability. As discussed in Section 4.1, their rule-based structure, where each path from the root to a leaf represents a clear sequence of conditions, makes their decisions inherently transparent. A security analyst can literally follow the logic: "If packet size is greater than X, and destination port is Y, then classify as Z attack". This direct explainability fosters immediate trust and allows for straightforward debugging and policy refinement. From the provided experimental results for the CICIDS2017 dataset trained by a Decision Tree with multi-classification (Table 2 and Fig. 1), we can observe its performance accuracy of 0.9968. This is remarkably high, indicating that the model correctly classified

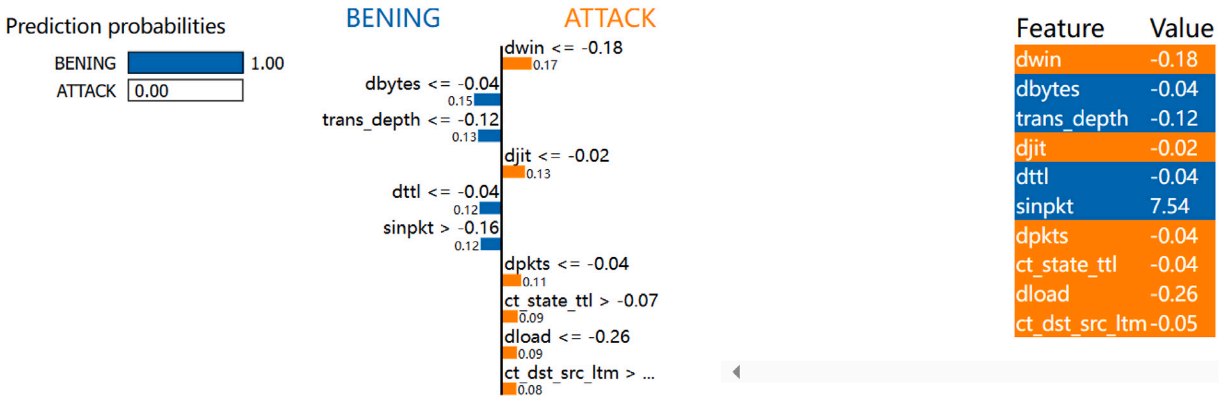


Fig. 24. LIME explain for a BENIGN sample using logistic regression on UNSW-NB15 dataset.

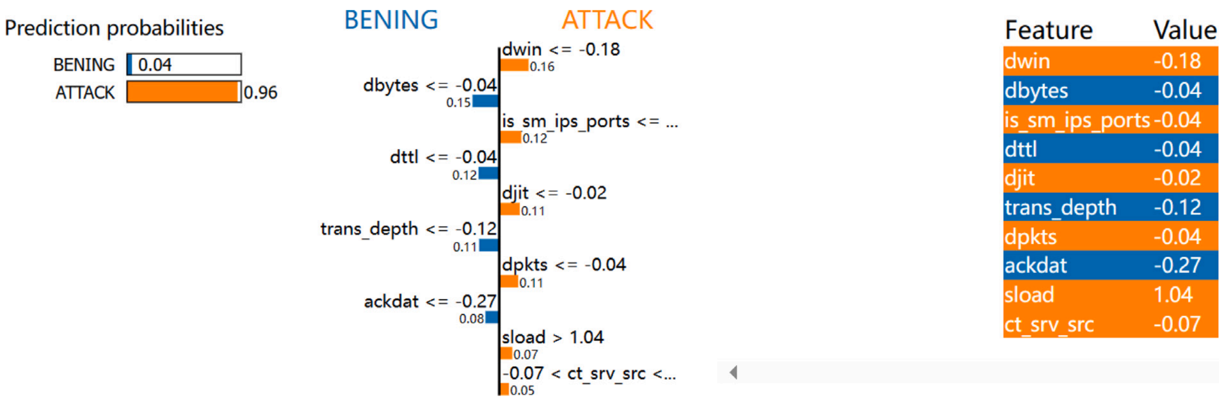


Fig. 25. LIME explain for an ATTACK sample Using logistic regression on UNSW-NB15 dataset.

nearly 99.7% of the network traffic instances. Such high accuracy is desirable for an IDS, as it suggests a strong ability to distinguish between normal and various attack types.

Logistic Regression, while also considered an interpretable model, offers a different kind of transparency compared to Decision Trees. As a linear model for binary classification, its interpretability stems directly from its coefficients. Each coefficient quantifies the change in the log-odds of the target variable (e.g., being an “intrusion”) for a one-unit increase in the corresponding feature, assuming all other features remain constant. This provides a clear, quantitative understanding of a feature’s influence. The direct relationship between features and the output (via coefficients) makes Logistic Regression highly transparent. Security analysts can easily see which features positively or negatively correlate with an intrusion classification and by how much. This simplicity allows for quick assessment of the model’s logic.

The core value of XAI, particularly techniques like SHAP and LIME, lies in their ability to mitigate the interpretability-accuracy trade-off. While Decision Trees and Logistic Regression are inherently interpretable, SHAP and LIME can still provide deeper, more nuanced explanations, especially for feature interactions or for understanding specific edge cases. More critically, for highly accurate “black-box” models SHAP and LIME become indispensable tools for injecting transparency without sacrificing the superior detection capabilities. By providing detailed local and global explanations, SHAP and LIME enable security professionals to have trust in the results and act fast accordingly. The goal is not to eliminate the interpretability-accuracy trade-off but to find an optimal balance that meets the specific requirements of the IoT IDS. For scenarios demanding extreme accuracy at all costs, highly complex models might be chosen, with XAI techniques then employed to provide post-hoc explanations. For resource-constrained environments or applications where immediate human comprehension is paramount, inherently interpretable models like Decision Trees and Logistic Regression, further enhanced by XAI for deeper insights, might be preferred.

7. Discussion and recommendations

This paper has demonstrated the effectiveness of various machine learning models in detecting cyberattacks and classifying network traffic through the use of comprehensive evaluation metrics, including accuracy, precision, recall, and the F1-score. While accuracy provides a general indication of overall model correctness, it becomes less reliable in imbalanced datasets, which are common in cybersecurity scenarios where benign traffic vastly outnumbers attack instances. In such contexts, relying on accuracy alone can result in misleading conclusions, as a model could achieve high accuracy by consistently predicting the majority class. Therefore, precision and recall become more critical. Precision assesses the proportion of correctly identified attack samples among all those predicted as attacks, while recall evaluates the model's ability to detect all actual attack instances. The F1-score harmonises precision and recall is valuable in IDS design where both false positives and false negatives can have serious operational and security consequences.

Beyond numerical performance metrics, we also place strong emphasis on model interpretability, recognising it as a cornerstone of trust and usability in cybersecurity systems. The interpretability techniques SHAP and LIME were employed to dissect the reasoning behind individual predictions. SHAP provides a consistent, theoretically grounded method to assess the global and local importance of features, while LIME offers localised approximations that help interpret individual predictions. These methods are particularly useful in security domains, where transparency and accountability are paramount, not only for improving system reliability but also for ensuring that analysts and end-users can make informed decisions based on model outputs. The interpretable models bolster user trust, facilitate system audits, and support model refinement by illuminating how features contribute to decision-making [12,40]. Although SHAP and LIME are typically used to explain predictions at the individual sample level, their outputs provide clear insights into both the direction and magnitude of feature influence. This ability to highlight key features contributing to a model's decision helps demystify the so-called black box nature of machine learning models. When applied effectively, these tools can substantially enhance the transparency of IDS decision processes and improve stakeholder confidence in automated detection systems.

Based on the insights derived from this study's results, several key recommendations are proposed to address current limitations and guide future improvements: First, improving model performance in multi-classification tasks is essential. This study found that decision tree and logistic regression models, while performing adequately in binary classification, struggle to accurately distinguish between multiple attack types, especially when those types exhibit overlapping features. To mitigate this, the implementation of more advanced feature engineering techniques and the adoption of complex or ensemble models are recommended. Enhancing the model's ability to extract and discriminate between subtle patterns will directly improve its multiclass classification capabilities. Notably, the multi-task, multi-objective feature selection algorithm exemplifies a promising approach [41]. It enables the identification of shared and task-specific relevant features, which significantly improves classification performance in complex datasets. Second, the interpretability and transparency of IDS must be further strengthened. While this study has demonstrated the usefulness of SHAP and LIME, wider integration of these interpretability tools into IDS platforms is encouraged. Doing so would provide consistent, user-accessible explanations for model decisions, facilitating not only technical audits but also increasing trust among end-users, including those without deep machine learning expertise. SHAP is widely regarded as the gold standard for local interpretability, offering rigorous mathematical underpinnings and scalability across models. In contrast, LIME's model-agnostic nature allows it to be applied across a broad range of machine learning algorithms. Building on these foundations, Mosca et al. [42] and Visani et al. [43] have proposed further optimisations to SHAP and LIME methodologies that enhance the clarity and utility of model explanations. Their work supports the proposition that explanatory tools are not optional, but essential components of effective, trustworthy IDS solutions.

These two recommendations – enhancing model performance and increasing interpretability – are not isolated concerns. Together, they ensure that IDS solutions are not only technically sound and accurate but also actionable and transparent in real-world cybersecurity environments. At the micro level, improved interpretability empowers cybersecurity teams operating in IoT environments to better understand, validate, and respond to potential threats. Greater transparency also enables non-technical users to build confidence in system outputs and manage cybersecurity risks more effectively. At the macro level, these improvements provide a competitive advantage for the IoT security market, where regulatory compliance, user trust, and explainability are increasingly critical differentiators. Transparent and explainable AI systems are also more likely to meet emerging compliance standards related to data governance and algorithmic accountability.

These recommendations are grounded in the study's empirical findings. For instance, experiments conducted using the CI-CIDS2017 dataset revealed that while binary classification tasks could be handled effectively by both decision tree and logistic regression models, these same models struggled with multi-class problems. This highlights the importance of advanced model selection and feature refinement in accurately differentiating between similar types of attacks. Furthermore, SHAP and LIME explanations provided valuable insights at the instance level, confirming their effectiveness in addressing the interpretability gap. Together, these findings reinforce the need to not only optimise model accuracy but also to ensure that model decisions are explainable and justifiable in the context of real-world intrusion detection systems.

8. Limitations and suggestions

One of the most significant limitations lies in data availability. The study relies on publicly accessible, secondary datasets, such as CICIDS2017 and UNSW-NB15—which, although widely adopted in academic research, do not fully capture the complexity, diversity, and dynamism of real-world network environments. These datasets are often curated and preprocessed under controlled conditions, lacking the unpredictability and noise of live network traffic. As a result, models trained and evaluated on such datasets may perform well in experimental settings but struggle to maintain accuracy or interpretability when deployed in operational networks. This limitation affects the external validity of the study and raises concerns about the practical applicability of the results across varying organisational or infrastructure contexts. Another key constraint is the limited sample size used in the experimental evaluation. A small dataset restricts the model's exposure to the full spectrum of potential attack behaviours and benign variations, particularly in heterogeneous or large-scale networks such as those found in IoT ecosystems. This limitation may undermine the reliability and robustness of the findings, as certain rare or emerging attack types may remain unrepresented in the training or testing phases. Consequently, the models may fail to generalise to new or unseen scenarios, diminishing their value in real-world deployments.

To overcome these limitations, future research should focus on enhancing data diversity and volume. This can be achieved through the collection of larger, real-time network traffic datasets from varied operational environments or through the generation of synthetic datasets that simulate a wide range of traffic patterns and attack vectors. Synthetic data, especially when generated using adversarial simulation or generative models, can help fill in the gaps left by publicly available datasets and provide more comprehensive coverage of the attack landscape.

Additionally, increasing the scale and diversity of experimental samples will be crucial. Repeating the analysis across different network environments – such as cloud-based infrastructures, industrial control systems, or smart home networks – can help verify the generalisation capability of the proposed methods. Incorporating longitudinal studies that examine how model performance and interpretability evolve over time in response to changes in network traffic can also offer deeper insights into the temporal robustness of interpretive AI tools in IDS.

9. Conclusion

This paper has analysed the application of Machine Learning, Intrusion Detection Systems, and Explainable Artificial Intelligence in the context of network security. By evaluating key models and interpretability tools, the paper highlights both the strengths and limitations of current approaches. Notably, SHAP and LIME proved effective in enhancing the transparency of ML-based IDS by offering valuable insights into the reasoning behind individual model predictions. These tools contribute significantly to addressing the black box problem by making complex model decisions more interpretable and actionable for security professionals. In terms of predictive performance, the study found that while binary classification tasks are generally well-handled by models such as decision trees and logistic regression, performance degrades in multi-class classification, especially under limited data conditions or when attack types share overlapping features. Additionally, the study successfully integrated visualisation techniques to communicate model decisions and data patterns in a clear, user-friendly manner. This is essential for enhancing interpretability and user trust in real-world applications.

CRediT authorship contribution statement

Yifan Wang: Writing – review & editing, Writing – original draft, Visualization, Validation, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Muhammad Ajmal Azad:** Writing – review & editing, Writing – original draft, Validation, Supervision, Resources, Project administration. **Maham Zafar:** Writing – review & editing, Writing – original draft, Investigation, Formal analysis. **Ammara Gul:** Writing – review & editing, Visualization, Methodology, Formal analysis.

Declaration of competing interest

We have no conflict of interest to declare.

Data availability

The authors do not have permission to share data.

References

[1] S. Sinha, State of iot 2024: Number of connected iot devices growing 13, 2025, Accessed 10 July 2025.

[2] S. Neupane, et al., Explainable intrusion detection systems (x-ids): A survey of current methods, challenges, and opportunities, IEEE Access 10 (2022) 112392–112415.

[3] G. Rjoub, J. Bentahar, O Abdel Wahab, R. Mizouni, A. Song, R. Cohen, H. Otrok, A. Mourad, A survey on explainable artificial intelligence for cybersecurity, IEEE Trans. Netw. Serv. Manag. 20 (4) (2023) 5115–5140.

[4] A. Barredo Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. Garcia, S. Gil-Lopez, D. Molina, R. Benjamins, R. Chatila, F. Herrera, Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai, Inf. Fusion 58 (2020) 82–115.

[5] S. Mohseni, N. Zarei, E.D. Ragan, A multidisciplinary survey and framework for design and evaluation of explainable ai systems, ACM Trans. Interact. Intell. Syst. 11 (2021) 3–4.

[6] W. Fei, H. Ohno, S. Sampalli, A systematic review of iot security: Research potential, challenges, and future directions, ACM Comput. Surv. 56 (5) (2023) 1–40.

[7] S. Arisdakessian, et al., A survey on iot intrusion detection: Federated learning, game theory, social psychology, and explainable ai as future directions, IEEE Internet Things J. 10 (5) (2023) 4059–4092.

[8] O. Chakir, Y. Sadqi, E.A.A. Alaoui, An explainable machine learning-based web attack detection system for industrial iot web application security, Inf. Secur. J.: A Glob. Perspect. (2024) 1–27.

[9] C. Kumar, M.S. Ansari, An explainable nature-inspired cyber attack detection system in software-defined iot applications, Expert Syst. Appl. 250 (2024) 123853.

[10] R. Kalakoti, S. Nõmm, H. Bahsi, Enhancing iot botnet attack detection in socs with an explainable active learning framework, in: Proc. 2024 IEEE World AI IoT Congress (AIIoT), 2024.

[11] J.V. Rani, H.A.S. Ali, A. Jakka, Iot network intrusion detection: An explainable ai approach in cybersecurity, in: Proc. 2023 4th Int. Conf. on Communication, Computing and Industry 6.0 (C2I6), 2023.

[12] L.A. Ahakonye, et al., Machine learning explainability for intrusion detection in the industrial internet of things, IEEE Internet Things Mag. 7 (3) (2024) 68–74.

[13] R.K. Sambandam, et al., Comparison of machine learning-based intrusion detection systems using unsw-nb15 dataset, Lecture Notes in Networks and Systems, 2024, pp. 311–324.

[14] S. Pansare, A. Malik, I. Batra, Comparative analysis of machine learning based intrusion detection systems, in: Proc. 2023 7th Int. Conf. on Image Information Processing, ICIIP, 2023.

[15] M.J. Rani, D. Singh, Machine learning algorithm for intrusion detection: Performance evaluation and comparative analysis, in: Proc. 2023 7th Int. Conf. on I-SMAC, 2023.

[16] H. Sayadi, et al., Redefining trust: Assessing reliability of machine learning algorithms in intrusion detection systems, in: Proc. 2024 IEEE Int. Symp. on Circuits and Systems, ISCAS, 2024.

[17] C. Jiang, J. Kuang, S. Wang, Home iot intrusion prevention strategy based on edge computing, in: Proc. 2019 IEEE 2nd Int. Conf. Electronics and Communication Engineering, ICECE, 2019.

[18] R.K. Vigneswaran, et al., Evaluating shallow and deep neural networks for network intrusion detection systems in cyber security, in: Proc. 2018 9th Int. Conf. on Computing, Communication and Networking Technologies, ICCCNT, 2018.

[19] A.A. Malibari, et al., A novel metaheuristics with deep learning enabled intrusion detection system for secured smart environment, Sustain. Energy Technol. Assess. 52 (2022) 102312.

[20] B. Xue, H. Zhao, W. Yao, Deep transfer learning for iot intrusion detection, in: Proc. 2022 3rd Int. Conf. on Computing, Networks and Internet of Things, CNIOT, 2022.

[21] S.T. Mehedi, et al., Dependable intrusion detection system for iot: A deep transfer learning based approach, IEEE Trans. Ind. Inform. 19 (1) (2023) 1006–1017.

[22] S. Saif, et al., Hiids: Hybrid intelligent intrusion detection system empowered with machine learning and metaheuristic algorithms for application in iot based healthcare, Microprocess. Microsyst. (2022) 104622.

[23] T. Hickling, et al., Explainability in deep reinforcement learning: A review into current methods and applications, ACM Comput. Surv. 56 (5) (2023) 1–35.

[24] S. Neupane, J. Ables, W. Anderson, S. Mittal, S. Rahimi, I. Banicescu, M. Seale, Explainable intrusion detection systems (x-ids): A survey of current methods, challenges, and opportunities, IEEE Access 10 (2022) 112392–112415.

[25] A. Nascita, G. Aceto, D. Ciuonzo, A. Montieri, V. Persico, A. Pescapé, A survey on explainable artificial intelligence for internet traffic classification and prediction, and intrusion detection, IEEE Commun. Surv. Tutor. (2024) 1–1.

[26] S.K. Jagatheesaperumal, et al., Explainable ai over the internet of things (iot): Overview, state-of-the-art and future directions, IEEE Open J. Commun. Soc. 3 (2022) 2106–2136.

[27] İ. Kök, et al., Explainable artificial intelligence (xai) for internet of things: A survey, IEEE Internet Things J. 10 (16) (2023) 14764–14779.

[28] M. Danilevsky, K. Qian, R. Aharonov, Y. Katsis, B. Kawas, P. Sen, A survey of the state of explainable AI for natural language processing, in: K.-F. Wong, K. Knight, H. Wu (Eds.), Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing (Suzhou, China, Dec. 2020), Association for Computational Linguistics, pp. 447–459.

[29] E. Tjoa, C. Guan, A survey on explainable artificial intelligence (xai): Toward medical xai, IEEE Trans. Neural Netw. Learn. Syst. 32 (11) (2021) 4793–4813.

[30] R. Machlev, L. Heistrene, M. Perl, K. Levy, J. Belikov, S. Mannor, Y. Levron, Explainable artificial intelligence (xai) techniques for energy and power systems: Review, challenges and opportunities, Energy AI 9 (2022) 100169.

[31] M. Siganos, et al., Explainable ai-based intrusion detection in the internet of things, in: Proc. 18th Int. Conf. on Availability, Reliability and Security, 2023.

[32] A. Lavrenovs, R. Graf, Explainable ai for classifying devices on the internet, in: Proc. 2021 13th Int. Conf. on Cyber Conflict (CyCon), 2021.

[33] Z.A. El Houda, B. Brik, S.-M. Senouci, A novel iot-based explainable deep learning framework for intrusion detection systems, IEEE Internet Things Mag. 5 (2) (2022) 20–23.

[34] Y. Djenouri, et al., When explainable ai meets iot applications for supervised learning, Clust. Comput. 26 (4) (2022) 2313–2323.

[35] S. Anjomshoae, K. Främling, A. Najjar, Explanations of black-box model predictions by contextual importance and utility, Lecture Notes in Computer Science, 2019, pp. 95–109.

[36] M. Keshk, N. Koroniotis, N. Pham, N. Moustafa, B. Turnbull, A.Y. Zomaya, An explainable deep learning-enabled intrusion detection framework in iot networks, Inform. Sci. 639 (2023) 119000.

[37] M. Zolanvari, Z. Yang, K. Khan, R. Jain, N. Meskin, Trust xai: Model-agnostic explanations for ai with a case study on iiot security, IEEE Internet Things J. 10 (4) (2023) 2967–2978.

[38] A. Ghubaish, Z. Yang, R. Jain, Hdrl-ids: A hybrid deep reinforcement learning intrusion detection system for enhancing the security of medical applications in 5 g networks, in: 2024 International Conference on Smart Applications, Communications and Networking (SmartNets), 2024, pp. 1–6.

[39] A. Ghubaish, Z. Yang, A. Erbad, R. Jain, Lemda: A novel feature engineering method for intrusion detection in iot systems, IEEE Internet Things J. 11 (8) (2024) 13247–13256.

[40] F. Hassan, et al., Achieving model explainability for intrusion detection in vanets with lime, PeerJ Comput. Sci. 9 (2023).

[41] J. Lin, et al., Multi-task optimisation for multi-objective feature selection in classification, in: Proc. Genetic and Evolutionary Computation Conf. Companion, 2022.

[42] E. Mosca, F. Szigeti, S. Tragianni, D. Gallagher, G. Groh, Shap-based explanation methods: A review for nlp interpretability, in: Proc. 29th Int. Conf. on Computational Linguistics, 2022, pp. 4593–4603.

[43] G. Visani, E. Bagli, F. Chesani, Optilime: Optimized lime explanations for diagnostic computer algorithms, 2020, arXiv preprint arXiv:2006.05714.