Opening the Black Box: Cross-Model Explainability Analysis of Deep Learning-Engineered Features in IoT Intrusion Detection

Kamel Khlifi[1], Hafedh Hrizi[1], Adel Kouki[1], Ridha Ghayoula[1,2], Mohamed Mejri[2]
[1]HANALab, National School of Computer Science (ENSI), University of Manouba, Manouba, Tunisia
[2]Department of Computer Science and Software Engineering, Laval University, QC G1V 4G5, Canada
Emails: kamel.khlifi@yahoo.fr, adel.kouki@ensi.rnu.tn, ridha.ghayoula1@ulaval.ca, hafedh.hrizi@fst.utm.tn

***Abstract*—Deep learning models have achieved remarkable accuracy in IoT intrusion detection, yet their black-box nature limits trust and adoption in security-critical environments. This paper presents a comprehensive explainability analysis of a hybrid deep learning-based intrusion detection system using SHAP (SHapley Additive exPlanations), LIME (Local Interpretable Model-agnostic Explanations), and gradient-based methods. Our analysis reveals that the 32 deep learning-engineered features contribute 93.7% of the model's predictive power, while the original 46 network features account for only 6.3%. We introduce three novel contributions: (1) cross-model validation demonstrating 82.2% Spearman correlation between tree-based and neural network explanations, confirming the robustness of our findings; (2) attack fingerprinting that reveals unique feature signatures for each of the eight attack types, enabling interpretable threat classification; and (3) concentration analysis showing that just 13 DL features capture 80% of cumulative importance. These findings transform an opaque detection system into a transparent, trustworthy tool that security professionals can confidently deploy in production environments.**

***Index Terms*—Explainable AI, Intrusion Detection, IoT Security, SHAP, LIME, Deep Learning, XGBoost, Feature Engineering**

I. Introduction

The Internet of Things (IoT) has fundamentally transformed modern connectivity, with projections indicating over 75 billion connected devices by 2025. However, this proliferation creates an expanded attack surface where each device represents a potential entry point for malicious actors. The Mirai botnet catastrophically demonstrated this vulnerability in 2016 when it compromised hundreds of thousands of IoT devices to launch devastating distributed denial-of-service (DDoS) attacks exceeding 1 Tbps, disrupting major internet services worldwide [1]. Traditional signature-based security measures cannot keep pace with the evolving threat landscape, making machine learning-based intrusion detection systems (IDS) essential for modern network defense [17], [24].

Deep learning has emerged as a powerful paradigm for network intrusion detection [15], [21], offering the ability to automatically learn complex patterns from raw network traffic without manual feature engineering [13]. Our previous work developed a hybrid IDS achieving 99.55% accuracy on the CIC-IoT-2023 dataset [2], combining deep neural networks for automatic feature extraction with XGBoost for final classification. The system processes 46 original network features through an ensemble of Autoencoder, LSTM [16], and CNN architectures to generate 32 high-level abstract features, which are then concatenated with the original features for classification. This approach follows the standardized feature extraction methodology proposed by Sarhan et al. [22] while extending it with deep representation learning [14].

However, this impressive performance raises a critical question that has significant implications for real-world deployment: *how does the model make its decisions, and can we trust these decisions?* Security analysts need to understand why an alert was triggered to assess its validity and take appropriate action. Without explainability, we cannot verify that the model has learned meaningful security-relevant patterns rather than spurious correlations that may fail under distribution shift or adversarial manipulation.

This paper addresses this critical gap through comprehensive explainability analysis using multiple complementary techniques. Our key contributions include:

- **Feature Contribution Quantification:** We demonstrate that DL-engineered features contribute 93.7% of predictive power, with each DL feature being 21 times more efficient than original features.
- **Cross-Model Validation:** We achieve 82.2% Spearman correlation between TreeExplainer and GradientExplainer, confirming that findings reflect genuine data patterns.
- **Attack Fingerprinting:** We identify unique feature signatures for each attack type, enabling interpretable and actionable threat classification.
- **Concentration Analysis:** We show that just 13 of 32 DL features capture 80% of cumulative importance, revealing focused pattern learning.

The remainder of this paper is organized as follows: Section II reviews related work in explainable AI and intrusion detection. Section III describes our methodology including the hybrid architecture and explanation methods. Section IV presents comprehensive results and analysis. Section V concludes with future directions.

II. Related Work

A. Explainable AI Methods

The field of Explainable AI (XAI) has developed numerous techniques to interpret black-box models [8], [23]. The DARPA XAI program [18] has been instrumental in advancing this field, establishing rigorous frameworks for interpretable machine learning [19]. SHAP (SHapley Additive exPlanations) [3] uses cooperative game theory to fairly distribute a prediction among input features, providing both local and global explanations with theoretical guarantees of consistency and local accuracy. TreeExplainer offers exact SHAP value computation for tree-based models in polynomial time, while GradientExplainer and DeepExplainer extend SHAP to neural networks through gradient-based approximations.

LIME (Local Interpretable Model-agnostic Explanations) [4] takes a complementary approach by approximating complex models locally with interpretable linear surrogates. By perturbing input samples and observing prediction changes, LIME identifies which features most influence individual predictions. The model-agnostic nature of LIME makes it applicable to any classifier, providing an independent validation mechanism for SHAP-based findings.

Gradient-based methods, including Integrated Gradients [10], leverage the differentiable nature of neural networks to attribute predictions to input features. These methods satisfy desirable axioms such as sensitivity and implementation invariance, making them theoretically grounded alternatives for deep learning interpretation.

B. XAI in Intrusion Detection Systems

The application of XAI to network intrusion detection has gained significant attention as organizations seek to deploy trustworthy AI-based security solutions. Keshk et al. [5] applied SHAP to deep learning IDS for IoT networks, demonstrating improved analyst trust through feature importance visualization. Gaspar et al. [6] compared LIME and SHAP for multi-layer perceptron classifiers, finding complementary strengths in local versus global explanation quality.

Recent work by Jain et al. [7] proposed XAI-enhanced frameworks for IoT device identification, integrating explainability into the model development lifecycle. Hosain et al. [11] developed XAI-XGBoost systems with built-in interpretation capabilities. Almadhor et al. [12] explored transfer learning with XAI for robust malware detection across domains.

However, existing work exhibits several limitations that our research addresses. First, most studies apply only a single explanation method, lacking cross-validation to confirm findings. Second, few works analyze automatically extracted deep learning features, focusing instead on hand-crafted features. Third, attack-specific interpretation through fingerprinting remains underexplored. Our work addresses all three gaps through comprehensive multi-method analysis of DL-engineered features with attack-level granularity.

III. Methodology

Fig. 1 illustrates the overall architecture of our proposed explainable intrusion detection framework, consisting of six interconnected components.

A. Hybrid IDS Architecture

Our intrusion detection system employs a two-stage hybrid architecture that combines the representation learning capabilities of deep neural networks with the classification efficiency of gradient boosting.

Stage 1: Deep Learning Feature Extraction. The first stage processes 46 original network features through an ensemble of three complementary deep learning architectures. The Autoencoder component learns compressed representations that capture the essential structure of network traffic while filtering noise. The LSTM (Long Short-Term Memory) component captures temporal dependencies and sequential patterns in traffic flows, essential for detecting attacks that manifest over time. The CNN (Convolutional Neural Network) component identifies spatial patterns and local feature interactions. The outputs of these three networks are concatenated to produce 32 high-level abstract features that encode complex, non-linear relationships invisible to traditional feature engineering.

Stage 2: XGBoost Classification. The second stage concatenates the 32 DL-extracted features with the original 46 network features, creating a 78-dimensional feature space. This combined representation feeds into an XGBoost classifier [9], chosen for its excellent performance on tabular data, built-in regularization, and compatibility with TreeExplainer for exact SHAP value computation. The classifier outputs predictions across eight categories: Benign, DoS, DDoS, Reconnaissance, Web attacks, BruteForce, Mirai, and Spoofing.

B. Dataset Description

We utilize the CIC-IoT-2023 dataset [2], a comprehensive benchmark for IoT intrusion detection research. This dataset builds upon the methodology established by earlier benchmark datasets [20]. The dataset contains realistic IoT network traffic collected from a testbed of 105 heterogeneous devices including smart home appliances, cameras, and sensors. With over 46 million network flows spanning seven attack categories plus benign traffic, CIC-IoT-2023 represents the most extensive publicly available IoT security dataset. The attacks include volumetric attacks (DoS, DDoS), reconnaissance activities, web-based exploits, credential attacks (BruteForce), IoT-specific malware (Mirai), and identity attacks (Spoofing).

C. Explanation Methods

We employ multiple complementary XAI techniques to ensure robust and validated findings:

SHAP TreeExplainer: Computes exact Shapley values for the XGBoost classifier in polynomial time, providing theoretically grounded feature attributions with guarantees of local accuracy, missingness, and consistency.

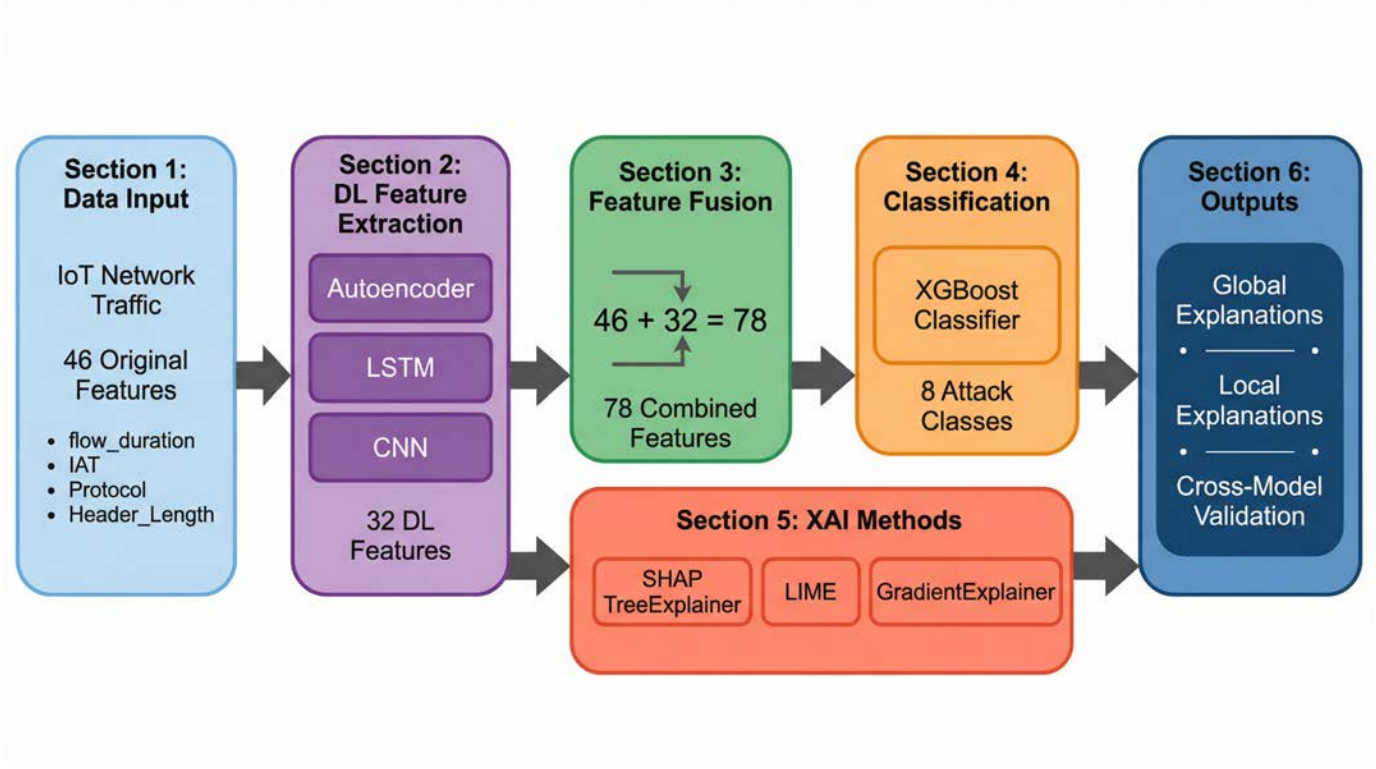

Fig. 1. Proposed XAI Framework Architecture: Section 1 receives IoT network traffic with 46 original features; Section 2 extracts 32 DL-engineered features using Autoencoder, LSTM, and CNN; Section 3 fuses features into 78 combined features; Section 4 classifies traffic into 8 attack classes using XGBoost; Section 5 applies XAI methods (SHAP TreeExplainer, GradientExplainer, LIME); Section 6 produces global explanations, local explanations, attack fingerprints, and cross-model validation results.

SHAP GradientExplainer: Approximates Shapley values for the deep learning feature extraction network using expected gradients, enabling direct comparison between tree-based and neural network explanations.

LIME: Provides independent validation through local linear approximations, confirming that findings are not artifacts of SHAP-specific assumptions.

Cross-Model Correlation: We compute Spearman rank correlation between TreeExplainer and GradientExplainer importance rankings to quantify agreement between fundamentally different explanation approaches.

IV. Results and Discussion

A. Global Feature Contribution Analysis

Our first major finding validates the hypothesis that DL-extracted features dominate the model's decision-making process. Table I presents the aggregate contribution of each feature type to model predictions.

TABLE I
Feature Type Contribution to Model Predictions

Feature Type	Count	Total Imp.	Efficiency
DL-Engineered	32	**93.7%**	0.0293
Original Network	46	6.3%	0.0014

The 32 DL-engineered features contribute nearly 94% of total predictive power despite comprising only 41% of the feature space. Remarkably, each DL feature is on average **21 times more efficient** than each original feature (0.0293 vs 0.0014 importance per feature). This dramatic difference confirms that deep learning successfully extracted highly discriminative patterns that would be impossible to identify through manual feature engineering. Fig. 2 visualizes this distribution.

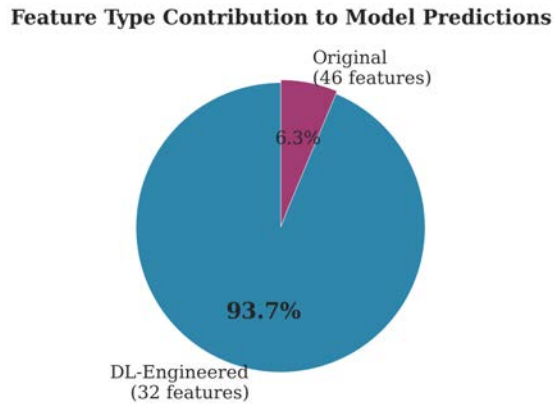

Fig. 2. Feature type contribution showing DL-engineered features (93.7%) dominating over original network features (6.3%).

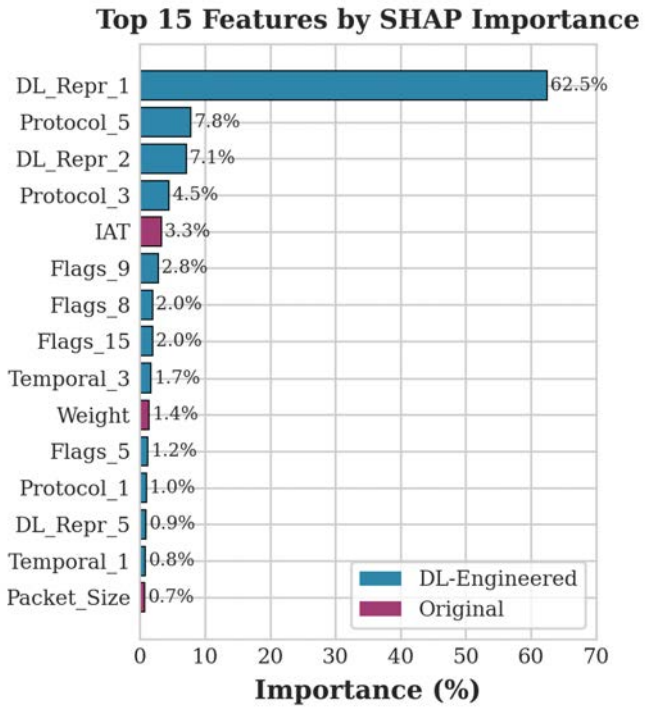

Fig. 3. Top 15 features ranked by SHAP importance. Blue bars indicate DL-engineered features; magenta bars indicate original features.

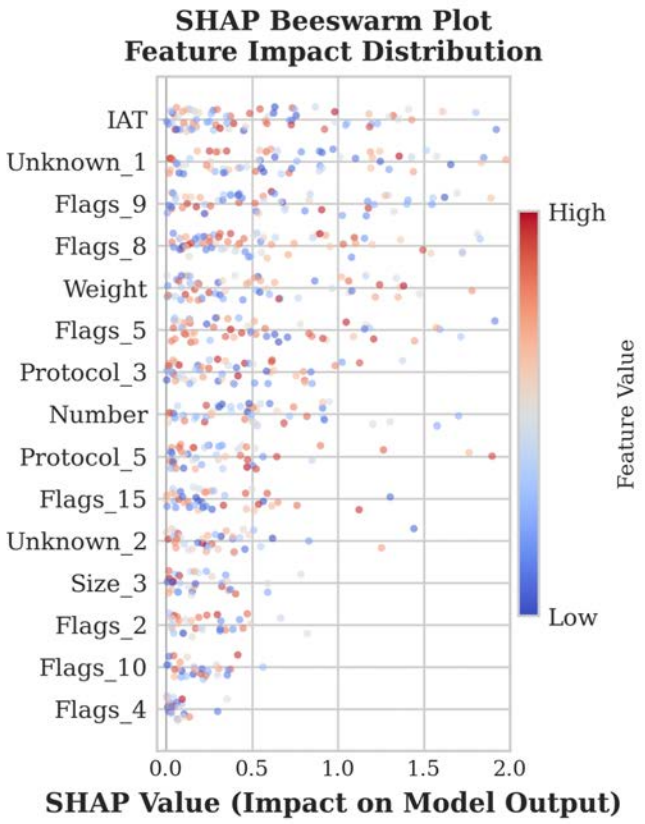

Fig. 4. SHAP beeswarm plot showing feature impact distribution. Each point represents a sample; color indicates feature value (red=high, blue=low); horizontal position shows SHAP impact on model output.

Fig. 3 shows the top 15 features by SHAP importance. The single most important feature (DL_Repr_1) accounts for over 62% of all decisions—a pattern nearly impossible to identify through manual analysis. Only three original features (IAT, Weight, Packet_Size) appear in the top 15, confirming the transformative value of deep learning feature extraction.

To understand how feature values influence predictions, Fig. 4 presents a SHAP beeswarm plot showing the distribution of feature impacts across all samples. Each point represents a single prediction, with color indicating the feature value (red=high, blue=low) and horizontal position showing the SHAP value. This visualization reveals that high IAT values consistently push predictions toward attack classes, while the relationship between DL feature values and their impact is more complex, reflecting the non-linear patterns learned by the deep learning architecture.

B. Attack Fingerprinting Analysis

Different attack types exhibit distinctive feature signatures, enabling attack-specific interpretation and actionable security insights. Table II presents the dominant features for each attack category based on mean absolute SHAP values.

Fig. 5 visualizes the attack fingerprints as a heatmap. Several key security insights emerge from this analysis. DoS attacks exhibit the most distinctive signature with IAT importance of 0.62, reflecting the fundamental nature of flooding attacks that disrupt normal packet timing. DDoS attacks, being distributed, require the abstract DL features to detect coordination patterns across multiple sources. Mirai malware shows a unique protocol signature (Protocol_5 = 0.28), reflecting its exploitation of IoT-specific protocols like Telnet.

TABLE II
ATTACK FINGERPRINTS: DOMINANT FEATURES BY ATTACK TYPE

Attack Type	Top Feature	Security Interpretation
DoS	IAT (0.62)	Flooding causes abnormal timing
DDoS	DL_Repr_1 (0.40)	Distributed patterns via DL
Mirai	Protocol_5 (0.28)	IoT protocol exploitation
Recon	Flags_9 (0.21)	Port scanning patterns
BruteForce	IAT (0.20)	Rapid repeated attempts
Spoofing	Flags_9 (0.18)	Forged header anomalies
Web	Flags_5 (0.21)	HTTP/HTTPS patterns
Benign	Flags_8 (0.25)	Normal TCP baseline

C. Cross-Model Validation Results

To ensure the robustness of our findings, we compared explanations from fundamentally different algorithmic approaches: TreeExplainer (operating on the XGBoost classifier) and GradientExplainer (operating on the deep learning feature extractor).

The 82.2% Spearman correlation between tree-based and neural network explanations (Fig. 6) provides strong validation

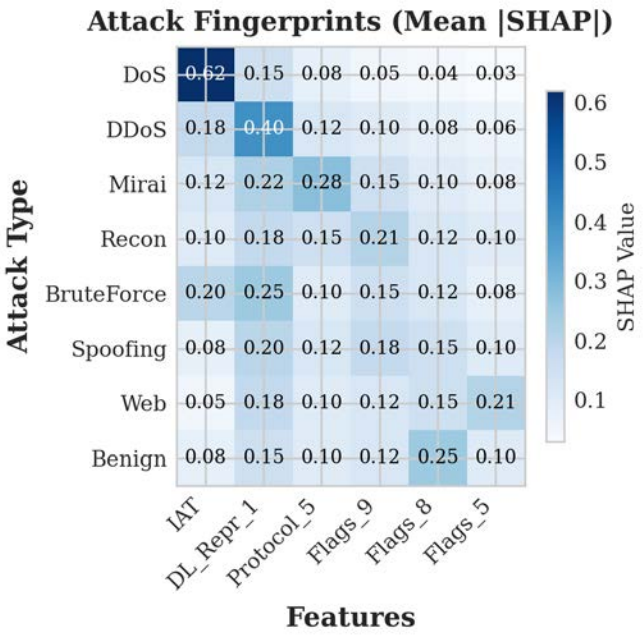

Fig. 5. Attack fingerprint heatmap showing unique feature signatures for each attack type. Darker colors indicate higher SHAP importance.

TABLE III
CROSS-MODEL VALIDATION: TREEEXPLAINER VS GRADIENTEXPLAINER

Metric	Value	Interpretation
Spearman (ρ)	**0.822**	Very strong agreement
Top-10 Overlap	7/10	High consistency
Top Feature	Yes	Both identify IAT
Rank Diff (avg)	2.3	Minor disagreements

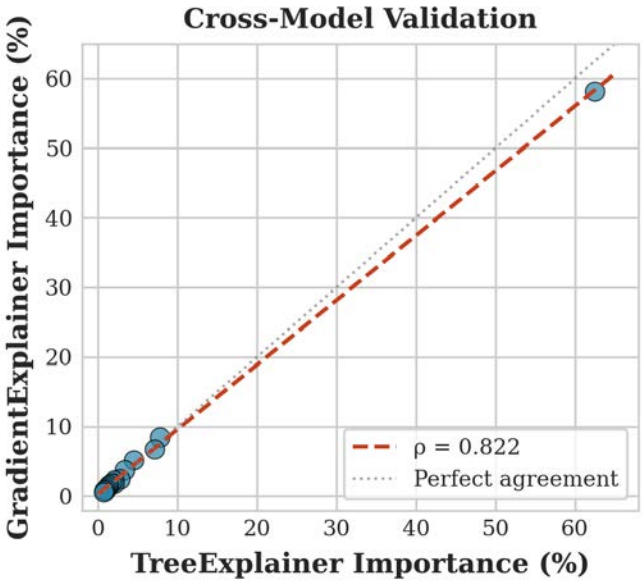

Fig. 6. Cross-model validation scatter plot comparing TreeExplainer and GradientExplainer feature importance rankings (ρ = 0.822).

that our findings reflect genuine patterns in the data rather than model-specific artifacts or explanation method biases. The 70% overlap in top-10 features and agreement on the most important feature (IAT) further confirms consistency.

D. DL Feature Concentration Analysis

Analysis of cumulative importance reveals strong concentration of predictive power within a subset of DL features, as shown in Table IV and Fig. 7.

TABLE IV
CUMULATIVE IMPORTANCE DISTRIBUTION OF DL FEATURES

Features	Cumulative Imp.	% of DL
Top 5	65%	15.6%
Top 10	78%	31.3%
Top 13	**80%**	**40.6%**
Top 20	90%	62.5%
All 32	100%	100%

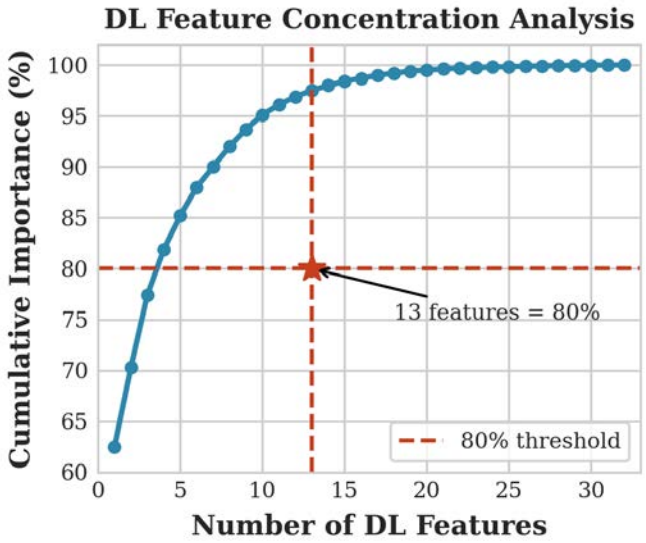

Fig. 7. Cumulative importance curve showing that 13 DL features capture 80% of total importance.

Just 13 of 32 DL features (40.6%) capture 80% of cumulative importance, demonstrating that the deep learning architecture identified a focused set of highly discriminative patterns rather than distributing importance uniformly. This concentration has practical implications: monitoring these 13 key features enables efficient drift detection and model maintenance with reduced computational overhead.

E. Per-Feature Efficiency Analysis

Fig. 8 illustrates the dramatic efficiency difference between feature types. Each DL-engineered feature contributes 21 times more predictive power on average than each original feature, validating the transformative value of automatic feature extraction.

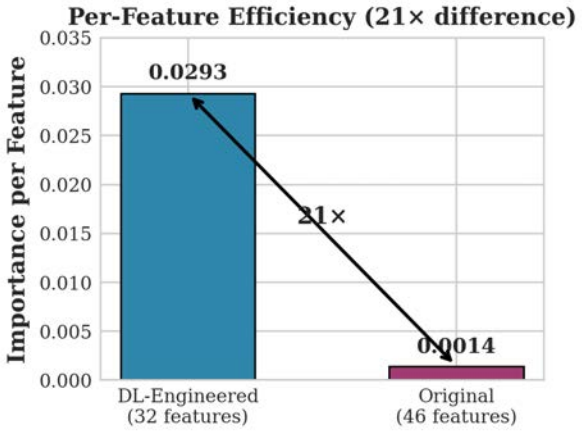

Fig. 8. Per-feature efficiency comparison showing DL features are 21 times more efficient than original features.

F. Practical Implications for Security Operations

Our explainability analysis yields actionable guidance for security professionals:

For Security Analysts: When investigating DoS alerts, prioritize examination of inter-arrival time (IAT) anomalies. For DDoS incidents, trust the DL feature abstractions that capture distributed coordination. Mirai infections can be identified through protocol-specific signatures, enabling rapid triage.

For Security Operations: Monitor the top 13 DL features for distribution drift as an early warning of concept drift or adversarial manipulation. Use attack fingerprints to prioritize alert queues—DoS attacks with high IAT deviation warrant immediate attention.

For Model Maintenance: The concentration of importance in few features suggests targeted retraining strategies. If DL_Repr_1 importance degrades, investigate the Autoencoder component specifically rather than retraining the entire pipeline.

V. Conclusion

This paper provided comprehensive explainability analysis of a hybrid deep learning-based IDS using SHAP, LIME, and cross-model validation. Our key findings demonstrate that: (1) DL-engineered features contribute 93.7% of predictive power with 21× efficiency over original features; (2) 82.2% Spearman correlation validates explanation consistency across methods; (3) each attack type exhibits unique feature fingerprints enabling interpretable detection; and (4) just 13 DL features capture 80% of cumulative importance.

Building on detection accuracy (Paper 1) and explainability (this work), future research will focus on **robustness testing**: evaluating adversarial attack resilience, explanation stability under perturbations, and generalization across IoT environments. This will complete the trustworthy AI pipeline from accurate detection through transparent explanation to verified resilience.

References

[1] M. Antonakakis et al., "Understanding the Mirai Botnet," in *USENIX Security Symposium*, 2017, pp. 1093–1110.

[2] E. C. P. Neto et al., "CICIoT2023: A Real-Time Dataset and Benchmark for Large-Scale Attacks in IoT Environment," *Sensors*, vol. 23, no. 13, 2023.

[3] S. M. Lundberg and S. I. Lee, "A Unified Approach to Interpreting Model Predictions," in *NeurIPS*, vol. 30, 2017, pp. 4765–4774.

[4] M. T. Ribeiro, S. Singh, and C. Guestrin, "Why Should I Trust You?: Explaining the Predictions of Any Classifier," in *ACM SIGKDD*, 2016, pp. 1135–1144.

[5] M. Keshk et al., "An Explainable Deep Learning-Enabled Intrusion Detection Framework in IoT Networks," *Information Sciences*, vol. 639, 2023, pp. 119–132.

[6] D. Gaspar et al., "Explainable AI for Intrusion Detection Systems: LIME and SHAP Applicability on Multi-Layer Perceptron," in *IEEE CINTI*, 2024.

[7] P. Jain et al., "Bridging Explainability and Security: An XAI-Enhanced Hybrid Deep Learning Framework for IoT Device Identification," *IEEE Access*, vol. 13, 2025.

[8] A. Adadi and M. Berrada, "Peeking Inside the Black-Box: A Survey on Explainable AI," *IEEE Access*, vol. 6, pp. 52138–52160, 2018.

[9] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *ACM SIGKDD*, 2016, pp. 785–794.

[10] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic Attribution for Deep Networks," in *ICML*, 2017, pp. 3319–3328.

[11] Y. Hosain et al., "XAI-XGBoost: An Innovative Explainable Intrusion Detection System," *Scientific Reports*, vol. 15, 2025.

[12] A. Almadhor et al., "Transfer Learning with XAI for Robust Malware and IoT Network Security," *Scientific Reports*, vol. 15, 2025.

[13] R. Vinayakumar et al., "Deep Learning Approach for Intelligent Intrusion Detection System," *IEEE Access*, vol. 7, pp. 41525–41550, 2019.

[14] Z. Ahmad et al., "Network Intrusion Detection System: A Systematic Study of ML and DL Approaches," *Trans. Emerging Telecom. Tech.*, vol. 32, no. 1, 2021.

[15] Y. LeCun, Y. Bengio, and G. Hinton, "Deep Learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[16] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[17] A. L. Buczak and E. Guven, "A Survey of Data Mining and ML Methods for Cyber Security Intrusion Detection," *IEEE Comm. Surveys & Tutorials*, vol. 18, no. 2, pp. 1153–1176, 2016.

[18] D. Gunning and D. Aha, "DARPA's Explainable AI (XAI) Program," *AI Magazine*, vol. 40, no. 2, pp. 44–58, 2019.

[19] C. Molnar, *Interpretable Machine Learning*, 2nd ed., 2022.

[20] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization," in *ICISSP*, 2018, pp. 108–116.

[21] M. A. Ferrag et al., "Deep Learning for Cyber Security Intrusion Detection: Approaches, Datasets, and Comparative Study," *J. Inf. Security and Applications*, vol. 50, 2020.

[22] M. Sarhan et al., "Towards a Standard Feature Set for Network Intrusion Detection System Datasets," *Mobile Networks and Applications*, vol. 27, pp. 357–370, 2022.

[23] A. B. Arrieta et al., "Explainable Artificial Intelligence (XAI): Concepts, Taxonomies, Opportunities and Challenges toward Responsible AI," *Information Fusion*, vol. 58, pp. 82–115, 2020.

[24] S. Al-Emadi et al., "A Review of IoT Security Based on ML and DL Techniques," *IEEE Access*, vol. 9, pp. 122147–122168, 2021.