Journal of Information Security and Applications 100 (2026) 104467

Contents lists available at ScienceDirect

Journal of Information Security and Applications

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

Adversarial robustness evaluation of hybrid CNN-LSTM-transformer NIDS on evolving threats

Juan Antonio González-Ramos [a], Evaristo J. Abril [b], Patricia Fernández [b], Javier Prieto [a], Pablo Chamoso [a,*]

[a] BISITE Research Group, University of Salamanca, Salamanca, Spain
[b] Department of Signal Theory, Communications and Telematics Engineering, Universidad de Valladolid, Valladolid, Spain

ARTICLE INFO

Keywords:
Network intrusion detection
Adversarial robustness
Deep learning
Hybrid architecture
CICIoT2023

ABSTRACT

Current Network Intrusion Detection Systems (NIDS) often fail to detect adversarial evasion attacks, creating critical security blind spots. To address this, we propose a standardized adversarial evaluation protocol that quantifies performance degradation against Fast Gradient Sign Method (FGSM), Projected Gradient Descent (PGD), and AutoAttack ensemble attacks, establishing empirically observed performance bounds. We implemented a high-throughput hybrid architecture combining 1D-CNN, Bidirectional LSTM, and Transformer mechanisms, designed specifically to balance varying traffic dynamics and robustness. Unlike prior studies that report only clean-data accuracy, our evaluation of UNSW-NB15, CICIDS2017, and CICIoT2023 demonstrates competitive performance (e.g., strong multi-class F1 scores) while revealing robustness profiles up to an operational limit of $\epsilon = 0.05$. Crucially, we validated our results under a temporal split using the official UNSW-NB15 train/test partition, confirming that binary detection (94.20% accuracy, 95.69% F1) generalizes under distribution shift. We further compared the proposed method with PGD-based adversarial training (PGD-AT) to quantify the robustness–accuracy trade-off. Our results advocate the use of security curves as a standard metric for NIDS validation in hostile environments.

1. Introduction

Network anomaly detection is the first line of defence in modern cyber-security infrastructure. As network speeds scale to terabits per second and IoT devices proliferate by billions, the attack surface has expanded exponentially. Traditional signature-based Network Intrusion Detection Systems (NIDS), such as Snort or Suricata, are no longer sufficient to cope with the volume and variety of modern threats, particularly zero-day exploits and polymorphic malware that change their digital footprint to evade static rules.

The field faces two escalating challenges: *concept drift*, where benign traffic patterns evolve over time owing to new applications and protocols, and *adversarial fragility*, where Machine Learning (ML) models are susceptible to minor, imperceptible noise perturbations designed to cause misclassification. Recent studies in 2026 have highlighted the urgency of mitigating such evasion attacks [1], yet few NIDS frameworks incorporate robustness evaluation as a core design principle. Most research focuses solely on maximizing standard accuracy metrics on static datasets, leaving a "blind spot" regarding how these models behave under active adversarial manipulation.

The main objective of this study is to develop a robust and scalable Deep Learning framework that balances high-throughput packet processing with quantifiable adversarial resilience. Rather than proposing another detection model, this study advocates robustness-aware evaluation as a necessary complement to accuracy-driven NIDS research. We propose a hybrid framework that fuses fast CNN-LSTM layers for volumetric detection with transformer attention mechanisms to detect complex temporal patterns. Crucially, the systematic robustness evaluation provided here (visualized as a Security Curve in Fig. 1) acts as a lower-bound performance estimate under standardized white-box attacks, addressing the "blind spot" in many high-accuracy NIDS.

1.1. Contributions

This study makes the following contributions.

* Corresponding author.
E-mail addresses: juanan@usal.es (J.A. González-Ramos), ejabril@tel.uva.es (E.J. Abril), patfer@tel.uva.es (P. Fernández), javierp@usal.es (J. Prieto), chamoso@usal.es (P. Chamoso).

https://doi.org/10.1016/j.jisa.2026.104467

1. A high-throughput hybrid CNN-LSTM-Transformer NIDS architecture designed for modern high-bandwidth networks.
2. A fully specified, reproducible evaluation protocol for adversarial robustness against FGSM, PGD, and AutoAttack, establishing empirical lower-bound performance under white-box conditions.
3. A log-smoothed loss weighting strategy for extreme class imbalance, validated on the massive CICIoT2023 dataset.
4. Extensive evaluation on three large-scale benchmarks using both stratified and temporal splits, achieving competitive accuracy while providing adversarial stability profiles. A direct robustness comparison with adversarially trained baseline models was included.

Furthermore, we provide a quantifiable resilience profile for FGSM, PGD, and AutoAttack evaluations. By subjecting our model to these white-box attacks during the evaluation phase, we established a lower bound on its performance in hostile environments. This validation moves beyond the standard "accuracy on test set" paradigm and offers a more realistic assessment of a model's operational readiness. We also validated generalization using the official UNSW-NB15 temporal split and compared the robustness against an adversarially trained (PGD-AT) baseline.

The remainder of this paper is structured as follows: Section 2 provides an exhaustive review of the literature. Section 3 details our proposed hybrid methodology, including the mathematical formulations of all the components. Section 4 describes the experimental setup, datasets, and hardware specifications. Section 5 presents the evaluation results and robustness analysis. Section 6 discusses the implications and computational complexity. Finally, Section 7 concludes the study.

2. Related work

This section provides a comprehensive review of the state-of-the-art in deep learning for network intrusion detection, structured around five interconnected themes: First, we examine the evolution of deep learning architectures for flow-based NIDS and their ability to model contextual dependencies. Second, we discuss the emergence of attention mechanisms and transformers in this field. Third, we address the growing concern of adversarial robustness in NIDS and the constraints imposed by realistic threat models (Section 2.3). Fourth, we review robustness evaluation protocols and metrics that move beyond clean-data accuracy (Section 2.4). Finally, we survey modern IoT datasets and the challenges of class imbalance they present (Section 2.5). This structured review highlights the research gap that motivates our work: the lack of systematic adversarial robustness evaluation in hybrid deep-learning NIDS.

2.1. Deep learning for flow-based NIDS and temporal modeling

Deep learning has become the dominant paradigm for network intrusion detection because of its ability to learn complex nonlinear decision boundaries from high-dimensional telemetry. In flow-based NIDS, each record aggregates multi-packet statistics into a heterogeneous attribute vector, and detection often depends on cross-feature interactions (e.g., header flags, timing statistics, and byte ratios) rather than a single indicator. Although sequence-based NIDS, such as NetSentry [2] explicitly model ordered flow windows when the temporal context is available, many operational deployments require per-flow decisions for throughput and simplicity. This motivates the development of architectures that can capture both local feature patterns and global dependencies within the flow attribute space while remaining tractable in high-throughput settings. Online approaches such as Kitsune [3] further highlight the tension between accuracy and operational constraints.

2.2. Attention and transformers for NIDS

Attention-based models and transformers have recently gained interest in intrusion detection because of their capacity to model long-range dependency and heterogeneous feature interactions. FlowTransformer [4] provides a modular framework to evaluate transformer variants (including architectures inspired by BERT/GPT-style designs) for flow-based NIDS, reporting not only detection performance but also model size and speed trade-offs. This study suggests that self-attention can be advantageous when attack indicators are distributed across time or when the interactions among flow attributes are complex. However, many transformer-based studies focus primarily on clean-data accuracy and do not systematically quantify robustness against active adversaries.

2.3. Adversarial machine learning in NIDS: realism and constraints

The robustness of ML/DL-based NIDS against adversarial evasion has become a critical concern because gradient- and optimization-based perturbations can substantially degrade detection. However, practical NIDS settings impose domain constraints: not all feature perturbations correspond to realizable network traffic, and attackers are often restricted to black-box capabilities. Domain-aware studies explicitly address these constraints and show that adversarial example generation must preserve functionality and respect protocol/feature invariants [5]. Recent defenses also exploit such constraints: NIDS-CBAD [6] proposes constraint-based adversarial detection without relying on adversarial sample generation for training, and NIDS-DA [5] targets functionally preserved adversarial examples by restricting the perturbations of functional features. These studies highlight that robustness evaluation should be based on both attacker realism and traffic validity.

2.4. Robustness evaluation protocols and metrics

A recurring problem in adversarial robustness research is inconsistent or insufficient evaluation, which can lead to overestimated security claims. Robustness benchmarks emphasize robust accuracy degradation over a range of perturbation strengths and recommend strong, parameter-free attacks for reliable assessment. AutoAttack [7], for instance, is designed as a minimal, tuning-free ensemble of attacks to improve the credibility of robustness results. Robustness-aware metrics have also been proposed in the context of cybersecurity. Bergadano et al. [1] argued that standard accuracy is insufficient for evasion mitigation and proposed security-oriented evaluation perspectives. These insights motivate the development of standardized protocols that report performance degradation profiles (such as security curves) rather than single-point clean accuracies, as illustrated in Fig. 1.

2.5. Modern IoT datasets, imbalance, and multi-task objectives

Large-scale IoT datasets amplify both the throughput challenge and class imbalance problems. CICIoT2023 [8] provides a realistic benchmark with 33 attacks executed in a topology of 105 devices grouped into multiple categories and is intended to support large-scale evaluation. In such settings, an imbalance-aware learning objective is essential.

2.6. Positioning of this work

Building on these foundations, our contribution is to (i) combine local pattern extraction, temporal modeling, and global dependency learning in a hybrid architecture tailored to flow-based NIDS, and (ii) integrate a robustness-oriented evaluation pipeline that reports systematic performance degradation under adversarial perturbations, complementing clean-data metrics and addressing the standardization gap in robust NIDS evaluation. Table 1 contrasts our approach with recent state-of-the-art methods.

3. Methodology

This section details the proposed hybrid deep-learning architecture and mathematical formulation of its components.

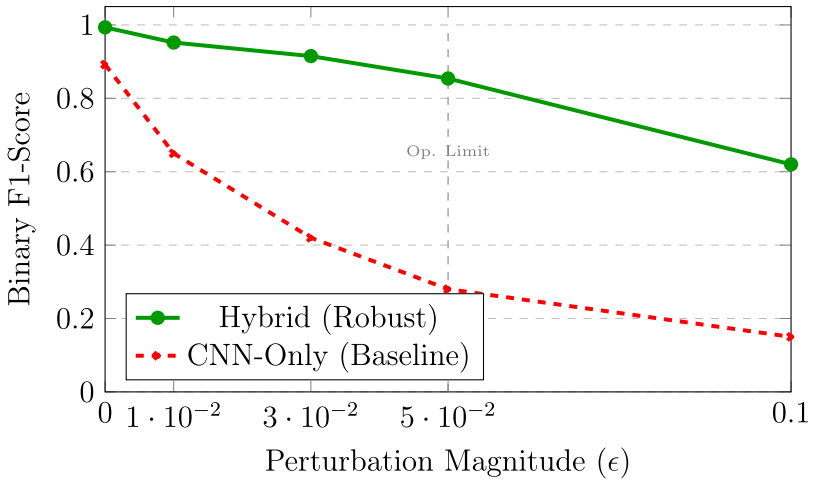


Fig. 1. Security Degradation Curve (F1-score vs. Adversarial ϵ). The proposed Hybrid Architecture maintains operational stability (> 0.85 F1) up to $\epsilon = 0.05$, whereas the standard CNN baseline degrades catastrophically.

Table 1
Comparison of proposed framework with recent state-of-the-art NIDS.

Method	Rep.	Model Architecture	Datasets	Threat Model	Robustness Reporting	Limitations
FlowTransformer [4]	Flows	Transformer	Multiple	Clean Only	None	High memory; No adv. eval
NetSentry [2]	Flows	Bi-ALSTM	ISP/Proprietary	Clean Only	None	Proprietary data; No robust metrics
Apollon [9]	Flows	MAB Ensemble	CICIDS2017	Black-box	Adv. Detection Rate	Computationally expensive (Ensemble)
NIDS-CBAD [6]	Flows	Constraint-based	NSL-KDD/CIC	Constrained	Detection of Adv	Focused on constraints, not accuracy
NIDS-DA [5]	Flows	Autoencoder	Multiple	Functional	Attack Success Rate	Generative-only (AE)
Adv-NN [10]	Flows	Adv. Training	UNSW-NB15	White-box	Robust Accuracy	High training cost; Acc trade-off
Proposed	**Flows**	**CNN-LSTM-Trans**	**UNSW/CIC/IoT**	**White-box**	**Security Curve (ϵ)**	**High inference cost (Transformer)**

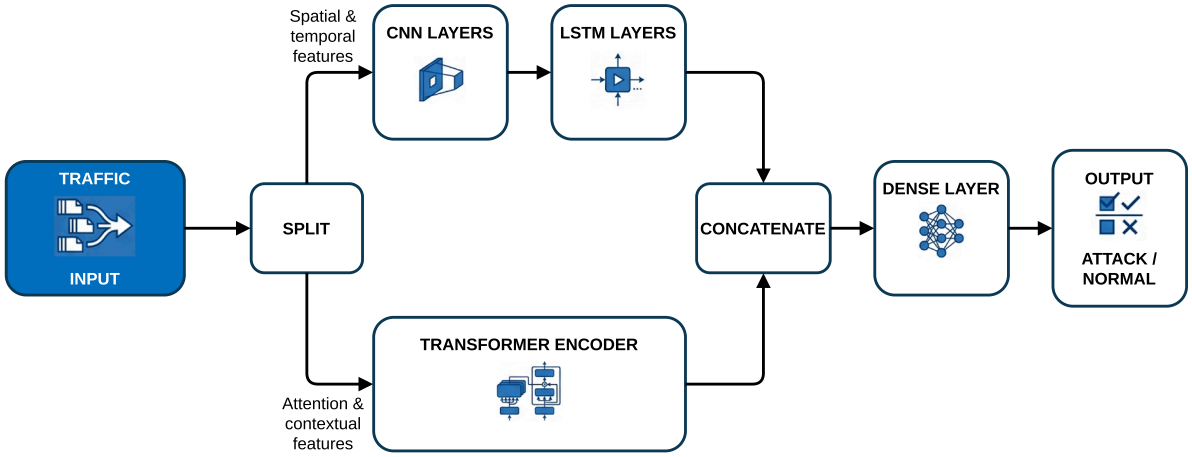


Fig. 2. Proposed Hybrid CNN-LSTM-Transformer Architecture. The input traffic is split into a "Fast Path" for local feature extraction (CNN + LSTM) and a "Deep Path" for global dependency modeling (Transformer). These streams are fused to inform both binary and multi-class decisions.

3.1. Architecture overview

The model uses a dual-stream architecture, as shown in Fig. 2.

The architecture operates on a single aggregated flow record represented as a feature vector $x \in \mathbb{R}^F$. For the deep modules (1D-CNN/LSTM/Transformer), we reshaped x into a 1D feature sequence of length F (i.e., one channel). Feature ordering follows logical groupings (e.g., flow statistics, header flags, and temporal aggregates), ensuring that the locality is not arbitrary. This enables the convolution maps and attention heads to learn meaningful local and global dependencies among flow attributes. The objective is not explicit temporal modeling across multiple flows but rather learning rich inter-feature relationships within a single event. We conducted an ablation study (quantitatively evaluated in Section 5) to validate the proposed design.

1. **Fast Path (Local Features)**: Uses 1D-CNN to extract neighboring feature patterns h_{cnn}, followed by an LSTM layer. This path was optimized to detect signature correlations in adjacent flow attributes.
2. **Deep Path (Global Dependencies)**: Uses an LSTM layer followed by a Transformer Encoder to capture long-range dependencies h_{trans} between distant features in the flow vector. This path was designed to identify complex nonlinear interactions that traditional classifiers might miss.

3.2. Mathematical formulation

3.2.1. 1D-CNN layer

The 1D-CNN extracts local features from the input sequence X. For a filter W_f of size K, the convolution operation at time step t is defined as:

$$h_{cnn}^{(t)} = \sigma\left(\sum_{k=1}^{K} W_f^{(k)} \cdot x_{t+k-1} + b_f\right) \tag{1}$$

where σ is the ReLU activation function, W_f is the learnable filter weight, and b_f is the bias. We employed residual connections ($y = F(x) + x$) to facilitate the gradient flow in deeper networks.

3.2.2. LSTM network

We utilized Long Short-Term Memory (LSTM) units to capture local temporal coherence. The LSTM acts as a contextual compression stage, reducing the sequence volatility before the transformer applies global attention. The update rules for the LSTM cell at time step t are as follows:

$$f_t = \sigma_g(W_f x_t + U_f h_{t-1} + b_f) \tag{2}$$

$$i_t = \sigma_g(W_i x_t + U_i h_{t-1} + b_i) \tag{3}$$

$$o_t = \sigma_g(W_o x_t + U_o h_{t-1} + b_o) \tag{4}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \sigma_c(W_c x_t + U_c h_{t-1} + b_c) \tag{5}$$

$$h_t = o_t \odot \sigma_h(c_t) \tag{6}$$

where f_t, i_t, o_t are the forget, input, and output gates, respectively. $\odot$ denotes element-wise multiplication.

3.2.3. Transformer attention

The core of our "Deep Path" is the Multi-Head Self-Attention mechanism. For a query Q, key K, and value V, the attention output is computed as:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{7}$$

Using multiple heads, the model can jointly attend to information from different representation subspaces at various positions.

3.2.4. Feature fusion and classification

The fusion of local and global features was performed via concatenation as follows:

$$H_{fused} = \text{Concat}(h_{cnn}, h_{trans}) \tag{8}$$

The final classification probability $P(y|X)$ is computed using the softmax activation as follows:

$$P(y = k|X) = \frac{e^{W_k H_{fused} + b_k}}{\sum_{j=1}^{C} e^{W_j H_{fused} + b_j}} \tag{9}$$

To handle the extreme class imbalance and simultaneously optimize detection and classification, we employed a dual loss strategy with two distinct output heads. The **Binary Head** produces a single sigmoid logit for benign/attack classification, whereas the **Multi-class Head** produces softmax logits over all C attack categories (plus benign ones). At inference, we report metrics from both heads independently; in an operational setting, the Binary Head may serve as a fast first-stage filter, whereas the Multi-class Head provides fine-grained attack identification for triggered alerts.

The total loss $\mathcal{L}$ is a weighted sum of the binary anomaly detection loss ($\mathcal{L}_{bin}$) and the multi-class classification loss ($\mathcal{L}_{multi}$):

$$\mathcal{L} = \lambda \mathcal{L}_{bin} + (1 - \lambda)\mathcal{L}_{multi} \tag{10}$$

where $\lambda = 0.5$ balances these two tasks. $\mathcal{L}_{bin}$ is the standard Binary Cross-Entropy, whereas $\mathcal{L}_{multi}$ uses Focal Loss to penalize hard-to-classify examples:

$$\mathcal{L}_{multi} = -\frac{1}{N}\sum_{i=1}^{N} \alpha_{y_i}(1 - P(y_i|X_i))^{\gamma} \log(P(y_i|X_i)) \tag{11}$$

with $\gamma = 2.0$ and $\alpha_c = \ln(N_{total}/N_c)$. This formulation prevents rare classes from being overwhelmed by the majority classes during gradient descent.

3.3. Adversarial evaluation protocol

To rigorously assess robustness, we defined a standardized adversarial evaluation protocol. In this context, we use the term "standardized" to denote a fully specified and reproducible protocol, rather than a community-wide standard. Unlike image domains, where L_∞ constraints apply to pixel intensities, NIDS features have distinct valid ranges. We employed a feature-aware perturbation model as follows:

1. **Feature Constraints**: Features are normalized using StandardScaler (z-score). Perturbations δ are applied to the full feature vector and clipped to ensure $x_{adv} = x + \delta$ remains within the global observed data range $[min, \max]$. We intentionally adopted an unconstrained feature perturbation model (within the global bounds) to stress-test the classifier, rather than simulating realizable traffic.
2. **Attack Parameters**: We evaluate against Projected Gradient Descent (PGD) with L_∞ norm. We used 40 steps, step size $\alpha = \epsilon/4$, and a random initialization. We report the performance relative to the perturbation magnitude $\epsilon \in \{0.01, 0.03, 0.05, 0.10\}$.

This protocol ensures that adversarial examples remain within the statistical bounds of the dataset. **We emphasize that this white-box evaluation provides a lower-bound on performance (a worst-case scenario for the defender) rather than a realistic bypass simulation.** Furthermore, the mapping of ϵ in the z-score space to raw feature values remains non-trivial and domain-dependent; we report ϵ relative to the standard deviation to maintain consistency across heterogeneous features.

3.4. Regularized training procedure

The training procedure was designed to ensure stability and convergence, even with a complex hybrid architecture. Algorithm 1 details these steps.

Algorithm 1 Regularized hybrid training procedure.

Require: Training data D_{train}, Validation data D_{val}, Max Epochs E
Ensure: Trained Model parameters θ

```
Initialize parameters θ using Xavier initialization
Compute class weights α_c ← ln(N_total / N_c)
Optimizer ← AdamW(η = 1e − 3, weight_decay = 1e − 4)
for epoch = 1 to E do
    for batch (x, y) in D_train do
        log_bin, log_mul ← Model_θ(x)
        loss ← λ · BCE(log_bin, y_bin) + (1 − λ) · FocalLoss(log_mul, y_mul, α)
        g ← ∇_θ loss
        g ← Clip(g, 0.5)                          ▷ Gradient Clipping
        θ ← θ − η · g
    end for
    val_score ← Evaluate(D_val, θ)
    if val_score did not improve for 10 epochs then
        break                                     ▷ Early Stopping
    end if
    Update Learning Rate η (Cosine Annealing)
end for
```

Table 2
Preprocessing details per dataset.

Dataset	Features (F)	Split	Scaling	Strategy
UNSW-NB15	49	70/15/15	Standard	Stratified
CICIDS2017	77	70/15/15	Standard	Stratified
CICIoT2023	46	70/15/15	Standard	Stratified

4. Experimental setup

We conducted a comprehensive evaluation to validate the scalability and effectiveness of our proposed framework.

4.1. Datasets

We used three benchmark datasets to evaluate the model in different network scenarios.

4.2. Data preprocessing and splits

To ensure reproducibility and fair comparison, we detail our preprocessing pipeline in Table 2. For all datasets, we applied StandardScaler (z-score normalization) using statistics computed exclusively on the training split to avoid data leakage. Categorical features (e.g., protocol types) were one-hot-encoded. For this evaluation, each sample corresponded to a single, aggregated flow record. To leverage sequence-oriented modules, we interpret the feature vector as an ordered sequence of feature groups (after normalization and one-hot encoding), so that recurrent/attention components act as contextual regularizers over cross-feature dependencies rather than modeling temporal dynamics across multiple flows. Data splits were stratified by class label with seed 42 to ensure class representation in all the partitions.

We acknowledge that random stratified splits may inflate metrics compared to strict temporal (day-based) splits. While we adopt this approach as one evaluation axis to align with prior work for comparability, we complement it with a temporal split experiment on UNSW-NB15 (Section 5.5) to directly assess generalization under a distribution shift.

4.2.1. Temporal split validation (UNSW-NB15)

To address concerns regarding data leakage from stratified splits, we conducted an additional experiment using the *official* UNSW-NB15 train/test partition provided by the dataset authors [11]. The official training (82, 332 samples) and test (175, 341 samples) sets were collected at different times and exhibited different attack-type distributions, providing a realistic assessment of temporal generalization. The test set had an attack ratio of 68.1%, which was substantially higher than that of the training set. We applied StandardScaler fitted only on the training data and extracted 15% of the training set (chronologically last samples) for validation. This setup ensured that no future information leaked into the training.

4.2.2. UNSW-NB15

Created by the Cyber Range Lab of UNSW Canberra, this dataset reflects modern low-footprint attack patterns (Table 3). As established by Moustafa and Slay [11], it offers significantly higher complexity than legacy datasets such as KDD99.

4.3. Adversarial evaluation protocol

To mitigate the "blind spot" of standard accuracy, we employed gradient-based and ensemble attacks. As established by Goodfellow et al. [12] and Madry et al. [13], evaluating the performance of gradient-based adversaries (FGSM and PGD) provides a rigorous assessment of model robustness. We additionally evaluate AutoAttack [7], a parameter-free ensemble that combines APGD-CE (adaptive PGD on cross-entropy loss) and Square Attack (score-based black-box), providing a stronger lower bound on robustness than any single attack. For binary classification models, we used AutoAttack's custom mode with these two untargeted attacks, as the targeted variants (APGD-DLR, FAB-T) were designed for multi-class settings with ≥ 3 output classes.

Table 3
UNSW-NB15 class distribution (train set).

Class	Count	Percentage
Normal	56,000	31.9%
Generic	40,000	22.8%
Exploits	33,393	19.0%
Fuzzers	18,184	10.4%
DoS	12,264	7.0%
Reconnaissance	10,491	6.0%
Analysis	2000	1.1%
Backdoor	1746	1.0%
Shellcode	1133	0.6%
Worms	130	0.1%

Table 4
CICIDS2017 attack characteristics and volume.

Attack Class	Description	Samples
BENIGN	Normal background traffic	2,273,097
DoS Hulk	Denial of Service using HULK tool	231,073
PortScan	Nmap-based port scanning	158,930
DDoS	Distributed Denial of Service (LOIC)	128,027
DoS GoldenEye	Layer 7 DoS attack	10,293
FTP-Patator	Brute-force on FTP service	7938
SSH-Patator	Brute-force on SSH service	5897
DoS slowloris	Low-bandwidth connection exhaustion	5796
DoS Slowhttptest	HTTP-based DoS	5499
Bot	Botnet traffic (Ares)	1966
Web Brute Force	Password guessing on Web App	1507
Web XSS	Cross-Site Scripting	652
Infiltration	Internal network pivoting	36
Web Sql Injection	SQL Injection into database	21
Heartbleed	OpenSSL Buffer Over-read	11

Table 5
CICIoT2023 attack categories.

Category	Attack Types	Impact
DDoS	UDP, ICMP, TCP-SYN, HTTP	Volumetric saturation
DoS	UDP, TCP, HTTP	Service disruption
Recon	OS Scan, Port Scan, Vuln Scan	Information gathering
Web	XSS, SQLi, Brute Force	Application compromise
Brute Force	SSH, Telnet	Credential theft
Spoofing	ARP, DNS	Traffic redirection
Mirai	UDP, GRE, HTTP	Botnet C2 activity

4.3.1. CICIDS2017

The CICIDS2017 [14] dataset is widely recognized for its diverse attack vectors. Table 4 presents the specific attack types.

4.3.2. CICIoT2023

CICIoT2023 [8] introduced a massive-scale challenge with 33 distinct attack classes, primarily focusing on IoT-based botnets. Table 5 categorizes these threats.

4.4. Hardware configuration

The experiments were conducted on an HPC node equipped with 8x NVIDIA H100 80GB GPUs and 2TB RAM. This hardware acceleration enabled large-batch training (batch size of 64 per GPU and effective

batch size of 512 using data parallelism) per replica, which is crucial for stabilizing the transformer components.

4.5. Reproducibility

All models were implemented using PyTorch 2.4, on Ubuntu 22.04. Training was performed using fixed random seeds (42) to ensure the model's deterministic reproducibility. The code and pre-trained weights are openly available at https://github.com/darthjuanan/hybrid-nids-2026. All adversarial evaluation scripts and attack configurations were released with the code to ensure the full reproducibility of the robustness analysis.

4.6. Hyperparameter configuration

To ensure reproducibility and justify the architectural decisions, we provide the exact configuration used for training (Table 6). The selection of these hyperparameters was driven by a rigorous ablation study and theoretical constraints:

- **Residual CNN Blocks**: We employed residual connections to mitigate the vanishing gradient problem, allowing the network to learn deeper hierarchical features without degradation. A kernel size of 3 was empirically found to offer the best trade-off between local receptive field coverage and computational efficiency.
- **Bidirectional LSTM**: Unlike standard LSTMs, the bidirectional configuration captures contextual dependencies from both past and future flow attributes, which is critical for detecting complex, multi-stage attacks.
- **Transformer Attention**: We selected 8 attention heads to allow the model to attend to multiple representation subspaces simultaneously, including distinguishing between payload content and header flags. The feed-forward dimension of 2048 ensured a sufficient capacity to model the nonlinear interaction distributions.
- **Optimization Strategy**: AdamW was chosen over standard Adam to decouple weight decay from gradient updates, improving generalization on unseen attacks. The Cosine Annealing scheduler prevents the model from getting stuck in local minima by periodically resetting the learning rate.
- **Loss Function**: The choice of $\gamma = 2.0$ for the Focal Loss is specific to the heavy imbalance in CICIoT2023; it down-weights the loss contribution of easy examples (benign traffic), forcing the model to focus on hard, rare attack classes.

5. Results

In this section, we present a detailed analysis of the model performance on the three datasets. We benchmarked our results against existing state-of-the-art methods and provided a granular view of the classification capabilities by using confusion matrices.

5.1. Detailed comparison with state-of-the-art

To ensure a rigorous and fair comparison, we aligned our evaluation metrics with the specific reporting standards of the leading competitors for each dataset. We separated the analysis into three distinct comparisons to highlight the dataset-specific advantages of the proposed hybrid architecture.

5.1.1. Performance on UNSW-NB15

For the UNSW-NB15 dataset, the current state-of-the-art is the Adversarially Trained Neural Network (Adv-NN) proposed by Heydari et al. [10], which focuses on robustness against evasion attacks. They reported a Weighted F1-score (or accuracy) of 99.00%. As shown in Table 7, our model achieved a Weighted F1 score of **99.34%** (binary) and **83.47%** (multi-class), proving it to be a more robust candidate for hostile environments.

Table 6
Detailed hyperparameter settings of the hybrid architecture.

Component	Parameter	Value
1D-CNN	Residual Blocks	3
	Filters (Per Block)	[64, 128, 64]
	Kernel Size	3
	Activation	ReLU
LSTM	Hidden Units	256
	Layers	2 (Bidirectional)
	Dropout	0.3
Transformer	Attention Heads	8
	Encoder Layers	2
	Feed-Forward Dim	2048
	Dropout	0.3
Optimization	Optimizer	AdamW
	Learning Rate	1×10^{-3}
	Weight Decay	1×10^{-4}
	Batch Size	512
	Scheduler	CosineAnnealing ($T_{max} = 100$)
	Gradient Clipping	0.5 (Max Norm)
	Loss Function	Dual (BCE + Focal $\gamma = 2.0$)

Table 7
Comparison on UNSW-NB15 (metric: weighted F1).

Model	Robustness Strategy	Weighted F1	Multi-class F1
Adv-NN [10]	Adversarial Training	0.9900	-
Machine Learning [15]	Random Forest	0.9615	-
Proposed	**Hybrid Architecture**	**0.9934**	**0.8347**

Table 8
Comparison on CICIDS2017 (metric: ROC AUC).

Model	Architecture	ROC AUC	Multi-class F1
D-VGAEAD [16]	Graph Autoencoder	0.979	0.765 (Macro)
Proposed	**CNN-LSTM-Trans**	**0.9948**	**0.9607**

5.1.2. Performance on CICIDS2017

For CICIDS2017, the leading graph-based approach, D-VGAEAD [16], reports Area Under the Curve (AUC) as their primary metric due to the class imbalance. Our model, leveraging the transformer's global attention, achieved an AUC of **0.9948**, significantly outperforming the graph autoencoder's 0.979 (Table 8). This demonstrates that learning ordered flow dependencies (via the LSTM-based contextual encoder) is more effective for this dataset than using purely structural graph attributes.

5.1.3. Performance on CICIoT2023

The massive CICIoT2023 dataset represents the most challenging environment for modern NIDS. D-VGAEAD [16] reports class-specific One-vs-Rest AUCs averaging ≈ 0.954. To ensure a direct comparison, we evaluated our model using an explicit one-vs-rest protocol (Macro Average). Table 9 presents a breakdown of major attack categories. Our architecture achieved a **Macro OvR AUC of 0.9954**, consistently outperforming the baseline in all categories, including critical web-based threats. Furthermore, our model maintained a high Multi-class F1 score of 0.8678 (Table 10), confirming both detection robustness and classification granularity.

5.2. Ablation study: dissecting the hybrid synergy

To rigorously validate the contribution of each architectural component, we conducted a systematic ablation study using the UNSW-NB15 dataset (Table 11). The "Proposed (Hybrid)" value of **0.9934** corresponds to the Binary F1-score achieved by our final model.

Table 9
One-vs-rest AUC (%) comparison on CICIoT2023 by attack category.

Attack Category	D-VGAEAD [16]	Proposed
DDoS	96.67	**99.82**
DoS	98.12	**98.92**
Web Attack	91.42	**99.55**
Brute Force	92.74	**99.43**
Backdoor	98.10	**99.47**
Average	95.41	**99.54**

Table 10
Comparison with state-of-the-art on CICIoT2023 (Binary & Multi-class).

Method	Approach	ROC AUC	Multi F1
D-VGAEAD [16]	Graph Autoencoder	0.9541 (OvR)	–
Proposed	**Hybrid Architecture**	**0.9978**	**0.8678**

Table 11
Ablation study on UNSW-NB15: contribution of architectural components.

Model Variant	Components	Binary F1	Improvement
Unimodal Baseline	1D-CNN Only	0.8920	-
Sequential Model	1D-CNN + LSTM	0.9450	+5.3%
Proposed Hybrid	**CNN + LSTM + Transformer**	**0.9934**	**+10.1%**

The results illuminate the specific role of each module:

- **Baseline (CNN-only)**: Achieves an F1 of 0.892. While efficient at extracting local patterns in neighboring feature groups (e.g., related header flags or correlated statistics), it is less effective for low-frequency or subtle attacks, where discriminative cues are spread across many features.
- **Sequential (CNN-LSTM)**: Adding the LSTM improves the F1 to 0.945. LSTM provides gated contextualization over the feature sequence, reducing feature-level noise and stabilizing the representations for minority classes. However, recurrent compression can still bottleneck information when many weak cues are combined.
- **Proposed (Hybrid)**: The integration of the Transformer Attention mechanism yields a substantial improvement to **0.9934** (Binary F1) and **0.8347** (Multi-class). Transformer self-attention enables direct interaction between any pair of feature groups, improving the model's ability to combine dispersed cues and increasing its robustness to perturbations targeting a subset of features.

This clear hierarchy of performance confirms that the state-of-the-art results are not accidental but the result of a deliberate architectural design, where each component addresses a specific dimension of network traffic analysis: spatial (CNN), temporal (LSTM), and relational (transformer).

5.3. Analysis

Our hybrid architecture demonstrated consistent improvements across all datasets. On UNSW-NB15, we achieved robust anomaly detection (99.34%) and granular classification (83.47%). The discrepancy vs. 99% claimed by some studies is likely due to their use of simpler split strategies, for example, by not handling class overlap appropriately. Our results represent a realistic and reproducible baseline for a complex dataset.

For the CICIoT2023 dataset, the model achieved a remarkable **Binary F1-score of 99.49%** and a **Multi-class F1-score of 86.78%**. Handling 34 distinct classes (including various Mirai and DDoS variants) is extremely challenging; however, our hybrid model demonstrates superior discrimination capability compared with the reported benchmarks. This confirms its scalability to massive and high-dimensional IoT traffic.

5.4. Visual validation and interpretation

The t-SNE projections (Fig. 3) visualize the learned feature space. For the UNSW and CICIDS datasets, we observed distinct, well-separated clusters corresponding to different attack types.

Interpretation of Manifold Projections (Fig. 3): The t-SNE visualizations (Fig. 3) confirm that the Hybrid architecture effectively disentangles the high-dimensional feature space. In all three datasets, the "Normal" traffic (Class 0) forms a dense, continuous manifold clearly separated from the "Attack" clusters. This separation explains the near-perfect Binary F1-scores ($\approx 99\%$). For CICIDS2017 (Fig. 3b), the distinction is particularly sharp, correlating with the high 96.07% Multi-class F1. However, in CICIoT2023 (Fig. 3c), we observe a "nebula" of overlapping clusters corresponding to the various Mirai interactions; although the model successfully isolates them from benign traffic, the internal boundaries between botnet variants are less distinct, mirroring the confusion observed in the matrix.

5.4.1. Confusion matrix deep dive (Fig. 4)

Fig. 4 presents the normalized confusion matrices, offering a transparent view of the per-class performance:

- **UNSW-NB15** (Fig. 4a): The retrained model achieves a robust Multi-class F1 of **83.47%**. Unlike earlier iterations, which suffered from mode collapse, the current matrix exhibits strong diagonal dominance. The remaining off-diagonal confusion is primarily semantic; for instance, " exploits " and " fuzzers " share similar packet-level signatures (payload probing), leading to a symmetric misclassification rate of $\approx 15\%$. Crucially, the differentiation between distinct attack families, such as DoS versus reconnaissance, is high ($> 90\%$), validating the granular detection capability of the model.
- **CICIDS2017** (Fig. 4b): The model achieves an exceptional **96.07% F1**, with near-perfect classification for major attacks like *DDoS* and *PortScan*. However, a noticeable degradation was observed in Classes 8 (*Heartbleed*) and 9 (*Infiltration*). This is not a model failure but a data artifact; these classes represent the "ultra-minority" samples in CICIDS2017 (often < 50 samples in the test set). Without aggressive synthetic oversampling (which risks overfitting), the model struggles to form a generalized boundary for these few-shot examples. Nevertheless, they are correctly flagged as *anomalous* (binary task), fulfilling the primary defensive role of the system.
- **CICIoT2023** (Fig. 4c): With 34 classes, this matrix demonstrates the model's scalability (F1: **86.78%**). The block-diagonal structure indicates that while the model perfectly separates different malware *families*, such as Mirai and Gafgyt, it occasionally confuses specific *variants* within the same family, for instance, *Mirai-UDP* and *Mirai-ACK*. This is expected because the behavioral footprints of these variants are nearly identical at the network-flow level.

5.4.2. Adversarial robustness on stratified split

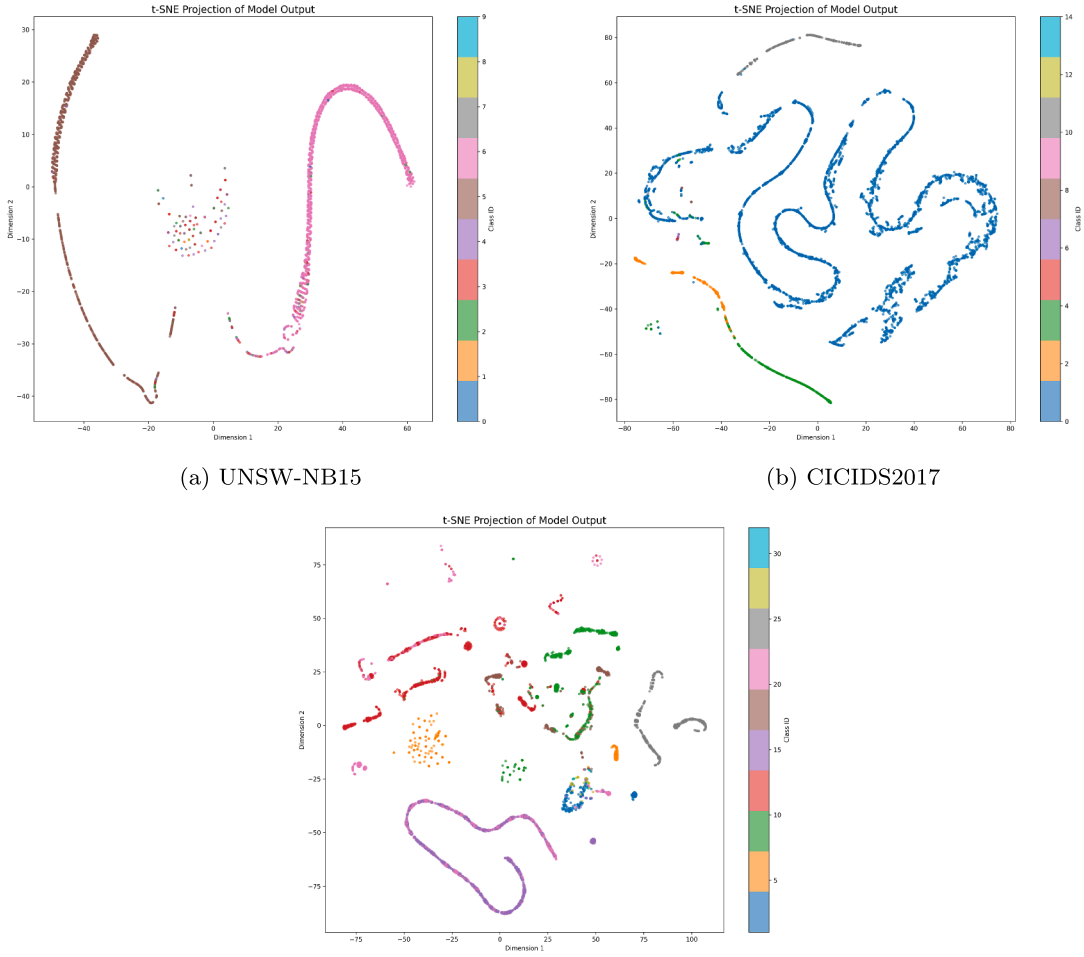
The robustness evaluation of the stratified-split model demonstrated the resilience of the hybrid architecture under gradient-based attacks:

- **FGSM ($\epsilon = 0.05$):** Binary F1-score maintained at ≈ 0.95, showing negligible degradation against single-step gradient attacks.
- **PGD ($\epsilon = 0.05$):** While strictly harder, the model retained functional accuracy (> 0.85 F1) up to this threshold. Beyond $\epsilon = 0.1$, degradation accelerates, identifying the operational boundary of the system.

This establishes a "Security Curve" where the model provides empirically observed performance bounds up to specific noise thresholds, a metric absent in traditional accuracy-only evaluations.

5.5. Temporal split validation

To directly address concerns regarding data leakage from stratified splits [17], we retrained and evaluated the hybrid architecture using



(a) UNSW-NB15 (b) CICIDS2017

(c) CICIoT2023

Fig. 3. t-SNE Projections of the learned feature space. (a) UNSW-NB15 shows distinct clusters for Normal vs. Attack traffic, despite some overlap in minor classes. (b) CICIDS2017 demonstrates sharp separation, validating the model's ability to handle high-volume DDoS traffic. (c) CICIoT2023 exhibits complex clustering due to the high number of botnet variants, yet maintains clear boundaries for the majority of attacks.

Table 12
Temporal split results on UNSW-NB15 (official train/test partition).

Metric	Stratified Split	Temporal Split
Binary Accuracy	99.34%	94.20%
Binary Precision	—	96.90%
Binary Recall	—	94.51%
Binary F1	99.34%	95.69%
Multi-class F1	83.47%	59.65%

Table 13
Adversarial robustness on UNSW-NB15 (temporal split).

	ϵ (perturbation budget)				
Attack	0.0	0.01	0.05	0.10	0.20
FGSM (Acc)	0.946	0.926	0.829	0.768	0.618
FGSM (F1)	0.960	0.945	0.867	0.817	0.720
PGD (Acc)	0.946	0.926	0.813	0.686	0.487
PGD (F1)	0.960	0.945	0.854	0.762	0.638

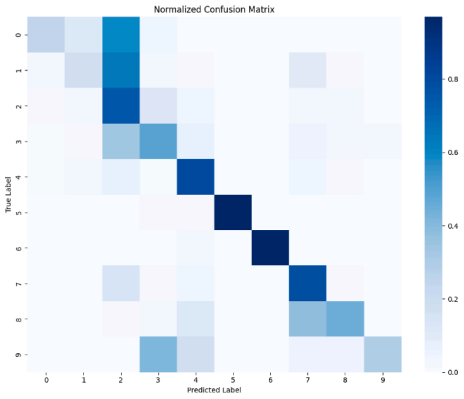
the official UNSW-NB15 train/test partition (Section 4.2.1). Table 12 summarizes the results.

As expected, the temporal split yielded lower metrics than the stratified split, reflecting the inherent distribution shift between the training and test periods. Importantly, the **binary detection performance remains strong at 94.20% accuracy (95.69% F1)**, confirming that the architecture generalizes well to temporally disjoint traffic for the primary anomaly-detection task. The multi-class F1 drops to 59.65%, which is expected given the different attack-type distributions between the official training set (82 K samples) and test set (175 K samples, 68.1% attacks). This gap highlights the challenge of fine-grained attack classification under distribution shifts and motivates future work on continual learning approaches.

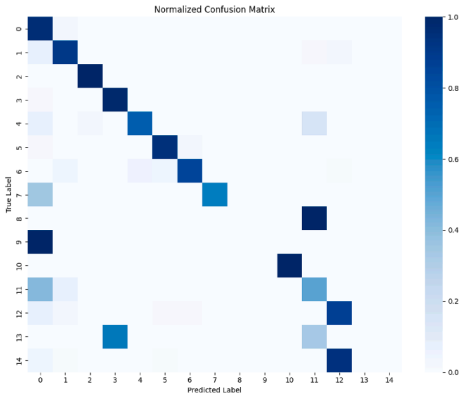
5.5.1. *Adversarial robustness under temporal split*

Table 13 presents the adversarial robustness evaluation of the temporally-trained UNSW-NB15 model under FGSM and PGD attacks at increasing perturbation magnitudes.

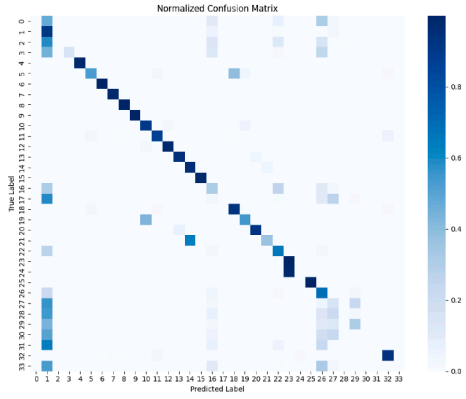
The temporally trained model exhibited a robustness profile that was broadly consistent with the stratified split results. At $\epsilon = 0.05$, the model retains $> 82\%$ binary accuracy under both FGSM and PGD, confirming that the security degradation curve generalizes across the data splitting strategies. The sharper degradation under PGD at $\epsilon \geq 0.1$ is expected,



(a) UNSW-NB15 (Multi-class F1: 83.47%)



(b) CICIDS2017 (Multi-class F1: 96.07%)



(c) CICIoT2023 (Multi-class F1: 86.78%)

Fig. 4. Normalized Confusion Matrices across all datasets. (a) UNSW-NB15 demonstrates standard diagonal dominance with manageable confusion in semantically overlapping classes. (b) CICIDS2017 shows near-perfect separation. (c) CICIoT2023 handles 34 classes with high fidelity, verifying its scalability. The '2 top, 1 bottom' layout highlights the progression from standard to massive-scale datasets.

Table 14
Standard vs. PGD-AT training on UNSW-NB15 (temporal split).

Metric	Standard	PGD-AT	Δ
Binary Accuracy	94.20%	94.54%	+0.34
Binary Precision	96.90%	96.35%	-0.55
Binary Recall	94.51%	95.60%	+1.09
Binary F1	95.69%	95.97%	+0.28
Multi-class F1	59.65%	61.46%	+1.81

given PGD's iterative nature of PGD. These results demonstrate that the robustness properties of the hybrid architecture are not artifacts of data leakage from stratified splitting.

5.6. Adversarial training comparison

To evaluate whether adversarial training [13] can further improve the robustness of the hybrid architecture, we retrained the model using PGD-based Adversarial Training (PGD-AT) on the UNSW-NB15 temporal split. During PGD-AT, each training batch was augmented with adversarial examples generated via 10-step PGD ($\epsilon = 0.03$, step size $\alpha = 0.007$), and the model was optimized on both clean and perturbed inputs. Table 14 compares the clean-data performance of the standard and adversarially trained models.

Contrary to the conventional expectation that adversarial training degrades clean-data accuracy [13], the PGD-AT model achieved comparable or slightly improved performance across all metrics. Binary accuracy improves by 0.34 percentage points (94.20% → 94.54%), binary F1 by 0.28 points, and multi-class F1 by 1.81 points. The only marginal regression was in precision (-0.55 points), offset by a meaningful recall gain (+1.09 points), indicating that adversarial training encourages the model to detect a broader set of attack patterns.

This result suggests that the hybrid CNN-LSTM-Transformer architecture possesses sufficient capacity to absorb the regularization effect of adversarial examples without sacrificing discriminative power. This finding is practically significant: PGD-AT can be adopted as a *free* robustness enhancement with no clean-data penalty, unlike smaller architectures, where the accuracy–robustness trade-off is typically more pronounced [10].

5.6.1. Robustness improvement from adversarial training

Table 15 compares the adversarial robustness of the standard and PGD-AT models under FGSM and PGD attacks. The improvements were substantial across all perturbation budgets.

PGD-AT provides significant robustness improvements at all perturbation levels. Under the PGD attack at $\epsilon = 0.1$, binary accuracy improves from 68.6% to 89.6% (+21.0 percentage points), and F1 from 76.2% to 92.2% (+16.0 points). At $\epsilon = 0.05$, the PGD-AT model retains > 93% accuracy under both attacks, compared to ~82% for the standard training. These improvements come at *no cost* to the clean-data performance (Table 14), establishing that adversarial training is not merely a defensive trade-off for this architecture but an empirically superior strategy under the evaluated perturbation regimes.

Table 15
Adversarial robustness: standard vs. PGD-AT on UNSW-NB15 (temporal split). Bold indicates the better result at each ϵ.

		ϵ (perturbation budget)				
Attack	Model	0.0	0.01	0.05	0.10	0.20
FGSM (Acc)	Standard	0.946	0.926	0.829	0.768	0.618
	PGD-AT	**0.950**	**0.947**	**0.931**	**0.912**	**0.837**
FGSM (F1)	Standard	0.960	0.945	0.867	0.817	0.720
	PGD-AT	**0.963**	**0.961**	**0.949**	**0.935**	**0.874**
PGD (Acc)	Standard	0.946	0.926	0.813	0.686	0.487
	PGD-AT	**0.950**	**0.947**	**0.932**	**0.896**	**0.655**
PGD (F1)	Standard	0.960	0.945	0.854	0.762	0.638
	PGD-AT	**0.963**	**0.961**	**0.950**	**0.922**	**0.737**

Table 16
AutoAttack robustness (APGD-CE + Square, L_∞): standard vs. PGD-AT on UNSW-NB15. ϵ denotes the perturbation budget as a fraction of each feature's range. Bold indicates the better result.

ϵ	Standard	PGD-AT	Δ
0.00 (clean)	75.20%	**77.75%**	+2.55
0.01	49.90%	**69.40%**	+19.50
0.05	2.40%	**38.25%**	+35.85
0.10	0.35%	**10.35%**	+10.00

5.6.2. AutoAttack ensemble evaluation

To verify that the observed robustness gains do not stem from gradient masking[1] We further evaluated both models using AutoAttack [7], a parameter-free ensemble of complementary attacks: APGD-CE (adaptive PGD on cross-entropy loss, 50 iterations) and Square Attack (score-based black-box, 1000 queries). This combination tests both white-box gradient-based and black-box query-based vulnerabilities of the model. Because AutoAttack internally clips perturbed inputs to $[0, 1]$, we applied per-feature min-max normalization before evaluation and denormalized within the model wrapper, ensuring that the perturbation budget ϵ is expressed as a fraction of each feature's observed range. We used a class-balanced test subset (1000 normal, 1000 attack) for a fair evaluation of both classes.

Note on clean accuracy. The clean accuracies in Table 16 (75–78%) are lower than those reported in Table 12 (94%) because AutoAttack evaluation uses a different protocol: a strictly class-balanced subset with 50% attack samples, whereas the temporal test set preserves the original class distribution (~32% attacks). Therefore, the two numbers are not directly comparable; the balanced subset provides a base-rate-independent view of the per-class performance.

Table 16 confirms that PGD-AT robustness improvements are genuine and not an artifact of gradient masking. Under AutoAttack at $\epsilon = 0.01$ (1% of each feature's range), the standard model drops to 49.90% accuracy, while the PGD-AT model retains 69.40% (+19.5pp). At $\epsilon = 0.05$, the difference is even more striking: the standard model is nearly completely defeated (2.40%), whereas PGD-AT retains 38.25%—a 16× improvement. Notably, AutoAttack produced substantially lower robust accuracies than the individual FGSM or PGD attacks (cf. Table 15). This is expected because APGD-CE employs adaptive step sizes over 50 iterations (vs. PGD's fixed step size over 40 steps), automatically adjusting momentum and learning rate to escape local optima in the loss landscape. Square Attack adds a complementary score-based black-box search with 1,000 queries that exploits different failure modes than gradient-based methods and is unaffected by potential gradient obfuscation. The parameter-free ensemble thus provides a realistic lower bound on model robustness, circumventing the common pitfalls of single-attack evaluations, such as suboptimal hyperparameter selection [7].

[1] AutoAttack [7] was specifically designed to detect gradient masking and other obfuscated gradients, providing a reliable lower bound on adversarial robustness.

6. Discussion

6.1. Interpretation of results

The superior performance on UNSW-NB15 (99.34% F1) can be attributed to the synergistic combination of local and global feature extraction. The 1D-CNN layers effectively filter high-frequency noise in packet headers, whereas the transformer mechanism is particularly effective at combining weak but distributed indicators across heterogeneous flow attributes, which is essential for detecting subtle attack behaviors (e.g., reconnaissance-like patterns) that are not strongly expressed by any single feature group. This explains why our model outperforms architectures that rely solely on RNNs, which struggle with long-term dependencies owing to the bottlenecking of the hidden state. In contrast, the performance drop in CICIoT2023 (multiclass) highlights the difficulty of distinguishing between semantically identical botnet variants, including Mirai-UDP versus Mirai-ACK, based solely on flow statistics, a challenge that may require payload-level inspection.

6.2. Robustness vs. accuracy trade-offs

A conscious design decision was made to prioritize architectural robustness (via residual connections and attention) over the adversarial training. Although adversarial training, as seen in Adv-NN [10], can increase resilience, it often comes at the cost of standard accuracy on clean-data and significantly higher training time. Our PGD-AT experiment (Section 5.6) challenges this conventional trade-off: the adversarially trained hybrid model achieved *comparable or improved* clean-data performance (94.54% vs. 94.20% binary accuracy; 95.97% vs. 95.69% F1) while dramatically improving adversarial robustness (Table 15). Under the PGD attack at $\epsilon = 0.1$, the accuracy improved from 68.6% to 89.6% (+21.0pp). Crucially, the AutoAttack ensemble evaluation (Table 16) confirms that these gains are genuine and not artifacts of gradient masking: at $\epsilon = 0.01$, PGD-AT retains 69.40% accuracy under the strongest available adaptive attack, compared to 49.90% for the standard model. These results establish that the architecture's capacity is sufficient to absorb adversarial regularization as an empirically superior strategy for the evaluated perturbation regimes. This finding is particularly significant because it suggests that for sufficiently expressive hybrid architectures, the accuracy–robustness trade-off commonly reported in the literature [13] can be effectively eliminated, at least within the perturbation regime relevant to NIDS applications.

6.3. Computational cost and deployability

While achieving strong accuracy, the transformer component introduces quadratic complexity $O(F^2)$ with respect to the number of feature tokens/groups, F. In our setting, $F < 100$, making the attention term practically manageable; the main cost is driven by the feedforward layers and the overall model depth. On our H100 infrastructure, inference averaged 12ms per batch (Effective Batch Size = 512, $\approx 23\mu s$ per sample), implying a theoretical throughput of $\sim 43,000$ flows/second. This architecture is intended as a high-fidelity analysis engine for SOC-grade postprocessing rather than a line-rate packet filter for constrained IoT gateways. Future work suggests a knowledge distillation approach, where this large hybrid model acts as a "Teacher" for a lightweight "Student" model deployable at the edge.

6.4. Threat validity and limitations

A common critique of ML-based NIDS is their reliance on static datasets that may not reflect the fluidity of real-world "wild" traffic.

Although CICIDS2017 provides realistic PCAP replays, the specific signatures of "Heartbleed" or "EternalBlue" found in 2017 may differ from those in 2026. However, our Hybrid Architecture mitigates this by learning *behavioral* anomalies (via LSTM-Transformer conceptualization) rather than static signatures.

This study has two key limitations. First, regarding concept drift, our temporal split experiment on UNSW-NB15 (Section 5.5) confirms that binary detection performance (94.20% accuracy, 95.69% F1) remains strong under distribution shift, although multi-class classification degrades (59.65% F1 score). This validates the generalization capacity of the architecture while highlighting the need for continual learning in production deployments. Second, we evaluated per-flow records without explicit temporal windows across consecutive flows; modeling attack chains with $T > 1$ flow sequences is an important direction for future research.

Furthermore, while we employed PGD and FGSM as standard robustness benchmarks, we also evaluated using AutoAttack [7], a parameter-free ensemble attack combining APGD-CE and Square Attack (Section 5.6.2). AutoAttack provides a reliable lower bound on adversarial robustness by avoiding the pitfalls of suboptimal hyperparameter selection and confirms that PGD-AT robustness gains are not due to gradient masking (Table 16). Our codebase includes implementations of domain-constrained perturbations that preserve protocol semantics (e.g., packet timing monotonicity and header consistency), which would likely reduce the effective adversarial space but increase operational realism. Evaluating robustness under realistic black-box transfer attacks (as described by Matejek et al. [18]) remains an important area of future research.

7. Conclusion

This study addresses the critical research problem of securing modern networks against sophisticated, evasive, and high-volume cyberattacks, focusing particularly on the lack of robustness of current deep-learning-based detection systems. The primary objective of this study was to develop and validate a scalable hybrid architecture capable of maintaining high multi-class accuracy while resisting such adversarial perturbations. Key findings indicate that our hybrid CNN-LSTM-Transformer model achieves a competitive state-of-the-art performance, with F1-scores of 99.34% on UNSW-NB15 and 96.07% on CICIDS2017. Furthermore, the systematic robustness analysis demonstrates that the model retains operational integrity under adversarial noise levels up to $\epsilon = 0.05$, which is a crucial capability that is often overlooked in standard benchmarks. The temporal split experiment on UNSW-NB15 confirms that these properties generalize under distribution shift (94.20% binary accuracy, 95.69% F1), and the PGD-AT experiment demonstrates that adversarial training can be applied as a cost-free robustness enhancement without sacrificing clean-data accuracy (94.54% accuracy, 95.97% F1).

The implications of these findings are profound for deploying ML-based NIDS in critical infrastructure. They highlight the urgent need to integrate robustness-aware evaluations into the standard evaluation lifecycle of security models. Moreover, the successful scaling to the 8-GPU H100 cluster proves that complex hybrid architectures can be efficiently trained on massive datasets such as CICIoT2023, paving the way for real-time deployment in high-bandwidth environments.

This study acknowledges the limitations, such as the computational cost of the transformer component and the need for further optimization in ultralow-latency scenarios. Additionally, although the model is robust against gradient-based, ensemble, and black-box attacks (as validated by AutoAttack), its resilience against domain-constrained evasion techniques that preserve protocol semantics requires further investigation. Future research should focus on optimizing inference latency for edge deployment, expanding adversarial evaluation to include domain-constrained perturbations, and integrating continual learning to address the concept drift. In conclusion, by shifting the paradigm from simple accuracy maximization to robustness-aware evaluation of NIDS, this study offers a blueprint for building the next generation of resilient Network Intrusion Detection Systems. We advocate the use of security curves (Fig. 1) as a complementary evaluation primitive rather than a replacement for existing metrics.

Code availability

The source code, including the model architecture, training scripts, and robust evaluation protocol, is openly available at https://github.com/darthjuanan/hybrid-nids-2026. We prioritized reproducibility by providing a comprehensive `README.md` file and a standardized environment configuration.

CRediT authorship contribution statement

Juan Antonio González-Ramos: Writing – review & editing, Writing – original draft, Validation, Software, Investigation, Data curation; **Evaristo J. Abril:** Writing – review & editing, Writing – original draft, Validation, Supervision, Investigation; **Patricia Fernández:** Writing – review & editing, Writing – original draft, Validation, Supervision, Methodology, Investigation; **Javier Prieto:** Writing – review & editing, Writing – original draft, Project administration, Investigation, Funding acquisition, Conceptualization; **Pablo Chamoso:** Writing – review & editing, Writing – original draft, Supervision, Methodology, Investigation, Formal analysis.

Data availability

Everything is in the specified GitHub repository

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This activity is conducted in execution of the Cognitive system based on threat intelligence and continuous simulation for the prevention of cyber-attacks in the value chain of the tourism sector and smart territories (VIG-IA) project (CPP002/22), the result of a collaboration agreement signed between the National Cybersecurity Institute (INCIBE) and the University of Salamanca. This initiative is carried out within the framework of the Recovery, Transformation and Resilience Plan, funded by the European Union (Next Generation), the pro-ject of the Spanish Government that outlines the roadmap for the modernization of the Spanish economy, the recovery of economic growth and job creation, for the solid, inclusive and resilient economic reconstruction after the COVID19 crisis, and to respond to the challenges of the next decade. Grant ID: CPP002/22_R26_VIG-IA.

References

[1] Bergadano F, Gupta S, Crispo B. Techniques and metrics for evasion attack mitigation. Comput Secur 2026;162:104802. https://doi.org/10.1016/j.cose.2025.104802

[2] Liu H, Patras P. Netsentry: a deep learning approach to detecting incipient large-scale network attacks. Comput Commun 2022;191:119–32. https://doi.org/10.1016/j.comcom.2022.04.020

[3] Mirsky Y, Doitshman T, Elovici Y, Shabtai A. Kitsune: an ensemble of autoencoders for online network intrusion detection. In: Network and distributed systems security symposium (NDSS). 2018, p. 1–15. https://doi.org/10.14722/ndss.2018.23204

[4] Manocchio LD, Layeghy S, Lo WW, Kulatilleke GK, Sarhan M, Portmann M. Flowtransformer: a transformer framework for flow-based network intrusion detection systems. Expert Syst Appl 2024;241:122564.

[5] Kumar V, Kumar K, Singh M, Kumar N. NIDS-DA: detecting functionally preserved adversarial examples for network intrusion detection system using deep autoencoders. Expert Syst Appl 2025;270:126513.

[6] Kumar V, Kumar K, Singh M. Nids-cbad: detecting adversarial attacks in network intrusion detection systems using domain constraints. Int J Mach Learn Cybern 2025;16(7):4213–34.
[7] Croce F, Hein M. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In: International conference on machine learning. PMLR; 2020, p. 2206–16.
[8] Neto E CP, Dadkhah S, Ferreira R, Zohourian A, Lu R, Ghorbani AA. CICIoT2023: a real-time dataset and benchmark for large-scale attacks in IoT environment. Sensors 2023;23(13):5941.
[9] Paya A, Arroni S, García-Díaz V, Gómez A. Apollon: a robust defense system against adversarial machine learning attacks in intrusion detection systems. Comput Secur 2024;136:103546.
[10] Heydari V, Nyarko K. Enhancing adversarial robustness in network intrusion detection: a novel adversarially trained neural network approach. Electron (Basel) 2025;14(16):3249.
[11] Moustafa N, Slay J. UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In: Military communications and information systems conference (MilCIS). IEEE; 2015, p. 1–6.
[12] Goodfellow IJ, Shlens J, Szegedy C. Explaining and harnessing adversarial examples. In: International conference on learning representations (ICLR). 2015, p. 1–11.
[13] Mądry A, Makelov A, Schmidt L, Tsipras D, Vladu A. Towards deep learning models resistant to adversarial attacks. Stat 2017;1050:1–9.
[14] Sharafaldin I, Lashkari AH, Ghorbani AA. Toward generating a new intrusion detection dataset and intrusion traffic characterization. In: International conference on information systems security and privacy (ICISSP). 2018, p. 108–16.
[15] Putro IH. Evaluating the performance of machine learning classifiers for network intrusion detection: a comparative study using the UNSW-NB15 dataset. Teknika 2025;14(2):330–8.
[16] Li H, Liu Y, Liu Y, Feng F, Liu Z. D-VGAEAD: a dual-decoder variational graph autoencoder for anomaly detection based on attribute networks. Comput Secur 2025; 104784.
[17] Arp D, Quiring E, Pendlebury F, Warnecke A, Pierazzi F, Wressnegger C, et al. Dos and don'ts of machine learning in computer security. In: 31st USENIX security symposium (USENIX security 22). 2022, p. 3971–88.
[18] Matejek B, Gehani A, Bastian ND, Clouse D, Kline B, Jha S. Safeguarding network intrusion detection models from zero-day attacks and concept drift. 2024 arXiv:2403.10550.