International Journal of Scientific Research & Engineering Trends
Volume 12, Issue 5, Sep-Oct-2026, ISSN (Online): 2395-566X

Explainable and Robust Deep Learning-Based Intrusion Detection for DoS Attacks

Gawade Ishwari Shyam[1], Prof. Bangar Abhishek P.[2], Prof. Karad Vandana A.[3]
[123]Department of Computer Engineering / Jaihind College of Engineering / SPPU, Pune, India

Abstract — Flooding-type attacks (DoS and DDoS) keep threatening network services and IoT infrastructure. Modern deeplearning intrusion detectors score above 99 % on public benchmarks, yet practitioners still face four obstacles: models cannot justify their alerts, rare attack classes are under-learned, tuning is costly, and small crafted perturbations can defeat the classifier. We examine 26 publications, of which five recent studies (2023–2026) form the core and 21 provide background, and group them into six themes: handling of imbalanced data, deep classifiers, optimisation, explanation techniques, efficient and trustworthy explanation, and adversarial resilience. Across the core studies, GAN-based balancing with attention networks reports 99.95–99.99 % accuracy on five datasets, but the gap to the best competing methods is a few tenths of a point at most. Selecting features by combining ANOVA and SHAP shrinks the input by roughly 70 % and shortens LIME run time by up to 87 %. Using SHAP to prune harmful inputs lowered missed detections by 26 % and 92 % on two DDoS sets. Recurrent detectors that reach about 96 % on clean data fall to 51–61 % under FGSM at ε = 0.1, and changes in SHAP attributions can serve as a partial warning signal (ROC-AUC 0.70–0.88). Evidence is mostly offline and limited to one dataset, the most accurate model was never attacked in testing, and no shared measure of explanation quality exists. We offer a thematic synthesis, a thirteen-study comparison, a conceptual seven-layer design and a prioritised agenda for future work.

Keywords— Intrusion Detection System; Denial of Service; Explainable AI; SHAP; LIME; Deep Learning; Class Imbalance; Metaheuristic Optimisation; Adversarial Robustness.

I. INTRODUCTION

An intrusion detection system (IDS) watches network or host activity and raises an alarm when behaviour looks malicious. Among the threats it must catch, DoS attacks stand out, because they overload bandwidth or protocol resources until genuine users are locked out; they may crash a service, flood it, or abuse a protocol to amplify traffic, as in DNS amplification [1], [5]. With IoT and smart-city networks adding huge numbers of cheap, varied devices, rule lists of known attack patterns can no longer keep up, since they cannot flag anything absent from their database [5]. Learning-based anomaly detectors fill this gap, and recurrent, convolutional and attention-based networks now lead the field [1], [4], [5].

Yet three difficulties persist. Attack records are normally far outnumbered by benign ones, so classifiers drift towards the majority class and overlook uncommon attacks [1]. Deep models also hide their reasoning, which frustrates analysts who must decide which alerts deserve action and complicates audit and compliance [2], [3]. Finally, good performance on clean test data says little about resistance to deliberate evasion: a detector near 96 % accuracy may fall to roughly half that after a single gradientbased perturbation step [4]. Explanation tools such as SHAP [6] and LIME [7] are widely attached to IDS to ease the second difficulty, but they bring their own issues: slow generation, disagreement between tools and, as shown in recent work, susceptibility to deliberate manipulation [2], [17].

A newer direction treats explanation not merely as a report but as a working instrument, used to choose features, speed up explanation, remove misleading inputs and flag tampered traffic [2], [3], [4]. In parallel, the best-scoring deep pipelines mix data augmentation, attention, multiscale residual networks and metaheuristic tuning with SHAP and LIME output [1]. No earlier review ties these strands together.

The goals of this review are to (i) summarise how deep learning, optimisation and explanation have been combined for DoS/DDoS classification, (ii) contrast the approaches, their reported results and weaknesses, (iii) propose a conceptual architecture grounded in the literature, and (iv) list the gaps to close before real deployment.

II. REVIEW METHODOLOGY

The review centres on five recent research papers [1]–[5]: a deep attention-based DoS classifier with SHAP and LIME [1]; a two-stage framework stressing fast, cross-checked explanations [2]; a SHAP-driven cycle for refining a DDoS detector [3]; a study of adversarial evasion and explanation drift in LSTM/GRU detectors [4]; and a framework mixing local and

© 2026 IJSRET

global explanation for IoT intrusion detection [5]. Twentyone further works were drawn from their bibliographies when they supplied a method, dataset or result the core studies depend on: foundations of SHAP, LIME, LSTM, FGSM, permutation importance and partial dependence [6]–[9], [23], [24]; earlier DL detectors of DoS and related attacks [10]–[15]; explanationbased adversarial detection and attacks on explainers [16]–[19]; surveys on XAI in IDS [20], [21], [25], [26]; and the ToN_IoT dataset [22].

Papers were kept if they concerned intrusion or DoS/DDoS detection with machine or deep learning and also dealt with tuning, explanation, explanation-based feature choice, or adversarial resistance. Every number quoted below is taken from the original authors and was not recomputed. Where an earlier study is known only through another paper's summary, the text says so. As no systematic database search was recorded, this is a narrative review. Fig. 1 outlines the organisation.

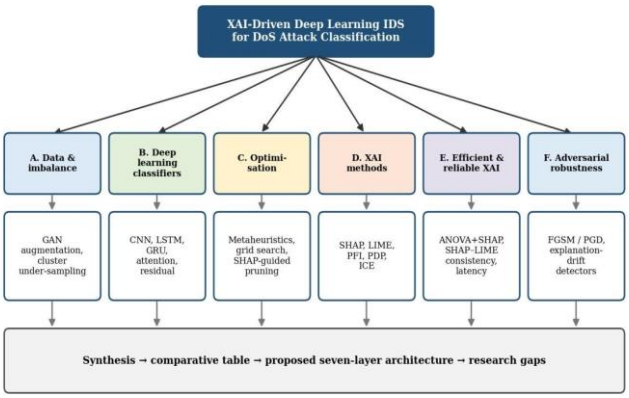

Fig. 1. Thematic organisation of the reviewed literature on XAI-driven deep-learning IDS for DoS classification.

III. LITERATURE REVIEW

1. DoS Detection and the Limits of Conventional IDS

IDS are usually split into signature, anomaly and hybrid types [5]. Signature engines are fast on known attacks yet can be dodged by obfuscation and are blind to zero-day attacks, while anomaly engines learn normal behaviour and can raise alarms on novel activity [5]. Keshk et al. add that DL techniques often generate many false positives and give opaque decisions [5]. Earlier DL-based detectors for DoS and related threats repeat a handful of weaknesses (see Table 4): balancing through synthetic data can encourage overfitting [10]; GAN training can be unstable and suffer mode collapse [11]; an explainable ensemble faced scalability limits [12]; a fuzzy temporal LSTM reached 99.58 % accuracy but needed laborious tuning [13]; federated CNN–BiLSTM detectors incurred communication overhead [14]; and an attention BiLSTM tuned by metaheuristics kept a fixed feature set [15].

2. Deep-Learning Classifiers for DoS and DDoS Traffic

Because attacks develop over time, recurrent models are a natural fit: LSTM cells carry earlier context into each new prediction [5], [8]. Maseno et al. fed sliding windows of five flows (stride 1) to LSTM and GRU layers of 64 units for ten ToN_IoT classes and obtained almost identical clean accuracy (0.9614 and 0.9615) [4]. Keshk et al. built LSTM-based classifiers for NSL-KDD, UNSW-NB15 and ToN_IoT with binary F1 of 0.815, 0.891 and 0.829; one-vs-all versions scored 0.991 accuracy on UNSW-NB15 Generic and 0.987 on ToN_IoT DDoS, but only 0.969 accuracy with F1 0.678 on XSS [5].

Convolution and attention aim for richer representations. Ghosh et al. chain an EfficientNet-B0 extractor with channel and spatial attention to a deep residual multiscale CNN driven by multiplicative Luong attention, whose 1×1, 2×2 and 3×3 kernels look at fine and coarse traffic structure; reported accuracy is 99.95–99.99 % on five datasets [1]. Komarchesqui et al. leave the detector design alone and instead explain an existing semi-supervised 1D-CNN-GAN that learns benign behaviour from Shannon entropy of IP addresses and ports per second [3]. At the light end, Nugraha et al. rely on XGBoost with TreeSHAP, which yields an explanation in 0.02–0.08 s [2]. Two cautions follow. The way [1] turns tabular flow records into an input suitable for an image backbone is not explained in detail, hampering replication. And choosing between a heavy deep network and a small tree model means trading peak accuracy against explanation speed [2].

3. Class Imbalance and Data Augmentation

Ghosh et al. rebalance data with a class-conditional GAN: an auxiliary classifier steers generation towards a requested class, dynamic class weights give rare classes more influence, and a gradient penalty steadies training and curbs mode collapse [1]. Their augmentation lifts the DoS class of KDD-Cup 99 from 97,278 to 391,458 samples and that of UNSW-NB15 from 12,264 to 56,000, and equalises classes at 67,343 (NSLKDD), 48,840 (CIC-DDoS2019) and 461,912 (CSE-CIC-IDS2018) [1]. Removing the GAN costs about 1.2–1.8 points of accuracy and removing EfficientNet-B0 about 1.2–2.3 points (Table 1) [1]. Only these two parts were ablated; the optimiser, the attention blocks and the explanation layer were not.

© 2026 IJSRET

Table 1. Ablation results reported in [1] (accuracy / precision / recall, %).

Dataset	Configuration	Accuracy	Precision	Recall
NSL-KDD (D1)	Entire architecture	99.99	99.98	99.97
	Without RACGAN	98.24	97.89	98.27
	Without EfficientNet-B0	98.78	97.98	97.71
CSE-CIC-IDS2018 (D2)	Entire / w/o RACGAN / w/o EfficientNet-B0	99.96 / 98.78 / 97.89	99.95 / 98.23 / 97.79	99.92 / 98.44 / 97.65
CIC-DDoS2019 (D3)	Entire / w/o RACGAN / w/o EfficientNet-B0	99.97 / 98.78 / 97.88	99.94 / 98.67 / 97.98	99.93 / 98.69 / 97.56
KDD-Cup 99 (D4)	Entire / w/o RACGAN / w/o EfficientNet-B0	99.96 / 98.65 / 98.56	99.97 / 98.12 / 98.16	99.95 / 98.34 / 98.18
UNSW-NB15 (D5)	Entire / w/o RACGAN / w/o EfficientNet-B0	99.95 / 98.61 / 97.67	99.97 / 98.54 / 97.81	99.93 / 98.76 / 98.67

Test sets in [1] look class-balanced for several datasets (equal per-class counts for CSE-CIC-IDS2018 and KDD-Cup 99), so the text leaves open whether only training data were augmented. Synthetic samples in the test data would inflate results. The other two core studies avoid synthetic data: [2] uses balanced subsets, such as 4,000 benign and 4,000 attack flows for CIC-DDoS2019 Portmap, without resampling, and [3] picks a representative subset by unsupervised hierarchical clustering so that SHAP stays affordable, comparing it with stratified random sampling.

4. Optimisation and Hyperparameter Tuning

Optimisation takes three forms in the reviewed work. Metaheuristic search is used in [1], where an enhanced starfish algorithm alternates broad exploration with local exploitation and a regeneration step to tune the classifier; earlier work tuned an attention BiLSTM with a salp swarm [15]. Plain grid search is used in [2] (stratified 10-fold search over XGBoost settings) and in [3] (window length, latent size, layer widths). The third form, proposed in [3], tunes the *inputs*: SHAP exposes features that mislead the detector, they are removed, and the model is retrained in a loop.

For the first form, [1] gives no final hyperparameters, convergence curves, run times or comparison with other optimisers, and its ablation (Table 1) leaves the optimiser out, so the claimed saving in complexity cannot be checked. The other two forms are reported more fully; for instance, tuning in [3] raised the window from 8 to 16 and cut the latent size from 64 to 32 on CIC-DDoS2019.

5. Explainable-AI Methods Used in IDS

Explanation can be local (one decision) or global (whole model). SHAP distributes a prediction among features using Shapley values with guaranteed efficiency, symmetry, dummy and additivity properties [6]; LIME trains a small interpretable surrogate near one instance [7]; permutation feature importance (PFI) tracks how much error grows when one feature is shuffled [23]; partial dependence (PDP) plots the average influence of a feature [24]; individual conditional expectation (ICE) plots one curve per record, exposing interactions that PDP averages away [5]. Table 2 lists these and two IDS-specific uses.

Table 2. XAI techniques used in the reviewed IDS studies.

Technique	Scope	How it explains	Strength in IDS	Limitation	Used in
SHAP [6]	Local + global	Shapley-value attribution of each feature to the prediction	Shows direction and size of effect; TreeSHAP takes 0.02–0.08 s per instance for XGBoost [2]	Kernel SHAP is costly and was limited to 100 samples in [4]	[1]–[5]
LIME [7]	Local	Local linear surrogate fitted on perturbed samples	Model-agnostic, sparse, easy for analysts to read	Time grows with feature count (36.4 s with all	[1], [2]

© 2026 IJSRET

				features, SYN Flood) [2]; can be fooled [17]	
PFI [23]	Global	Error increase after permuting a feature	One number per feature; simple for non-experts [5]	Misses features that matter greatly for only a few records [5]	[5]
PDP [24] / ICE	Global / local	Average / per-record effect of one feature	Reveals direction and groups of records that behave differently [5]	PDP assumes independent features [5]	[5]
Explanation drift and top-k overlap	Local, paired	Distance between clean and perturbed SHAP vectors	Extra signal for adversarial inputs: ROC-AUC 0.70– 0.88 [4]	Classes overlap; weaker at larger scale [4]	[4]
SHAP–LIME consistency κ	Local, paired	Overlap of top-5 features from both explainers; flag if κ < 0.5	Median κ = 0.94; about 2 % of flows need review [2]	Agreement does not prove correctness	[2]

The SPIP framework of Keshk et al. pairs SHAP and ICE for single decisions with PFI, SHAP and PDP for the whole model [5]. Their findings show why one explainer is not enough: SHAP and PFI ranked different top20 features on all datasets, and PFI, which averages over shuffles, underrated ct_src_dport_ltm, a feature that matters for only some records [5]. For ToN_IoT DDoS, conn_state, dns_RD and proto were the main drivers, and SHAP dependence plots showed detected DDoS flows using UDP, which fits the nature of the attack [5]. Surveys of XAI in IDS agree that explanation has become a central topic while its dependability under attack is still open [20], [21], [26].

6 XAI for Feature Selection and Efficient Explanation

Efficient XAI methods are commonly grouped as non-amortised (exact per sample, hard to scale) or amortised (faster but less faithful) [25]. Two reviewed studies cut explanation cost by shrinking the input.

Keshk et al. retrained on SHAP- or PFI-chosen subsets with little change in accuracy; detection time dropped about 14 % on NSL-KDD (507 to 434 ms), 12 % on UNSW-NB15 (1,444 to 1,268 ms) and 12 % on ToN_IoT (1,664 to 1,458 ms), while training time fell only 2–4 % [5].

Nugraha et al. blend a normalised ANOVA score (40 %) with global SHAP importance (60 %) to retain ten features, about 70 % fewer, and retrain a compact XGBoost model [2]. Their ablation favours the blend: ANOVA alone was fast but lost accuracy on harder cases such as Slowloris, whereas SHAP-only and the blend stayed near the full-feature baseline, the blend adding small, steady speed gains. PCA and mutual information scored slightly higher F1 (0.999) in a preliminary test but discard feature meaning or model awareness [2]. Fig. 2 shows the trade-off. With ten features F1 stays at or above 99 % for SYN Flood, HTTP Flood and Slowloris, but is 98.12 % for Portmap, and five features is clearly too few (82.04 % on Portmap). LIME time for SYN Flood falls from 36.38 s (all features) to 9.48 s (ten, −74 %) and 4.85 s (five, −87 %) [2].

The “36 to 4.9 s” figure in the abstract of [2] thus refers to five features rather than the ten-feature setting.

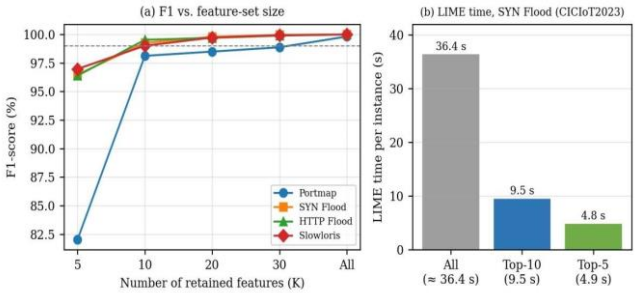

Fig. 2. (a) F1-score against the number of retained features K (Portmap from CIC-DDoS2019; SYN Flood, HTTP Flood and Slowloris from CICIoT2023); (b) LIME explanation time per instance on SYN Flood. Plotted from the results reported in [2]. The same framework adds a reliability test: the top-5 features from SHAP and LIME are compared per flow, and a flow is flagged if the overlap is below 0.5, which happened for roughly

© 2026 IJSRET

2 % of flows [2]. One limit is that the 5G PFCP Session Deletion subset has a full-feature F1 of just 0.857, so the claim of F1 ≥ 99 % does not cover every dataset. All data in [2] are balanced subsets (8,000 flows for Portmap, 120,000 for three CICIoT2023 variants), leaving realistic imbalance untested.

7. Explanation-Guided Optimisation of Detector Performance

Komarchesqui et al. run explanation as a cycle: SHAP beeswarm plots, split by traffic category and by time step (including decay and lag of influence), show which inputs push decisions the wrong way; those inputs are dropped and the model is retrained until the explanations no longer show biased effects [3]. On CICDDoS2019, destination- and source-IP entropies were tied to false alarms and source-port entropy was unclear, so only destination-port entropy remained. F1 rose from 0.865 to 0.901, MCC from 0.804 to 0.851 and recall from 0.773 to 0.833, with missed attacks falling from 693 to 509 and inference time increasing from 2.525 to 3.192 s [3]. On CSE-CIC-IDS2018, F1 went from 0.726 to 0.925, MCC from 0.703 to 0.916 and recall from 0.647 to 0.972; missed attacks dropped from 1,159 to 89 (−92.32 %) and false alarms from 445 to 428 [3] (Fig. 3). Two cautions: the base model in [3] is an entropy-based anomaly detector, not a multiclass DoS classifier, so transfer to models like [1] is unproven; and the large IDS2018 gain came from keeping a single feature, hinting that a confounding input had dominated the original model.

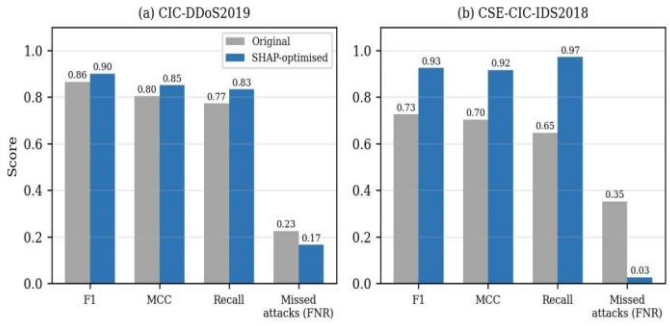

Fig. 3. Effect of SHAP-guided feature pruning on a 1D-CNN-GAN DDoS detector: (a) CIC-DDoS2019, (b) CSE-CICIDS2018. Plotted from Tables 4 and 5 of [3]. Lower is better for FNR.

8. Adversarial Robustness and the Reliability of Explanations

Maseno et al. applied white-box FGSM [9] with ε = 0.1: LSTM accuracy fell from 0.9614 to 0.6094 (evasion 37.38 %) and GRU accuracy from 0.9615 to 0.5130 (evasion 47.02 %) [4]. A sweep (Fig. 4a) showed damage rising with ε and the iterative PGD attack being stronger: at ε = 0.10 accuracy dropped to about 0.33 (LSTM) and 0.35 (GRU), with success rates on originally correct samples of 0.67 and 0.64 [4]. Losses were uneven across classes. Under FGSM, backdoor, DDoS, injection and password traffic reached F1 near zero for both models (DDoS 0.05 for LSTM, 0.02 for GRU), normal traffic stayed at 0.81–0.88, and the MITM class showed high recall but very low precision because altered samples were pushed into it [4]. Weighted averages therefore conceal serious loss on the attack classes that matter most.

Explanations offered a partial remedy. Adversarial samples moved SHAP attributions towards larger drift and smaller top-k overlap with the clean explanation. A rule combining these two signals reached ROCAUC 0.8150 (LSTM) and 0.7523 (GRU) on 100 Kernel-SHAP samples, and a small logistic-regression detector reached 0.8843 and 0.8705 [4]. On 2,000 samples with GradientExplainer the rule-based detector scored only 0.7047 and 0.7323 (Fig. 4b), so the small-sample results were optimistic [4]. The idea echoes earlier use of SHAP signatures to separate adversarial from clean inputs [16]. Explainers themselves can be attacked: LIME and SHAP can be tricked into plausible but misleading output [17]; realistic problem-space attacks on network IDS are harder to craft than feature-space ones [18]; and adversarial training has been combined with explanation in NIDS [19]. Reviews of adversarial XAI conclude that explanation reliability under attack is unsettled [20].

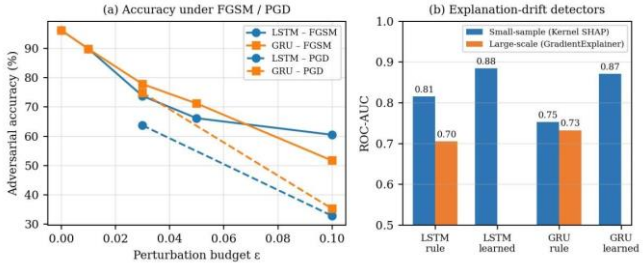

Fig. 4. (a) Adversarial accuracy of LSTM and GRU IDS on ToN_IoT under FGSM and PGD (clean accuracy plotted at ε = 0); (b) ROC-AUC of explanation-drift detectors. Plotted from Table 3 and Section 4 of [4].

This matters for the most accurate systems. Ghosh et al. describe their pipeline as naturally resistant to perturbation and layered in defence, yet report no adversarial experiment, noting only that resilience tests with crafted samples are ongoing [1]. Considering the sharp fall seen for simpler recurrent models in [4], robustness of such pipelines should be regarded as unverified.

© 2026 IJSRET

9. Datasets Used

The studies draw on six dataset families (Table 3). KDD-Cup 99 and NSL-KDD have 41 features and four attack groups; UNSW-NB15, ToN_IoT, CIC-DDoS2019 and CSE-CIC-IDS2018 provide newer attacks and far more flows. ToN_IoT was produced on an edge–fog–cloud testbed that imitates IIoT networks [22]. For DoS work, CIC-DDoS2019 is the most relevant (for example SYN, UDP, UDP-lag and Portmap) [1], [2], [3], whereas NSL-KDD and KDD-Cup 99 are old and may not mirror current traffic. Most experiments use one dataset at a time and cross-dataset checks are nearly absent, which restricts generalisation.

Table 3. Datasets used in the reviewed studies (sizes as reported in the papers).

Dataset	Size / features (as reported)	Classes relevant to this review	Used in
NSL-KDD	125,973 train / 22,544 test; 41 features	DoS (back, land, neptune, pod, smurf, teardrop, …), probe, U2R, R2L	[1], [5]
KDD-Cup 99	About 5 million records; 41 features	DoS, probe, U2R, R2L; [1] uses DoS vs normal	[1]
UNSW-NB15	175,341 train / 82,332 test; 44–47 features depending on source	Analysis, Backdoor, DoS, Exploits, Fuzzers, Generic, Reconnaissance, Shellcode, Worms	[1], [5]
ToN_IoT	322,731 train / 138,313 test (network data)	Backdoor, DoS, DDoS, injection, MITM, normal, password, ransomware, scanning, XSS	[4], [5]
CIC-DDoS2019	Over 50 million entries; 86 flow features (full set); [2] uses a balanced Portmap subset of 8,000 flows (59 features)	DDoS (SYN, UDP, UDP-lag, Portmap, …) and benign	[1], [2], [3]
CSE-CIC-IDS2018	About 16 million records, 17 % attack; 79–83 attributes	Benign, GoldenEye, Hulk, SlowHTTPTest, Slowloris	[1], [3]
CICIoT2023 ; 5G PFCP	CICIoT2023: 120,000 balanced flows in three variants (35 features); PFCP: 6,206 flows (14 features, 9 after pre-processing)	SYN Flood, HTTP Flood, Slowloris; PFCP Session Establishment/Deletion	[2]

10. Synthesis

Four observations stand out. Accuracy has saturated: the best pipelines report 99.95–99.99 % on benchmarks, so progress must be judged on latency, explainability and resilience instead [1], [2].

Explanation has become a tool for selecting features [2], [5], pruning confounders [3] and spotting tampering [4], but each use sits in a different study. The studies with the most complete explanation evidence (speed, consistency, drift) use lightweight models [2], [4], while the top-accuracy deep pipeline offers only qualitative SHAP and LIME plots [1]. Lastly, validation is narrow: offline benchmarks, balanced subsets and single datasets dominate, and no study attacks the most accurate architecture or runs it on live traffic.

IV. COMPARATIVE ANALYSIS

Table 4 contrasts thirteen representative studies. Their metrics cannot be compared directly because tasks, datasets, class definitions, splits and protocols differ, and several earlier works are known only from the summaries in [1] and [4].

© 2026 IJSRET

Table 4. Comparison of representative XAI and DL-based intrusion detection studies.

Author, year	Objective	Method / technology	Major reported findings	Limitations
Fidel et al., 2020 [16]	Detect adversarial inputs	SHAP signatures as secondary detector	Explanation patterns separate clean from adversarial inputs (as summarised in [4])	Explanation-driven detection less developed for IDS [4]
Sauka et al., 2022 [19]	Robust and explainable NIDS	DL with adversarial training and explanation	Adversarial training improves resilience to FGSM and PGD (as summarised in [4])	Focus on classifier, little on explanation traces [4]
Keshk et al., 2023 [5]	Local + global explanation of IoT IDS	LSTM + SHAP, PFI, ICE, PDP; NSL-KDD, UNSW-NB15, ToN_IoT	Binary F1 0.815 / 0.891 / 0.829; SHAP/PFI subsets cut detection time 12–14 %; Generic one-vs-all acc. 0.991	Moderate binary accuracy (0.81–0.87); no adversarial test; explanations do not name the exploited weakness
Ullah et al., 2024 [10]	Imbalanced traffic	IDS-INT: semantic anchors, CNN, CNN-LSTM	Handles imbalance, captures feature interactions (as summarised in [1])	Synthetic balancing risks overfitting [1]
Sathishkumar et al., 2024 [13]	DoS in wireless sensor networks	Fuzzy temporal deep LSTM	99.58 % accuracy	Heavy parameter tuning [1]
Javeed et al., 2024 [14]	Privacy-preserving IDS	Horizontal federated CNN + BiLSTM, zero-trust	Data stay on devices; weights shared	Communication overhead, slow convergence [1]
Li et al., 2024 [11]	Botnet and DoS detection	HDA-IDS: semi-supervised CL- GAN	High accuracy, reduced training time	GAN training instability, mode collapse [1]
Ahmed et al., 2024 [12]	Explainable ensemble IDS	HAEnID with SHAP and LIME	Fewer false alarms, adaptive control	Scalability in large networks [1]
Aljehane et al., 2024 [15]	Optimised DL IDS	Golden jackal optimisation, attention BiLSTM tuned with SSA	Improved classification	Static feature selection [1]
Nugraha et al., 2025 [2]	Efficient, consistent explanation	ANOVA + SHAP (K = 10), XGBoost, SHAP–LIME check; CIC-DDoS2019, CICIoT2023, 5G PFCP	F1 ≥ 99 % on most subsets; LIME 36.4 → 9.5 s (K = 10), 4.9 s (K = 5); median κ = 0.94	Balanced subsets; Portmap F1 98.1 % at K = 10; 5G Session Deletion baseline F1 0.857; no adversarial test

© 2026 IJSRET

Komarchesqui et al., 2026 [3]	Optimise IDS via SHAP	Cluster under-sampling, perclass and temporal SHAP, 1D- CNN-GAN	CIC-DDoS2019 recall 0.773 → 0.833; IDS2018 MCC 0.703 → 0.916, missed attacks −92 %	Anomaly detector on entropy features; single retained feature on IDS2018
Maseno et al., 2026 [4]	Adversarial evasion, explanation drift	LSTM/GRU on flow windows; FGSM, PGD; SHAP drift and top-k overlap	Clean acc. 0.96 → 0.51–0.61 (FGSM, ε = 0.1); detector ROCAUC 0.70–0.88	One dataset (ToN_IoT); featurespace attacks; Kernel SHAP on 100 samples
Ghosh et al., 2026 [1]	DoS classification with XAI	RACGAN-GP, EfficientNet-B0 + hybrid attention, MLADRMCNN, ESOA, SHAP, LIME	99.99 / 99.96 / 99.97 / 99.96 / 99.95 % accuracy on five datasets	No adversarial test; ESOA not ablated; augmentation/test split unclear; tabular-to-CNN input not described

Fig. 5 places the headline accuracy of [1] in context. The model is best on four of five datasets, but only by 0.01–0.33 percentage points, and trails MRLC on CSE-CIC-IDS2018 (99.99 %). With rivals this close to the ceiling, such gaps are unlikely to be meaningful without repeated runs and confidence intervals, which are absent. The contribution of the framework therefore seems to lie more in imbalance handling and explanation than in raw accuracy.

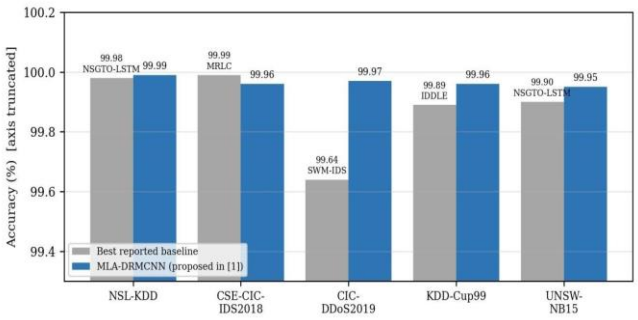

Fig. 5. Accuracy of the proposed MLA-DRMCNN in [1] against the best baseline listed for each dataset in the same paper (accuracy axis truncated at 99.3 %).

V. PROPOSED CONCEPTUAL ARCHITECTURE

Fig. 6 sketches a seven-layer conceptual architecture assembled from the reviewed studies; it has not been built or tested. It joins elements that the literature shows only in isolation.

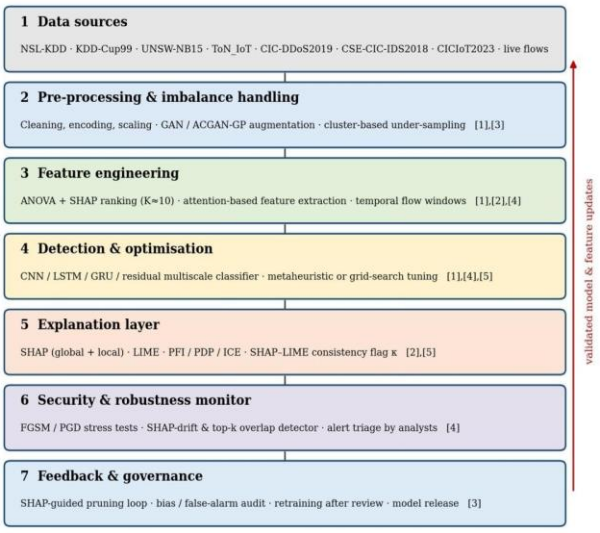

Fig. 6. Proposed conceptual architecture for an explainable, optimised and robustness-aware DL-based DoS detection system (conceptual, not experimentally validated).

Layer 1 collects benchmark and live flow data. **Layer 2** cleans and encodes the data and tackles imbalance by GAN augmentation [1] or cluster-based under-sampling [3]; synthetic samples must stay out of evaluation data. **Layer 3** engineers features by combining a statistical score with SHAP importance to keep a small set [2], plus attention-based extraction [1] and temporal flow windows [4] for deep models.

© 2026 IJSRET

Layer 4 hosts the classifier (CNN, LSTM/GRU or residual multiscale network) tuned by metaheuristic or grid search [1], [3], with the optimiser included in ablation. **Layer 5** produces SHAP and LIME explanations [2], [5] and raises a flag when they disagree [2]. **Layer 6** checks security: FGSM/PGD stress tests before release, and SHAP drift and top-k overlap as an extra adversarial signal for analysts [4]. **Layer 7** closes the loop by auditing explanations for biased inputs and releasing a retrained model only after review [3]. Layers 5 and 6 can run offline for audit and, with lightweight models, online at bounded latency [2].

VI. RESEARCH GAPS AND FUTURE DIRECTIONS

Table 5 lists the gaps visible in the reviewed evidence with matching research directions, ordered by priority.

Table 5. Research gaps and future directions (priority: High / Medium).

Priority	Gap supported by evidence	Studies	Future direction
High	The most accurate DL + XAI pipeline has no adversarial evaluation, while simpler LSTM/GRU models collapse under FGSM/PGD	[1], [4]	Run FGSM, PGD and problem-space attacks on attention/multiscale IDS; pair adversarial training with explanation-drift monitoring
High	Unclear separation of synthetic and real data; balanced test sets	[1], [2]	Augment training folds only; test on untouched, imbalanced, time-ordered data
High	Validation on a single dataset or balanced subset; no live or cross-dataset tests	[1]–[5]	Cross-dataset and online evaluation with measured false-alarm and missed-attack rates
High	No standard metrics for explanation quality (faithfulness, stability, consistency)	[2], [4], [17]	Adopt κ consistency, drift and overlap plus faithfulness tests; test explainers against manipulation
Medium	Optimisation contribution unverified (values, convergence, run time not reported or ablated)	[1], [15]	Report tuned values and run time, compare with grid/Bayesian search, include optimiser in ablation
Medium	XAI-guided pruning shown only for an entropy-based anomaly detector	[3]	Extend explain–prune–retrain loops to multiclass DoS classifiers and more attack families
Medium	Explanation latency and hardware cost reported in one study only	[2], [25]	Benchmark explanation time on edge hardware for deep models, including amortised explainers
Medium	Explanations describe features, not the weakness an attack exploits	[5]	Link explanations to protocol-level causes and analyst workflows (triage, false-alarm audit)

Further open topics include federated and continual learning for privacy-preserving explanation [14], costaware evaluation of false alarms in security-operations work, and human-factors studies on whether analysts decide better with SHAP, LIME or consistency flags.

VII. CONCLUSION

Work on explainable deep-learning DoS detection has moved from black-box classifiers to pipelines that rebalance data, extract attention-based features, tune models with metaheuristics and explain outputs with SHAP and LIME. Benchmark accuracy is very high (99.95–99.99 %), which leaves little room to separate methods. More useful progress comes from putting explanation to work: ANOVA–SHAP selection cut features by about 70 % and sharply reduced LIME time, SHAP-guided pruning lowered missed attacks, and explanation drift gave a supporting signal against adversarial traffic. Evidence is still mostly offline, singledataset or balanced-subset; the strongest architectures have not faced attackers; and explanation quality lacks a shared metric. Dependable deployment will require leakage-free evaluation, adversarial testing, transparent reporting of optimisation, and validated explanation monitoring, as the proposed architecture reflects.

© 2026 IJSRET

REFERENCES

1. Ghosh, S., Goyal, R.K., Chowdhury, K.: "Explainable AI-driven intrusion detection system for DoS attack classification using deep learning and optimization techniques". IEEE Access 14, 5618–5642 (2026). https://doi.org/10.1109/ACCESS.2026.3651187
2. Nugraha, B., Jnanashree, A.V., Bauschert, T.: "A versatile XAI-based framework for efficient and explainable intrusion detection systems". Ann. Telecommun. 80, 1095–1120 (2025). https://doi.org/10.1007/s12243-025-01118-9
3. Komarchesqui, M., Ruffo, V.G.S., Carvalho, L.F., Proença Jr., M.L.: "eXplainable Artificial Intelligence for transparent optimization of deep learning-based intrusion detection systems". J. Netw. Syst. Manag. 34, 94 (2026). https://doi.org/10.1007/s10922-026-10084-z
4. Maseno, E.M., Sun, Y., Wang, Z.: "Detecting adversarial evasion in deep learning intrusion detection systems using explainable AI". Int. J. Inf. Secur. 25, 125 (2026). https://doi.org/10.1007/s10207-026-01299-x
5. Keshk, M., Koroniotis, N., Pham, N., Moustafa, N., Turnbull, B., Zomaya, A.Y.: "An explainable deep learning-enabled intrusion detection framework in IoT networks". Inf. Sci. 639, 119000 (2023). https://doi.org/10.1016/j.ins.2023.119000
6. Lundberg, S.M., Lee, S.-I.: "A unified approach to interpreting model predictions". Adv. Neural Inf. Process. Syst. 30, 4765–4774 (2017).
7. Ribeiro, M.T., Singh, S., Guestrin, C.: "'Why should I trust you?': Explaining the predictions of any classifier". In: Proc. 22nd ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining, pp. 1135–1144 (2016).
8. Hochreiter, S., Schmidhuber, J.: "Long short-term memory". Neural Comput. 9(8), 1735–1780 (1997). https://doi.org/10.1162/neco.1997.9.8.1735
9. Goodfellow, I.J., Shlens, J., Szegedy, C.: "Explaining and harnessing adversarial examples". In: 3rd Int. Conf. Learning Representations (ICLR 2015). https://doi.org/10.48550/arXiv.1412.6572
10. Ullah, F., Ullah, S., Srivastava, G., Lin, J.C.-W.: "IDS-INT: Intrusion detection system using transformer-based transfer learning for imbalanced network traffic". Digit. Commun. Netw. 10(1), 190–204 (2024). https://doi.org/10.1016/j.dcan.2023.03.008
11. Li, S., Cao, Y., Liu, S., Lai, Y., Zhu, Y., Ahmad, N.: "HDA-IDS: A hybrid DoS attacks intrusion detection system for IoT by using semi-supervised CL-GAN". Expert Syst. Appl. 238, 122198 (2024). https://doi.org/10.1016/j.eswa.2023.122198
12. Ahmed, U., Jiangbin, Z., Almogren, A., Khan, S., Sadiq, M.T., Altameem, A., Rehman, A.U.: "Explainable AI-based innovative hybrid ensemble model for intrusion detection". J. Cloud Comput. 13(1), 150 (2024). https://doi.org/10.1186/s13677-024-00712-x
13. Sathishkumar, P., Gnanabaskaran, A., Saradha, M., Gopinath, R.: "DoS attack detection using fuzzy temporal deep long short-term memory algorithm in wireless sensor network". Ain Shams Eng. J. 15(12), 103052 (2024). https://doi.org/10.1016/j.asej.2024.103052
14. Javeed, D., Saeed, M.S., Adil, M., Kumar, P., Jolfaei, A.: "A federated learning-based zero trust intrusion detection system for Internet of Things". Ad Hoc Netw. 162, 103540 (2024). https://doi.org/10.1016/j.adhoc.2024.103540
15. Aljehane, N.O., Mengash, H.A., Eltahir, M.M., Alotaibi, F.A., Aljameel, S.S., Yafoz, A., Alsini, R., Assiri, M.: "Golden jackal optimization algorithm with deep learning assisted intrusion detection system for network security". Alexandria Eng. J. 86, 415–424 (2024). https://doi.org/10.1016/j.aej.2023.11.078
16. Fidel, G., Bitton, R., Shabtai, A.: "When explainability meets adversarial learning: Detecting adversarial examples using SHAP signatures". In: 2020 Int. Joint Conf. Neural Networks (IJCNN), pp. 1–8. IEEE (2020). https://doi.org/10.1109/IJCNN48605.2020.9207637
17. Slack, D., Hilgard, S., Jia, E., Singh, S., Lakkaraju, H.: "Fooling LIME and SHAP: Adversarial attacks on post hoc explanation methods". In: Proc. 2020 AAAI/ACM Conf. AI, Ethics, and Society, pp. 180–186 (2020). https://doi.org/10.1145/3375627.3375830
18. Apruzzese, G., Andreolini, M., Ferretti, L., Marchetti, M., Colajanni, M.: "Modeling realistic adversarial attacks against network intrusion detection systems". Digit. Threats Res. Pract. 3(3), 24 (2022). https://doi.org/10.1145/3469659
19. Sauka, K., Shin, G.Y., Kim, D.W., Han, M.M.: "Adversarial robust and explainable network intrusion detection systems based on deep learning". Appl. Sci. 12(13), 6451 (2022). https://doi.org/10.3390/app12136451
20. Khan, N., Ahmad, K., Al Tamimi, A., Alani, M.M., Bermak, A., Khalil, I.: "Explainable AI-based intrusion detection systems for Industry 5.0 and adversarial XAI: a systematic review". Information 16(12), 1036 (2025). https://doi.org/10.3390/info16121036
21. Mohale, V.Z., Obagbuwa, I.C.: "A systematic review on the integration of explainable artificial intelligence in intrusion detection systems to enhancing transparency and interpretability in cybersecurity". Front. Artif. Intell. 8,

© 2026 IJSRET

1526221 (2025). https://doi.org/10.3389/frai.2025.1526221
22. Moustafa, N.: “A new distributed architecture for evaluating AI-based security systems at the edge: Network TON_IoT datasets”. Sustain. Cities Soc. 72, 102994 (2021). https://doi.org/10.1016/j.scs.2021.102994
23. Fisher, A., Rudin, C., Dominici, F.: “All models are wrong, but many are useful: learning a variable’s importance by studying an entire class of prediction models simultaneously”. J. Mach. Learn. Res. 20(177), 1–81 (2019).
24. Friedman, J.H.: “Greedy function approximation: a gradient boosting machine”. Ann. Stat. 29(5), 1189–1232 (2001).
25. Chuang, Y.-N., Wang, G., Yang, F., Liu, Z., Cai, X., Du, M., Hu, X.: “Efficient XAI techniques: a taxonomic survey”. arXiv:2302.03225 (2023). Preprint.
26. Neupane, S., et al.: “Explainable intrusion detection systems (X-IDS): a survey of current methods, challenges, and opportunities”. IEEE Access 10, 112392–112415 (2022). https://doi.org/10.1109/ACCESS.2022.3216617

© 2026 IJSRET