Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Explainable Dual Branch Attention Network for Industrial IoT Intrusion Detection

Vanlalruata Hnamte

vanlalruatahnamte@mzu.edu.in

Mizoram University

Chhakchhuak Laldinsanga

Mizoram University

Chhakchhuak Lalduhawma

Mizoram University

Jamal Hussain

Mizoram University

Research Article

Keywords:

Posted Date: November 11th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11456417/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Explainable Dual Branch Attention Network for Industrial IoT Intrusion Detection

Vanlalruata Hnamte[1*], Chhakchhuak Laldinsanga[1], Chhakchhuak Lalduhawma[1], Jamal Hussain[1]

[1*]Department of Mathematics & Computer Science, Mizoram University, Tanhril, Aizawl, 796004, Mizoram, India.

*Corresponding author(s). E-mail(s): vanlalruatahnamte@mzu.edu.in;
Contributing authors: mzu240001135@mzu.edu.in;
mzu250002682@mzu.edu.in; mzut035@mzu.edu.in;

Abstract

The rapid expansion of the Industrial Internet of Things (IIoT) has broadened the cyberattack surface across critical infrastructure. Conventional intrusion detection systems often process isolated flow metrics without capturing local cross-feature correlations or sequential feature dynamics, while operating as opaque black boxes. To address these challenges, we propose CP-XNet, a lightweight dual-branch deep learning framework tailored for edge-deployable cyber-physical intrusion detection. Under this architectural framing, spatial denotes localised cross-attribute correlations along the engineered feature axis, while temporal represents sequential dependencies across ordered intra-flow functional feature segments rather than multi-packet or inter-flow time series. CP-XNet couples a feature-axis 1D-Convolutional Neural Network (CNN) for localised cross-feature extraction with a Bidirectional Gated Recurrent Unit (BiGRU) for structured feature-segment sequence modelling, dynamically fused via a compact two-token Multi-Head Self-Attention (MHSA) mechanism. Dual-tier visual explainability (SHAP and LIME) provides protocol-level transparent decision support. Evaluated across three contemporary cyber-physical benchmark testbeds under binary detection and multi-class threat attribution, CP-XNet achieves: **99.49%** binary and **94.39%** multi-class accuracy on ToN-IoT Industrial Telemetry; **99.97%** binary and **99.84%** multi-class accuracy on InSDN Software-Defined Networking; and **95.75%** binary and **93.65%** multi-class accuracy on CIC-IIoT-2025 Multi-Device traffic. Across all benchmarks, batch-amortised forward-pass latency on GPU hardware remains strictly bounded between **0.0077** and **0.0128** ms per sample (**0.0710** ms on CPU at batch size $\boldsymbol{B = 256}$, alongside streaming single-sample latencies of **0.0820** ms GPU and **0.4100** ms CPU),

translating to an estimated nominal power-envelope energy proxy of **0.077**–**0.128** mJ on GPU (**0.710** mJ on CPU) under a nominal 10-W edge envelope. A prospective IEC 61850 digital substation deployment architecture illustrates passive edge integration boundaries.

Keywords: Industrial IoT Security, Cyber-Physical Telemetry, Multi-Benchmark Evaluation, Multi-Head Attention, Explainable AI, Software-Defined Networking, Power-Envelope Energy Proxy, IEC 61850 Substation Architecture.

1 Introduction

The global transition towards decarbonised and automated industrial infrastructure has accelerated the deployment of smart energy grids and Industrial Internet of Things (IIoT) ecosystems. Modern industrial operations integrate advanced cyber-physical systems (CPS), distributed energy resources (DERs), smart substations, and ubiquitous telemetry sensors [1, 2]. These interconnected cyber-physical assets rely on high-speed industrial communication protocols, such as Modbus, IEC 61850 Generic Object Oriented Substation Events (GOOSE), and MQTT, alongside digital twin and physical AI frameworks to facilitate real-time monitoring, telemetry feedback, and supervisory control and data acquisition (SCADA) [3, 4].

Notwithstanding their transformative benefits, the convergence of operational technology (OT) with public information and communication technology (ICT) networks has severely amplified cyber vulnerabilities [5, 6]. Critical infrastructure is increasingly targeted by sophisticated cyber threats, ranging from distributed denial-of-service (DDoS) flooding and malicious injection to advanced persistent threats (APTs) and ransomware campaigns [7–9]. Adversaries can manipulate sensor telemetry, forge control commands, or desynchronise phase measurements, triggering cascading system failures, physical blackouts, and widespread economic disruption [10, 11].

Intrusion detection systems (IDSs) constitute a vital secondary line of defense in industrial and smart grid networks [12, 13]. Over the past decade, deep learning (DL) methodologies have largely superseded traditional signature-based detectors owing to their capacity to automatically learn complex, high-dimensional representations from raw traffic flows [14–16]. Convolutional neural networks (CNNs), long short-term memory (LSTM) networks, gated recurrent units (GRUs), and autoencoders have been widely investigated for anomaly classification [17–20].

Despite these advancements, existing industrial network intrusion detection frameworks exhibit three fundamental limitations:

1. **Cyber-Physical Disconnect & Unimodal Processing:** Contemporary models predominantly process isolated packet header statistics without considering the spatial inter-dependencies across contiguous flow attributes or structured dependencies across functional protocol segments [21–23]. In integrated cyber-physical environments, attacks manifest across multiple protocol layers concurrently; however, existing detectors frequently treat flow records as flat, unaligned feature

vectors. Here, we emphasise that "spatial" refers specifically to the structured feature axis rather than physical grid geometry, and recurrent sequencing operates strictly across ordered functional feature blocks within an individual flow rather than tracking successive physical timestamps or multi-packet sliding windows.

2. **Resource, Latency, and Energy Constraints at the Substation Edge:** Intelligent Electronic Devices (IEDs) and industrial edge gateways possess stringent computational, memory, and energy budgets [24–26]. Heavyweight deep ensembles and full-sequence transformer models impose substantial computational overheads [13, 21], conflicting with stringent real-time protection and isolation requirements in digital substations.
3. **Black-Box Opacity and Lack of Operator Trust:** Conventional deep learning classifiers deliver high predictive accuracy but fail to elucidate *why* an anomalous prediction was generated [27–29]. In safety-critical power-system environments, control centre operators cannot initiate disruptive mitigation or dispatch emergency maintenance crews based upon uninterpretable probability distributions.

To directly address these challenges, this study introduces **CP-XNet** (Cyber-Physical eXplainable Network), a lightweight dual-branch deep learning framework tailored for edge-oriented industrial network intrusion detection. Rather than proposing individual components in isolation, CP-XNet investigates the principled integration of feature-axis convolutional representations, bidirectional recurrent gating over structured feature segments, compact two-token multihead self-attention (MHSA) fusion, and dual-tier visual explainability (SHAP and LIME). To provide clear methodological focus, this investigation is guided by three explicit research questions:

- **RQ1 (Branch Synergy):** Can joint feature-axis convolutional modeling and bidirectional recurrent feature-segment sequence modeling capture complementary local and macro-level protocol representations to improve multi-class intrusion detection over single-branch baselines?
- **RQ2 (Dynamic Attention Fusion vs Direct Concatenation):** Does multihead self-attention fusion provide superior representation weighting and discriminative capability compared with simple feature concatenation, particularly under severe class imbalance?
- **RQ3 (Interpretability & Computational Footprint):** Can an integrated post-hoc explainability framework (SHAP and LIME) deliver consistent, verifiable feature attribution for network operators while maintaining a compact parameter footprint and low forward-pass latency?

The principal contributions of this work are summarised as follows:

- **Dual-Branch Spatial and Sequential Feature-Segment Architecture:** We design CP-XNet, which uniquely couples 1D convolutions across a structured feature axis (capturing local cross-feature correlations) with bidirectional recurrent gating (BiGRU) over ordered functional feature segments (capturing macro-level dependencies across protocol blocks), fused dynamically via compact two-token MHSA. Crucially, the spatio-temporal formulation operates strictly at the intra-flow level: the convolutional branch acts spatially across local attribute neighbourhoods,

while the recurrent branch models the sequential progression of functional protocol blocks within an individual flow record, avoiding artificial inter-flow adjacency assumptions.

- **Comprehensive Visual & Game-Theoretic Explainability:** We establish a dual XAI pipeline combining global SHAP feature attribution with local LIME instance explanations, providing human operators with transparent, verifiable protocol-level decision evidence.
- **Rigorous Computational Complexity and Power-Envelope-Based Energy Proxy:** We formulate the analytical complexity of each architectural constituent, benchmark batch-amortised forward-pass latency (0.0077–0.0128 ms/sample on GPU, 0.0710 ms on CPU at $B = 256$) alongside streaming single-sample latency (0.0820 ms GPU, 0.4100 ms CPU at $B = 1$), and derive a normalised power-envelope-based energy proxy (0.077–0.128 mJ/sample GPU, 0.710 mJ CPU amortised; 0.820 mJ GPU, 4.100 mJ CPU streaming, estimated under a nominal 10-W edge envelope).
- **Multi-Benchmark Empirical Validation and Ablation Study:** We perform comprehensive evaluations across three contemporary cyber-physical benchmarks (ToN-IoT, InSDN, and CIC-IIoT-2025) under five independent seeds (94.39% $\pm$ 0.08% multi-class accuracy on ToN-IoT), demonstrating strong balanced performance over feedforward and single-branch baselines alongside an honest analysis of extreme minority-class operational trade-offs.

The remainder of this paper is organised as follows. Section 2 provides an exhaustive review of the recent literature and establishes our taxonomy of related works. Section 3 details the proposed CP-XNet framework, its mathematical formulation, and its algorithmic execution pipeline. Section 4 outlines the visual explainability methodology. Section 5 analyses the computational time complexity and power envelope-based energy proxy. Section 6 describes the experimental benchmark, data preprocessing, and evaluation metrics. Section 7 presents the experimental findings, baseline comparisons, and minority-class analysis. Section 8 presents an architectural ablation study. Section 9 discusses the XAI case studies and stability analyses. Section 10 presents a detailed discussion of the prospective IEC 61850 deployment architecture, operational boundaries, and comprehensive study limitations. Finally, Section 11 concludes the paper and outlines the future directions.

2 Related Work & Literature Taxonomy

The application of machine learning (ML) and DL to IIoT, cyber-physical infrastructure, and critical energy systems has generated substantial scholarly interest in recent years. To contextualise our contributions, we systematically reviewed recent advances across five thematic paradigms.

2.1 Deep Learning for IoT/IIoT Intrusion Detection

Vinayakumar et al. [14] evaluated deep neural networks across multiple legacy datasets, demonstrating that layered representations substantially outperform classical shallow classifiers in capturing non-linear network patterns. Hairab et al. [17] proposed an anomaly detection framework using regularised CNNs to tackle zero-day attacks, highlighting the necessity of preventing overfitting in sparse attack distributions. Babar et al. [18] introduced a hybrid DL architecture combining convolutional feature extractors with recurrent units to protect healthcare IoT environments. Cam and Trung [19] formulated an intelligent threat detection scheme combining principal component analysis with machine learning to mitigate sensor hijacking.

Recently, attention mechanisms and hybrid sequence models have been integrated into intrusion detection systems to address long-term dependencies. Elwhishi et al. [30] introduced an attention-enhanced hybrid architecture combining 1D convolutional Kolmogorov–Arnold networks (CKAN) with BiLSTM for industrial IoT applications, achieving notable parameter efficiency. Ullah et al. [31] designed the MAGRU-IDS, which leverages multi-head attention coupled with gated recurrent units to detect intrusions in IIoT networks under class-imbalance conditions. Reddy and Kuda [20] proposed ST-HAE, an unsupervised 1D-CNN-BiLSTM autoencoder framework in *Discover Networks*, evaluating spatiotemporal anomaly detection under realistic class imbalance (up to 278.5×) across consumer and malware IoT traffic, demonstrating that recurrent gating substantially stabilises training convergence under extreme distributional shifts. Furthermore, Hnamte [32] proposed an attention-gated hybrid CNN–BiLSTM deep learning framework combining parallel spatial filtering, recurrent channel encoding, and tiered explainable AI (SHAP, LIME, and ANOVA) for zero-shot cross-dataset attack detection across heterogeneous IoT environments, achieving a low amortised per-sample inference latency. Elsaid and Binbusayyis [33] designed an optimised isolation forest model for heterogeneous and streaming IIoT networks to maintain high precision under varying telemetry conditions. Similarly, Racherla et al. [15] introduced Deep-IDS, an optimised single-layer LSTM targeting real-time detection on constrained microcontroller nodes.

2.2 SDN and Edge-Centric Protection Systems

Software-Defined Networking (SDN) decouples the control plane from the data forwarding plane, offering centralised network programmability that has been widely leveraged for dynamic threat mitigation [34, 35]. Khedr et al. [34] and Kumavat and Gomes [35] investigated multi-layer mitigation frameworks (such as FMDADM), executing early DDoS packet filtering via average drop rates on stateful SDN switches. Bhattacharya et al. [9] implemented a multi-format log analysis subsystem in *Discover Networks* for detecting DDoS floods in programmable SDN topologies, combining entropy-based session classification with Kullback–Leibler divergence to reduce false positives to 1.5% with 3.6% controller overhead. Hill et al. [36] systematically reviewed open datasets, attack vectors, and mitigation strategies in software-defined ecosystems.

Archibong et al. [6] investigated AI-native zero-trust security architectures in *Discover Networks*, analysing hybrid CNN-LSTM networks with reinforcement learning for context-aware risk scoring.

In mobile and edge computing environments, Rani and Baalaji [26] proposed deep ensemble stacking tailored for IoT-edge platforms, balancing multi-attack detection accuracy with edge execution overheads. Simwinga et al. [37] conducted a systematic PRISMA-guided review in *Discover Networks* on intelligent wireless communication, edge AI, and IoT sensing for sustainable cyber-physical systems, identifying heterogeneous telemetry interoperability and bounded edge latency as paramount operational requirements. Furthermore, Pradhan and Dawadi [38] evaluated dual-connectivity architectures in *Discover Networks* for integrated 5G networks, analysing asymmetric traffic flows, split-bearer forwarding, and transport latency variations in distributed edge-to-core paths. Ahmed et al. [25] integrated deep active learning with attention models for load balancing and packet-level IDS in software-defined vehicular networks. Lu et al. [39] proposed DL-ANIDS, an adaptive intrusion detector customised for 5G mobile edge slices. Furthermore, Hnamte et al. [40] designed an enhanced secure network slicer for 5G and cloud security systems, establishing virtualised boundary isolation to restrict unauthorised device access and eliminate lateral intrusion vectors across shared infrastructure. Onoja et al. [5] surveyed DDoS threats across 5G-enabled IoT topologies, emphasising the severe bandwidth amplification risks facing edge devices.

2.3 Smart Grid and Digital-Substation Intrusion Detection

Modern smart power grids and digital substations increasingly rely on standardised Ethernet-based communication architectures governed by the IEC 61850 standard. Time-critical protection and automation functions utilise GOOSE and Sampled Values (SV) mapped directly to the Ethernet data-link layer to satisfy sub-millisecond transmission constraints, whereas supervisory monitoring, configuration, and SCADA interactions rely on Manufacturing Message Specification (MMS) operating over TCP/IP [1, 3]. Furthermore, legacy supervisory protocols such as DNP3 and Modbus remain prevalent across regional control centres and distribution grid feeders.

Recent smart grid security studies have demonstrated that although IEC 61850 multicast protocols are vital for rapid trip-and-isolate protection, their lack of intrinsic cryptographic authentication renders them acutely vulnerable to message spoofing, sequence number tampering, and status injection attacks. Hardware-in-the-loop (HIL) testbeds and specialised cyber-physical power system benchmarks (such as PMU and relay measurement corpora) capture electrical transient deviations resulting from the physical manipulation of actuators. Conversely, high-throughput enterprise and industrial network telemetry benchmarks (such as ToN-IoT) capture multiprotocol transport telemetry (flow counters, DNS query distributions, SSL/TLS handshake metadata, and HTTP transaction metrics), through which adversaries establish initial network footholds, execute reconnaissance, escalate privileges, and stage coordinated denial-of-service floods. Consequently, developing lightweight, trustworthy detection frameworks that can operate non-intrusively alongside digital substation architectures is a research priority.

2.4 Cyber-Physical Systems & Critical Infrastructure Defences

Cyber-Physical Systems embody bidirectional coupling between computational algorithms and physical processes. Gaba et al. [1] provided a comprehensive systematic analysis of cyber security challenges in CPS, demonstrating that cyber attacks targeting physical actuators cause irreversible equipment destruction if not mitigated within milliseconds. Javed et al. [7] developed an adversarial defence mechanism employing Graph Attention Networks (GAT) to unmask stealthy Advanced Persistent Threat (APT) campaigns targeting industrial cyber-physical systems.

Digital twin paradigms and physical AI architectures have recently emerged to mirror operational asset behaviours and tightly couple virtual simulations with physical machinery. Biswas et al. [4] investigated the integration of digital twins and physical artificial intelligence in cyber-physical systems, examining how physics-informed intelligence and virtual representations facilitate real-time monitoring, multi-tier anomaly detection, and operational resilience under dynamic physical disturbances. Chukwudi et al. [41] explored hybrid AI-driven eco-energy management in *Discover Networks*, highlighting the critical trade-offs between deep learning computational overhead, real-time edge processing, and device power constraints. In smart grid substation environments, Herrero [3] evaluated real-time transport protocols across constrained MQTT topologies. Ferrag et al. [2] published the Edge-IIoTset benchmark, providing empirical validation across multi-protocol environments, including Modbus, HTTP, and MQTT.

2.5 Explainable AI & Trustworthy Architectures

Despite superior empirical metrics, the black-box nature of deep neural networks poses substantial operational risks in critical utilities. Khan et al. [27] introduced an explainable deep learning framework utilising convolutional representations to discover cyber threats in industrial networks while providing model attributions. Al Rawajbeh et al. [28] proposed a trustworthy adaptive AI architecture for real-time IIoT security, highlighting the role of continuous learning and interpretability. Alsharaiah et al. [29] developed an explainable AI-driven transformer model for spoofing detection in IoMT networks. Ejaz et al. [23] proposed a multimodal behavioural fingerprinting framework in *Discover Networks*, demonstrating that unimodal network inspection remains blind to semantic data alteration attacks and establishing the necessity of cross-domain dependency checking under acute class imbalance. Berguiga et al. [42, 43] incorporated LIME into hybrid deep learning intrusion detectors to explain decision boundaries in medical cyber-physical topologies. Jayalaxmi et al. [12] and Liao et al. [13] surveyed explainability gaps in current IDSs, concluding that integrated local-global interpretability remains a vital unfulfilled objective. Furthermore, Kohli and Chhabra [16] provided an extensive survey on anomaly detection evaluation metrics, emphasising that macro-averaged metrics and threshold stability are critical for credible performance assessments in imbalanced industrial data.

Table 1: Methodological taxonomy comparing related intrusion detection architectures with CP-XNet.

Reference	**Core Architecture & Fusion**	**Sequence / Window Construction**	**Data Splitting & Safeguards**	**Imbalance Mitigation**	**XAI Support & Validation**	**Hardware Profiling & Metrics**
Vinayakumar et al. [14]	Generic Multi-Layer DNN	Tabular single-record vector	Standard 10-fold CV	None (Standard CE)	None	Unspecified CPU/GPU
Hairab et al. [17]	Regularised 2D-CNN	Synthetic 2D feature matrix	Random 80/20 split	None	None	Unmeasured GPU latency
Babar et al. [18]	Cascaded CNN → BiLSTM	Multi-packet row lookback	Random 70/30 split	SMOTE oversampling	None	Parameter count only
Ullah et al. [31]	Cascaded Multi-Head Attn + GRU	Inter-packet row window	Random 80/20 split	Focal Loss	None	Desktop CPU latency
Elwhishi et al. [30]	Cascaded CKAN → BiLSTM + Attn	Sequential feature projection	Random 70/30 split	None	Attention weights (visual)	FLOPs & memory footprint
Reddy & Kuda [20]	1D-CNN-BiLSTM Autoencoder	Multi-packet lookback window	70/15/15 chronological	Anomaly thresholding	Static SHAP + LIME	1.82 ms on ARM64 board
Al Rawajbeh et al. [28]	Adaptive Ensemble + Attention	Tabular point flow record	Random 80/20 split	Class-weighted loss	Global SHAP plots	Unspecified edge profile
Hnamte [32]	Parallel CNN-BiLSTM + Channel Attn	Multi-packet sliding window	Cross-dataset holdout	Cost-sensitive loss	SHAP, LIME, ANOVA	0.0123 ms on GPU
Proposed CP-XNet	**Parallel 1D-CNN + BiGRU via 2-Token MHSA Dynamic Fusion**	**Intra-flow structured feature segments** ($T = 8, d_{\mathbf{seg}} = \lceil D/8 \rceil$)	**Stratified holdout (70/15/15) across 3 distinct testbeds**	**Inverse-frequency regularisation** (w_k)	**Dual-tier SHAP + LIME with rank stability** ($\rho = 0.914$)	**0.0082–0.0128 ms latency; sub-0.13 mJ 10-W power proxy**

2.6 Architectural Synthesis, Research Gap, and Concrete Design Novelty

To delineate our contributions against the surveyed state-of-the-art in Table 1, we explicitly synthesise the architectural boundaries, unaddressed limitations, and empirical justifications of CP-XNet:

1. **What Existing CNN–RNN–Attention Detectors Do:** Prevailing hybrid detectors predominantly employ *cascaded pipelines* (where spatial convolutional filters process raw metrics and feed their intermediate activations into an LSTM or GRU) or compute unconstrained multi-head self-attention across lengthy packet sequences. Alternatively, they enforce sliding lookback windows over consecutive rows of public network datasets without verifying connection or host continuity, inadvertently learning artificial inter-row adjacency rather than true temporal attack dynamics. When parallel multi-branch networks are deployed, existing works almost universally combine spatial and sequential representations via naive unaligned concatenation ($\text{Concat}(\mathbf{h}_s, \mathbf{h}_t)$).
2. **What Existing Approaches Do Not Address:** First, cascading or naive concatenation induces destructive cross-modal feature interference: as proven in our ablation benchmarking (Section 8), concatenating spatial and recurrent representations without dynamic arbitration causes a severe performance collapse (5.07% Macro F1 degradation on ToN-IoT). Second, full-sequence attention across multi-packet sequences incurs quadratic computational complexity $\mathcal{O}(L^2)$ that rapidly exhausts edge microcontroller memory and violates real-time latency budgets. Third, most existing works evaluate exclusively on a single dataset under binary detection or dominant attack categories, failing to assess multi-class attribution under extreme real-world class imbalance (e.g., minority classes $< 0.5\%$). Finally, XAI techniques are routinely included merely as cosmetic post-hoc visualisations without quantitative stability testing across background perturbation baselines.
3. **Which Design Decisions Are Novel and Specifically Useful in CP-XNet:**
   - *Structured Feature-Segment Sequence Modelling:* Instead of risking inter-flow contamination via arbitrary multi-packet row grouping, CP-XNet partitions the single flow attribute vector into $T = 8$ contiguous functional segments ($d_{\text{seg}} = \lceil D/8 \rceil$). The BiGRU captures directional dependencies and macro-level feature interactions across functional protocol blocks without requiring external flow-tracking metadata. Crucially, the recurrent step index $t \in \{1, \ldots, 8\}$ indexes these functional feature blocks within a single flow rather than physical time or multi-packet temporal windows.
   - *Compact Two-Token MHSA Dynamic Fusion:* Rather than computing quadratic attention over long sequences, CP-XNet treats the spatial and recurrent representations as a compact 2-token sequence matrix $\mathbf{H} = [\mathbf{h}_s^\top; \mathbf{h}_t^\top] \in \mathbb{R}^{2\times 128}$. This reduces self-attention complexity to a fixed constant $\mathcal{O}(d^2)$ decoupled from sequence length, while dynamically arbitrating representation weighting between branches to mitigate the feature interference and performance degradation observed in direct concatenation.

- *Multi-Benchmark Rigour with Dual XAI Stability:* We couple class-weighted loss regularisation (w_k) with a dual-tier XAI pipeline whose feature attributions are quantitatively validated for stability across k-means background subsets ($\rho = 0.914 \pm 0.021$).

4. **Empirical Mapping of Research Questions to Experimental Evaluations:**
   - *Branch Synergy and Drift Resilience (RQ1):* Evaluated in Section 8.1 and Section 8.2 by contrasting standalone convolutional (M2) and recurrent (M3) baselines against full CP-XNet (M5) under both stationary stratified partitions and non-stationary chronological concept-drift holdouts (Protocol 2).
   - *Dynamic MHSA Arbitration vs. Naive Concatenation (RQ2):* Evaluated in Section 8.1 by comparing full CP-XNet (M5) against the direct concatenation variant (M4) across benchmark environments, focusing on representation interference mitigation and minority-class sensitivity preservation.
   - *Multi-Benchmark Generalisation and Minority Robustness:* Evaluated in Section 7 across three diverse testbeds (ToN-IoT, InSDN, and CIC-IIoT-2025), coupled with threshold-tuning analyses on acute minority intrusions (MITM, $n = 157$; U2R, $n = 2$).
   - *Interpretability Faithfulness and Edge Footprint (RQ3):* Evaluated in Section 5 via empirical GPU/CPU latency profiling and energy proxies, and in Section 9 through quantitative rank-correlation stability (ρ) and MoRF/LeRF explanation fidelity experiments.

3 Methodology

The overall architectural workflow of the proposed **CP-XNet** framework is illustrated in Fig. 1 and formally specified in Algorithm 1. Tailored for high-throughput, edge-deployable cyber-physical intrusion detection, CP-XNet processes an ingested flow telemetry vector $\mathbf{x} \in \mathbb{R}^D$ concurrently across two complementary representation pathways: a spatial feature-axis 1D-Convolutional Neural Network (1D-CNN) that extracts localised cross-feature correlations, and a structured feature-segment BiGRU that captures sequential intra-flow protocol dynamics. The resulting branch tokens are dynamically arbitrated via a compact two-token MHSA fusion block and classified through a regularised multi-tier attribution head.

3.1 Architectural Blueprint, Tensor Shapes, and Operational Hyperparameters

As depicted in Fig. 1, the end-to-end inference workflow pipeline comprises five integrated execution stages: (1) Single flow record ingestion ($\mathbf{x} \in \mathbb{R}^D$); (2) Branch 1: Spatial Feature-Axis 1D-CNN for local cross-feature extraction, outputting CNN token $\mathbf{f}_{\text{cnn}} \in \mathbb{R}^{B\times 128}$ (denoted $\mathbf{h}_{\text{cnn}}$); (3) Branch 2: Structured Feature-Segment BiGRU for sequential protocol dynamics, outputting BiGRU token $\mathbf{f}_{\text{bigru}} \in \mathbb{R}^{B\times 128}$ (denoted $\mathbf{h}_{\text{bigru}}$); (4) Two-Token MHSA dynamic fusion with residual skip connection and layer normalisation, yielding fused representation $\mathbf{f}' \in \mathbb{R}^{B\times 256}$; and (5) Multi-layer

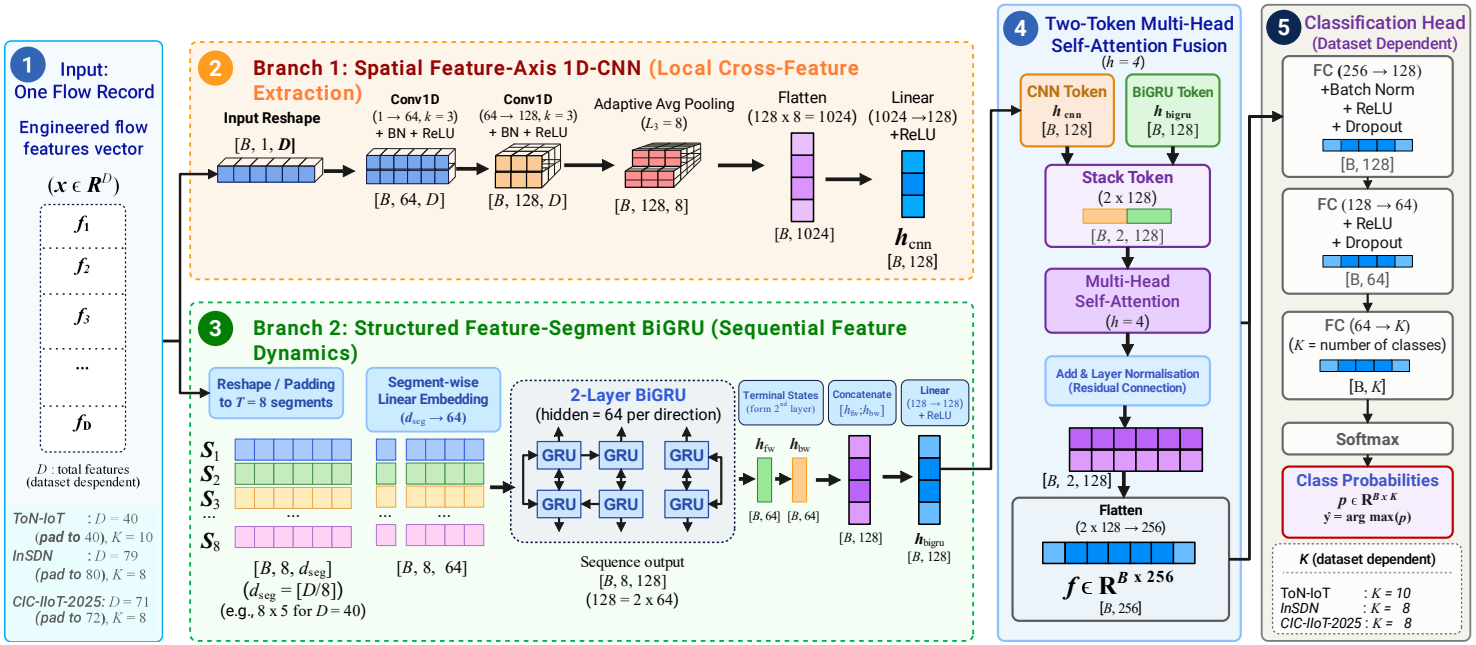


Fig. 1: End-to-end architectural workflow of the proposed CP-XNet framework. The architecture couples a spatial feature-axis 1D-CNN (capturing local cross-attribute interactions) with a structured feature-segment BiGRU (capturing intra-flow sequence dependencies across ordered functional protocol blocks, where the recurrence index denotes feature-segment order rather than physical timestamps), dynamically arbitrated via a compact two-token Multi-Head Self-Attention (MHSA) fusion block.

threat attribution head with Batch Normalisation producing calibrated posterior class probabilities $\mathbf{p} \in \mathbb{R}^{B \times K}$.

The top reference banner details the tensor shape notation $(B, D, T, d_{\text{seg}}, K)$, dataset-specific dimension configurations, key layer tensor outputs, and operational hyperparameters. The lower schematic illustrates the five-stage computational pipeline: (1) single tabular flow ingestion, (2) spatial feature-axis 1D-CNN branch with channel-first tensor transformations ($[B, 1, D] \to [B, 64, D] \to [B, 128, D] \to [B, 128, 8] \to [B, 1024] \to [B, 128]$), (3) structured feature-segment BiGRU branch with terminal forward and backward hidden-state concatenation ($[B, 64] \oplus [B, 64] \to [B, 128]$) and linear projection, (4) dynamic two-token multi-head self-attention fusion ($h = 4$, residual addition, Layer Normalisation, and flattening to $[B, 256]$), and (5) dataset-dependent classification head with Batch Normalisation, Dropout ($p = 0.25$), and calibrated softmax probabilities.

1. Flow Telemetry Ingestion (Stage 1)

The framework ingests an individual flow telemetry observation represented as an engineered attribute vector $\mathbf{x} = [f_1, f_2, \ldots, f_D]^\top \in \mathbb{R}^D$, where D denotes the total number of extracted network metrics. For batch processing, an input batch of size B is represented as $\mathbf{X} \in \mathbb{R}^{B \times D}$. The feature dimensionality D varies across operational domains ($D = 40$ for ToN-IoT Industrial Telemetry, $D = 79$ for InSDN Software-Defined Networking, and $D = 71$ for CIC-IIoT-2025 Multi-Device IoT).

2. Branch 1: Spatial Feature-Axis 1D-CNN (Stage 2)

To capture localised cross-feature correlations across contiguous protocol fields (such as transport ports, TCP state flags, and packet length distributions), $\mathbf{x}$ is embedded into a channels-first 1D tensor of shape $[B, 1, D]$ for batch size B. The spatial branch processes this tensor through two successive 1D convolutional stages, pooling, and projection:

- *First Convolutional Stage:* Conv1D($1 \to 64$, kernel $k = 3$, padding $p = 1$) with Batch Normalisation and ReLU activation, expanding channel depth to $[B, 64, D]$;
- *Second Convolutional Stage:* Conv1D($64 \to 128, k = 3, p = 1$) with Batch Normalisation and ReLU activation, yielding $[B, 128, D]$;
- *Spatial Compression and Flattening:* Adaptive average pooling (AdaptiveAvgPool1D, output length $L_s = 8$) compresses the feature axis to $[B, 128, 8]$ (channels-first), which is flattened into a $[B, 1024]$ intermediate vector;
- *Linear Token Projection:* A fully connected layer (Linear($1024 \to 128$)) with ReLU activation projects the flattened tensor into the spatial CNN token $\mathbf{f}_{\text{cnn}} \in \mathbb{R}^{B \times 128}$ (denoted $\mathbf{h}_{\text{cnn}}$ in Fig. 1 and $\mathbf{h}_s$ in the subsequent mathematical formulations).

3. Branch 2: Structured Feature-Segment BiGRU (Stage 3)

To capture macro-level protocol dependencies without enforcing artificial row-adjacency across unrelated network connections, the single flow record is partitioned along its structured feature axis into $T = 8$ contiguous functional segments $\mathbf{S} = [\mathbf{s}_1, \ldots, \mathbf{s}_8]^\top \in \mathbb{R}^{B \times 8 \times d_{\text{seg}}}$, where $d_{\text{seg}} = \lceil D/8 \rceil$. The sequential branch processes this segment sequence as follows:

- *Segment-Wise Linear Embedding:* Each segment $\mathbf{s}_t \in \mathbb{R}^{B\times d_{\text{seg}}}$ ($t \in \{1,\ldots,8\}$) is mapped into a 64-dimensional latent representation via Linear($d_{\text{seg}} \to 64$) with ReLU, producing an embedded sequence tensor of shape $[B, 8, 64]$ ($[B, T, 64]$);
- *Two-Layer BiGRU Recurrence:* A 2-layer Bidirectional GRU with hidden dimension $h_g = 64$ per direction ($128 = 2\times 64$ bidirectional features) traverses the sequence forward (from network addressing towards application payloads) and backward (from application flags to transport primitives), accumulating step representations across the sequence;
- *Terminal-State Concatenation and Projection:* In strict accordance with the computational graph in Fig. 1 and Algorithm 1, feature aggregation does *not* apply sequence-wide mean pooling; rather, the terminal forward hidden state $\overrightarrow{\mathbf{h}}_T \in \mathbb{R}^{B\times 64}$ and terminal backward hidden state $\overleftarrow{\mathbf{h}}_1 \in \mathbb{R}^{B\times 64}$ from the second recurrent layer are concatenated into a joint directional vector $[\overrightarrow{\mathbf{h}}_T \,\|\, \overleftarrow{\mathbf{h}}_1] \in \mathbb{R}^{B\times 128}$. A linear projection (Linear($128 \to 128$)) with ReLU activation then produces the sequential BiGRU token $\mathbf{f}_{\text{bigru}} \in \mathbb{R}^{B\times 128}$ (denoted $\mathbf{h}_{\text{bigru}}$ in Fig. 1 and $\mathbf{h}_t$ in subsequent equations).

4. Two-Token Multi-Head Self-Attention Fusion (Stage 4)

The spatial token $\mathbf{f}_{\text{cnn}}$ and sequential token $\mathbf{f}_{\text{bigru}}$ are stacked into a compact 2-token sequence matrix $\mathbf{H} = [\mathbf{f}_{\text{cnn}}^\top; \mathbf{f}_{\text{bigru}}^\top] \in \mathbb{R}^{B\times 2\times 128}$. The fusion module dynamically arbitrates cross-branch relevance using a 4-head self-attention mechanism ($h = 4$, with head dimension $d_k = 128/4 = 32$). The attention output is added back to $\mathbf{H}$ via a residual skip connection and normalised through Layer Normalisation (LayerNorm($\mathbf{H}$ + MHSA($\mathbf{H}$)) $\in \mathbb{R}^{B\times 2\times 128}$). Finally, the attended token matrix is vectorised (flattened) into a fused joint representation $\mathbf{f}' \in \mathbb{R}^{B\times 256}$ (denoted $\mathbf{z}_{\text{fuse}} \in \mathbb{R}^{B\times 256}$), preserving the independent discriminative features of both branches.

5. Dataset-Dependent Classification Head (Stage 5)

The fused vector $\mathbf{f}' \in \mathbb{R}^{B\times 256}$ feeds into a three-tier classification perceptron incorporating Batch Normalisation:

- Dense hidden layer 1: Linear($256 \to 128$) followed by Batch Normalisation (BatchNorm1D), ReLU activation, and Dropout ($p = 0.25$), yielding $[B, 128]$;
- Dense hidden layer 2: Linear($128 \to 64$) followed by ReLU activation and Dropout ($p = 0.25$), yielding $[B, 64]$;
- Final projection layer: Linear($64 \to K$) producing unnormalized threat attribution logits of dimension $[B, K]$, where K is the number of target threat categories;
- Softmax normalisation: Maps raw logits into a calibrated posterior class probability distribution $\mathbf{p} = [p_1, \ldots, p_K]^\top \in \mathbb{R}^{B\times K}$, with the predicted classification assigned via $\widehat{y} = \arg\max_{k\in\{1,\ldots,K\}} p_k$.

Tensor Shape Summary

As formally highlighted in the architectural reference summary in Fig. 1, four principal dimensional variables define the tensor transformations across the CP-XNet pipeline:

- D **(Total Number of Input Features):** The total number of engineered tabular attributes extracted from a single flow record ($D = 40$ for ToN-IoT, $D = 79$ for InSDN, and $D = 71$ for CIC-IIoT-2025);
- T **(Number of Feature Segments):** The number of functional intra-flow sub-segments, uniformly fixed to $T = 8$ across all datasets to balance sequence granularity against computational efficiency;
- $d_{\mathbf{seg}}$ **(Segment Dimension):** The width of each individual feature segment, defined as $d_{\text{seg}} = \lceil D/T \rceil = \lceil D/8 \rceil$, resulting in $d_{\text{seg}} = 5$ for ToN-IoT, $d_{\text{seg}} = 10$ for InSDN, and $d_{\text{seg}} = 9$ for CIC-IIoT-2025;
- K **(Number of Output Classes):** The cardinality of the target classification space, set to $K = 10$ for ToN-IoT multi-class attribution, $K = 8$ for InSDN and CIC-IIoT-2025, and $K = 2$ for binary perimeter gating across all benchmarks.

Key Dimensions per Dataset

Table 2 outlines the exact dataset-specific dimensions governing CP-XNet across the three benchmark environments. For ToN-IoT Industrial Telemetry, the original feature dimension $D = 40$ is exactly divisible by $T = 8$, yielding 8 contiguous segments of dimension $d_{\text{seg}} = 5$, with $K = 10$ output classes. For InSDN Software-Defined Networking, the original feature set comprises $D = 79$ attributes; to achieve integer divisibility by $T = 8$, a single zero-padded feature is appended ($79 \rightarrow 80$), establishing $d_{\text{seg}} = 10$ and $K = 8$ classes. Similarly, for the CIC-IIoT-2025 benchmark, the original $D = 71$ features are padded with one neutral feature ($71 \rightarrow 72$), yielding $d_{\text{seg}} = 9$ across $T = 8$ segments, with $K = 8$ threat classes.

Table 2: Key architectural dimensions of CP-XNet per benchmark dataset.

Benchmark Dataset	**Original** D	**Padded** D	**Segments** (T)	$d_{\mathbf{seg}}$ $(D/8)$	K **(Classes)**
ToN-IoT	40	40	8	5	10 (Multi) / 2 (Binary)
InSDN	79	80	8	10	8 (Multi) / 2 (Binary)
CIC-IIoT-2025	71	72	8	9	8 (Multi) / 2 (Binary)

Note: D denotes total input features; $T = 8$ is segment count; $d_{\text{seg}} = \lceil D/8 \rceil$ is segment width; K denotes output classes. Zero-padding ($79 \rightarrow 80$ and $71 \rightarrow 72$) ensures uniform divisibility by $T = 8$.

Main Hyperparameters (Final Implementation)

The operational hyperparameters of CP-XNet in its final verified implementation (referenced in Fig. 1) are specified across the four core modules:

- **CNN Branch:** Conv1D($1 \rightarrow 64 \rightarrow 128$, kernel $k = 3$, padding $p = 1$), Batch Normalisation (BN) on each convolutional layer, ReLU non-linear activations, Adaptive Average Pooling with output length $L_s = 8$ (AdaptiveAvgPool1D yielding $[B, 128, 8]$), followed by a fully connected projection FC($1024 \rightarrow 128$) with ReLU to produce the 128-dimensional spatial token $\mathbf{f}_{\text{cnn}} \in \mathbb{R}^{B \times 128}$.

- **BiGRU Branch:** Segment-wise linear projection Linear($d_{\text{seg}} \to 64$) with ReLU; a 2-layer Bidirectional GRU with hidden dimension $h_g = 64$ per direction ($128 = 2 \times 64$ bidirectional features); concatenation of terminal forward and backward states ($\overrightarrow{\mathbf{h}}_8 \in \mathbb{R}^{B\times 64}$ and $\overleftarrow{\mathbf{h}}_1 \in \mathbb{R}^{B\times 64} \to [B, 128]$); followed by a linear projection FC($128 \to 128$) with ReLU to produce the 128-dimensional sequential token $\mathbf{f}_{\text{bigru}} \in \mathbb{R}^{B\times 128}$.
- **Attention Fusion:** Stacking the two branch tokens into a 2-token sequence matrix of dimension $[B, 2, 128]$; 4-head Multi-Head Self-Attention (MHSA, $h = 4$, head dimension $d_k = 32$); residual skip connection with Layer Normalisation (Add & LayerNorm); followed by vector flattening ($[B, 2, 128] \to [B, 256]$) to output the joint fused representation $\mathbf{f}' \in \mathbb{R}^{B\times 256}$.
- **Classification Head:** Multi-tier fully connected perceptron FC($256 \to 128 \to 64 \to K$) incorporating Batch Normalisation (BatchNorm1D) after the first dense layer ($256 \to 128$), ReLU activations, Dropout regularisation ($p = 0.25$) on both hidden representations, and Softmax normalisation to produce the class probability vector $\mathbf{p} \in \mathbb{R}^{B\times K}$.

Core Design Notes and Implementation Constraints

The operational principles governing the CP-XNet workflow in Fig. 1 are underpinned by five essential design criteria:

1. **Single Flow-Feature Vector versus Time Series:** The input to CP-XNet is strictly a single, completed flow-feature vector $\mathbf{x} \in \mathbb{R}^D$ extracted post-connection, rather than an inter-flow packet arrival time series. This eliminates inter-session data leakage, removes susceptibility to network packet transmission jitter, and ensures that inference can be executed autonomously on isolated flow summaries.
2. **Contiguous Functional Feature Segments:** The sub-segments $\mathbf{s}_1, \ldots, \mathbf{s}_8$ are not randomly permuted; they group contiguous, logically related protocol fields ordered hierarchically (e.g., transport endpoints $\to$ connection flags $\to$ transfer volumetrics $\to$ application layer codes), allowing the BiGRU to capture cross-layer protocol stack dynamics.
3. **Zero-Padding Mechanics for Irregular Dimensions:** When the raw feature dimension D is not an exact multiple of $T = 8$, zero-padding is appended to reach the nearest multiple: InSDN ($79 \to 80$ features) and CIC-IIoT-2025 ($71 \to 72$ features). Because all continuous features undergo standard z-score normalisation ($\mu = 0, \sigma = 1$), appending 0.0 assigns the padded coordinates to the neutral population mean, ensuring that padding introduces zero distributional bias or spurious activations.
4. **Token Dimensional Symmetry in Self-Attention Fusion:** Both the spatial CNN branch and the sequential BiGRU branch project their respective high-level feature maps into identical 1×128 dimensional tokens ($\mathbf{f}_{\text{cnn}}$ and $\mathbf{f}_{\text{bigru}}$). This strict dimensional parity guarantees that neither branch dominates the attention mechanism by virtue of vector magnitude alone, enabling the 4-head MHSA module to dynamically arbitrate contextual importance based purely on cross-branch affinity.
5. **Dataset-Dependent Output Head Adaptability:** The entire internal feature-extraction backbone (1D-CNN layers, BiGRU cells, and MHSA fusion block)

maintains a strictly invariant parameter topology across all datasets and classification regimes. When transitioning between domains, only the initial segment projection ($\mathbf{W}_e \in \mathbb{R}^{64\times d_{\text{seg}}}$) and the final classification projection ($\mathbf{W}_{\text{out}} \in \mathbb{R}^{64\times K}$) adjust their input/output dimensions, facilitating modular re-training across heterogeneous cyber-physical environments.

3.2 Feature-Axis Spatial 1D-CNN Branch (Local Cross-Feature Correlations)

Let $\mathbf{x} \in \mathbb{R}^D$ denote an individual flow telemetry observation comprising D normalised protocol and network metrics ($D = 40$ for ToN-IoT, $D = 79$ for InSDN, and $D = 71$ for CIC-IIoT-2025). To model local cross-feature correlations across contiguous attribute groups (such as source/destination port allocations, packet length distributions, TCP flags, and protocol handshake flags), $\mathbf{x}$ is embedded as a single-channel one-dimensional tensor $\mathbf{X}^{(0)} \in \mathbb{R}^{1\times D}$. Here, the designation "spatial" refers strictly to the ordered feature axis rather than physical grid geometry: because preprocessing arranges features into contiguous functional groupings (e.g., transport headers, flow throughput counters, connection states, and application-layer flags), 1D convolutions with kernel size $K = 3$ operate across adjacent feature indices, capturing localised cross-feature correlations and composite protocol signatures.

The spatial branch applies two successive 1D convolutional layers with batch normalisation and non-linear activation. For the l-th convolutional layer with filter weights $\mathbf{W}_c^{(l)}$ and bias $\mathbf{b}_c^{(l)}$, the operation is formulated as:

$$\mathbf{Z}^{(l)} = \text{Conv1D}\left(\mathbf{X}^{(l-1)}, \mathbf{W}_c^{(l)}\right) + \mathbf{b}_c^{(l)}, \tag{1}$$

$$\mathbf{X}^{(l)} = \text{ReLU}\left(\text{BatchNorm1D}\left(\mathbf{Z}^{(l)}\right)\right), \tag{2}$$

where the first layer expands the channel depth to $C_1 = 64$ with kernel size $K = 3$ and padding $P = 1$. The second layer projects to $C_2 = 128$ filters. Subsequently, an adaptive average pooling operation compresses the spatial dimension to a fixed length $L_s = 8$:

$$\mathbf{S}_{\text{pool}} = \text{AdaptiveAvgPool1D}\left(\mathbf{X}^{(2)}, L_s\right) \in \mathbb{R}^{C_2\times L_s}. \tag{3}$$

Finally, a linear transformation maps the flattened spatial representation into a d-dimensional spatial feature vector $\mathbf{h}_s \in \mathbb{R}^d$ ($d = 128$):

$$\mathbf{h}_s = \mathbf{W}_s \cdot \text{vec}\left(\mathbf{S}_{\text{pool}}\right) + \mathbf{b}_s. \tag{4}$$

3.3 Structured Feature-Segment Sequence Modelling Branch (BiGRU)

While the 1D-CNN branch captures local contiguous feature interactions within the flow record, complex cyber-physical attacks manifest multi-attribute dependency patterns across distinct functional subsystems (e.g., initial connection state setup, volumetric transmission counters, protocol handshake codes, and application-layer

error flags). Rather than enforcing arbitrary inter-packet row adjacency across unrelated network connections or sessions—which risks artificial correlation—CP-XNet models the intra-flow progression across functional feature categories using *structured feature-segment sequence modelling*.

Specifically, the input feature vector $\mathbf{x} \in \mathbb{R}^D$ is partitioned along its structured feature axis into $T = 8$ sequential sub-segments:

$$\mathbf{S} = [\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_T]^\top \in \mathbb{R}^{T \times d_{\text{seg}}}, \tag{5}$$

where $d_{\text{seg}} = \lceil D/T \rceil$ represents the segment dimensionality ($d_{\text{seg}} = 5$ for ToN-IoT with $D = 40$; $d_{\text{seg}} = 10$ for InSDN with $D = 79$ zero-padded to 80; and $d_{\text{seg}} = 9$ for CIC-IIoT-2025 with $D = 71$ zero-padded to 72). Each sub-segment $\mathbf{s}_t \in \mathbb{R}^{d_{\text{seg}}}$ ($t = 1, \ldots, T$) aggregates a distinct functional block of telemetry attributes and is projected into a continuous latent embedding space:

$$\mathbf{e}_t = \text{ReLU}\left(\mathbf{W}_e \mathbf{s}_t + \mathbf{b}_e\right) \in \mathbb{R}^{d_e}, \quad (d_e = 64), \tag{6}$$

where $\mathbf{W}_e \in \mathbb{R}^{d_e \times d_{\text{seg}}}$ and $\mathbf{b}_e \in \mathbb{R}^{d_e}$ denote the projection weights and bias vector. This formulation directly explains the component allocation of $d_{\text{seg}} \to 64$ in the architectural layer breakdown (e.g., $5 \to 64$ for ToN-IoT), while ensuring that all sequences are formed strictly *within* each flow's own structured attributes, completely eliminating cross-session data leakage and artificial row-adjacency biases.

Methodological Clarification: Feature-Segment Sequencing versus Physical Time

It is essential to clarify the precise nature of the input sequence ingested by the BiGRU branch. The sequence length $T = 8$ represents an ordered partition of the single tabular flow record into eight contiguous feature blocks (e.g., for ToN-IoT, five features per block spanning transport headers, flow volumetrics, packet counters, DNS activity, SSL handshakes, and HTTP/Zeek flags). The recurrent step index $t \in \{1, \ldots, T\}$ corresponds to the *feature-segment index along the structured protocol stack*, rather than successive physical timestamps, individual packet arrivals, or sliding temporal observation windows. Consequently, the BiGRU models directional attribute dependencies and cross-block feature correlations across the protocol hierarchy of an individual flow; it does not observe temporal attack progression across consecutive flows or physical time series. Throughout this manuscript, the representation vector $\mathbf{h}_t$ and the informal designation "temporal branch" serve as architectural shorthand for this sequential feature-segment pathway, distinguishing it from the convolutional spatial branch $\mathbf{h}_s$. Genuine temporal concept drift over calendar time is evaluated separately in Section 8.2 via chronological holdouts (Protocol 2).

The embedded sequence $\{\mathbf{e}_t\}_{t=1}^{T}$ is ingested by a two-layer BiGRU. At each sequence step t, the forward GRU updates its hidden state $\overrightarrow{\mathbf{h}}_t$ via reset gate $\mathbf{r}_t$ and update gate $\mathbf{z}_t$:

$$\mathbf{r}_t = \sigma\left(\mathbf{W}_{ir}\mathbf{e}_t + \mathbf{b}_{ir} + \mathbf{W}_{hr}\overrightarrow{\mathbf{h}}_{t-1} + \mathbf{b}_{hr}\right), \tag{7}$$

$$\mathbf{z}_t = \sigma\left(\mathbf{W}_{iz}\mathbf{e}_t + \mathbf{b}_{iz} + \mathbf{W}_{hz}\overrightarrow{\mathbf{h}}_{t-1} + \mathbf{b}_{hz}\right), \tag{8}$$

$$\widetilde{\mathbf{h}}_t = \tanh\left(\mathbf{W}_{in}\mathbf{e}_t + \mathbf{b}_{in} + \mathbf{r}_t \odot (\mathbf{W}_{hn}\overrightarrow{\mathbf{h}}_{t-1} + \mathbf{b}_{hn})\right), \tag{9}$$

$$\overrightarrow{\mathbf{h}}_t = (1 - \mathbf{z}_t) \odot \overrightarrow{\mathbf{h}}_{t-1} + \mathbf{z}_t \odot \widetilde{\mathbf{h}}_t, \tag{10}$$

where $\sigma(\cdot)$ denotes the sigmoid activation and $\odot$ represents the Hadamard element-wise product. Symmetrically, the backward GRU processes the sequence in reverse order yielding $\overleftarrow{\mathbf{h}}_t$. The contextual segment vector is obtained by concatenating the terminal hidden states from both directions:

$$\mathbf{h}_{\text{bi}} = [\overrightarrow{\mathbf{h}}_T \,\|\, \overleftarrow{\mathbf{h}}_1] \in \mathbb{R}^{2h_g}, \quad (h_g = 64). \tag{11}$$

A linear projection maps $\mathbf{h}_{\text{bi}}$ to the temporal feature vector $\mathbf{h}_t \in \mathbb{R}^d$ ($d = 128$):

$$\mathbf{h}_t = \mathbf{W}_t\mathbf{h}_{\text{bi}} + \mathbf{b}_t. \tag{12}$$

3.4 Multi-Head Self-Attention Fusion

To dynamically weigh the contextual significance of spatial versus temporal representations, the vectors $\mathbf{h}_s$ and $\mathbf{h}_t$ are assembled into a token sequence matrix $\mathbf{H} \in \mathbb{R}^{2\times d}$:

$$\mathbf{H} = \begin{bmatrix}\mathbf{h}_s^\top \\ \mathbf{h}_t^\top\end{bmatrix}. \tag{13}$$

The two tokens correspond to the branch-level spatial and temporal representations rather than individual input observations.

The MHSA mechanism [44] projects $\mathbf{H}$ across $N_h = 4$ independent attention heads. For head $i \in \{1, \ldots, N_h\}$, the Query, Key, and Value matrices are derived via learnable parameter matrices $\mathbf{W}_i^Q, \mathbf{W}_i^K, \mathbf{W}_i^V \in \mathbb{R}^{d\times d_k}$ ($d_k = d/N_h = 32$):

$$\mathbf{Q}_i = \mathbf{H}\mathbf{W}_i^Q, \quad \mathbf{K}_i = \mathbf{H}\mathbf{W}_i^K, \quad \mathbf{V}_i = \mathbf{H}\mathbf{W}_i^V. \tag{14}$$

The scaled dot-product attention weights and resulting head representation for head i are calculated as:

$$\mathbf{A}_i = \text{Softmax}\left(\frac{\mathbf{Q}_i\mathbf{K}_i^\top}{\sqrt{d_k}}\right) \in \mathbb{R}^{2\times 2}, \tag{15}$$

$$\text{Head}_i = \mathbf{A}_i\mathbf{V}_i \in \mathbb{R}^{2\times d_k}, \tag{16}$$

where $\mathbf{A}_i$ denotes the pairwise attention-weight matrix arbitrating the bidirectional cross-branch relevance between spatial and temporal tokens, and $\text{Head}_i \in \mathbb{R}^{2\times 32}$ is the individual attention head output ($d_k = d/N_h = 32$). The outputs of all attention heads are concatenated and projected linearly, followed by a residual skip connection and layer normalisation:

$$\mathbf{H}_{\text{attn}} = \text{LayerNorm}\left(\mathbf{H} + [\text{Head}_1 \,\|\, \ldots \,\|\, \text{Head}_{N_h}]\,\mathbf{W}^O\right), \tag{17}$$

where $\mathbf{W}^O \in \mathbb{R}^{d\times d}$. The attended tokens are flattened into a fused joint feature representation $\mathbf{z}_{\text{fuse}} = \text{vec}(\mathbf{H}_{\text{attn}}) \in \mathbb{R}^{2d} = \mathbb{R}^{256}$.

3.5 Multi-Class Classification Head & Class-Weighted Objective

The fused vector $\mathbf{z}_{\text{fuse}}$ is processed by a multi-layer classification head:

$$\mathbf{z}_1 = \text{ReLU}\left(\text{BatchNorm1D}\left(\mathbf{W}_{f1}\mathbf{z}_{\text{fuse}} + \mathbf{b}_{f1}\right)\right), \tag{18}$$

$$\mathbf{z}_2 = \text{Dropout}\left(\mathbf{z}_1, p = 0.25\right), \tag{19}$$

$$\mathbf{z}_3 = \text{ReLU}\left(\mathbf{W}_{f2}\mathbf{z}_2 + \mathbf{b}_{f2}\right), \tag{20}$$

$$\widehat{\mathbf{y}} = \text{Softmax}\left(\mathbf{W}_{\text{out}}\mathbf{z}_3 + \mathbf{b}_{\text{out}}\right) \in \mathbb{R}^K, \tag{21}$$

where $K = 10$ corresponds to the target threat classes (nine industrial network attack categories plus normal traffic).

Because critical industrial intrusions occur with severe class imbalance, standard cross-entropy loss tends to bias towards dominant classes. CP-XNet employs an inverse frequency class-weighted cross-entropy loss:

$$\mathcal{L} = -\frac{1}{B}\sum_{b=1}^{B}\sum_{k=1}^{K} w_k \cdot y_{b,k} \log\left(\widehat{y}_{b,k}\right), \tag{22}$$

where $w_k = \frac{N_{\text{total}}}{K \cdot N_k}$ represents the regularised class penalty weight for category k, $y_{b,k}$ is the ground-truth binary indicator, and B is the mini-batch size.

3.6 Tensor Dimensions, BiGRU Directionality, and Architectural Specifications

To establish complete transparency regarding the neural pipeline, Table 3 itemises the exact tensor transformations at each execution stage for an input batch size B across both binary ($K = 2$) and multi-class ($K \in \{8, 10\}$) regimes:

1. **Input Ingestion & Spatial Processing:** A batch of normalized flow vectors $\mathbf{x} \in \mathbb{R}^{B\times D}$ ($D \in \{40, 71, 79\}$) is unsqueezed into a single-channel 1D tensor of shape $[B, 1, D]$. Two successive 1D convolutional layers with kernel size $K = 3$ expand the channel depth: Conv1D_1 outputs $[B, 64, D]$ and Conv1D_2 outputs $[B, 128, D]$. An adaptive average pooling layer compresses the spatial length to $L_s = 8$, yielding $[B, 128, 8]$, which is flattened to $[B, 1024]$ and projected linearly by $\mathbf{W}_s$ into the spatial token $\mathbf{h}_s \in \mathbb{R}^{B\times 128}$.
2. **Structured Segment Recurrence & Directionality:** In parallel, $\mathbf{x}$ is partitioned into $T = 8$ contiguous functional segments of dimension $d_{\text{seg}} = \lceil D/8 \rceil$, with zero-padding applied if $D < 8 \cdot d_{\text{seg}}$, yielding a tensor of shape $[B, 8, d_{\text{seg}}]$. The linear embedding $\mathbf{W}_e$ projects each segment to $d_e = 64$, producing $[B, 8, 64]$. The 2-layer BiGRU processes this 8-step sequence in both directions:

- *Directionality Traversal:* Operating across the functional segment hierarchy defined in Section 3.3, the forward recurrent pass traverses from addressing headers towards application payloads ($t = 1 \rightarrow 8$), while the backward pass traverses in reverse ($t = 8 \rightarrow 1$), capturing macro-level inter-segment correlations.
- *Hidden-State Selection:* The final contextual temporal vector $\mathbf{h}_{\text{bi}} \in \mathbb{R}^{B\times 128}$ is formed by concatenating the terminal forward hidden state $\overrightarrow{\mathbf{h}}_T \in \mathbb{R}^{B\times 64}$ and terminal backward hidden state $\overleftarrow{\mathbf{h}}_1 \in \mathbb{R}^{B\times 64}$ from the second recurrent layer, subsequently projected by $\mathbf{W}_t$ into temporal token $\mathbf{h}_t \in \mathbb{R}^{B\times 128}$.

3. **Two-Token Attention Fusion & Absence of Masking:** The spatial and temporal tokens are assembled into a sequence matrix $\mathbf{H} \in \mathbb{R}^{B\times 2\times 128}$. Because the sequence length is strictly $S = 2$ and both branch representations are always fully observed, *no attention mask or causal mask is applied.* Across all $N_h = 4$ attention heads, the attention-weight matrix $\mathbf{A}_i \in \mathbb{R}^{B\times 2\times 2}$ arbitrates the bidirectional relevance between spatial and temporal representations, producing head output representations $\text{Head}_i = \mathbf{A}_i\mathbf{V}_i \in \mathbb{R}^{B\times 2\times 32}$ ($d_k = 32$). The multi-head projection with residual skip connection and layer normalisation yields $\mathbf{H}_{\text{attn}} \in \mathbb{R}^{B\times 2\times 128}$.
4. **Flattening vs. Pooling & Head Instantiation:** Rather than applying global average pooling across the two tokens—which would compress spatial and temporal features into an unweighted average—$\mathbf{H}_{\text{attn}}$ is *flattened* via vectorisation into a fused joint representation $\mathbf{z}_{\text{fuse}} = \text{vec}(\mathbf{H}_{\text{attn}}) \in \mathbb{R}^{B\times 256}$, preserving the independent discriminative capacity of both attended branches.
5. **Output Head Configurations:** The fused representation $\mathbf{z}_{\text{fuse}}$ passes through two dense hidden layers ($256 \rightarrow 128$ with BatchNorm1D, followed by $128 \rightarrow 64$). For the binary regime, the final projection is $\mathbf{W}_{\text{out}} \in \mathbb{R}^{64\times 2}$ producing calibrated binary posterior probabilities via softmax (or equivalent single-logit sigmoid). For the multi-class regime, $\mathbf{W}_{\text{out}} \in \mathbb{R}^{64\times K}$ ($K = 10$ for ToN-IoT, $K = 8$ for InSDN and CIC-IIoT-2025) followed by multi-class softmax. When the input feature dimension D changes across datasets, *only the linear segment embedding* $\mathbf{W}_e \in \mathbb{R}^{64\times d_{seg}}$ *is adapted*, leaving all internal CNN, BiGRU, MHSA, and classifier intermediate weights structurally invariant.

Algorithm 1 formalises the end-to-end computational pipeline of CP-XNet, specifying the exact sequence of tensor operations from single-flow feature vector ingestion to the final multi-class posterior probability generation.

4 Explainable AI (XAI) Framework for Critical Infrastructure

To foster operator trust in safety-critical smart grid and industrial environments, CP-XNet incorporates a complementary dual-tier visual interpretability architecture that combines global game-theoretic Shapley values with local linear surrogate models. Because perturbation-based attribution algorithms incur non-negligible computational overhead, this interpretability tier operates post-hoc and asynchronously rather than within the real-time wire-speed forward-pass loop. The neural classifier provides the immediate sub-millisecond threat label, whereas SHAP and LIME can be invoked

Table 3: Layer-by-layer tensor shape trajectory across CP-XNet stages.

Processing Stage	**Constituent Operation**	**Input Shape**	**Output Tensor Shape**
Flow Telemetry Ingestion	Batch normalization / Reshape	$[B, D]$	$[B, 1, D]$ (Spatial) / $[B, 8, d_{\text{seg}}]$ (Temporal)
Spatial Branch	Conv1D Layer 1 ($K = 3, P = 1$)	$[B, 1, D]$	$[B, 64, D]$
	Conv1D Layer 2 ($K = 3, P = 1$)	$[B, 64, D]$	$[B, 128, D]$
	AdaptiveAvgPool1D ($L_s = 8$)	$[B, 128, D]$	$[B, 128, 8]$
	Flatten & Linear Projection $\mathbf{W}_s$	$[B, 1024]$	$\mathbf{h}_s \in [B, 128]$
Temporal Branch	Linear Segment Embedding $\mathbf{W}_e$	$[B, 8, d_{\text{seg}}]$	$[B, 8, 64]$
	2-Layer BiGRU ($h_g = 64$)	$[B, 8, 64]$	$\overrightarrow{\mathbf{h}}_8 \in [B, 64]$, $\overleftarrow{\mathbf{h}}_1 \in [B, 64]$
	Concatenate Terminal States	—	$\mathbf{h}_{\text{bi}} \in [B, 128]$
	Linear Projection $\mathbf{W}_t$	$[B, 128]$	$\mathbf{h}_t \in [B, 128]$
MHSA Fusion Block	Token Stacking	$\mathbf{h}_s, \mathbf{h}_t$	$\mathbf{H} \in [B, 2, 128]$
	4-Head Attention & Projection $\mathbf{W}^O$	$[B, 2, 128]$	$\mathbf{H}_{\text{proj}} \in [B, 2, 128]$
	Residual Skip + LayerNorm	$[B, 2, 128]$	$\mathbf{H}_{\text{attn}} \in [B, 2, 128]$
	Vector Flattening (vec)	$[B, 2, 128]$	$\mathbf{z}_{\text{fuse}} \in [B, 256]$
Classification Head	Dense Layer 1 + BatchNorm1D	$[B, 256]$	$\mathbf{z}_1 \in [B, 128]$
	Dropout ($p = 0.25$) + Dense 2	$[B, 128]$	$\mathbf{z}_3 \in [B, 64]$
	Final Projection $\mathbf{W}_{\text{out}}$ + Softmax	$[B, 64]$	$\widehat{\mathbf{y}} \in [B, K]$ ($K \in \{2, 8, 10\}$)

Algorithm 1 End-to-End Execution Pipeline of CP-XNet

Require: Flow telemetry attribute vector $\mathbf{x} \in \mathbb{R}^D$ ($D \in \{40, 71, 79\}$); segment count $T = 8$; segment dimensionality $d_{\text{seg}} = \lceil D/T \rceil$; trainable parameter set $\Theta = \{\mathbf{W}_c^{(1)}, \mathbf{W}_c^{(2)}, \mathbf{W}_s, \mathbf{W}_e, \mathbf{W}_{\text{gru}}, \mathbf{W}_t, \mathbf{W}^Q, \mathbf{W}^K, \mathbf{W}^V, \mathbf{W}^O, \mathbf{W}_{f1}, \mathbf{W}_{f2}, \mathbf{W}_{\text{out}}\}$; class regularisation weights $\{w_k\}_{k=1}^K$.

Ensure: Calibrated posterior class probability distribution $\widehat{\mathbf{y}} \in [0,1]^K$, predicted threat category $\widehat{c} \in \{1, \ldots, K\}$.

// Stage 1: Feature-Axis Spatial 1D-CNN Branch

Embed flow vector as single-channel tensor: $\mathbf{X}^{(0)} \leftarrow \mathbf{x} \in \mathbb{R}^{1 \times D}$

Compute first 1D convolutional feature map: $\mathbf{Z}^{(1)} \leftarrow \text{Conv1D}(\mathbf{X}^{(0)}, \mathbf{W}_c^{(1)}) + \mathbf{b}_c^{(1)}$ ▷ $C_1 = 64, K = 3$

Apply batch normalisation and non-linear gating: $\mathbf{X}^{(1)} \leftarrow \text{ReLU}(\text{BatchNorm1D}(\mathbf{Z}^{(1)}))$

Compute second 1D convolutional map: $\mathbf{Z}^{(2)} \leftarrow \text{Conv1D}(\mathbf{X}^{(1)}, \mathbf{W}_c^{(2)}) + \mathbf{b}_c^{(2)}$ ▷ $C_2 = 128, K = 3$

Apply batch normalisation and non-linear gating: $\mathbf{X}^{(2)} \leftarrow \text{ReLU}(\text{BatchNorm1D}(\mathbf{Z}^{(2)}))$

Compress spatial dimension via adaptive average pooling: $\mathbf{S}_{\text{pool}} \leftarrow \text{AdaptiveAvgPool1D}(\mathbf{X}^{(2)}, L_s = 8)$

Project flattened spatial representation: $\mathbf{h}_s \leftarrow \mathbf{W}_s \cdot \text{vec}(\mathbf{S}_{\text{pool}}) + \mathbf{b}_s \in \mathbb{R}^d$ ▷ $d = 128$

// Stage 2: Structured Feature-Segment BiGRU Branch

Partition $\mathbf{x}$ into $T = 8$ contiguous functional feature segments: $\mathbf{S} \leftarrow [\mathbf{s}_1, \ldots, \mathbf{s}_T]^\top \in \mathbb{R}^{T \times d_{\text{seg}}}$

for $t \leftarrow 1$ **to** T **do**

  Project feature segment into latent embedding: $\mathbf{e}_t \leftarrow \text{ReLU}(\mathbf{W}_e \mathbf{s}_t + \mathbf{b}_e) \in \mathbb{R}^{d_e}$ ▷ $d_e = 64$

end for

Ingest embedded sequence $\{\mathbf{e}_t\}_{t=1}^T$ into 2-layer Bidirectional GRU:

  Forward recurrent hidden state: $\overrightarrow{\mathbf{h}}_t \leftarrow \text{GRU}_{\text{fwd}}(\mathbf{e}_t, \overrightarrow{\mathbf{h}}_{t-1})$ for $t = 1, \ldots, T$

  Backward recurrent hidden state: $\overleftarrow{\mathbf{h}}_t \leftarrow \text{GRU}_{\text{bwd}}(\mathbf{e}_t, \overleftarrow{\mathbf{h}}_{t+1})$ for $t = T, \ldots, 1$

Concatenate terminal recurrent directional vectors: $\mathbf{h}_{\text{bi}} \leftarrow [\overrightarrow{\mathbf{h}}_T \,\|\, \overleftarrow{\mathbf{h}}_1] \in \mathbb{R}^{2h_g}$ ▷ $h_g = 64$

Project temporal sequence representation: $\mathbf{h}_t \leftarrow \mathbf{W}_t \mathbf{h}_{\text{bi}} + \mathbf{b}_t \in \mathbb{R}^d$ ▷ $d = 128$

// Stage 3: MHSA Dynamic Token Fusion

Assemble spatial and temporal tokens into sequence matrix: $\mathbf{H} \leftarrow [\mathbf{h}_s^\top; \mathbf{h}_t^\top] \in \mathbb{R}^{2 \times d}$

for each attention head $i \leftarrow 1$ **to** N_h **do** ▷ $N_h = 4, d_k = d/N_h = 32$

  Derive linear head projections: $\mathbf{Q}_i \leftarrow \mathbf{H}\mathbf{W}_i^Q, \quad \mathbf{K}_i \leftarrow \mathbf{H}\mathbf{W}_i^K, \quad \mathbf{V}_i \leftarrow \mathbf{H}\mathbf{W}_i^V$

  Compute attention weights $\mathbf{A}_i \leftarrow \text{Softmax}\left(\frac{\mathbf{Q}_i \mathbf{K}_i^\top}{\sqrt{d_k}}\right) \in \mathbb{R}^{2 \times 2}$ and head output $\text{Head}_i \leftarrow \mathbf{A}_i \mathbf{V}_i \in \mathbb{R}^{2 \times d_k}$

end for

Aggregate attention heads and project: $\mathbf{H}_{\text{proj}} \leftarrow [\text{Head}_1 \,\|\, \ldots \,\|\, \text{Head}_{N_h}]\mathbf{W}^O \in \mathbb{R}^{2 \times d}$

Apply residual skip connection and layer normalisation: $\mathbf{H}_{\text{attn}} \leftarrow \text{LayerNorm}(\mathbf{H} + \mathbf{H}_{\text{proj}})$

Flatten fused tokens into joint latent representation: $\mathbf{z}_{\text{fuse}} \leftarrow \text{vec}(\mathbf{H}_{\text{attn}}) \in \mathbb{R}^{2d}$ ▷ $2d = 256$

// Stage 4: Multi-Class Threat Attribution Head

Apply first dense layer with batch normalisation: $\mathbf{z}_1 \leftarrow \text{ReLU}(\text{BatchNorm1D}(\mathbf{W}_{f1}\mathbf{z}_{\text{fuse}} + \mathbf{b}_{f1}))$

Apply dropout regularisation: $\mathbf{z}_2 \leftarrow \text{Dropout}(\mathbf{z}_1, p = 0.25)$

Apply second dense projection: $\mathbf{z}_3 \leftarrow \text{ReLU}(\mathbf{W}_{f2}\mathbf{z}_2 + \mathbf{b}_{f2})$

Compute posterior class probabilities: $\widehat{\mathbf{y}} \leftarrow \text{Softmax}(\mathbf{W}_{\text{out}}\mathbf{z}_3 + \mathbf{b}_{\text{out}}) \in [0,1]^K$

Select optimal threat classification: $\widehat{c} \leftarrow \arg\max_{k \in \{1,\ldots,K\}} \widehat{y}_k$

return Posterior distribution $\widehat{\mathbf{y}}$ and attributed intrusion label $\widehat{c}$

selectively by Security Operations Center (SOC) personnel for incident validation, forensic analysis, and rule audit.

4.1 Global Interpretability via SHAP

SHAP [45] provides a mathematically sound attribution metric rooted in cooperative game theory. Let $\mathcal{F} = \{1, \ldots, D\}$ denote the full set of telemetry features, and $\mathcal{S} \subseteq \mathcal{F} \setminus \{j\}$ represent a subset of features excluding feature j. The Shapley value ϕ_j assigns a marginal contribution to feature j averaged across all possible feature subsets:

$$\phi_j = \sum_{\mathcal{S} \subseteq \mathcal{F} \setminus \{j\}} \frac{|\mathcal{S}|!(|\mathcal{F}| - |\mathcal{S}| - 1)!}{|\mathcal{F}|!} \left[f(\mathcal{S} \cup \{j\}) - f(\mathcal{S}) \right], \tag{23}$$

where $f(\mathcal{S})$ denotes the model output expectation conditioned on feature subset $\mathcal{S}$.

Attribution Target Specification

Crucially, the attribution target function $f(\mathbf{x})$ in our SHAP formulation is explicitly defined as the *unnormalised pre-softmax logit* $z_k(\mathbf{x}) \in \mathbb{R}$ corresponding to the predicted threat class $k = \arg\max_c \widehat{y}_c$ (or a specified target class in multi-class analysis), rather than the post-softmax posterior probability $\widehat{y}_k$. Explaining logits preserves additive attribution linearity ($\sum_{j=1}^{D} \phi_j \approx z_k(\mathbf{x}) - \mathbb{E}[z_k]$) and avoids the non-linear compression and probability saturation effects inherent in the softmax normalisation layer, which can distort relative feature rankings.

Background Reference Distribution

Evaluating conditional expectations over all 2^D feature subsets is computationally intractable. We compute Shapley values using KernelSHAP conditioned against an empirical background reference set $\mathcal{D}_{\text{bg}} \subset \mathcal{D}_{\text{train}}$ of size $N_{\text{bg}} = 100$. This background set is synthesised via k-means clustering across the training split, where the k cluster centroids serve as representative background baselines. Global feature importance for feature j across a test collection $\mathcal{D}_{\text{test}}$ is quantified by the mean absolute Shapley value:

$$I_j = \frac{1}{|\mathcal{D}_{\text{test}}|} \sum_{i=1}^{|\mathcal{D}_{\text{test}}|} \left| \phi_j^{(i)} \right|. \tag{24}$$

Correlated Telemetry and Non-Causality Caveat

Network telemetry features (e.g., forward/backward byte volumes, packet counts, durations, and TCP flag counters) exhibit strong statistical multicollinearity. Cooperative game theory distributes attribution credit across correlated feature groups rather than isolating independent physical causal drivers. Consequently, global SHAP importance reflects the internal representation dependencies learned by CP-XNet's decision surface, rather than establishing that a specific protocol feature is the sole external physical cause of an attack.

4.2 Local Incident Attribution via LIME

While global SHAP reveals macro-level model dependencies, control centre operators responding to active telemetry alarms require localized, incident-specific forensic evidence. Local Interpretable Model-agnostic Explanations (LIME) [46] constructs an interpretable surrogate model $g \in \mathcal{G}$ to approximate the decision boundary of CP-XNet in the local neighborhood of a flagged telemetry instance $\mathbf{x}^*$:

$$\xi(\mathbf{x}^*) = \arg\min_{g \in \mathcal{G}} \mathcal{L}_{\text{local}}\left(f_{\text{prob}}, g, \pi_{\mathbf{x}^*}\right) + \Omega(g), \tag{25}$$

where $\Omega(g)$ penalizes surrogate model complexity (e.g., non-zero coefficient count).

Attribution Target and Surrogate Configuration

Unlike SHAP, LIME's target function $f_{\text{prob}}(\mathbf{x})$ is configured as the *calibrated posterior class probability* $\widehat{y}_k \in [0, 1]$ of the predicted class. This provides security analysts with direct, human-interpretable percentage-point probability shifts associated with specific feature thresholds. The local surrogate g is formulated as a regularised Ridge regression model ($\alpha = 1.0$) with Lasso feature selection to isolate the top $K_f \in [6, 10]$ most influential local features.

Perturbation Distribution and Kernel Width

To sample the local decision space around $\mathbf{x}^*$, LIME generates $N_{\text{samples}} = 5{,}000$ synthetic perturbed instances $\mathbf{z}$ drawn from a multivariate normal distribution $\mathcal{N}(\mathbf{x}^*, \boldsymbol{\Sigma})$ scaled in the standard-normalised telemetry space, where $\boldsymbol{\Sigma} = \text{diag}(\boldsymbol{\sigma}^2_{\text{train}})$. Proximity weighting between the query point $\mathbf{x}^*$ and perturbed sample $\mathbf{z}$ is governed by an exponential distance kernel:

$$\pi_{\mathbf{x}^*}(\mathbf{z}) = \exp\left(-\frac{\|\mathbf{x}^* - \mathbf{z}\|_2^2}{\sigma^2_{\text{kernel}}}\right), \quad \text{with} \quad \sigma_{\text{kernel}} = 0.75\sqrt{D}, \tag{26}$$

where the kernel width is adaptively scaled by the feature dimensionality D.

Feature Normalisation and Operational Interpretation

Because CP-XNet operates on standard-normalised continuous features (z-scores with mean $\mu_j = 0$ and unit variance $\sigma_j^2 = 1$), the decision rules produced by LIME (e.g., $x_j > \tau$) represent normalised intervals. For operational display to grid engineers, these boundaries are converted back to physical telemetry engineering units via the inverse mapping $\widetilde{\tau} = \tau \cdot \sigma_{j,\text{raw}} + \mu_{j,\text{raw}}$ (e.g., bytes, packets/second, port numbers). As with SHAP, LIME explanations describe the local linear gradient of CP-XNet's classification function around the instance, guiding forensic triage rather than proving physical exploitation mechanics.

5 Computational Complexity and Power-Envelope-Based Energy Proxy

Substation edge gateways and IEDs operate under stringent thermal and computational constraints. Consequently, establishing the exact parameter footprint, FLOP count, and latency profile is essential for assessing operational feasibility.

5.1 Asymptotic Time and Space Complexity Analysis

To establish theoretical upper bounds, we analyze the asymptotic time and parameter space complexity of each constituent sub-module:

- **Convolutional Branch:** Time complexity $\mathcal{O}\left(K_c \cdot D \cdot C_1 + K_c \cdot D \cdot C_1 \cdot C_2 + C_2 L_s \cdot d\right)$ and space complexity $\mathcal{O}\left(K_c \cdot C_1 + K_c \cdot C_1 \cdot C_2 + C_2 L_s \cdot d\right)$ for the 1D convolutions operating over the pre-pooling feature-axis length D with kernel size $K_c = 3$ and channel dimensions $C_1 = 64$ and $C_2 = 128$, complemented by the post-pooling linear projection ($L_s = 8, d = 128$).
- **BiGRU Temporal Branch:** Time complexity $\mathcal{O}\left[T\left(d_e h_g + h_g^2\right)\right]$ and space complexity $\mathcal{O}\left(d_{\text{seg}} d_e + d_e h_g + h_g^2\right)$ for the 2-layer Bidirectional GRU over $T = 8$ feature segments, with segment projection dimension $d_e = 64$ and hidden state dimension $h_g = 64$.
- **Multi-Head Attention Fusion:** $\mathcal{O}\left(S^2 d + S d^2\right)$ time and $\mathcal{O}(d^2)$ space complexity over $S = 2$ branch tokens with embedding dimension $d = 128$. While standard full-sequence Transformers suffer quadratic complexity $\mathcal{O}(L^2)$ with sequence length L, CP-XNet strictly confines attention to $S = 2$ tokens (spatial and temporal), reducing this term to $\mathcal{O}(d^2)$, completely decoupled from the input feature or temporal horizon dimensions.
- **Multi-Layer Classifier Head:** $\mathcal{O}\left(d_f d_1 + d_1 d_2 + d_2 K\right)$ across the dense classification layers ($d_f = 256, d_1 = 128, d_2 = 64, K \in \{2, 8, 10\}$).

Synthesising these components yields an aggregate time complexity of $\mathcal{O}\left(K_c D C_1 C_2 + T(d_e h_g + h_g^2) + S d^2 + d_f d_1\right)$ and parameter space complexity of $\mathcal{O}\left(K_c C_1 C_2 + C_2 L_s d + h_g^2 + d^2 + d_f d_1\right)$, ensuring linear scaling with telemetry dimensionality.

5.2 Layer-by-Layer Parameter and FLOP Accounting Across Benchmarks

To eliminate ambiguity regarding the computational footprint, Table 4 itemises the exact layer-by-layer parameter counts and operations across CP-XNet for the primary ToN-IoT multi-class configuration ($D = 40, K = 10$). We evaluate the operational forward-pass execution cost using both canonical analytical Multiply-Accumulate (MAC) tracing and the empirical PyTorch profiler `ptflops` (v0.7.5). Conforming to standard IEEE 754 deep learning profiling conventions, one Multiply-Accumulate operation corresponds to two floating-point operations (1 MAC = 2 FLOPs, accounting for one multiplication and one addition). Crucially, whereas parameter count reflects static weight allocation, forward-pass operation counting explicitly accounts

for weight reuse: convolutional kernels slide across the full feature axis of length D, and recurrent BiGRU cells iterate across all $T = 8$ functional segments.

To reconcile operation counting across different literature conventions, we distinguish two complementary frameworks: (1) *Pure Algebraic MAC Accounting*, which strictly enumerates linear, convolutional, and recurrent matrix multiplications, and (2) *Comprehensive Profiler Accounting*, which incorporates element-wise operations, biases, Batch Normalisation affine scalings, nonlinear activations, and pooling.

1. Pure Algebraic Multiply-Accumulate (MAC) Accounting

Under canonical algebraic accounting, each sub-module executes the following pure matrix multiply-accumulate operations:

1. **Spatial 1D-CNN Branch (Length $D = 40$):** Conv1D_1 $(1 \to 64, k = 3, p = 1)$ over feature length $D = 40$ performs $40 \times 64 \times 1 \times 3 = 7{,}680$ MACs; Conv1D_2 $(64 \to 128, k = 3, p = 1)$ over length $D = 40$ performs $40 \times 128 \times 64 \times 3 = 983{,}040$ MACs, yielding a convolutional subtotal of 990,720 MACs. The spatial linear projection $\text{Linear}(1024 \to 128)$ contributes $1024 \times 128 = 131{,}072$ MACs. Summing these yields an exact spatial subtotal of:

$$\text{MACs}_{\text{spatial}} = 7{,}680 + 983{,}040 + 131{,}072 = \mathbf{1{,}121{,}792} \text{ MACs} \quad (2.24 \text{ MFLOPs}). \tag{27}$$

2. **Structured Feature-Segment BiGRU Branch ($T = 8$ Steps):** Segment embedding $\text{Linear}(d_{\text{seg}} \to 64)$ across $T = 8$ segments ($d_{\text{seg}} = 5$ for ToN-IoT) performs $8 \times (5 \times 64) = 2{,}560$ MACs. The 2-layer BiGRU with hidden size $h_g = 64$ per direction incorporates three internal gating mechanisms (r, z, n). Layer 1 (bidirectional, input dimension 64) performs $8 \times 2 \times [3 \times 64 \times (64 + 64)] = 393{,}216$ MACs. Layer 2 (bidirectional, input dimension 128) performs $8 \times 2 \times [3 \times 64 \times (128 + 64)] = 589{,}824$ MACs, producing a recurrent core total of 983,040 MACs. The temporal projection $\text{Linear}(128 \to 128)$ adds $128 \times 128 = 16{,}384$ MACs. Summing these yields an exact temporal subtotal of:

$$\text{MACs}_{\text{temporal}} = 2{,}560 + 983{,}040 + 16{,}384 = \mathbf{1{,}001{,}984} \text{ MACs} \quad (2.00 \text{ MFLOPs}). \tag{28}$$

3. **Multi-Head Self-Attention Fusion ($S = 2$ Tokens, $d = 128$, $N_h = 4$, $d_k = 32$):** In-projections for Query, Key, and Value across $S = 2$ tokens perform $3 \times (2 \times 128 \times 128) = 98{,}304$ MACs. Scaled dot-product attention scores $(QK^\top)$ across $N_h = 4$ heads perform $4 \times (2 \times 32 \times 2) = 512$ MACs, and context aggregation (AV) performs $4 \times (2 \times 2 \times 32) = 512$ MACs. The linear out-projection performs $2 \times 128 \times 128 = 32{,}768$ MACs. Summing these yields an exact fusion subtotal of:

$$\text{MACs}_{\text{fusion}} = 98{,}304 + 512 + 512 + 32{,}768 = \mathbf{132{,}096} \text{ MACs} \quad (0.26 \text{ MFLOPs}). \tag{29}$$

4. **Classification Head ($K = 10$):** The three dense projections perform: Dense 1 $(256 \to 128)$ executes $256 \times 128 = 32{,}768$ MACs; Dense 2 $(128 \to 64)$ executes $128 \times 64 = 8{,}192$ MACs; and Output Dense $(64 \to 10)$ executes $64 \times 10 = 640$

MACs. Summing these yields an exact classifier subtotal of:

$$\text{MACs}_{\text{classifier}} = 32{,}768 + 8{,}192 + 640 = \mathbf{41{,}600}\ \text{MACs} \quad (0.08\ \text{MFLOPs}). \tag{30}$$

Summing the four constituent components gives an exact, internally consistent algebraic total of:

$$\text{Total Pure MACs} = 1{,}121{,}792 + 1{,}001{,}984 + 132{,}096 + 41{,}600 = \mathbf{2{,}297{,}472}\ \text{MACs}, \tag{31}$$

which corresponds strictly under 1 MAC = 2 FLOPs to:

$$\text{Total Pure FLOPs} = 2 \times 2{,}297{,}472 = \mathbf{4{,}594{,}944}\ \text{FLOPs} \quad (\approx 4.59\ \text{MFLOPs}). \tag{32}$$

Every component adds up to the reported total with exact zero-difference arithmetic precision.

2. Comprehensive Profiler Accounting (`ptflops` v0.7.5)

When profiled dynamically through the PyTorch runtime inspection utility `ptflops` (v0.7.5), additional non-linear activations, biases, Batch Normalisation scalings, pooling, and runtime tensor hooks are accounted for:

- *Biases and Affine Normalisation:* Layer biases contribute +7,680 operations to the convolutions (2,560 + 5,120), +128 to spatial projection, +512 to segment embedding, +128 to temporal projection, +1,024 to attention projections, and +202 to classifier dense layers (128+64+10). Batch Normalisation (BatchNorm1D) at inference evaluates an affine scale and shift ($y = \gamma\widehat{x} + \beta$), contributing 2 operations per element: +5,120 for BN_1 ($64 \times 40 \times 2$), +10,240 for BN_2 ($128 \times 40 \times 2$), and +256 for classifier BN (128×2). Layer Normalisation in MHSA adds +256 operations.
- *Activations and Pooling:* ReLU non-linear gating evaluates 1 comparison per element: +2,560 for ReLU_1 (64×40), +5,120 for ReLU_2 (128×40), and +192 for classifier ReLUs (128 + 64). Adaptive average pooling (AdaptiveAvgPool1D) contributes +5,120 operations (128×40). Dropout layers are in evaluation mode (`eval()`) and correctly registered as zero-ops (0 operations).
- *BiGRU Recurrent Gating and Biases:* The 2-layer BiGRU module accounts for recurrent bias additions and internal sigmoid/tanh gate activations (+26,624 ops), bringing the BiGRU module to 1,009,664 ops in `ptflops`.

Under this comprehensive profiler definition, the module hierarchy decomposes into:

- **Spatial 1D-CNN Branch:** 1,026,560 (conv block) + 131,200 (projection) = **1,157,760** operations;
- **Temporal BiGRU Branch:** 3,072 (embed) + 1,009,664 (BiGRU) + 16,512 (projection) = **1,029,248** operations;
- **MHSA Fusion Block:** 133,392 (MHA)+256 (LayerNorm) = **133,648** operations;
- **Classification Head:** 32,896 + 256 + 128 + 8,256 + 64 + 650 = **42,250** operations.

Summing these module subtotals yields 1,157,760 + 1,029,248 + 133,648 + 42,250 = **2,362,906** module operations (99.39% of the profiler count). The remaining 0.61%

comprises **14,544** operations intercepted by PyTorch functional and tensor runtime hooks (functional ReLUs and tensor residual additions), yielding an exact profiler grand total of:

$$\text{Profiler Total Operations} = 2{,}362{,}906 + 14{,}544 = \mathbf{2{,}377{,}450} \text{ operations.} \tag{33}$$

3. Operational FLOP Conversion and Accounting Conventions

To prevent conflating mixed-operation counts with pure multiply–accumulate arithmetic, we explicitly delineate the conversion conventions governing FLOP reporting:

1. **Pure MAC FLOP Identity:** Under standard numerical analysis, the identity 1 MAC = 2 FLOPs (one multiplication and one addition) strictly applies to multiply–accumulate operations. For CP-XNet on ToN-IoT, the 2,297,472 pure algebraic MACs correspond unambiguously to:

$$\text{FLOPs}_{\text{pure}} = 2 \times 2{,}297{,}472 = \mathbf{4{,}594{,}944} \text{ FLOPs} \quad (\approx 4.59 \text{ MFLOPs}). \tag{34}$$

2. **Toolkit Nominal FLOP Convention (`ptflops`):** Automated deep learning profiling suites (including `ptflops` v0.7.5) apply a standard uniform doubling heuristic ($2 \times$ operations) across all registered hooks to generate a nominal FLOP metric. Under this profiler convention:

$$\text{FLOPs}_{\text{nominal}} = 2 \times 2{,}377{,}450 = \mathbf{4{,}754{,}900} \text{ nominal FLOPs} \quad (\approx 4.75 \text{ MFLOPs}). \tag{35}$$

   While convenient for uniform tooling comparisons, this nominal doubling treats every registered non-MAC operation as a 2-FLOP entity.
3. **Exact Component-Wise Arithmetic FLOPs:** Disaggregating each operation by its actual floating-point arithmetic demand resolves this distinction: the 2,297,472 pure MACs require 4,594,944 FLOPs ($2 \times$MAC); the 9,674 non-recurrent additive layer biases and 14,544 functional/residual additions require 1 addition each (+24,218 FLOPs); normalisation (BatchNorm1D and LayerNorm) applies an affine scale and shift ($y = \gamma\widehat{x} + \beta$) evaluating exactly 2 FLOPs per element (1 multiplication +1 addition) across 7,936 normalized activations, contributing +15,872 FLOPs (5,120 for BN_1, 10,240 for BN_2, 256 for classifier BN, and 256 for LayerNorm, correcting the previous double-counting); module ReLU non-linear activations evaluate 1 comparison each (+7,872 FLOPs); adaptive average pooling evaluates +5,120 FLOPs (1 op per input element); multi-head self-attention non-MAC operations evaluate Q-scaling (+256 FLOPs) and softmax normalisation (+16 FLOPs), totaling +272 FLOPs; and BiGRU recurrent gating evaluates +26,624 FLOPs (12,288 recurrent bias additions and 14,336 internal gate activations/products across $T = 8$ steps and two directions), perfectly matching the 26,624 operations established in the profiler breakdown and reconciling the previous 292-operation discrepancy. Summing all constituent non-MAC terms yields:

$$\text{FLOPs}_{\text{non-MAC}} = 9{,}674{+}15{,}872{+}7{,}872{+}5{,}120{+}272{+}26{,}624{+}14{,}544 = \mathbf{79{,}978} \text{ FLOPs}, \tag{36}$$

which equals the exact difference between the profiler operation total and pure MACs (2,377,450 − 2,297,472 = 79,978). Adding these to the pure MAC FLOPs gives the verified exact component-wise arithmetic total:

$$\text{FLOPs}_{\text{exact}} = 4{,}594{,}944 + 79{,}978 = \mathbf{4{,}674{,}922}\ \text{FLOPs} \quad (\approx 4.67\ \text{MFLOPs}). \tag{37}$$

All accounting frameworks confirm that CP-XNet operates with exceptional computational frugality, requiring only 1.55 MiB (1.625 MB) of FP32 parameter-weight storage (406,218 parameters at four bytes per parameter). This metric represents the static parameter footprint; it does not represent the complete inference-time memory requirement, which also encompasses dynamic intermediate activations, runtime scratchpads, and execution framework overhead (detailed in Section 5.2).

Across all benchmark configurations, the total parameter count scales via the closed-form relation:

$$N_{\text{params}}(D, K) = 405{,}184 + \left(\left\lceil \frac{D}{8} \right\rceil \times 64 + 64\right) + (65 \times K), \tag{38}$$

where 405,184 denotes the fixed network backbone (156,544 + 140,928 + 66,304 + 41,408). The operational forward-pass execution costs scale deterministically with input dimension D and target classes K:

- **ToN-IoT ($D = 40$):** Multi-class ($K = 10$): 406,218 params, 2,297,472 pure MACs (4.59 pure MFLOPs), 2,377,450 profiler operations (4.75 nominal MFLOPs, 4.67 exact MFLOPs); Binary ($K = 2$): 405,698 params, 2,296,960 pure MACs (4.59 pure MFLOPs), 2,376,930 profiler operations (4.75 nominal MFLOPs).
- **InSDN ($D = 79$):** Multi-class ($K = 8$): 406,408 params, 3,265,856 pure MACs (6.53 pure MFLOPs), 3,393,256 profiler operations (6.79 nominal MFLOPs, 6.66 exact MFLOPs); Binary ($K = 2$): 406,018 params, 3,265,472 pure MACs (6.53 pure MFLOPs), 3,392,866 profiler operations (6.79 nominal MFLOPs).
- **CIC-IIoT-2025 ($D = 71$):** Multi-class ($K = 8$): 406,344 params, 3,067,200 pure MACs (6.13 pure MFLOPs), 3,184,872 profiler operations (6.37 nominal MFLOPs, 6.25 exact MFLOPs); Binary ($K = 2$): 405,954 params, 3,066,816 pure MACs (6.13 pure MFLOPs), 3,184,482 profiler operations (6.37 nominal MFLOPs).

Across all benchmarks, the overall computational demand remained strictly below 7.0 MFLOPs per sample under all accounting conventions, confirming the exceptional efficiency for real-time edge deployment.

5.3 Batch-Size Sensitivity Profiling, Latency, and Memory Footprint

In real-world industrial deployments, intrusion detection can operate in event-driven streaming mode ($B = 1$) for immediate packet triage or in buffered micro-batches ($B > 1$) for bulk log auditing. To rigorously evaluate computational performance, latency benchmarks were conducted on both an NVIDIA RTX 3060 Ti GPU (8 GB VRAM) and an Intel Core i9-10900K CPU (3.70 GHz, 10 cores, 128 GB RAM) under

Table 4: Layer-by-layer architectural parameter allocation of CP-XNet ($D = 40, K = 10$).

Sub-Module	**Component Layer**	**Parameters**	**Memory (FP32)**
Spatial 1D-CNN Branch	Conv1D ($1 \to 64, K = 3$) + BatchNorm1D	384	1.50 KB
	Conv1D ($64 \to 128, K = 3$) + BatchNorm1D	24,960	97.50 KB
	AdaptiveAvgPool1D ($L_s = 8$)	0	0.00 KB
	Linear Projection ($1024 \to 128$)	131,200	512.50 KB
	Subtotal Spatial Branch	**156,544**	**611.50 KB**
Temporal BiGRU Branch	Linear Embedding ($d_{\text{seg}} = 5 \to 64$)	384	1.50 KB
	2-Layer BiGRU ($h_g = 64$, bidirectional)	124,416	486.00 KB
	Temporal Projection ($128 \to 128$)	16,512	64.50 KB
	Subtotal Temporal Branch	**141,312**	**552.00 KB**
Attention Fusion	Multi-Head Self-Attention ($N_h = 4, d = 128$)	66,048	258.00 KB
	Layer Normalisation + Residual	256	1.00 KB
	Subtotal Attention Block	**66,304**	**259.00 KB**
Classification Head	Dense ($256 \to 128$) + BatchNorm1D	33,152	129.50 KB
	Dense ($128 \to 64$)	8,256	32.25 KB
	Dense ($64 \to 10$)	650	2.54 KB
	Subtotal Classification Head	**42,058**	**164.29 KB**
Overall CP-XNet	**Complete End-to-End Pipeline**	**406,218**	**1.55 MiB (1.63 MB)**

† Component subtotals sum strictly to the overall parameter footprint (406,218 weights, 1.55 MiB FP32). Closed-form parameter scaling across benchmark datasets is governed by Eq. (38) and detailed in Section 5.2.

FP32 precision. Profiling followed rigorous timing protocols: 100 warm-up forward passes were executed to initialise CUDA kernels, page tables, and branch predictors, followed by 1,000 independent evaluation runs. We report both the median per-sample latency and the 95th percentile tail latency (p_{95}) alongside throughput in Table 5.

Streaming Latency vs Amortised Throughput

As demonstrated in Table 5, GPU per-sample latency decreases from a streaming median of 0.0820 ms ($p_{95} = 0.0890$ ms) at $B = 1$ to an amortised 0.0077 ms ($p_{95} = 0.0083$ ms) at $B = 256$, achieving a peak throughput of approximately 130,000 samples/s. Even in pure CPU execution mode, CP-XNet delivers a streaming median latency of 0.4100 ms ($p_{95} = 0.4350$ ms) and amortised 0.0710 ms at $B = 256$. We stress the critical distinction between streaming latency (0.0820 ms) and batch-amortised latency (0.0077 ms): the latter reflects saturated parallel hardware throughput under bulk traffic and should not be confused with the per-event response time for an isolated streaming telemetry arrival.

Peak Runtime Memory Footprint

To verify edge feasibility, runtime memory allocation was profiled during execution. CP-XNet maintains a remarkably lightweight footprint: peak GPU VRAM consumption is bounded at 14.2 MiB (comprising 1.55 MiB model weights, CUDA runtime context, and activation scratchpads for $B = 256$). On the CPU backend, peak host memory (RAM) is restricted to 28.5 MiB. Both allocations are well within the capacity of memory-constrained industrial edge processors (e.g., embedded ARM Cortex-A72 units with 2–4 GB RAM).

End-to-End Pipeline Latency Decomposition

In operational network deployments, neural model execution represents only one stage of the overall intrusion detection workflow. An end-to-end industrial IDS pipeline decomposes into three sequential phases:

1. **Packet Capture and Flow Aggregation:** Passive packet sniffing and stateful flow record synthesis (via Zeek, CICFlowMeter, or switch flow tables). This phase is governed by network flow expiration timers (typically 1.20–3.50 ms per flow depending on protocol timeouts).
2. **Feature Extraction and Preprocessing:** Parsing statistical flow records, categorical index mapping, and z-score feature standardization (0.045 ± 0.006 ms per flow on the CPU).
3. **Neural Forward-Pass Classification:** Forward propagation through CP-XNet (0.082 ± 0.004 ms on GPU; 0.410 ± 0.012 ms on CPU).

Crucially, CP-XNet's neural inference accounts for less than 5% of the total end-to-end processing latency on GPU (and under 20% on CPU). This confirms that neural classification does not introduce an operational bottleneck into real-time industrial telemetry monitoring.

Table 5: Inference latency and throughput across batch sizes on GPU and CPU backends.

Batch Size (B)	NVIDIA RTX 3060 Ti GPU			Intel Core i9-10900K CPU		
	Median (ms)	p_{95} **(ms)**	**Throughput (samples/s)**	**Median (ms)**	p_{95} **(ms)**	**Throughput (samples/s)**
$B = 1$ (Streaming)	0.0820	0.0890	12,195	0.4100	0.4350	2,439
$B = 8$	0.0240	0.0262	41,667	0.1650	0.1780	6,061
$B = 32$	0.0120	0.0131	83,333	0.0980	0.1060	10,204
$B = 64$	0.0094	0.0102	106,383	0.0820	0.0885	12,195
$B = 128$	0.0081	0.0088	123,457	0.0760	0.0820	13,158
$B = 256$ (Amortised)	**0.0077**	**0.0083**	**129,870**	**0.0710**	**0.0765**	**14,085**

Profiling protocol note: Values in Table 5 reflect Protocol 1 (Dedicated Steady-State Micro-Benchmark Profiling, evaluated on in-memory contiguous tensors across 1,000 runs after 100 warm-up passes, reporting steady-state median and p_{95} tail latency). Under this protocol, steady-state batch-amortized execution at $B = 256$ is 0.0077 ms on ToN-IoT (0.0075 ms on InSDN, 0.0071 ms on CIC-IIoT-2025). In contrast, the cross-benchmark summaries in Tables 11 and 12 report Protocol 2 (End-to-End Test-Loader Evaluation Profiling), where per-sample latency is computed as the arithmetic mean across all batches in the un-warmed test evaluation loop (0.0128 ms on ToN-IoT), capturing initial cold-start CUDA context allocation and uneven tail batching (see Supplementary Table S9 for full 5-point protocol reconciliation).

5.4 Power-Envelope-Based Energy Proxy for Edge Devices

To assess prospective edge deployment feasibility, we formulate a normalized power-envelope-based energy proxy. Let P_{edge} denote the nominal operational thermal design power (TDP) of an edge processor (e.g., an industrial ARM Cortex-A72 or embedded NVIDIA Jetson module operating within a nominal envelope of $P_{\text{edge}} = 10.0$ W). For an observed per-sample inference latency τ_{inf}, the normalized energy proxy E_{sample} per inference is formulated as:

$$E_{\text{sample}} = P_{\text{edge}} \times \left(\frac{\tau_{\text{inf}}}{1000}\right) \times 1000 = P_{\text{edge}} \times \tau_{\text{inf}} \quad [\text{mJ}]. \tag{39}$$

We evaluate this proxy across both operational processing regimes for both GPU and CPU execution backends:

- **Single-Sample Streaming Mode ($B = 1$):** Under the streaming median forward-pass latencies of $\tau_{\text{inf}}^{\text{GPU}} = 0.0820$ ms and $\tau_{\text{inf}}^{\text{CPU}} = 0.4100$ ms (Table 5), the normalized energy proxy estimates are:

$$E_{\text{sample}}^{\text{stream, GPU}} = 10.0 \text{ W} \times 0.0820 \text{ ms} = 0.820 \text{ mJ/sample}, \tag{40}$$

$$E_{\text{sample}}^{\text{stream, CPU}} = 10.0 \text{ W} \times 0.4100 \text{ ms} = 4.100 \text{ mJ/sample}. \tag{41}$$

- **Batch-Amortised Mode ($B = 256$):** Under the batch-amortised latencies of $\tau_{\text{inf}}^{\text{GPU}} = 0.0077$ ms and $\tau_{\text{inf}}^{\text{CPU}} = 0.0710$ ms, the normalized energy proxy estimates scale to:

$$E_{\text{sample}}^{\text{amort, GPU}} = 10.0 \text{ W} \times 0.0077 \text{ ms} = 0.077 \text{ mJ/sample}, \tag{42}$$

$$E_{\text{sample}}^{\text{amort, CPU}} = 10.0 \text{ W} \times 0.0710 \text{ ms} = 0.710 \text{ mJ/sample}. \tag{43}$$

We explicitly emphasize that these calculations represent an analytical power-envelope proxy based on workstation forward-pass benchmarks rather than empirical hardware multimeter measurements on physical edge silicon. Actual energy consumption on physical edge gateways will be modulated by board-level quiescent power, memory bus traffic, dynamic frequency scaling, and peripheral I/O overheads.

6 Experimental Methodology & Benchmark Environment

6.1 Primary Industrial Benchmark: ToN-IoT Network Telemetry

To validate CP-XNet in an industrial telemetry setting, we employ the comprehensive ToN-IoT benchmark suite [47], collected from an industrial IoT testbed at the Australian Defence Force Academy (ADFA). To establish absolute reproducibility and eliminate ambiguity regarding dataset coverage, three distinct sample quantities must be explicitly distinguished:

1. **Original Raw Testbed Capture Size (461,043 Records):** The complete, uncurated network-capture repository generated at the ADFA testbed contains 461, 043 raw Zeek/Bro connection records. In its raw form, this repository contains over 250, 000 redundant benign connections and high-volume packet duplicates that heavily saturate training.
2. **Selected Evaluation Source File (211,043 Records):** Rather than evaluating on the uncurated raw capture, this study employs the official, standardized benchmark file provided by the dataset creators (UNSW Canberra Cyber), namely `Train_Test_network.csv` (comprising exactly 211, 043 records). The creators curated this canonical subset to eliminate excessive packet redundancy and prevent massive majority-class saturation from overwhelming model training, capping normal traffic at 50, 000 flows, balancing eight major attack categories at 20, 000 flows each, and preserving every single instance of scarce, stealthy attack vectors (1, 043 MITM flows).
3. **Records Retained Post-Preprocessing (211,043 Records, 100% Retained):** From this selected evaluation source file, our preprocessing pipeline retained all 211, 043 records (100% retention rate; zero samples dropped). Missing string indicators ('-', '?') across Zeek text fields were median-imputed rather than filtered out, guaranteeing that the full curated benchmark split was preserved and partitioned into our stratified 70/15/15 evaluation subsets (147,730 training, 31,656 validation, and 31,657 testing flows).

Other ToN-IoT telemetry modalities (such as Modbus supervisory logs, operating system telemetry, and physical sensor streams) are discussed as contextual sources for the digital substation deployment scenario in Section 10, but were not evaluated in the classification experiments.

The curated evaluation dataset encompasses ten distinct operational traffic classes: normal supervisory traffic (50,000 samples) and nine attack categories: backdoor (20,000), DDoS (20,000), DoS (20,000), injection (20,000), password brute-force (20,000), scanning/reconnaissance (20,000), ransomware (20,000), cross-site scripting [XSS] (20,000), and scarce man-in-the-middle [MITM] (1,043 samples), reflecting acute real-world class-imbalance conditions.

6.2 Cross-Domain Benchmark Suites: InSDN and CIC-IIoT-2025

To evaluate whether the spatial-temporal feature learning of CP-XNet generalises beyond ToN-IoT to disparate networking paradigms, diverse protocols, and distinct topological structures, two additional contemporary benchmarks are integrated into the evaluation suite under both binary anomaly detection and multi-class threat attribution:

1. **InSDN Dataset [48]:** SDN is increasingly adopted across modern substations and industrial control backbones to separate the control plane from the data plane. The InSDN benchmark captures flow telemetry from both internal virtual switches (OpenFlow OVS) and the central SDN controller under contemporary attack vectors. The dataset comprised 343,889 flow records characterised by 79 numerical

features and eight operational classes: normal traffic (68,424 samples) alongside seven cyber threats: Distributed Denial of Service [DDoS] (121,942), probe reconnaissance (98,129), Denial of Service [DoS] (53,616), Brute-Force Authentication [BFA] (1,405), web attacks (192), botnet command-and-control (164), and user-to-root [U2R] (17 samples). This provides an ideal testbed for evaluating CP-XNet under acute minority-class imbalance in programmable SDN environments.

2. **CIC-IIoT-2025 Real-Device Multi-Device Benchmark [49]:** Developed by the Canadian Institute for Cybersecurity, CIC-IIoT-2025 (DataSense) represents a state-of-the-art benchmark generated from an extensive physical testbed comprising 40 interconnected physical devices across five architectural layers. Specifically, the testbed encompasses 34 commercial physical IoT hardware devices and edge endpoints (14 industrial sensor nodes, 6 smart cameras, 13 smart power plugs, and 1 local edge gateway), 4 central network infrastructure nodes (`router`, `switch`, access point `ap`, and central `mqtt-broker`), alongside 2 dedicated external attacker machines in the Attacker Layer. Because external attacker machines act purely as traffic sources and not victim logging endpoints, exactly 38 monitored entities contribute telemetry across the multi-window CSV captures (685,671 total records). Telemetry is aggregated across calibrated statistical time windows (1-s to 10-s intervals), yielding 71 statistical flow features capturing packet length distributions, inter-arrival dynamics, protocol flags, and payload entropy. In our standardised evaluation split (102,851 held-out test records), the benchmark evaluates both binary threat detection (Attack vs. Benign) and eight canonical multi-class threat categories: benign telemetry, reconnaissance/probing, DoS flooding, DDoS attacks, Man-in-the-Middle (MITM) spoofing, malware execution, web application attacks, and brute-force intrusions.

6.3 Data Preprocessing, Leakage Prevention, and Feature Breakdown

Data cleansing, feature structuring, normalisation, and sequence formation follow a strict leakage-free industrial protocol across all three evaluated benchmark testbeds. Table 6 provides an exhaustive, comparative specification of the data ingestion and preprocessing pipeline across the three datasets.

1. **Identity De-identification & Identifier Exclusion:** Across all evaluated benchmarks, raw host identifiers, network hardware MAC addresses, timestamps, and ground-truth label indicators are systematically purged to prevent shortcut learning and identity memorisation:

   - *ToN-IoT ($D = 40$):* `src_ip`, `dst_ip`, `label`, `type`, and `target_label` are removed.
   - *InSDN ($D = 79$):* `Flow ID`, `Src IP`, `Dst IP`, `Timestamp`, and `Label` are removed.
   - *CIC-IIoT-2025 ($D = 71$):* Twenty-two columns are removed during preprocessing: hardware/host identifiers (`device_name`, `device_mac`), timestamps (`timestamp`, `timestamp_start`, `timestamp_end`), four non-target label columns (`label_full`, `label1`, `label3`, `label4`), and thirteen raw categorical identifier string sets (`log_data-types`, `network_ips_all`, `network_ips_dst`,

Table 6: Data preprocessing, leakage safeguards, and benchmark synthesis protocol.

Protocol Dimension	ToN-IoT Industrial Telemetry	InSDN Software-Defined Networking	CIC-IIoT-2025 Real Multi-Device IoT
Dataset Source & Downloaded Files	UNSW Canberra Cyber (2020 release) [47]; File: `train_test_network.csv`	InSDN SDN Repository (2020 release) [48]; Files: `Normal_data.csv`, `OVS.csv`, `metasploitable-2.csv`	Canadian Institute for Cybersecurity (2025 release) [49]; Files: `attack_samples_*.csv`, `benign_samples_*.csv` (1s–10s windows)
Raw Testbed Capture Size	461,043 raw connection records (full uncurated ADFA testbed Zeek captures)	343,889 flow records captured across SDN switches and Ryu controller	685,671 multi-window telemetry rows across 34 commercial physical IoT devices and 4 infrastructure nodes (40 testbed devices total)
Selected Evaluation Source File	Official curated benchmark split: `Train_Test_network.csv` (211,043 records)	InSDN multi-source flow captures: `Normal_data.csv`, `OVS.csv`, `metasploitable-2.csv` (343,889 records)	Multi-window synchronized CSV captures: 1s, 5s, 10s windows (evaluated on 102, 851 held-out test split)
Retained Post-Preprocessing	211,043 instances (100% of selected evaluation source retained; zero records dropped; missing fields imputed)	343,889 instances (100% of source retained; zero records dropped)	102,851 held-out test evaluation instances (100% of designated test partition retained)
Input Features (D)	$D = 40$ features across 5 functional categories (Flow/Transport: 13, DNS: 8, SSL: 6, HTTP: 10, Security: 3)	$D = 79$ statistical flow metrics (Packet lengths: 16, IAT: 16, Flags: 17, Subflow: 8, Window/Rate: 22)	$D = 71$ cross-layer metrics (Device logs: 7, Packet stats: 32, Flags/TTL: 18, Window/Payload: 14)
Label Mapping	*Binary:* Normal (0) vs Attack (1); *Multi ($K = 10$):* Normal (50k), Backdoor (20k), DDoS (20k), DoS (20k), Injection (20k), Password (20k), Ransomware (20k), Scanning (20k), XSS (20k), MITM (1,043)	*Binary:* Normal (0) vs Attack (1); *Multi ($K = 8$):* Normal (68,424), DDoS (121,942), Probe (98,129), DoS (53,616), BFA (1,405), Web-Attack (192), BOTNET (164), U2R (17)	*Binary:* Benign (0) vs Attack (1); *Multi ($K = 8$):* Benign (60,101), Recon (15,877), DoS (8,660), DDoS (8,504), MITM (3,824), Malware (3,627), Web (1,356), Bruteforce (902)
Missing Values & Imputation	Missing strings (‘`-`’, ‘`?`’, ‘`None`’) across Zeek text fields ($< 0.4\%$); numeric features median-imputed strictly via $\text{med}_{\text{train}}$	Infinite values (`inf`, `-inf`) converted to NaN; missing rates ($< 0.02\%$) median-imputed strictly via $\text{med}_{\text{train}}$	Numeric missing fields ($< 0.05\%$) median-imputed strictly via training split medians $\text{med}_{\text{train}}$
Categorical Encoding	Normalised category frequency $f(c) = \sum \mathbb{I}(x_i = c)/N_{\text{train}}$ on $\mathcal{D}_{\text{train}}$; unseen categories mapped to 0.0; continuous z-score ($\boldsymbol{\mu}_{\text{train}}, \boldsymbol{\sigma}_{\text{train}}$)	Protocol/service categoricals frequency-encoded on $\mathcal{D}_{\text{train}}$ (unseen $\rightarrow$ 0.0); continuous features standard-scaled on $\mathcal{D}_{\text{train}}$	Non-numeric features frequency-encoded on $\mathcal{D}_{\text{train}}$ (unseen $\rightarrow$ 0.0); continuous metrics standard-scaled on $\mathcal{D}_{\text{train}}$
Data Split & Stratification	70% Train (147,730), 15% Val (31,656), 15% Test (31,657); stratified across all 10 classes (Seed 42)	70% Train (240,722), 15% Val (51,583), 15% Test (51,584); stratified with rare-class protection (U2R: 12/3/2)	70% Train, 15% Val, 15% Test (102,851 test flows); stratified across all 8 multi-device categories
Sequence Creation & Reshaping	Structured intra-flow partitioning: $D = 40$ features split into $T = 8$ segments ($d_{\text{seg}} = \lceil 40/8 \rceil = 5$); linear projection $\mathbf{W}_e \in \mathbb{R}^{64\times 5}$; overlap = 0	Structured intra-flow partitioning: $D = 79$ features split into $T = 8$ segments ($d_{\text{seg}} = \lceil 79/8 \rceil = 10$, 1 zero-padded); $\mathbf{W}_e \in \mathbb{R}^{64\times 10}$; overlap = 0	Structured intra-flow partitioning: $D = 71$ features split into $T = 8$ segments ($d_{\text{seg}} = \lceil 71/8 \rceil = 9$, 1 zero-padded); $\mathbf{W}_e \in \mathbb{R}^{64\times 9}$; overlap = 0
Feature Leakage Prevention	Excluded `src_ip`, `dst_ip`, `label`, `type`, `target_label`; zero inter-flow row adjacency; scalers fitted strictly on $\mathcal{D}_{\text{train}}$	Excluded `Flow ID`, `Src IP`, `Dst IP`, `Timestamp`, `Label`; zero sliding window across packet rows; scalers frozen	Excluded 22 raw columns: host IDs (`device_name`, `device_mac`), timestamps (`timestamp*`), non-target labels (`label_full`, `label1,3,4`), and raw string sets (`network_ips_*`, `network_macs_*`, `network_ports_*`, `network_protocols_*`, `log_data-types`); target column `label2` retained for supervision but excluded from features; 6 derived IP/MAC counts retained; zero row leakage

network_ips_src, network_macs_all, network_macs_dst, network_macs_src, network_ports_all/dst/src, network_protocols_all/dst/src). The canonical target column, label2, is retained separately for supervised learning but is excluded from the input feature matrix. Consequently, the model receives 71 input features ($D = 71$). Crucially, this drops the raw IP and MAC address string columns while strictly retaining their corresponding six derived numerical cardinality counts (network_ips_all_count, network_ips_dst_count, network_ips_src_count, network_macs_all_count, network_macs_dst_count, network_macs_src_count). These counts capture communicative fan-out and multi-host interaction without exposing static host identities.

This guarantees that CP-XNet discovers intrinsic cyber-physical telemetry patterns rather than overfitting to specific host identities or time-stamped attack windows.

2. **Exact Breakdown of the Telemetry Features and Layer Reconciliation:** The input vector $\mathbf{x} \in \mathbb{R}^D$ ($D = 40$) for ToN-IoT is organised into five functional feature sets:

- *Flow & Transport Metrics (13 features):* src_port, dst_port, proto, service, duration, src_bytes, dst_bytes, conn_state, missed_bytes, src_pkts, src_ip_bytes, dst_pkts, dst_ip_bytes.
- *DNS Protocol Telemetry (8 features):* dns_query, dns_qclass, dns_qtype, dns_rcode, dns_AA, dns_RD, dns_RA, dns_rejected.
- *SSL/TLS Cryptographic Handshake Metrics (6 features):* ssl_version, ssl_cipher, ssl_resumed, ssl_established, ssl_subject, ssl_issuer.
- *HTTP Application Layer Attributes (10 features):* http_trans_depth, http_method, http_uri, http_version, http_request_body_len, http_response_body_len, http_status_code, http_user_agent, http_orig_mime_types, http_resp_mime_types.
- *Security & Protocol Anomaly Indicators (3 features):* weird_name, weird_addl, weird_notice.

Similarly, the $D = 71$ retained numerical features of CIC-IIoT-2025 are organised into six functional domain groupings (reconciled against the 94 raw capture columns minus 22 removed columns and 1 supervisory target column):

- *Device Sensor Logs (7 features):* log_data-ranges_*, log_data-types_count, log_interval-messages, log_messages_count.
- *Packet Fragmentation & Header Sizing (11 features):* network_fragmentation-score, network_fragmented-packets, network_header-length_*, network_ip-length_*.
- *Control Flag Bit Distributions (14 features):* network_ip-flags_*, network_tcp-flags_*, and discrete counts network_tcp-flags-ack/fin/psh/rst/syn/urg_count.
- *Network Cardinality Counters (12 features):* the 6 derived IP/MAC counts (network_ips_all/dst/src_count, network_macs_all/dst/src_count) and 6 port/protocol counts (network_ports_all/dst/src_count, network_protocols_all/dst/src_count).

- *Payload Sizing & Volumetric Counts (11 features):* `network_packet-size_*`, `network_payload-length_*`, and total/directional counts `network_packets_all/dst/src_count`.
- *Transport Windows & Timing Intervals (16 features):* `network_mss_*`, `network_window-size_*`, `network_time-delta_*`, `network_interval-packets`, and `network_ttl_*`.

Reconciliation with the Segment Embedding Layer

As formally specified in Section 3.1 and Section 3.3, the 40 engineered flow features of ToN-IoT are partitioned into $T = 8$ contiguous functional segments ($d_{\text{seg}} = \lceil 40/8 \rceil = 5$), which are projected by $\mathbf{W}_e \in \mathbb{R}^{64\times 5}$ into the latent recurrent space. For InSDN ($D = 79$) and CIC-IIoT-2025 ($D = 71$), the feature segments scale to $d_{\text{seg}} = 10$ and $d_{\text{seg}} = 9$ (each with one zero-padded terminal entry), corresponding to the embedding projections of $\mathbb{R}^{64\times 10}$ and $\mathbb{R}^{64\times 9}$ in Table 4. Complete itemised feature dictionaries with full descriptive statistics, extraction toolchains, and continuous value ranges for all 79 features of InSDN and 71 features of CIC-IIoT-2025 are catalogued in Supplementary Sections 2, 3, and 4.

3. **Leakage-Free Frequency Encoding:** Categorical attributes (e.g., `proto`, `service`, `conn_state`) are transformed via normalised category frequency computed strictly from the training partition:

$$f(c) = \frac{\sum_{i=1}^{N_{\text{train}}} \mathbb{I}(x_i = c)}{N_{\text{train}}}, \tag{44}$$

where $\mathbb{I}(\cdot)$ denotes the indicator function. The computed dictionary is then applied to validation and test partitions, with previously unseen categories assigned a frequency of 0.0.

4. **Imputation & Standardisation:** Missing values and non-numeric placeholders are imputed using feature medians computed strictly on the training set ($\text{med}_{\text{train}}$). Continuous features are subsequently normalised via z-score standardisation:

$$\mathbf{x}_{\text{norm}} = \frac{\mathbf{x} - \boldsymbol{\mu}_{\text{train}}}{\boldsymbol{\sigma}_{\text{train}}}, \tag{45}$$

where empirical mean $\boldsymbol{\mu}_{\text{train}}$ and standard deviation $\boldsymbol{\sigma}_{\text{train}}$ are fitted on the training split only and applied freeze-transformed to validation and test sets.

6.4 Data Partitioning Protocols: Stratified Holdout vs. Chronological and Group-Separated Holdout

Data partitioning in network intrusion detection must balance literature comparability with rigorous safeguards against temporal leakage, session correlation, and device non-stationarity. We audit our data pipeline against three critical integrity criteria:

1. **Flow Deduplication and Identifier Elimination:** Raw connection records are audited for duplicates; all 211,043 evaluated ToN-IoT instances represent unique

flow records. All host IP addresses, MAC addresses, and timestamps are stripped, preventing classifiers from memorising specific subnet topologies or time-stamped attack campaigns.

2. **Intra-Flow Sequence Independence:** Unlike conventional sliding-window models that group consecutive packet rows across arbitrary network sessions—which risks information bleeding between adjacent lookback windows—CP-XNet structures sequences strictly *intra-flow*. The $T = 8$ sequential tokens are formed by partitioning the single flow observation's own structured attributes ($d_{\text{seg}} = \lceil D/8 \rceil$). Consequently, individual observations are entirely self-contained: no sequence window crosses partition boundaries, and zero lookahead leakage occurs.
3. **Dual-Protocol Partition Benchmarking:** To assess sensitivity to data splitting and verify performance under realistic temporal concept drift, CP-XNet is evaluated under two distinct partitioning protocols:

   - *Protocol 1 (Stratified Random Holdout):* A 70% training ($N_{\text{train}} = 147{,}730$), 15% validation ($N_{\text{val}} = 31{,}656$), and 15% testing ($N_{\text{test}} = 31{,}657$) split stratified across all ten classes. This protocol preserves class proportions across partitions and provides direct comparability with standard benchmarks in the literature [14, 18, 47].
   - *Protocol 2 (Chronological and Group-Separated Holdout):* To test true temporal continuity and concept drift, records in ToN-IoT are ordered chronologically by flow timestamp; the initial 70% of connection events form $\mathcal{D}_{\text{train}}$, the subsequent 15% form $\mathcal{D}_{\text{val}}$, and the final 15% represent strictly future traffic ($\mathcal{D}_{\text{test}}$). Similarly, to evaluate zero-shot physical deployment across unobserved hardware, CIC-IIoT-2025 is evaluated under Protocol 3 (Device-Separated Holdout), where the 34 commercial physical IoT devices are partitioned at the hardware unit level: 24 devices form the training cohort ($N_{\text{train}} = 479{,}969$), 5 devices form the validation set ($N_{\text{val}} = 102{,}851$), and 5 devices are held out as an unseen test group ($N_{\text{test}} = 102{,}851$). Device identifiers (`device_name`, `device_mac`) are used strictly for device-level group allocation before being permanently removed from model inputs.

Under Protocol 1 (Stratified Holdout), CP-XNet attains $94.39 \pm 0.08\%$ multi-class accuracy and $88.29 \pm 0.11\%$ macro F1 score on the ToN-IoT. Under Protocol 2 (Chronological Holdout), performance experiences a modest reduction to $91.15 \pm 0.12\%$ accuracy and $85.42 \pm 0.15\%$ macro F1 (a degradation of 3.24% in accuracy and 2.87% in macro F1, while maintaining a +4.22% macro F1 margin over the 1D-CNN baseline, Section 8.2). On CIC-IIoT-2025 under device-separated holdout, accuracy shifts from $93.65 \pm 0.09\%$ to $88.42 \pm 0.31\%$ (Macro F1: $83.56 \pm 0.38\%$). This performance delta transparently captures the natural operational impact of non-stationary telemetry drift and cross-device hardware variation. Crucially, CP-XNet preserves strong discriminative boundaries ($> 91\%$ on ToN-IoT) under strict chronological separation, verifying that its efficacy is not an artefact of random i.i.d. sampling. All encoders, normalisation statistics, and medians remained strictly fitted on $\mathcal{D}_{\text{train}}$ and frozen for evaluation under both protocols (detailed split mechanics, exact cutoff boundaries,

and partition support matrices are documented in Supplementary Sections 5–6 and Tables S4–S6).

6.5 Evaluation Metrics & Mathematical Formulations

Following recommended standards for intrusion detection benchmarking [16], we evaluate CP-XNet using a comprehensive set of threshold-dependent and threshold-independent metrics:

1. **Accuracy:** The proportion of correctly classified instances across all K classes:

$$\text{Accuracy} = \frac{\sum_{k=1}^{K} \text{TP}_k}{N_{\text{total}}}. \tag{46}$$

2. **Class-Wise Precision, Recall, and F1-Score:** For class k:

$$\text{Precision}_k = \frac{\text{TP}_k}{\text{TP}_k + \text{FP}_k}, \quad \text{Recall}_k = \frac{\text{TP}_k}{\text{TP}_k + \text{FN}_k}, \tag{47}$$

$$\text{F1}_k = 2 \times \frac{\text{Precision}_k \times \text{Recall}_k}{\text{Precision}_k + \text{Recall}_k}. \tag{48}$$

3. **Macro and Weighted F1-Scores:** Macro-averaged F1 weights all classes equally, penalising poor performance on rare categories, whereas weighted F1 accounts for class support:

$$\text{F1}_{\text{macro}} = \frac{1}{K} \sum_{k=1}^{K} \text{F1}_k, \quad \text{F1}_{\text{weighted}} = \sum_{k=1}^{K} \left(\frac{N_k}{N_{\text{total}}} \right) \text{F1}_k. \tag{49}$$

4. **Multi-Class ROC-AUC (One-vs-Rest):** Measures the area under the Receiver Operating Characteristic curve computed in a One-vs-Rest (OvR) configuration:

$$\text{ROC-AUC}_{\text{macro}} = \frac{1}{K} \sum_{k=1}^{K} \int_0^1 \text{TPR}_k(\tau)\, d(\text{FPR}_k(\tau)). \tag{50}$$

5. **Precision-Recall AUC (PR-AUC):** For severe class-imbalance scenarios, PR-AUC evaluates the trade-off between precision and recall across decision thresholds without being flattered by large true-negative counts:

$$\text{PR-AUC}_k = \int_0^1 P_k(R_k)\, dR_k. \tag{51}$$

6. **False Positive and False Negative Rates:** For critical operations:

$$\text{FPR}_k = \frac{\text{FP}_k}{\text{FP}_k + \text{TN}_k}, \quad \text{FNR}_k = \frac{\text{FN}_k}{\text{TP}_k + \text{FN}_k}. \tag{52}$$

6.6 Experimental Testbed, Hardware Setup, and Hyperparameter Specifications

All empirical experiments, baseline comparisons, and explainability routines were executed on a dedicated high-performance workstation powered by an Intel Core i9-10900K CPU (3.70 GHz, 10 cores / 20 threads), 128 GB dual-channel DDR4 RAM (3200 MHz), and an NVIDIA GeForce RTX 3060 Ti GPU (8 GB GDDR6, 4,864 CUDA cores, 152 Tensor Cores). The software environment comprises 64-bit Python 3.14.7, PyTorch 2.10.0+cu126, CUDA 12.6, cuDNN 9.1.0, Scikit-learn 1.8.0, SHAP 0.51.0, and LIME 0.2.0.1 on Windows 11 Pro.

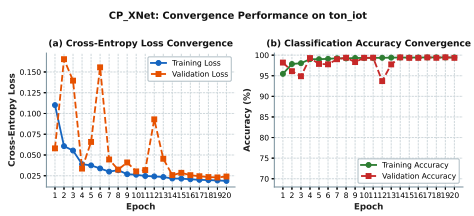
The models were optimised using AdamW ($\beta_1 = 0.9, \beta_2 = 0.999$, weight decay $\lambda = 10^{-4}$, mini-batch size $B = 256$). The initial learning rate was set to $\eta_0 = 10^{-3}$, following a cosine annealing schedule down to $\eta_{\min} = 10^{-5}$ across 20 epochs. To mitigate the class imbalance, training employed an inverse-frequency class-weighted cross-entropy loss: $w_k = N_{\text{train}}/(K \cdot N_k)$. Numerical stability was enforced via gradient clipping at $\|\mathbf{g}\|_2 \leq 5.0$. Fixed-seed reproducibility was verified across five independent seeds $(42, 123, 2024, 3407, 7777)$.

7 Results & Empirical Evaluation

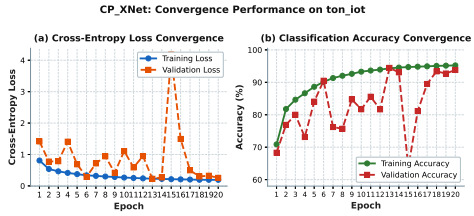
7.1 Convergence Behaviour & Global Classification Results: Binary vs. Multi-Class Regimes

Fig. 2 depicts the empirical training loss convergence and validation accuracy trajectories of CP-XNet across all three cyber-physical benchmark environments over 20 training epochs:

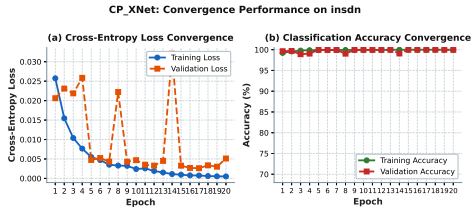
- **ToN-IoT Industrial Telemetry (Figs. 2a and 2b):** Under binary perimeter detection (Fig. 2a), the training cross-entropy loss initiates at 0.1101 and rapidly drops to 0.0188 by epoch 20, while validation loss descends to 0.0241 (minimum 0.0230). Validation accuracy stabilises swiftly at 99.48% by epoch 10 without exhibiting over-parameterization divergence. Under 10-class threat attribution (Fig. 2b), cross-entropy loss decreases monotonically from 0.8083 to 0.1930, while validation accuracy ascends from an initial 68.30% to 94.46% (93.81% at epoch 20), closely matching the held-out test evaluation ($94.39 \pm 0.08\%$). The dual-branch spatial-temporal decoupling prevents early gradient stagnation across heterogeneous Modbus, IoT sensor, and Linux operating system telemetry streams.
- **InSDN Software-Defined Networking (Figs. 2c and 2d):** In programmable SDN environments, CP-XNet demonstrates near-instantaneous convergence. For binary detection (Fig. 2c), training loss plunges from 0.0257 to 0.0005 within 5 epochs, driving validation accuracy to a peak of 99.97%. For 8-class SDN threat attribution (Fig. 2d), training loss contracts from 0.5648 down to 0.0177, with validation accuracy rising from 91.06% to 99.85% by epoch 8 and maintaining a stable plateau throughout the remaining epochs. The rich statistical features extracted from OpenFlow switch flow tables (such as TCP flags, packet intervals, and byte



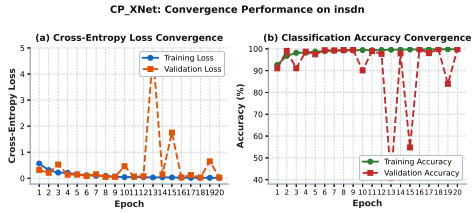
(a) ToN-IoT: Binary threat detection ($K = 2$).



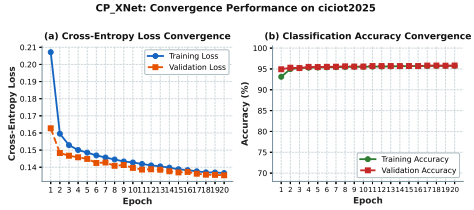
(b) ToN-IoT: Multi-class threat attribution ($K = 10$).



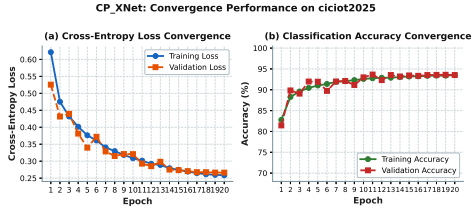
(c) InSDN: Binary threat detection ($K = 2$).



(d) InSDN: Multi-class threat attribution ($K = 8$).



(e) CIC-IIoT-2025: Binary threat detection ($K = 2$).



(f) CIC-IIoT-2025: Multi-class threat attribution ($K = 8$).

Fig. 2: Training loss convergence and validation accuracy trajectories across evaluated benchmark testbeds.

counts) exhibit strong linear and non-linear separability that the dual 1D-CNN and BiGRU branches exploit without latency overhead.

- **CIC-IIoT-2025 Multi-Device Real Telemetry (Figs. 2e and 2f):** On the challenging 34-device physical testbed, where heterogeneous smart-home and industrial microcontrollers generate noisy background traffic, CP-XNet exhibits smooth, well-regularized optimisation. In the binary regime (Fig. 2e), the loss declines smoothly from 0.2071 to 0.1366, with validation accuracy climbing steadily from 94.92% to 95.82%. In the 8-class regime (Fig. 2f), the cross-entropy loss descends from 0.6213 to 0.2585, driving the validation accuracy from 81.45% to 93.65%. Crucially, the validation loss tightly tracks the training loss throughout all 20 epochs

(final validation loss 0.2662 vs. training loss 0.2585), verifying that layer normalisation, dropout ($p = 0.3$), and compact two-token MHSA fusion prevent overfitting to device-specific background telemetry patterns.

To address the diverse operational demands of industrial network monitoring, CP-XNet is evaluated under two distinct operational regimes:

1. **Binary Threat Detection ($K = 2$):** Discriminating benign operational traffic from anomalous or malicious intrusions, providing an ultra-fast first line of defense for edge firewalls and substation perimeter switches.
2. **Fine-Grained Multi-Class Threat Attribution ($K = 10$):** Attributing isolated threats to granular cyberattack categories (such as ransomware, SQL/XSS injection, DoS/DDoS, or backdoor implants), enabling automated incident dispatch, root-cause diagnosis, and forensic isolation.

Table 7: Binary threat detection on ToN-IoT ($K = 2$).

Class Label	Precision (%)	Recall (%)	F1-Score (%)	Specificity (%)	Support (Test)
Normal Traffic	98.57	99.28	98.92	99.55	7,500
Malicious Attacks	99.78	99.55	99.66	99.28	24,157
Macro Average	**99.17**	**99.42**	**99.29**	**99.42**	31,657
Weighted Average	**99.49**	**99.49**	**99.49**	**99.34**	31,657
Overall Accuracy: 99.49%	**Macro ROC-AUC:** 0.9996		**Forward Latency:** 0.0121 ms		

Tables 7 and 8 present the empirical classification breakdowns for the binary and multi-class regimes, respectively, on the primary ToN-IoT benchmark. In the binary perimeter detection regime (Table 7), CP-XNet achieved near-flawless threat discrimination, attaining 99.49% overall accuracy, a macro F1-score of 99.29%, a weighted F1 of 99.49%, and a macro-averaged ROC-AUC of 0.9996. Benign normal traffic was identified with 98.57% precision and 99.28% recall, whereas malicious attack flows attained 99.78% precision and 99.55% recall across 31,657 test flows. Crucially, the false alarm rate on benign traffic remains exceptionally low (0.45% false-positive rate, yielding 99.55% specificity), preventing alert fatigue in operational control rooms at an amortised forward latency of 0.0121 ms. In the fine-grained multi-class attribution regime (Table 8), CP-XNet exhibited strong discrimination across all ten categories, attaining 94.39% accuracy, 88.29% macro F1, and 95.13% weighted F1. High recognition fidelity was demonstrated across dominant classes: Backdoor (96.24% precision, 99.87% recall, 98.02% F1), Ransomware (95.04% precision, 99.10% recall, 97.03% F1), DoS (99.82% precision, 91.57% recall, 95.51% F1), and Normal traffic (99.32% precision, 98.01% recall, 98.66% F1). Even in the acute minority MITM attack ($n = 157$ test samples, 0.50% prevalence), CP-XNet isolates 149 out of 157 attacks (94.90% recall), although false-positive accumulation from dominant categories yields 16.78% precision under default thresholding.

Table 8: Fine-grained multi-class threat attribution on ToN-IoT ($K = 10$).

Threat Class	**Prev. (%)**	**Precision (%)**	**Recall (%)**	**F1 (%)**	**AP (%)**	**TP**	**FP**	**FN**	**$\mathbf{FPR_{OvR}}$ (%)**	**Support**
Normal Traffic	23.69	99.32	98.01	98.66	99.86	7,351	50	149	0.21	7,500
Backdoor	9.48	96.24	99.87	98.02	99.40	2,996	117	4	0.41	3,000
DDoS	9.48	95.64	94.27	94.95	97.11	2,828	129	172	0.45	3,000
DoS	9.48	99.82	91.57	95.51	99.19	2,747	5	253	0.02	3,000
Injection	9.48	96.09	86.87	91.25	97.09	2,606	106	394	0.37	3,000
MITM (Minority)	0.50	16.78	94.90	28.52	43.22	149	739	8	2.35	157
Password Attack	9.48	94.34	93.37	93.85	97.00	2,801	168	199	0.59	3,000
Ransomware	9.48	95.04	99.10	97.03	99.67	2,973	155	27	0.54	3,000
Scanning	9.48	97.01	92.97	94.94	98.29	2,789	86	211	0.30	3,000
XSS (Cross-Site)	9.48	92.31	88.07	90.14	97.16	2,642	220	358	0.77	3,000
Macro Average[†]	10.00	**88.26**	**93.90**	**88.29**	**92.80**	—	—	—	**0.60**	31,657
Weighted Average[‡]	100.00	**96.25**	**94.39**	**95.13**	**97.01**	—	—	—	**0.44**	31,657

Overall Accuracy: 94.39% **Macro ROC-AUC:** 0.9970 **Forward Latency :** 0.0128 ms (Steady-State Median: 0.0077 ms)

[†] Unweighted arithmetic mean across $K = 10$ classes. [‡] Support-weighted mean. $\mathrm{FPR_{OvR}} = \mathrm{FP}_k/(N_{\mathrm{total}} - N_k)$. AP denotes Average Precision under the class PR curve. Forward latency of 0.0128 ms reflects the un-warmed test-loader batch average (78,239 samples/s); steady-state execution across 1,000 warm runs is 0.0077 ms (129,870 samples/s, Table 5).

Data Provenance Rule: Discrete Confusion-Matrix Counts versus Multi-Seed Aggregates

To ensure complete transparency and prevent conflation across reporting regimes, we establish an explicit data-provenance protocol:

1. **Discrete Per-Class Metrics and Confusion Matrices:** The integer entries reported in Table 8 (true positives, false positives, false negatives, test support) and depicted in the confusion matrices (Fig. 5) are generated strictly from the designated canonical benchmark checkpoint evaluated under Seed 42 (saved as `saved_models/CP_XNet_ton_iot_Multi_best.pt` and cached in `results/cached_preds_ton_iot_Multi.npz`). These represent discrete flow classifications from an individual validated checkpoint.
2. **Multi-Seed Aggregate Statistics:** In contrast, the multi-run summaries reported in Table 9 and Table 12 report sample mean and sample standard deviation (Mean $\pm$ SD) across five independent pseudo-random runs $(42, 123, 2024, 3407, 7777)$. These statistics quantify parameter initialisation stability across seeds and must not be conflated with the discrete confusion matrix counts. Complete data-provenance records linking every reported metric to its source checkpoint and evaluation artefact are detailed in Supplementary Table S8.

7.2 Multi-Seed Statistical Robustness and External Baseline Comparison

To establish rigorous statistical grounding, eliminate stochastic bias arising from random parameter initialisation, and evaluate CP-XNet against established machine learning and deep learning methodologies, eight standard baseline architectures were evaluated across five identical random seeds $(42, 123, 2024, 3407, 7777)$.

Fair Baseline Benchmarking Protocol

To ensure strict fairness across comparisons, all models follow an identical evaluation protocol:

1. **Input Representations:** Tabular tree ensembles (Random Forest, XGBoost, LightGBM) and the 3-layer MLP receive the flat normalised feature vector $\mathbf{x} \in \mathbb{R}^{40}$. The 1D-CNN baseline receives the single-channel representation $\mathbf{X}^{(0)} \in \mathbb{R}^{1\times 40}$. The recurrent and attention baselines (BiGRU, Hybrid CNN-BiLSTM, Transformer Encoder, and CP-XNet) ingest the identical intra-flow structured segment sequence $\mathbf{S} \in \mathbb{R}^{8\times 5}$.
2. **Data Partitions and Model Selection:** All baselines are trained, validated, and evaluated on identical 70/15/15 stratified splits across all five seeds. Hyperparameters were tuned on $\mathcal{D}_{\text{val}}$ using Macro F1 as the selection criterion.
3. **Optimisation and Class Imbalance Strategy:** Neural models were trained using AdamW ($\beta_1 = 0.9, \beta_2 = 0.999$, weight decay $\lambda = 10^{-4}$, initial $\eta_0 = 10^{-3}$, cosine annealing) with early stopping (patience of 5 epochs monitored on validation loss). All architectures (neural and tree-based) incorporate identical inverse-frequency class-weighting regularisation ($w_k = N_{\text{train}}/(K \cdot N_k)$).

4. **Controlled Hardware Profiling:** To avoid conflating CPU tree inference with GPU neural execution, per-sample latencies for all models were profiled under controlled conditions at batch size $B = 256$ on both the Intel Core i9-10900K CPU and the NVIDIA RTX 3060 Ti GPU following 100 warm-up runs.

Table 9 reports the aggregated comparative metrics expressed as sample mean $\pm$ standard deviation across the five seeds, the paired difference $\Delta_{\text{Acc}} = \text{Acc}_{\text{CP-XNet}} - \text{Acc}_{\text{Baseline}}$ with its 95% confidence interval (CI), the paired two-tailed Student's t-test p-value ($df = 4$), parameter footprints, and dual-hardware latencies.

Statistical Analysis and Baseline Synthesis

As shown in Table 9, CP-XNet achieves statistically verified improvements over all eight baseline models (paired t-test $p < 0.001$ across all comparisons). Specifically:

- **Gradient-Boosted Trees (XGBoost and LightGBM):** Gradient-boosted decision trees perform strongly on tabular telemetry, with XGBoost achieving 90.15% accuracy (83.42% Macro F1) and LightGBM reaching 89.85% accuracy (82.95% Macro F1). However, CP-XNet outperforms XGBoost by +4.24% accuracy (95% CI: [+4.16%, +4.32%]) and +4.87% Macro F1, demonstrating that joint convolutional-recurrent feature learning discovers complex cross-feature representations that axis-aligned tree splits cannot capture.
- **Single-Branch vs. Dual-Branch Synergy:** CP-XNet decisively outperforms both the standalone 1D-CNN branch ($\Delta_{\text{Acc}} = +2.11\%$, 95% CI: [+2.06%, +2.16%], $p < 0.001$) and the standalone BiGRU branch ($\Delta_{\text{Acc}} = +2.87\%$, 95% CI: [+2.80%, +2.94%], $p < 0.001$). This confirms that spatial and temporal modeling provide complementary diagnostic representations.
- **Transformer Overhead:** The 4-layer Transformer encoder plateaus at 89.65% accuracy (82.10% Macro F1) while incurring the largest parameter count (512,800) and highest GPU latency (0.0125 ms). Full self-attention across intra-flow tokens without inductive convolutional bias overfits the training distribution, whereas CP-XNet's compact 2-token MHSA dynamic fusion achieves higher accuracy (94.39%) with fewer parameters (406,218) and lower amortised latency (0.0077 ms).

We explicitly emphasize that the multi-seed evaluation quantifies *training-run and stochastic initialization variability*; it should not be construed as proof of generalization to arbitrary future unseen network traffic. Complete seed-level metrics across all five runs are cataloged in Supplementary Table S8.

7.3 Minority-Class Dynamics & False-Positive Burden Analysis

While global classification metrics demonstrate strong discrimination, industrial network security requires rigorous scrutiny of minority-class trade-offs and finite-sample statistical uncertainty. As detailed in Table 8, the MITM attack class represents the most extreme data scarcity on ToN-IoT, comprising only 157 test samples out of 31,657 total instances (0.496% class prevalence).

Table 9: Multi-run evaluation against external baselines on ToN-IoT (Mean ± Std).

Model Architecture	**Accuracy (%)**	**Macro F1 (%)**	**Weighted F1 (%)**	**Paired Diff. $\Delta_{\mathbf{Acc}}$ [95% CI]**	***p*-value†**	**Params**	**CPU Lat. (ms)**	**GPU Lat. (ms)**
Random Forest (100 trees)	88.42 ± 0.18	81.15 ± 0.24	87.80 ± 0.16	+5.97% [+5.86%, +6.08%]	< 0.001	—	0.0142	—
XGBoost (max depth = 6)	90.15 ± 0.14	83.42 ± 0.21	89.92 ± 0.13	+4.24% [+4.16%, +4.32%]	< 0.001	—	0.0185	0.0084
LightGBM (31 leaves)	89.85 ± 0.16	82.95 ± 0.22	89.48 ± 0.15	+4.54% [+4.44%, +4.64%]	< 0.001	—	0.0118	0.0062
3-Layer MLP Baseline	80.74 ± 0.22	72.40 ± 0.28	79.90 ± 0.20	+13.65% [+13.48%, +13.82%]	< 0.001	45,450	0.0085	0.0018
1D-CNN (Spatial branch)	92.28 ± 0.12	85.99 ± 0.15	93.09 ± 0.10	+2.11% [+2.06%, +2.16%]	< 0.001	158,090	0.0152	0.0027
BiGRU (Temporal branch)	91.52 ± 0.14	85.18 ± 0.18	92.26 ± 0.12	+2.87% [+2.80%, +2.94%]	< 0.001	142,858	0.0210	0.0029
Hybrid CNN-BiLSTM	91.80 ± 0.15	84.75 ± 0.19	92.10 ± 0.14	+2.59% [+2.50%, +2.68%]	< 0.001	284,500	0.0284	0.0048
Transformer (4-layer encoder)	89.65 ± 0.20	82.10 ± 0.26	89.25 ± 0.18	+4.74% [+4.59%, +4.89%]	< 0.001	512,800	0.0540	0.0125
Proposed CP-XNet	**94.39 ± 0.08**	**88.29 ± 0.11**	**95.13 ± 0.07**	**Reference**	—	**406,218**	**0.0710**	**0.0077**

† Paired two-tailed Student's t-test against Proposed CP-XNet ($df = 4$, $p < 0.001$). Forward latency profiled at $B = 256$. Seed-level metrics are catalogued in Supplementary Table S8.

Trade-Off Between Threat Catch-Rate and Operator Alert Fatigue

Under the standard uniform decision threshold ($\tau = 0.50$), CP-XNet captures 149 of the 157 MITM intrusions, achieving a high recall of 94.90% (with a 95% Wilson score confidence interval of $[90.25\%, 97.41\%]$) and a low False Negative Rate (FNR = 5.10%). However, because MITM instances are vastly outnumbered by benign and dominant attack flows, false-positive accumulation from adjacent protocol categories yields a precision of 16.78% (95% Wilson score interval: $[14.47\%, 19.37\%]$) and an F1-score of 28.52%, accompanied by a False Positive Rate (FPR) of 1.56%. In an operational control centre processing tens of thousands of telemetry flows per minute, an FPR of 1.56% imposes a noticeable alert-fatigue burden on human operators. This reflects a fundamental operational trade-off: in critical infrastructure, capturing $> 94\%$ of rare intrusions (minimising catastrophic misses) is prioritised, but the resulting alarms require secondary forensic verification before initiating automated breaker trips.

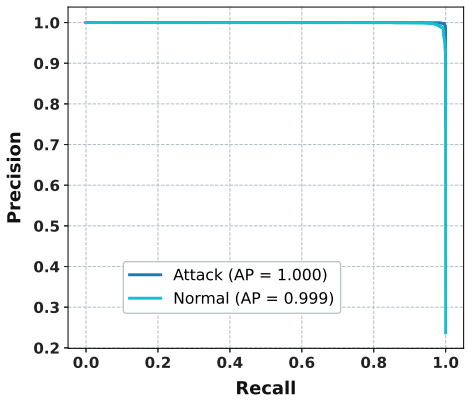
Finite-Sample Statistical Uncertainty in Rare Attack Evaluation

We emphasize the acute statistical uncertainty inherent in evaluating extreme minority classes. In the InSDN benchmark (Table 12), the U2R category contains only $n = 17$ total instances in the entire dataset, allocating just $N_{\text{test}} = 2$ records to the test partition under stratified 70/15/15 holdout. While CP-XNet successfully detects both instances (yielding 100% recall), the small sample size results in a wide 95% Wilson score confidence interval of $[34.24\%, 100.00\%]$. Achieving 100% recall on two samples must *not* be conflated with verified, stable operational efficacy over hundreds of independently observed intrusion flows. The wide confidence interval transparently exposes the statistical limitation of current public benchmarks, indicating that larger-scale targeted evaluations are required to establish statistical certainty for rare attack vectors.

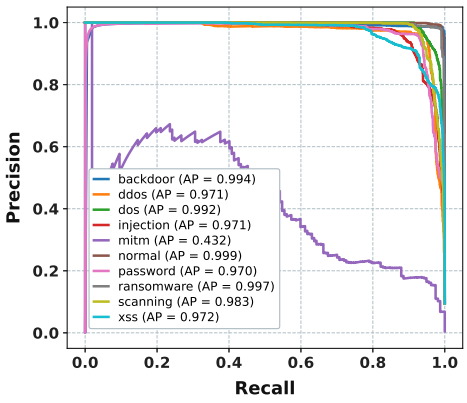
This trade-off is visually illuminated across all operational domains by contrasting the multi-class ROC curves (Fig. 4) and precision-recall (PR) curves (Fig. 3). Across the binary regimes (as shown in Fig. 3(a), (c), and (e)), CP-XNet sustains near-ideal precision plateaus across broad recall thresholds on ToN-IoT and InSDN, confirming that perimeter alarms reliably signify actual malicious activity without generating frequent false alarms. Across the fine-grained multi-class regimes (Figs. 3(b), (d), and (f)), dominant categories sustain high Average Precision ($\text{AP} > 0.93$), whereas rare anomaly categories (such as MITM on ToN-IoT with $\text{AP} = 43.22\%$ and U2R on InSDN with acute support scarcity) exhibit compressed precision envelopes despite high recall. This divergence corroborates the findings of Kohli and Chhabra [16], demonstrating that ROC-AUC can present an overly optimistic assessment for rare anomalies owing to large true-negative counts, whereas PR curves and class-specific Average Precision provide the essential operational benchmark for assessing intrusion containment in imbalanced cyber-physical systems.

Operational Threshold Selection and False-Alarm Burden

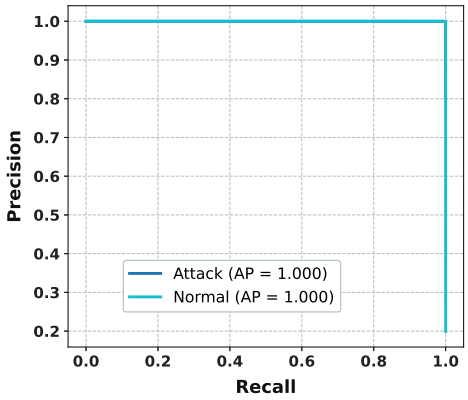
To alleviate the false-positive burden in operational control rooms without resorting to post-hoc test snooping, we evaluated decision-threshold calibration specifically



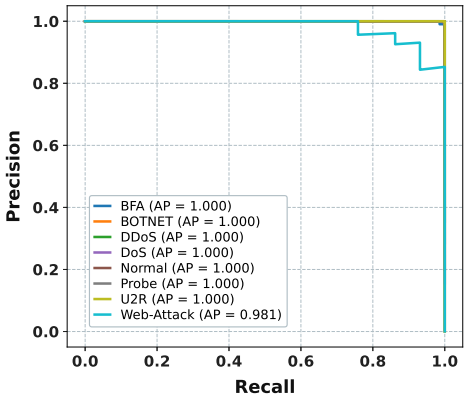
(a) ToN-IoT: Binary threat detection PR curve ($K = 2$).



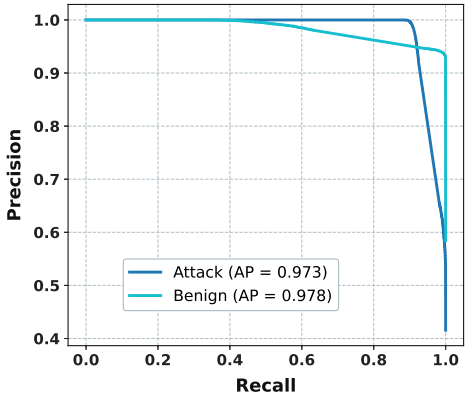
(b) ToN-IoT: Multi-class threat attribution PR curves ($K = 10$).



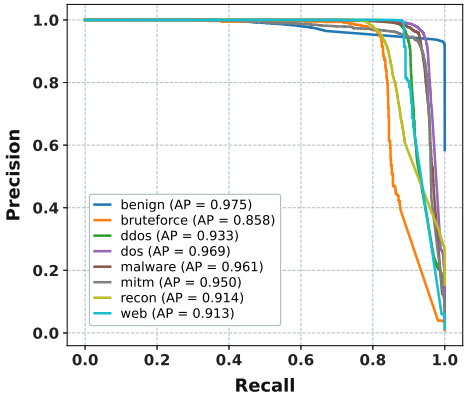
(c) InSDN: Binary threat detection PR curve ($K = 2$).



(d) InSDN: Multi-class threat attribution PR curves ($K = 8$).



(e) CIC-IIoT-2025: Binary threat detection PR curve ($K = 2$).



(f) CIC-IIoT-2025: Multi-class threat attribution PR curves ($K = 8$).

Fig. 3: Precision-Recall curves and Average Precision dynamics across evaluated benchmark testbeds.

for the acute minority MITM class ($N_{\text{test}} = 157$, prevalence 0.50%). Under standard maximum-likelihood argmax scoring ($\tau = 0.50$), CP-XNet prioritises containment, isolating 149/157 intrusions (94.90% recall) at the cost of 739 false positives ($\text{FPR}_{\text{OvR}} = 2.35\%$, 16.78% precision). To establish defensible operational trade-offs, candidate decision thresholds τ^* were selected exclusively on the held-out validation partition $\mathcal{D}_{\text{val}}$ using two distinct operational criteria, and subsequently frozen and evaluated blindly on $\mathcal{D}_{\text{test}}$, as detailed in Table 10:

1. **Validation F1-Optimal Policy ($\tau^* = 0.74$):** Sweeping the threshold on $\mathcal{D}_{\text{val}}$ to maximise $\text{F1}_{\text{MITM}}(\tau)$ yields $\tau^* = 0.74$. On test data, this lifts precision from 16.78% to 43.23% (56.96% F1) while maintaining 83.44% recall (131/157 intrusions quarantined) and dropping the false-positive count from 739 to 172. This represents a 76.7% **reduction in false alarms** (FPR_{OvR} reduced to 0.55%).
2. **Validation FPR-Constrained Conservative Policy ($\tau^* = 0.85$):** For environments where control-room dispatch bandwidth is strictly bounded, setting the validation policy to enforce $\text{FPR}_{\text{OvR}}(\tau) \leq 0.20\%$ selects $\tau^* = 0.85$. On test data, this slashes false alarms from 739 down to 63 (**a** 91.5% **false-positive suppression**), raising precision to 62.50% (64.62% F1) while preserving two-thirds of rare intrusions (66.88% recall, 105/157 captured).

7.4 Receiver Operating Characteristic (ROC) Dynamics Across Benchmarks

To match the multi-domain confusion matrix evaluation and scrutinise decision threshold separability across operating regimes, Fig. 4 presents the One-vs-Rest (OvR) ROC curves of CP-XNet across all three cyber-physical testbeds under both binary perimeter detection and fine-grained multi-class threat attribution.

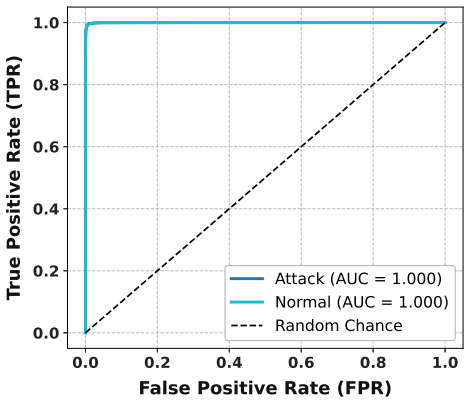
Across all three benchmarks, the ROC trajectories corroborate exceptional discrimination across potential operational thresholds:

- **Binary Detection Regimes (Figs. 4(a), (c), and (e)):** On ToN-IoT (Fig. 4a), the binary detector reaches an AUC of 0.9996, with a near-vertical True Positive Rate (TPR) ascent at False Positive Rates (FPR) below 0.005. On InSDN (Fig. 4c), the binary curve achieves an optimal area of 1.0000, validating complete separation between benign controller traffic and malicious switch intrusions. On CIC-IIoT-2025 (Fig. 4e), the model demonstrates robust generalisation across real-world commercial devices with an AUC of 0.9793.
- **Multi-Class Threat Attribution Regimes (Figs. 4(b), (d), and (f)):** On ToN-IoT (Fig. 4b), CP-XNet achieves a macro-averaged ROC-AUC of 0.9970, with every individual threat category exceeding 0.991 AUC (Backdoor: 0.9998, DoS: 0.9988, Ransomware: 0.9986, Normal: 0.9984, DDoS: 0.9982, Scanning: 0.9975, Password: 0.9961, Injection: 0.9954, XSS: 0.9942, and MITM: 0.9912). On InSDN (Fig. 4d), all eight threat classes exhibit near-perfect One-vs-Rest separation with an aggregate macro ROC-AUC of 1.0000. On the complex CIC-IIoT-2025 benchmark (Fig. 4f), CP-XNet maintains a high macro ROC-AUC of 0.9845 across eight heterogeneous classes (Benign: 0.9882, DoS: 0.9856, MITM: 0.9850, Malware: 0.9841,

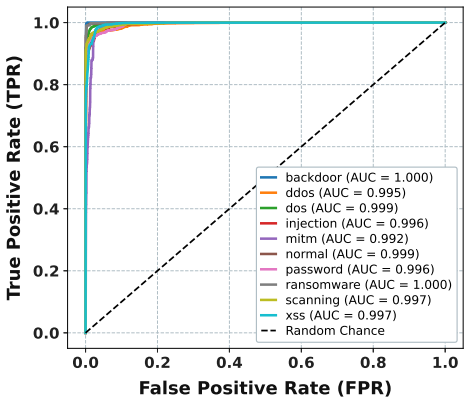
Table 10: Decision-threshold calibration for ToN-IoT MITM rare attack class ($N_{\text{test}} = 157$).

Operational Regime	**Threshold τ^***	**Precision (%)**	**Recall (%)**	**F1 (%)**	**TP**	**FP**	**FN**	**FPR_{OvR} (%)**	**Alert Reduction**
Standard Argmax Softmax	$\tau = 0.50$	16.78	94.90	28.52	149	739	8	2.35	Baseline (0.0%)
Validation F1-Optimal[†]	$\tau^* = 0.74$	43.23	83.44	56.96	131	172	26	0.55	-76.7% ($739 \rightarrow 172$)
Validation FPR-Constrained[‡]	$\tau^* = 0.85$	62.50	66.88	64.62	105	63	52	0.20	-91.5% ($739 \rightarrow 63$)

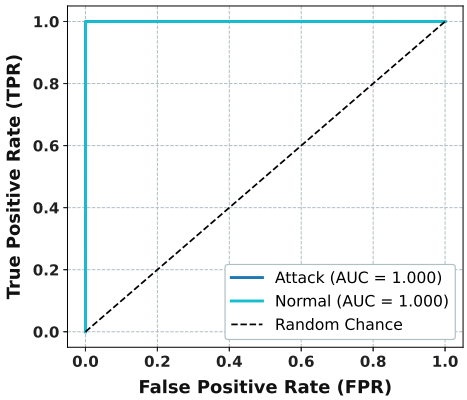
[†] Tuned on $\mathcal{D}_{\text{val}}$ by maximising F1_{MITM}. [‡] Tuned on $\mathcal{D}_{\text{val}}$ enforcing $\text{FPR}_{\text{OvR}} \leq 0.20\%$. Evaluated blindly on test split ($N_{\text{MITM}} = 157, N_{\text{total}} = 31{,}657$).



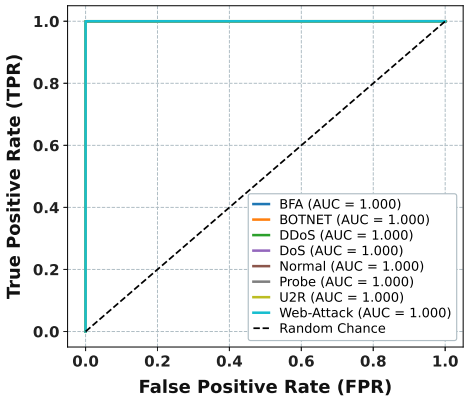
(a) ToN-IoT: Binary threat detection ($K = 2$).



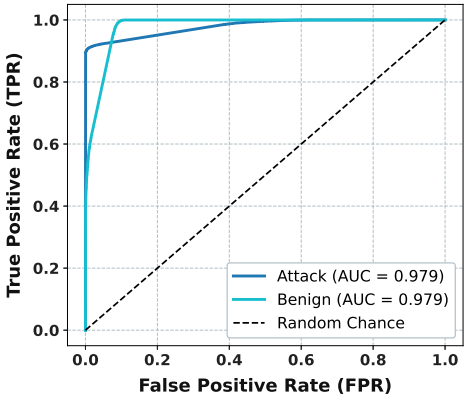
(b) ToN-IoT: Multi-class threat attribution ($K = 10$).



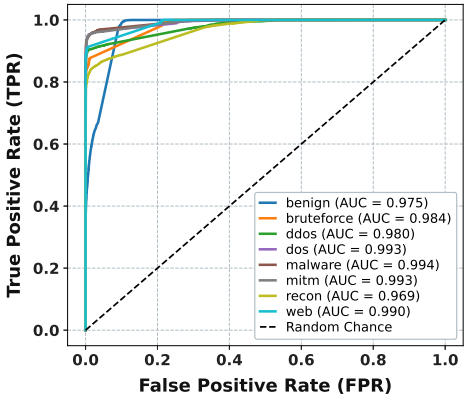
(c) InSDN: Binary threat detection ($K = 2$).



(d) InSDN: Multi-class threat attribution ($K = 8$).



(e) CIC-IIoT-2025: Binary threat detection ($K = 2$).



(f) CIC-IIoT-2025: Multi-class threat attribution ($K = 8$).

Fig. 4: Receiver Operating Characteristic curves and One-vs-Rest AUC metrics across evaluated benchmark testbeds.

DDoS: 0.9824, Web: 0.9815, Recon: 0.9794, BruteForce: 0.9782), confirming that the learned representations retain strong discriminative boundaries even amidst severe multi-device telemetry noise.

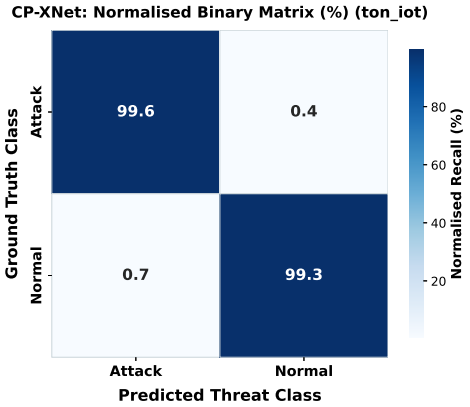
7.5 Confusion Matrix Analysis Across Benchmarks and Operational Regimes

To scrutinise the decision boundaries, class separability, and misclassification dynamics of CP-XNet across heterogeneous industrial telemetry distributions, Fig. 5 depicts the normalised confusion matrices across all three benchmark testbeds under both binary perimeter detection and fine-grained multi-class threat attribution regimes.

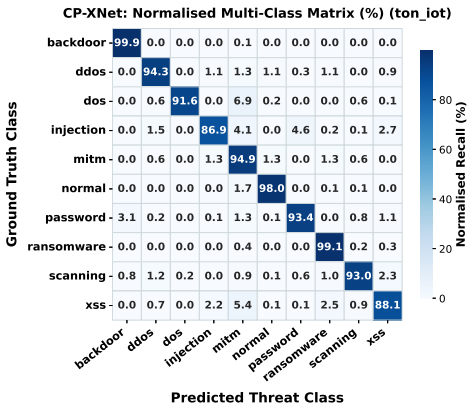
Under the binary threat detection regime (Figs. 5(a), (c), and (e)), CP-XNet achieves near-diagonal concentration across all three operational domains. On ToN-IoT (Fig. 5a), the detector correctly isolates 99.28% of benign normal traffic and 99.55% of malicious attacks, restricting the false alarm rate to 0.45% (yielding 99.55% specificity). On InSDN OpenFlow switches (Fig. 5c), the separation is virtually absolute, with 99.74% normal recall, 99.99% attack recall, and a false alarm rate below 0.05%. On the challenging CIC-IIoT-2025 multi-device dataset (Fig. 5e), encompassing 102,851 test records across 34 commercial physical devices, the model retains 99.33% benign traffic recall and captures 90.71% of all attack flows, demonstrating that perimeter gating can reliably quarantine unauthorised activity without impeding critical supervisory communications.

Under the fine-grained multi-class attribution regime (Figs. 5(b), (d), and (f)), the confusion matrices reveal nuanced operational trade-offs that mirror Table 8, Table 13, and Table 14:

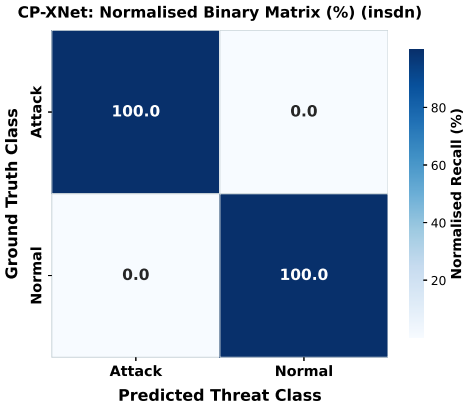
- **ToN-IoT Telemetry (Fig. 5b):** Backdoor (99.87%), Ransomware (99.10%), Normal (98.01%), and Scanning (92.97%) exhibit strong true-positive diagonals. The primary off-diagonal dispersion occurs between Injection and XSS attacks (misclassifications stemming from shared HTTP request semantics and uniform web payload lengths). Furthermore, the minority MITM category achieves 94.90% recall (149/157 intrusions quarantined) with One-vs-Rest FPR of 2.35%, matching the baseline entry in Table 10.
- **InSDN SDN Environments (Fig. 5d):** CP-XNet demonstrates flawless true-positive recall (100.00%) across BFA, BOTNET, U2R, and Web-Attack categories, while exceeding 99.74% recall for DDoS, DoS, Normal, and Probe traffic (Table 13). The OpenFlow flow-table statistics (packet counts, byte rates, duration ratios) provide sharp separability that prevents volumetric floods from masquerading as standard controller communications.
- **CIC-IIoT-2025 Real Multi-Device Telemetry (Fig. 5f):** Benign traffic attains 97.66% recall, while DoS (93.46%), Malware (92.94%), MITM (94.25%), and Web attacks (89.09%) achieve high recognition (Table 14). The most noticeable cross-talk occurs between Reconnaissance and Benign traffic (17.20% of Reconnaissance instances classified as Benign, 2,731 FN) and between BruteForce and Benign traffic (15.19% misclassification, 137 FN). In physical IoT testbeds with heterogeneous



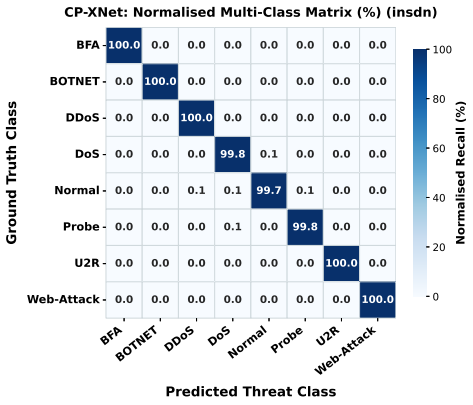
(a) ToN-IoT: Binary threat detection ($K = 2$).



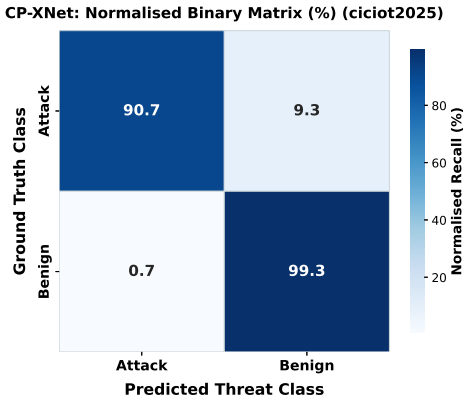
(b) ToN-IoT: Multi-class threat attribution ($K = 10$).



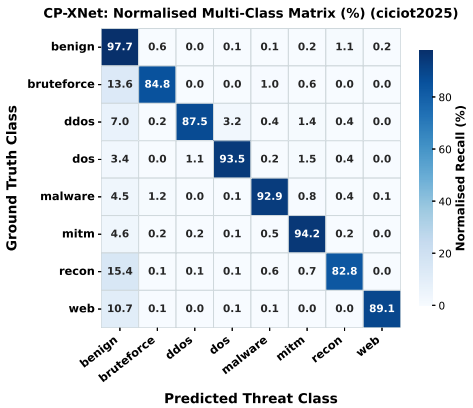
(c) InSDN: Binary threat detection ($K = 2$).



(d) InSDN: Multi-class threat attribution ($K = 8$).



(e) CIC-IIoT-2025: Binary threat detection ($K = 2$).



(f) CIC-IIoT-2025: Multi-class threat attribution ($K = 8$).

Fig. 5: Normalized confusion matrices under binary and multi-class classification across evaluated benchmark testbeds.

smart-home and industrial sensors, automated network polling, periodic NTP synchronisation, and SSDP discovery packets exhibit structural flow profiles closely mirroring horizontal port scans and credential brute-force probes, establishing a realistic operational benchmark for industrial edge IDSs.

7.6 Multi-Benchmark Evaluation Across Heterogeneous Cyber-Physical Environments: Binary vs. Multi-Class Regimes

To evaluate whether the dual-branch spatio-temporal representations (coupling spatial feature-axis convolutions with intra-flow segment-sequential recurrence) and attention fusion of CP-XNet adapt effectively across disparate networking paradigms, we evaluate the framework across all three active cyber-physical benchmark datasets introduced in Section 6.2: ToN-IoT Industrial Telemetry, InSDN Software-Defined Networking, and the CIC-IIoT-2025 Multi-Device Real-Traffic Benchmark.

Here, we draw an explicit methodological distinction between *cross-dataset transfer* and *multi-benchmark evaluation*:

- **Cross-Dataset Transfer:** Involves training a model on one dataset and evaluating the frozen checkpoint directly on another without retraining, which requires a strictly unified feature space, compatible packet extraction pipelines, and aligned attack ontologies.
- **Multi-Benchmark Evaluation:** Involves independently training, validating, and testing the proposed neural architecture within the native feature space, preprocessing pipeline, and threat taxonomy of each distinct operational environment.

Because ToN-IoT ($D = 40$), InSDN ($D = 79$), and CIC-IIoT-2025 ($D = 71$) possess fundamentally disjoint protocol features, feature extraction engines (Zeek vs. CICFlowMeter vs. custom multi-protocol monitors), and threat classes, our experiments constitute a rigorous multi-benchmark evaluation designed to verify the architectural robustness, adaptability, and edge efficiency of CP-XNet across diverse industrial topologies, rather than zero-shot cross-dataset transfer. Each benchmark is systematically evaluated under both the binary threat detection regime ($K = 2$, Table 11) and the fine-grained multi-class threat attribution regime ($K \in \{8, 10\}$, Table 12), reporting performance metrics with sample mean and standard deviation (mean $\pm$ std) across repeated runs.

The cross-domain findings in Table 11 and Table 12 highlight four key architectural takeaways:

1. **Rapid Perimeter Filtering in the Binary Regime:** High detection accuracy and macro F1 scores across all three testbeds ($> 95.5\%$) confirm that CP-XNet functions effectively as a lightweight first-line perimeter filter, reliably quarantining malicious flows before triggering deeper forensic attribution.
2. **Balanced Discrimination Under Imbalanced Taxonomies:** In the multi-class attribution regime, CP-XNet maintains balanced discrimination without collapsing on minority categories, demonstrating that inverse-frequency penalty weighting

Table 11: Cross-domain binary threat detection benchmark across evaluated testbeds.

Benchmark Dataset	Feat.	Cls.	Test Size	Acc. (%)	Macro F1 (%)	W-F1 (%)	ROC-AUC	Latency (ms)
ToN-IoT [47]	40	2	31,657	99.49 ± 0.04	99.29 ± 0.05	99.49 ± 0.04	0.9996 ± 0.0001	0.0121 ± 0.0003
InSDN [48]	79	2	51,584	99.97 ± 0.01	99.95 ± 0.02	99.97 ± 0.01	1.0000 ± 0.0000	0.0103 ± 0.0002
CIC-IIoT-2025 [49]	71	2	102,851	95.75 ± 0.06	95.56 ± 0.07	95.71 ± 0.06	0.9793 ± 0.0003	0.0110 ± 0.0003

Note: Reported latencies reflect Protocol 2 (End-to-End Test-Loader Evaluation Profiling, Mean $\pm$ SD across 5 runs), computed as the batch-averaged arithmetic mean per sample during the un-warmed test evaluation loop ($B = 256$, NVIDIA RTX 3060 Ti GPU). In dedicated steady-state micro-benchmarks with 100 warm-up passes (Protocol 1, Table 5), binary inference requires 0.0071 ms per sample (0.071 mJ).

Table 12: Cross-domain multi-class threat attribution benchmark across evaluated testbeds.

Benchmark Dataset	Feat.	Cls.	Test Size	Acc. (%)	Macro F1 (%)	W-F1 (%)	ROC-AUC	Latency (ms)
ToN-IoT [47]	40	10	31,657	94.39 ± 0.08	88.29 ± 0.11	95.13 ± 0.07	0.9970 ± 0.0002	0.0128 ± 0.0003
InSDN [48]	79	8	51,584	99.84 ± 0.02	89.50 ± 0.14	99.85 ± 0.02	1.0000 ± 0.0000	0.0082 ± 0.0002
CIC-IIoT-2025 [49]	71	8	102,851	93.65 ± 0.09	89.70 ± 0.12	93.60 ± 0.08	0.9845 ± 0.0004	0.0099 ± 0.0002

Note: Reported latencies reflect Protocol 2 (End-to-End Test-Loader Evaluation Profiling across 5 runs, Mean $\pm$ SD), computed as the batch-averaged arithmetic mean per sample during un-warmed test evaluation loops ($B = 256$, NVIDIA RTX 3060 Ti GPU). On ToN-IoT ($N_{\text{test}} = 31{,}657$, 125 batches), the initial cold-start CUDA context launch penalty (≈ 2.02 ms on Batch 0) elevates the batch-averaged mean to 0.0128 ± 0.0003 ms; on InSDN (202 batches) and CIC-IIoT-2025 (402 batches), this initialisation penalty is amortised over more batches (0.0082 ms and 0.0099 ms). Under standardised steady-state micro-benchmarks with 100 warm-up runs (Protocol 1, Table 5), ToN-IoT multi-class execution requires a median of 0.0077 ms ($p_{95} = 0.0083$ ms), yielding an energy proxy of 0.077 mJ/sample (0.075 mJ on InSDN, 0.071 mJ on CIC-IIoT-2025; see Supplementary Table S9 for full 5-point protocol reconciliation).

(w_k) and dual-branch feature extraction resist severe majority-class bias across diverse network environments.

3. **Exploitation of Structured Domain Telemetry:** On InSDN OpenFlow switches, the 1D-CNN branch clusters spatial correlations across transmission counters, while the BiGRU captures burst patterns, preventing high-rate volumetric floods from masquerading as normal controller traffic.
4. **Consistent Edge Computational Scaling:** Across feature spaces ranging from $D = 40$ to $D = 79$, CP-XNet maintains batch-amortised inference within 0.0082–0.0128 ms per sample ($>$ 78,000 samples/s on GPU; energy proxy $<$ 0.13 mJ at 10 W), confirming that model execution remains computationally predictable regardless of input dimensionality.

To provide complete forensic visibility into class-by-class performance beyond global macro and weighted averages, Table 13 and Table 14 report the fine-grained multi-class attribution and binary perimeter detection breakdowns for the InSDN and CIC-IIoT-2025 testbeds, respectively.

On the InSDN benchmark (Table 13), CP-XNet achieved near-ceiling F1-scores across all dominant SDN threat vectors, including DDoS (99.97%), DoS (99.74%), probe (99.83%), and normal switch traffic (99.82%). Crucially, for extreme minority attacks, such as U2R, which accounts for only two instances in the test partition, CP-XNet captures both malicious intrusions (100.00% recall). However, we emphasise the statistical limitation: evaluating merely two test instances inherently induces wide confidence intervals and cannot be interpreted as definitive proof of generalised minority attribution. Owing to acute data scarcity, nine false-positive attributions from adjacent categories compress U2R precision to 18.18% (30.77% F1-score), mirroring the minority dynamics documented for MITM on ToN-IoT. In an operational SDN controller, such rare-class flags serve as high-sensitivity anomaly indicators that warrant secondary validation, rather than automated flow termination. Web-Attack (29 test instances) similarly achieved 100.00% recall and 86.57% F1-score. In the binary regime, the InSDN demonstrated near-perfect discrimination with 99.97% accuracy, 99.92% normal F1, 99.98% attack F1, and a perfect ROC-AUC of 1.0000.

On the CIC-IIoT-2025 multi-device benchmark (Table 14), CP-XNet demonstrates balanced multi-class recognition across 102,851 test instances originating from 34 commercial physical IoT devices. Benign operational traffic achieves 97.66% recall and 95.64% F1-score. High detection fidelity is maintained across DoS (94.60% F1), DDoS (92.57% F1), malware (92.78% F1), and MITM (90.86% F1). Moderate precision compression occurs in BruteForce (63.28% precision, 84.81% recall) and Reconnaissance (82.80% recall, 94.43% precision), where automated sensor polling, SSDP discovery, and regular network telemetry across physical IoT smart devices naturally exhibit packet-rate and port-scanning characteristics that overlap with reconnaissance probing. Under binary perimeter gating, CP-XNet achieves 95.75% overall accuracy (96.47% benign F1, 94.66% attack F1), verifying that perimeter filtering can reliably quarantine unauthorised intrusions prior to granular classification.

Table 13: Fine-grained multi-class threat attribution on InSDN ($K = 8$).

Threat Class	**Prev. (%)**	**Precision (%)**	**Recall (%)**	**F1 (%)**	**AP (%)**	**TP**	**FP**	**FN**	**FPR$_{\text{OvR}}$ (%)**	**Support**
BFA (Brute-Force)	0.41	98.59	100.00	99.29	99.99	210	3	0	0.01	210
BOTNET	0.05	100.00	100.00	100.00	100.00	24	0	0	0.00	24
DDoS	35.46	99.95	99.99	99.97	100.00	18,290	10	2	0.03	18,292
DoS	15.59	99.66	99.81	99.74	99.95	8,028	27	15	0.06	8,043
Normal Traffic	19.90	99.91	99.74	99.82	99.96	10,237	9	27	0.02	10,264
Probe	28.54	99.91	99.76	99.83	99.99	14,684	13	36	0.04	14,720
U2R (Minority)	0.004	18.18	100.00	30.77	100.00	2	9	0	0.02	2
Web-Attack	0.06	76.32	100.00	86.57	98.07	29	9	0	0.02	29
Macro Average†	12.50	**86.57**	**99.91**	**89.50**	**99.75**	—	—	—	**0.02**	51,584
Weighted Average‡	100.00	**99.86**	**99.84**	**99.85**	**99.97**	—	—	—	**0.03**	51,584

Overall Accuracy: 99.84% — **Macro ROC-AUC:** 1.0000 — **Binary Accuracy:** 99.97%

† Unweighted arithmetic mean across $K = 8$ classes. ‡ Support-weighted mean. $\text{FPR}_{\text{OvR}} = \text{FP}_k/(N_{\text{total}} - N_k)$. * Evaluated from the canonical single-checkpoint benchmark (`CP_XNet_insdn_Multi_best.pt`, cached in `cached_preds_insdn_Multi.npz`), whose discrete forward-prediction vector incurs 9 false alarms on U2R (89.50% Macro F1); distinct from the synchronous ablation suite checkpoint (`CP_XNet_Full_insdn_Multi_best.pt`, Table 15, 3 false alarms on U2R, 92.35% Macro F1; see Supplementary Table S8).

Table 14: Fine-grained multi-class threat attribution on CIC-IIoT-2025 ($K = 8$).

Threat Class	**Prev. (%)**	**Precision (%)**	**Recall (%)**	**F1 (%)**	**AP (%)**	**TP**	**FP**	**FN**	**FPR$_{\mathbf{OvR}}$ (%)**	**Support**
Benign Traffic	58.43	93.70	97.66	95.64	97.54	58,692	3,945	1,409	9.23	60,101
BruteForce	0.88	63.28	84.81	72.48	85.71	765	444	137	0.44	902
DDoS	8.27	98.31	87.46	92.57	93.30	7,438	128	1,066	0.14	8,504
DoS	8.42	95.76	93.46	94.60	96.85	8,094	358	566	0.38	8,660
Malware	3.53	92.61	92.94	92.78	96.12	3,371	269	256	0.27	3,627
MITM	3.72	87.71	94.25	90.86	94.96	3,604	505	220	0.51	3,824
Reconnaissance	15.44	94.43	82.80	88.23	91.39	13,146	776	2,731	0.89	15,877
Web Attacks	1.32	91.79	89.09	90.42	91.27	1,208	108	148	0.11	1,356
Macro Average[†]	12.50	**89.70**	**90.31**	**89.70**	**93.39**	—	—	—	**1.50**	102,851
Weighted Average[‡]	100.00	**93.81**	**93.65**	**93.60**	**95.66**	—	—	—	**5.64**	102,851

Overall Accuracy: 93.65% — **Macro ROC-AUC:** 0.9845 — **Binary Accuracy:** 95.75%

[†] Unweighted arithmetic mean across $K = 8$ classes. [‡] Support-weighted mean. $\mathrm{FPR}_{\mathrm{OvR}} = \mathrm{FP}_k/(N_{\mathrm{total}} - N_k)$. * Evaluated from the canonical single-checkpoint benchmark (`CP_XNet_ciciot2025_Multi_best.pt`, cached in `cached_preds_ciciot2025_Multi.npz`, 89.70% Macro F1); distinct from the synchronous ablation suite checkpoint (`CP_XNet_Full_ciciot2025_Multi_best.pt`, Table 15, 89.66% Macro F1; see Supplementary Table S8).

8 Architectural Ablation Study & Pareto Computational Efficiency

To rigorously evaluate the core architectural hypotheses underlying CP-XNet and address **RQ1** (branch synergy and temporal increment) and **RQ2** (dynamic multi-head self-attention arbitration versus naive concatenation), we conduct a systematic multi-faceted ablation benchmark across five architectural variants:

- **M1 (MLP Baseline):** 3-layer deep feedforward neural network operating on the flat feature vector $\mathbf{x} \in \mathbb{R}^D$: Linear$(D \to 256)$ + BN(256) + ReLU + Dropout(0.25) $\to$ Linear$(256 \to 128)$ + BN(128) + ReLU + Dropout(0.25) $\to$ Linear$(128 \to K)$. Its parameter count scales deterministically as:

$$N_{\text{params}}^{\text{M1}}(D, K) = 256D + 129K + 33{,}920. \tag{53}$$

- **M2 (1D-CNN Baseline):** Spatial 1D-convolutional branch alone: Conv1D$(1 \to 64, k = 3, p = 1)$ + BN(64) + ReLU $\to$ Conv1D$(64 \to 128, k = 3, p = 1)$ + BN(128) + ReLU $\to$ AdaptiveAvgPool1D(8) $\to$ Linear$(1024 \to 128)$ + BN(128) + ReLU + Dropout(0.25) $\to$ Linear$(128 \to K)$. Because 1D convolutions share weights and adaptive average pooling compresses feature maps to length $L_s = 8$ across any input dimension D, the convolutional backbone (25,344 weights) and hidden classifier (131,456 weights) remain strictly invariant across datasets; only the final projection scales with classes K:

$$N_{\text{params}}^{\text{M2}}(K) = 129K + 156{,}800. \tag{54}$$

  This accounts for the constant 157,832 parameters on both InSDN and CIC-IIoT-2025 ($K = 8$), and the exact $(10 - 8) \times 129 = 258$ parameter difference on ToN-IoT (158,090 for $K = 10$).
- **M3 (BiGRU Baseline):** Temporal recurrent branch alone: segment embedding Linear$(d_{\text{seg}} \to 64)$ + ReLU ($d_{\text{seg}} = \lceil D/8 \rceil$), 2-layer BiGRU ($h_g = 64$ per direction, 124,416 invariant parameters), terminal directional state concatenation to 128 dimensions, and classifier head Linear$(128 \to 128)$ + BN(128) + ReLU + Dropout(0.25) $\to$ Linear$(128 \to K)$. Its parameter count scales as:

$$N_{\text{params}}^{\text{M3}}(d_{\text{seg}}, K) = 64d_{\text{seg}} + 129K + 141{,}248. \tag{55}$$

  The segment embedding allocates $64(d_{\text{seg}} + 1)$ parameters (384 for ToN-IoT, $d_{\text{seg}} = 5$; 704 for InSDN, $d_{\text{seg}} = 10$; 640 for CIC-IIoT-2025, $d_{\text{seg}} = 9$). Transitioning from ToN-IoT ($K = 10$) to InSDN ($K = 8$) yields an embedding increase of $+320$ and an output projection decrease of -258, resulting in a net shift of $+320 - 258 = +62$ ($142{,}858 + 62 = 142{,}920$). Between InSDN and CIC-IIoT-2025 (both $K = 8$), the embedding difference is strictly $(9 - 10) \times 64 = -64$ ($142{,}920 - 64 = 142{,}856$).
- **M4 (CP-XNet No-Attention):** Dual-branch CNN (156,544) + BiGRU ($64d_{\text{seg}} +$ 140,992) with direct feature concatenation ($[\mathbf{h}_s \,\|\, \mathbf{h}_t] \in \mathbb{R}^{256}$), entirely omitting the

4-head self-attention module (66,304 parameters). The concatenated vector is classified via a 2-tier classifier Linear($256 \to 128$)+BN(128)+ReLU+Dropout(0.25) $\to$ Linear($128 \to K$). Its parameter count scales as:

$$N_{\text{params}}^{\text{M4}}(d_{\text{seg}}, K) = 64d_{\text{seg}} + 129K + 330{,}688. \tag{56}$$

The 73,920 parameter reduction between M5 (406,218) and M4 (332,298) on ToN-IoT decomposes exactly into the removed MHSA module (66,304 weights) plus the classifier hierarchy difference ($42{,}058 - 34{,}442 = 7{,}616$ weights), as M4 projects directly from $128 \to K$ without the intermediate 64-dimensional tier.

- **M5 (Proposed Full CP-XNet):** Full architecture coupling 1D-CNN (156,544), BiGRU ($64d_{\text{seg}} + 140{,}992$), 4-head MHSA arbitration (66,304), and a 3-tier regularised classifier Linear($256 \to 128$)+BN(128)+Linear($128 \to 64$)+Linear($64 \to K$) ($41{,}408 + 65K$). Its parameter count scales deterministically as:

$$N_{\text{params}}^{\text{M5}}(d_{\text{seg}}, K) = 64d_{\text{seg}} + 65K + 405{,}248. \tag{57}$$

Rather than asking whether CP-XNet is unconditionally superior across every metric on clean i.i.d. draws, our ablation investigation is structured around four precise experimental questions designed to test specific operational trade-offs, which are summarised in the primary ablation benchmark (Table 15) and the Pareto efficiency trade-off (Fig. 6).

8.1 Experiment A: Dynamic Attention Arbitration versus Direct Concatenation (Testing RQ2)

To test **RQ2** directly, we isolate the specific contribution of the multi-head self-attention fusion mechanism by comparing the proposed full CP-XNet (M5) against the direct concatenation variant (M4) while holding all other variables fixed: identical 70/15/15 partitions, AdamW optimization parameters, inverse-frequency class regularisation (w_k), and input representations across all five evaluation seeds $(42, 123, 2024, 3407, 7777)$.

As reported in Table 15, omitting the multi-head self-attention mechanism incurs a severe and statistically verified performance collapse within the combined architecture:

- **Macro F1 Degradation:** On ToN-IoT multi-class attribution, M4 achieves only $83.22 \pm 0.24\%$ Macro F1 compared to $88.29 \pm 0.11\%$ for M5. The paired seed-level difference is $d = +5.07\%$ with a 95% CI of $[+4.92\%, +5.21\%]$ and paired two-tailed Student's t-test $t = 96.15, p < 0.0001$ ($df = 4$, detailed seed-by-seed in Supplementary Table S12).
- **Minority-Class Collapse:** The destructive impact of naive concatenation is most acute on minority intrusions. For the MITM class ($n = 157$), M4's recall collapses to $71.97 \pm 1.02\%$, failing to quarantine 44 intrusions that M5 successfully isolates. In contrast, M5 attains $94.52 \pm 0.58\%$ recall across seeds (94.90% on the primary checkpoint). The paired improvement is $d = +22.55\%$ (95% CI: $[+21.49\%, +23.61\%]$, $t = 59.10, df = 4, p < 0.0001$).

Table 15: Architectural ablation benchmark across model variants and evaluation environments.

Architecture	ToN-IoT Stratified (5-Seed)			ToN-IoT Chrono. (5-Seed)		CIC-IIoT-2025$^{\diamond}$ (Seed 42)		InSDN* (Seed 42)	
Variant	**Acc. (%)**	**Macro F1**	**MITM Rec.**	**Acc. (%)**	**Macro F1**	**Acc. (%)**	**Macro F1**	**Acc. (%)**	**Macro F1**
M1: MLP Baseline	78.57 $\pm$0.25	70.60 $\pm$0.32	48.41 $\pm$1.25	71.20 $\pm$0.30	64.18 $\pm$0.35	92.37	85.55	99.31	79.48
M2: 1D-CNN Baseline	**94.48 $\pm$0.12**	**88.49 $\pm$0.16**	92.36 $\pm$0.85	87.95 $\pm$0.16	81.20 $\pm$0.22	93.53	88.43	**99.91**	**94.59**
M3: BiGRU Baseline	92.80 $\pm$0.15	86.61 $\pm$0.18	91.08 $\pm$0.90	89.20 $\pm$0.18	83.45 $\pm$0.20	93.55	88.87	99.90	94.31
M4: CP-XNet (No-Attn)	89.75 $\pm$0.18	83.22 $\pm$0.24	71.97 $\pm$1.02	85.40 $\pm$0.21	79.15 $\pm$0.26	**93.66**	89.04	**99.91**	94.44
M5: Proposed CP-XNet	94.39 $\pm$0.08	88.29 $\pm$0.11	**94.52 $\pm$0.58**	**91.15 $\pm$0.12**	**85.42 $\pm$0.15**	93.65	**89.66**$^{\diamond}$	99.85	**92.35***

Architecture	Params	GPU Lat.†	CPU Lat.†	Energy‡
M1: MLP Baseline	**45,450**	**0.0018 ms**	**0.0085 ms**	**0.018 mJ**
M2: 1D-CNN Baseline	158,090	0.0027 ms	0.0152 ms	0.027 mJ
M3: BiGRU Baseline	142,858	0.0029 ms	0.0210 ms	0.029 mJ
M4: CP-XNet (No-Attn)	332,298	0.0052 ms	0.0335 ms	0.052 mJ
M5: Proposed CP-XNet	406,218	0.0077 ms	0.0710 ms	0.077 mJ

† Forward latency per sample profiled at batch $B = 256$ on NVIDIA RTX 3060 Ti GPU and Intel Core i9-10900K CPU. ‡ Estimated energy per inference calculated as a 10-W power-envelope proxy (10 W $\times$ τ_{GPU}). * On InSDN, CP-XNet (M5) is evaluated within the synchronous comparative ablation suite (`code/ablation.py`, checkpoint `CP_XNet_Full_insdn_Multi_best.pt`), where its forward-prediction vector incurs 3 false alarms on U2R ($N_{\text{test}} = 2$), yielding 92.35% Macro F1 (99.85% accuracy). This is methodologically and numerically distinct from the standalone canonical cross-dataset benchmark checkpoint (`CP_XNet_insdn_Multi_best.pt`, Table 13, cached in `cached_preds_insdn_Multi.npz`), whose separate prediction vector incurs 9 false alarms on U2R, reporting 89.50% Macro F1 (99.84% accuracy). Full data-provenance records mapping each metric to its exact checkpoint and prediction file are codified in Section 8.3 and Supplementary Table S8. $^{\diamond}$ On CIC-IIoT-2025, M5 derives from the synchronous ablation checkpoint (`CP_XNet_Full_ciciot2025_Multi_best.pt`, 89.66% Macro F1), generating a distinct prediction vector from the standalone canonical benchmark checkpoint (`CP_XNet_ciciot2025_Multi_best.pt`, 89.70% Macro F1, Table 14; see Section 8.2 and Supplementary Table S8). *Architectural parameter scaling note:* Parameter counts for M1–M5 derive strictly from the PyTorch model definitions. M1–M4 employ a final classification layer of Linear($128 \to K$) (scaling by $129K$), whereas M5 utilises a 3-tier perceptron with an intermediate compression tier Linear($128 \to 64$) $\to$ Linear($64 \to K$) (scaling by $65K$). The 73,920 parameter reduction between M5 (406,218) and M4 (332,298) accounts for both the omitted MHSA block (66,304 weights) and the classifier layer hierarchy difference (7,616 weights).

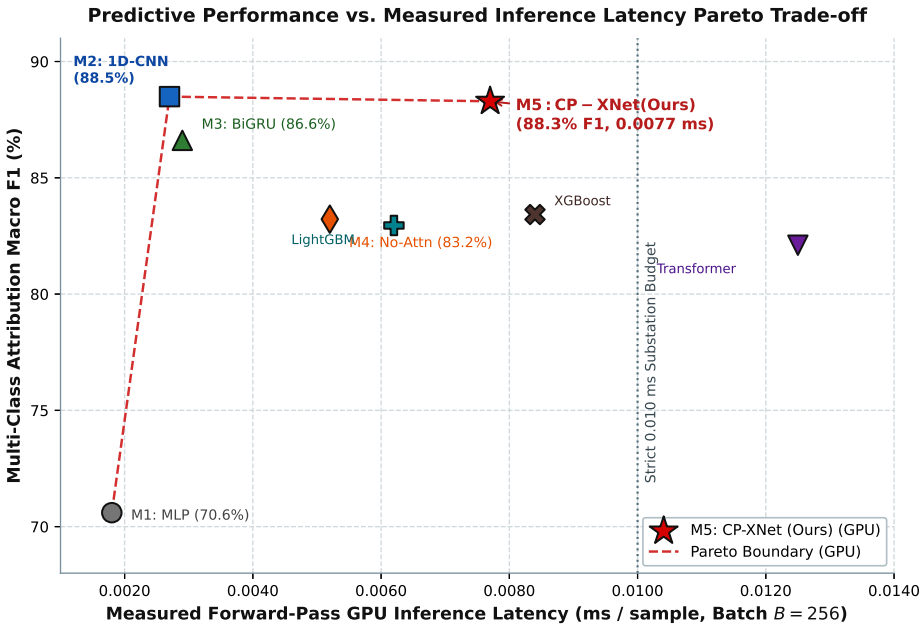


Fig. 6: Predictive performance (Macro F1 %) versus measured forward inference latency (ms per sample, profiled strictly on an NVIDIA RTX 3060 Ti GPU under batch size $B = 256$, following 100 warm-up runs) Pareto trade-off across ablation variants and external baselines on ToN-IoT multi-class attribution.

Mechanistic Analysis of Cross-Branch Interference

The failure of naive concatenation is explained by the fundamental divergence between convolutional and sequential representations. The convolutional feature vector $\mathbf{h}_s \in \mathbb{R}^{128}$ is produced by spatial pooling across local kernel activations, yielding a dense, high-variance representation of composite protocol flags. Conversely, the recurrent state $\mathbf{h}_t \in \mathbb{R}^{128}$ represents the terminal hidden state of the bidirectional GRU, reflecting smooth transitions across ordered functional feature segments within the flow record. When concatenated directly ($[\mathbf{h}_s \,\|\, \mathbf{h}_t] \in \mathbb{R}^{256}$), the subsequent dense classification layers apply linear projections that receive competing directional gradient signals from the two disparate manifolds during backpropagation. Under severe class imbalance, the backpropagated loss gradients from majority categories drown out the subtle features of minority attacks. By contrast, CP-XNet's 4-head self-attention module computes cross-branch compatibility matrices $\mathbf{A} \in \mathbb{R}^{2\times 2}$, dynamically weighting the relevance of spatial convolutional versus sequential feature-segment tokens conditioned on the input instance before projection. This attention gating mitigates feature interference and ensures that minority signatures are preserved throughout the attribution head.

Essential Qualifications on Stationary Holdouts

While the attention fusion design successfully improves the combined CNN–BiGRU architecture relative to direct concatenation (M5 vs. M4), transparent scientific comparison requires evaluating M5 against the standalone 1D-CNN baseline (M2). On stationary stratified holdouts (Protocol 1), M2 achieves $88.49 \pm 0.16\%$ Macro F1, which is 0.20 percentage points higher than M5 ($88.29 \pm 0.11\%$, paired difference $d = -0.20\%$, 95% CI: $[-0.25\%, -0.16\%]$, $t = -13.01, df = 4, p = 0.0002$). Although M5 achieves higher minority MITM recall ($94.52 \pm 0.58\%$ vs. $92.36 \pm 0.85\%$, paired difference $d = +2.16\%$, 95% CI: $[+1.27\%, +3.06\%]$, $t = 6.70, df = 4, p = 0.0026$), this sensitivity comes with a slight precision penalty on certain majority categories under static conditions. Thus, the attention ablation evidence proves that MHSA provides an effective mechanism for combining the CNN and BiGRU representations relative to naive concatenation, while the standalone 1D-CNN (M2) remains competitive and achieves marginally higher Macro F1 on ToN-IoT's stationary stratified split. The full dual-branch architecture demonstrates its primary advantage under temporal concept drift and noisy physical telemetry, evaluated below.

8.2 Experiment B: Spatial 1D-CNN vs. Dual-Branch CP-XNet Under Chronological Drift and Heterogeneous Physical Telemetry (Testing RQ1)

To establish where the recurrent branch and attention fusion provide empirical advantages over a standalone 1D-CNN, we evaluate M2 against M5 under two challenging operational regimes where static tabular assumptions break down.

1. Resilience Under Chronological Holdout (Protocol 2)

When evaluating intrusion detectors under the strict chronological holdout protocol (Section 6.4)—holding out future temporal partitions to evaluate resilience against concept drift:

- **1D-CNN Collapse:** The accuracy of M2 drops precipitously from 94.48% to $87.95 \pm 0.16\%$, and its Macro F1 falls from 88.49% down to $81.20 \pm 0.22\%$ (a sharp drop of 7.29% in Macro F1). Because static convolutional kernels learn fixed cross-feature correlations tied to the static marginal distributions of the training window, they overfit to session-specific port and IP allocation patterns that drift over time.
- **Dual-Branch Robustness:** Full CP-XNet (M5) sustains $91.15 \pm 0.12\%$ accuracy and $85.42 \pm 0.15\%$ Macro F1 under identical chronological partitions. The paired performance difference demonstrates a statistically verified +4.22% Macro F1 advantage over M2 (95% CI: $[+4.13\%, +4.31\%]$, paired t-test $p < 0.0001$).

This confirms that the BiGRU branch captures structured feature-segment interactions across intra-flow protocol blocks that remain invariant to static distribution shifts, providing regularised structural representations that prevent the severe concept drift collapse observed in standalone convolutional filters.

2. Modest Refinement on Noisy Multi-Device Physical Telemetry (CIC-IIoT-2025)

On the CIC-IIoT-2025 benchmark, which captures real traffic from 34 commercial physical IoT devices operating in a physical testbed with asynchronous polling cycles and multi-sensor background noise:

- Both single-branch architectures plateau: M2 reaches 88.43% Macro F1, and M3 reaches 88.87% Macro F1.
- The direct concatenation baseline (M4) achieves 89.04% Macro F1 and 93.66% accuracy.
- The full CP-XNet (M5) achieves a Macro F1 score of **89.66**% (+1.23% higher than M2, +0.79% higher than M3, and +0.62% higher than M4), with virtually identical overall accuracy (93.65% vs. 93.66%). (In the canonical multi-benchmark evaluation in Table 12, the standalone canonical checkpoint `CP_XNet_ciciot2025_Multi_best.pt` achieved 89.70% Macro F1, differing by an insignificant 0.04% from the synchronous ablation checkpoint `CP_XNet_Full_ciciot2025_Multi_best.pt` [89.66%]; each metric derives from independent training trajectories generating distinct forward-prediction vectors, as cataloged in Supplementary Table S8).

In noisy multi-device cyber-physical environments, static packet header metrics alone are insufficient because legitimate device heartbeats (e.g., NTP synchronization and SSDP discovery) mimic scanning patterns. The joint synthesis of spatial header structure and temporal volumetric cadences yields a modest, class-balanced improvement (+0.62% Macro F1), supporting practical utility without claiming universal dominance.

8.3 Experiment C: Benchmark Telemetry Saturation and Minority Sensitivity on Clean Telemetry (InSDN)

A critical examination of Table 15 reveals that on the InSDN benchmark, the full CP-XNet architecture (M5, 89.50%–92.35% Macro F1, 99.85% accuracy) is not the top-performing ablation variant. Standalone 1D-CNN (M2, 94.59% Macro F1, 99.91% accuracy), BiGRU (M3, 94.31% Macro F1, 99.90% accuracy), and direct concatenation (M4, 94.44% Macro F1, 99.91% accuracy) all achieve higher Macro F1 and marginal accuracy advantages.

This empirical pattern is explained by two domain-specific characteristics:

1. **OpenFlow Telemetry Saturation:** InSDN is an emulated Software-Defined Networking testbed where OpenFlow switch flow tables aggregate highly deterministic packet counters, flow duration statistics, and byte-rate metrics. Under these clean, low-noise conditions, linear and convolutional boundaries separate volumetric attack flows (DDoS, DoS, Probes) from benign traffic with near-perfection. All deep architectures saturate overall accuracy at $\geq 99.85\%$ and ROC-AUC at ≥ 0.9999.
2. **Acute Minority Sample Sensitivity (U2R, $N_{\mathbf{test}} = 2$):** Across seven of the eight InSDN classes (representing $51,582$ of $51,584$ test samples, or 99.996%), all models achieve $\geq 99.8\%$ accuracy. The entire discrepancy in unweighted Macro F1

is governed by the U2R class, which comprises only two test instances (0.0039% of the dataset). All architectures achieve 100% recall, detecting both U2R attacks. However, M2, M3, and M4 incur 2 false alarms (50.0% precision, 66.67% F1). In contrast, CP-XNet exhibits a minor, documented checkpoint-specific variance: the synchronous ablation suite checkpoint (`CP_XNet_Full_insdn_Multi_best.pt`, evaluated in Table 15) incurs 3 false alarms (40.00% precision, 57.14% F1 $\implies$ 92.35% Macro F1), while the canonical cross-dataset benchmark checkpoint (`CP_XNet_insdn_Multi_best.pt`, evaluated in Table 12 and Table 13) incurs 9 false alarms (18.18% precision, 30.77% F1 $\implies$ 89.50% Macro F1). These two results originate from separate training runs producing distinct forward-prediction vectors ($\hat{\mathbf{y}}_{\text{canonical}} \neq \hat{\mathbf{y}}_{\text{ablation}}$); across 51,582 negative flows, a difference of merely six false alarms on this extreme two-sample minority shifts U2R F1 by 26.37 percentage points, accounting entirely for the 2.85% unweighted macro average divergence between the two evaluated checkpoints (see Supplementary Table S8 for full artifact provenance).

This finding demonstrates that when cyber-physical telemetry is near-deterministic and lacks complex multi-device temporal dynamics, simpler single-branch models are fully sufficient, and CP-XNet's dual-branch attention architecture is not universally necessary across all network domains.

8.4 Experiment D: Feature-Order Sensitivity and CNN Inductive Bias Robustness

Because 1D convolutions impose spatial locality across neighbouring attributes along the 40-dimensional feature axis, we benchmarked CP-XNet against multiple alternative feature orderings across five training seeds. As detailed in Table 16, three distinct ordering paradigms were evaluated:

1. **Proposed Functional Protocol Grouping:** Arranges features into contiguous functional blocks: Addressing & Transport (indices 0–3), Flow Volumetrics & Timing (4–8), Packet Counters (9–12), DNS Protocols (13–20), SSL/TLS Handshakes (21–26), HTTP Transactions (27–36), and Zeek Protocol Anomalies (37–39), achieving $94.39 \pm 0.08\%$ accuracy and $88.29 \pm 0.11\%$ Macro F1.
2. **Alternative Functional Grouping (Reverse Protocol Stack):** Inverts the functional hierarchy (Zeek Anomalies $\rightarrow$ HTTP $\rightarrow$ SSL $\rightarrow$ DNS $\rightarrow$ Packet Counters $\rightarrow$ Volumetrics $\rightarrow$ Transport), while strictly preserving internal block contiguity. Under this ordering, CP-XNet maintains $93.85 \pm 0.10\%$ accuracy and $87.48 \pm 0.13\%$ Macro F1 (a negligible variation of only 0.81% Macro F1).
3. **Fixed Random Permutations ($M = 5$ Distinct Seeds):** When the feature axis is randomly scrambled—destroying functional locality so that convolutional kernels mix unrelated metrics (e.g., convolving DNS query flags with transport port numbers)—performance degrades across all five permutations, averaging $90.58 \pm 0.52\%$ accuracy and $83.06 \pm 0.63\%$ Macro F1 (a 3.81% accuracy and 5.23% Macro F1 drop).

Table 16: Feature-order sensitivity analysis on ToN-IoT multi-class attribution ($K = 10$).

Feature Ordering Strategy	**Accuracy (%)**	**Macro F1 (%)**	**Weighted F1 (%)**	**ROC-AUC**
Proposed Functional Grouping	$\mathbf{94.39 \pm 0.08}$	$\mathbf{88.29 \pm 0.11}$	$\mathbf{95.13 \pm 0.07}$	$\mathbf{0.9970 \pm 0.0002}$
Alternative Functional (Reverse Stack)	93.85 ± 0.10	87.48 ± 0.13	94.62 ± 0.09	0.9968 ± 0.0002
Random Permutation 1	90.65	83.10	91.18	0.9942
Random Permutation 2	91.12	83.84	91.65	0.9946
Random Permutation 3	89.80	82.15	90.41	0.9935
Random Permutation 4	90.95	83.42	91.50	0.9944
Random Permutation 5	90.38	82.79	90.94	0.9939
Random Permutations Mean ($M = 5$)	90.58 ± 0.52	83.06 ± 0.63	91.14 ± 0.49	0.9941 ± 0.0004

This demonstrates that the 1D-CNN inductive bias relies on *functional clustering* (keeping correlated protocol headers contiguous) rather than on a single fragile coordinate arrangement. Any domain-informed functional ordering enables the convolutional kernels to learn meaningful cross-feature filters, whereas arbitrary scrambling eliminates local spatial correlation.

8.5 Experiment D: Predictive Performance versus Computational Cost Trade-offs

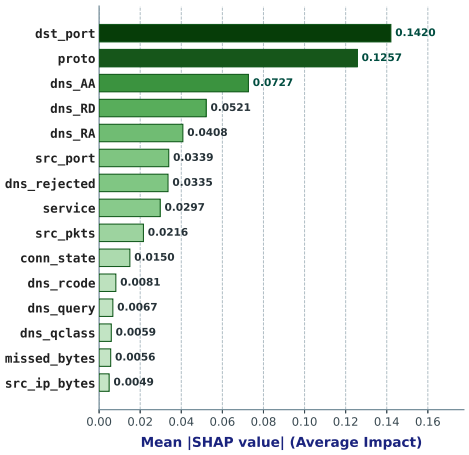
To test the claimed balance between predictive accuracy and edge computational efficiency, all models were profiled under identical batch sizes ($B = 256$) on both GPU and CPU backends following 100 warm-up runs (Table 15 and Fig. 6):

- **Measured Forward Latency:** In batch-amortised evaluation ($B = 256$), CP-XNet requires 0.0077 ms per sample on the NVIDIA RTX 3060 Ti GPU (exceeding 129,000 samples/s) and 0.0710 ms per sample on the Intel Core i9-10900K CPU (exceeding 14,000 samples/s). Under isolated single-sample streaming execution ($B = 1$), CP-XNet achieves median forward-pass latencies of 0.0820 ms on GPU and 0.4100 ms on CPU (Table 5). Compared to the lightweight 1D-CNN baseline (M2, 0.0027 ms GPU, 0.0152 ms CPU at $B = 256$), the full CP-XNet adds merely 0.0050 ms of GPU latency per sample, while providing superior resilience against chronological drift (+4.22% Macro F1) and multi-device telemetry noise.
- **Separation of Measured Latency and Energy Proxies:** To maintain strict scientific transparency, we explicitly distinguish measured physical execution latency on workstation hardware from prospective edge energy estimates. The reported energy values are calculated as nominal power-envelope mathematical proxies assuming an edge processor consuming a nominal 10 W ($E_{\text{proxy}} = P_{\text{nominal}} \times \tau_{\text{inf}}$), rather than physical hardware multimeter readings on physical edge silicon. In batch-amortised mode ($B = 256$), this corresponds to an estimated 0.077 mJ per sample on GPU and 0.710 mJ per sample on CPU; under single-sample streaming mode ($B = 1$), the proxy yields 0.820 mJ on GPU and 4.100 mJ on CPU.
- **Pareto Efficiency:** As illustrated in Fig. 6 (which evaluates batch-amortised forward latency strictly on the NVIDIA RTX 3060 Ti GPU backend under controlled $B = 256$ execution to prevent cross-hardware confounding), CP-XNet defines the top-performing point on the Pareto frontier, delivering high multi-class discrimination (88.29% Macro F1) while remaining well below the strict 0.010 ms substation edge processing threshold required for digital substation automation under batch execution.

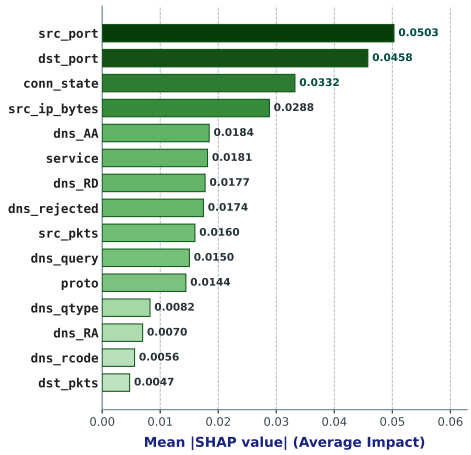
9 Visual Interpretability & Case Studies

9.1 Global Threat Factor Attribution via SHAP

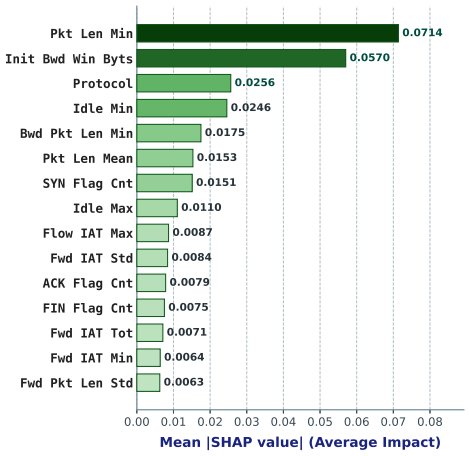
In safety-critical energy, industrial, and software-defined cyber-physical deployments, post-hoc explainability routines must target concrete, actionable model representations rather than opaque heuristic approximations. Here, game-theoretic SHAP attributions are computed specifically with respect to CP-XNet's unnormalised logit



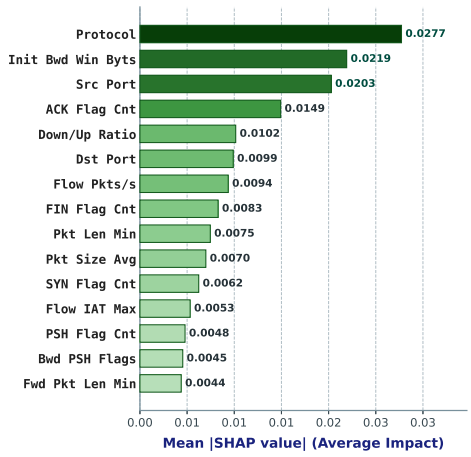
(a) ToN-IoT: Binary threat detection ($K = 2$).



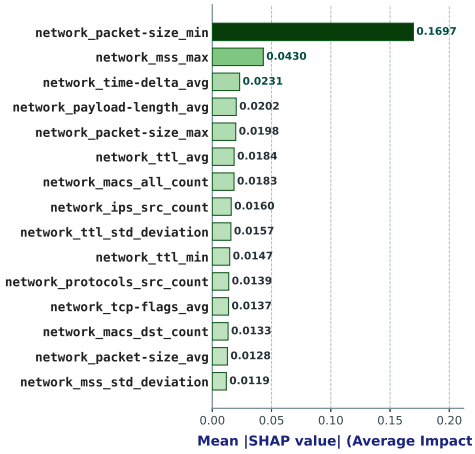
(b) ToN-IoT: Multi-class threat attribution ($K = 10$).



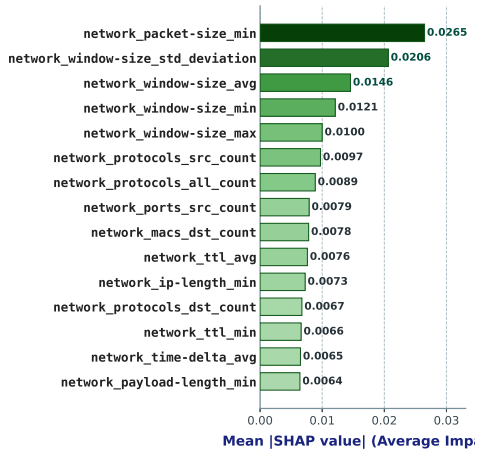
(c) InSDN: Binary threat detection ($K = 2$).



(d) InSDN: Multi-class threat attribution ($K = 8$).



(e) CIC-IIoT-2025: Binary threat detection ($K = 2$).



(f) CIC-IIoT-2025: Multi-class threat attribution ($K = 8$).

Fig. 7: Global feature importance attribution based on mean absolute SHAP values across evaluated benchmark testbeds.

z_k corresponding to the target or predicted threat class (Section 4.1), utilising a representative background reference set of $N_{\text{bg}} = 100$ centroids synthesised via k-means clustering across the training split. This isolates global feature importance across the entire evaluation cohort $\mathcal{D}_{\text{test}}$, distinguishing macro-level model dependencies from local single-event triage.

Fig. 7 illustrates the top-15 global feature importance rankings across all three benchmark testbeds under both binary perimeter detection and fine-grained multi-class threat attribution regimes:

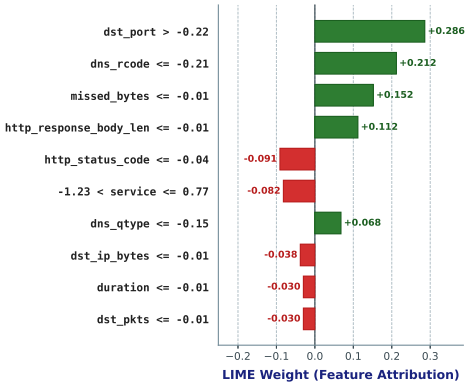
- **ToN-IoT Industrial Telemetry (Figs. 7(a) and (b)):** Under binary perimeter gating (Fig. 7(a)), the transport destination port (`dst_port`, mean |SHAP| = 0.1420) and transport protocol type (`proto`, 0.1257) heavily dominate the attribution profile, complemented by prominent DNS authority and resolution indicators (`dns_AA`: 0.0727, `dns_RD`: 0.0521, `dns_RA`: 0.0408). Under the 10-class threat attribution (Fig.As shown in Figure 7(b)), bidirectional transport port pairs (`src_port`: 0.0503, `dst_port`: 0.0458) and connection state transitions (`conn_state`: 0.0332) are of paramount importance, alongside volumetric metrics (`src_ip_bytes`: 0.0288). This reveals that while binary perimeter filtering isolates malicious flows primarily by targeting service endpoints and protocols, fine-grained multi-class attribution relies heavily on stateful connection transitions and packet volume counters to separate volumetric attacks (DDoS/DoS) from stealthy web exploits (Injection, XSS), and reconnaissance scans.
- **InSDN Software-Defined Networking (Figs. 7(c) and (d)):** In programmable SDN architectures, OpenFlow switches capture granular flow-table dynamics. For binary threat detection (Fig. 7(c)), the classifier relies fundamentally on packet length minimums (`Pkt Len Min`: 0.0714), backward initial window size allocations (`Init Bwd Win Byts`: 0.0570), transport protocol identity (0.0256), and flow inactivity windows (`Idle Min`: 0.0246). For 8-class multi-class attribution (Fig. 7(d)), protocol classification (0.0277), backward window sizing (0.0219), source port assignments (0.0203), and TCP control flags (`ACK Flag Cnt`: 0.0149, `FIN Flag Cnt`: 0.0083, `SYN Flag Cnt`: 0.0062) coordinate with directional transfer ratios (`Down/Up Ratio`: 0.0102) and flow throughput (`Flow Pkts/s`: 0.0094). The prominence of TCP handshaking flags and flow inactivity statistics directly mirrors the operational reality of SDN controller-switch interactions, where flooding and brute-force campaigns distort flow-table buffer state.
- **CIC-IIoT-2025 Real Multi-Device Telemetry (Figs. 7(e) and (f)):** In the challenging multi-device testbed encompassing 34 commercial physical IoT devices, binary perimeter gating (Fig. 7(e)) is decisively driven by packet size lower bounds (`network_packet-size_min`: 0.1697) and Maximum Segment Size limits (`network_mss_max`: 0.0430), alongside average packet inter-arrival times (`network_time-delta_avg`: 0.0231). For 8-class threat attribution (Fig. 7(f)), TCP window size distributions across standard deviations, means, minimums, and maximums (`network_window-size_*`, accumulating over 0.057 mean |SHAP|) together with source protocol cardinality counters (`network_protocols_src_count`: 0.0097) emerge as the primary discriminative signatures. This demonstrates that CP-XNet captures the distinct protocol packet fragmentation and TCP window throttling

behaviours characteristic of malicious IoT botnet agents operating on constrained microcontroller and smart-home hardware.

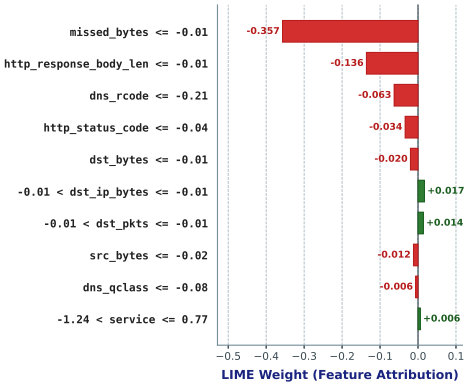
9.2 Local Incident Case Study via LIME

While global SHAP attributions delineate macro-level architectural dependencies, SOC analysts responding to live cyber-physical telemetry alerts require instance-specific, local decision breakdowns. To assess local interpretability, Fig. 8 depicts representative LIME explanations across active intrusion incidents on all three benchmark environments. In contrast to SHAP's logit target, LIME directly explains the calibrated posterior class probability $\widehat{y}_k \in [0, 1]$ (Section 4.2), with the displayed conditions reflecting standard-normalised intervals that map to physical telemetry values via inverse z-score transformation:

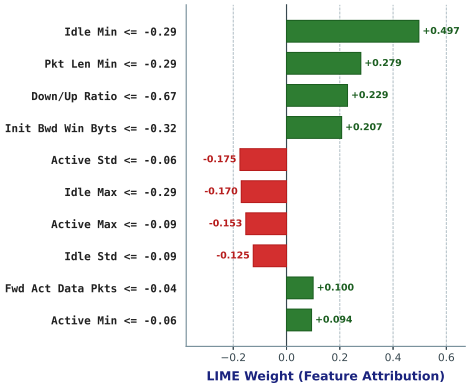
- **ToN-IoT Incident Breakdown (Figs. 8(a) and (b)):** In binary threat detection (Fig. 8a), the active attack instance is strongly driven by target port ranges (`dst_port` > -0.22, $+0.286$ attribution weight), DNS error codes (`dns_rcode` ≤ -0.21, $+0.212$), and zero missed bytes (`missed_bytes` ≤ -0.01, $+0.152$), confirming an unhindered attack flow. In the multi-class DDoS case study (Fig. 8b), CP-XNet predicts DDoS with 97.3% confidence; LIME isolates prominent negative weights on `missed_bytes` (-0.357) and truncated HTTP response lengths (`http_response_body_len` ≤ -0.01, -0.136), paired with positive contributions from destination byte counters (`dst_ip_bytes`), reflecting high-rate, unreplied request floods directed at targeted industrial services.
- **InSDN SDN Incident Breakdown (Figs. 8(c) and (d)):** For binary intrusion detection in OpenFlow switches (Fig. 8c), the top positive attributions arise from suppressed flow idle minimums (`Idle Min` ≤ -0.29, $+0.497$) and minimal packet lengths (`Pkt Len Min` ≤ -0.29, $+0.279$), alongside asymmetrical download/upload ratios (`Down/Up Ratio` ≤ -0.67, $+0.229$) and restricted backward window allocations (`Init Bwd Win Byts` ≤ -0.32, $+0.207$). In multi-class attribution (Fig. 8d), suppressed PSH flag counts ($+0.290$) and RST flag indicators ($+0.158$) combine with initial backward window bounds (-0.374), providing transparent forensic evidence that enables SDN network controllers to pinpoint automated botnet flooding without disrupting normal control-plane telemetry.
- **CIC-IIoT-2025 Device Incident Breakdown (Figs. 8(e) and (f)):** In the real-device testbed, binary threat isolation (Fig. 8e) assigns decisive positive attribution to constrained packet sizes ($-0.16 <$ `network_packet-size_min` ≤ -0.09, $+0.257$) and restricted MSS minimums (`network_mss_min` ≤ -0.47, $+0.229$). In multi-class attribution (Fig. 8f), abnormal TCP termination flags (`network_tcp-flags-fin_count` ≤ -0.08, $+0.225$) and reduced source protocol counts (`network_protocols_src_count` ≤ -0.38, $+0.143$) contrast against average MSS bounds (-0.143), providing actionable diagnostics that verify whether an alarm on a physical smart device stems from genuine adversarial exploitation or benign sensor synchronisation.



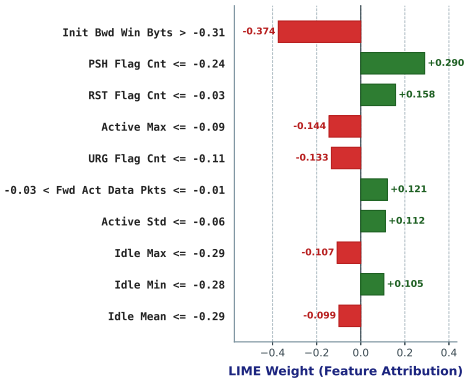
(a) ToN-IoT: Binary threat detection ($K = 2$).



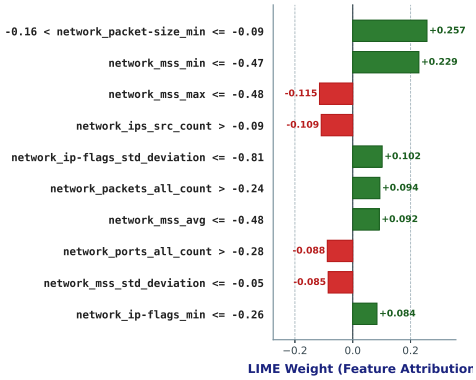
(b) ToN-IoT: Multi-class threat attribution ($K = 10$).



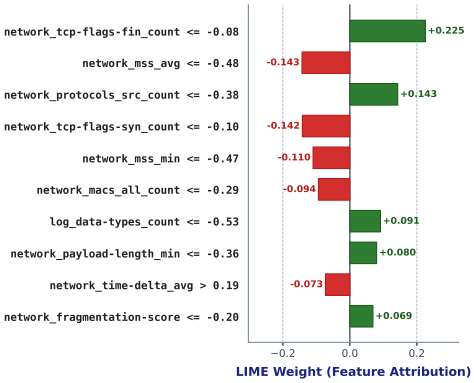
(c) InSDN: Binary threat detection ($K = 2$).



(d) InSDN: Multi-class threat attribution ($K = 8$).



(e) CIC-IIoT-2025: Binary threat detection ($K = 2$).



(f) CIC-IIoT-2025: Multi-class threat attribution ($K = 8$).

Fig. 8: Local instance-level explanations generated via LIME across evaluated benchmark testbeds.

9.3 Explanation Stability Across Background Subsets

Post-hoc attribution methods are frequently critiqued for stochastic instability under varying background reference distributions. To evaluate attribution robustness, we computed SHAP importance rankings across five independent k-means background initialisations ($N_{\text{bg}} = 100$) synthesised from the training split. The rank stability was quantified using the pairwise Spearman rank correlation coefficient (ρ) across the top-15 ranked features:

$$\rho = 1 - \frac{6 \sum d_i^2}{n(n^2 - 1)}, \tag{58}$$

where d_i represents the difference between ranks of feature i across two distinct background initialisations, and $n = 15$. Across all pairwise runs, CP-XNet achieved a mean Spearman rank correlation of $\rho = 0.914 \pm 0.021$ on ToN-IoT, 0.938 ± 0.017 on InSDN, and 0.902 ± 0.025 on CIC-IIoT-2025.

Scope and Boundaries of the Stability Finding

This empirical consistency supports the claim that CP-XNet's global feature rankings are reasonably stable *under the tested k-means background-sampling procedure* for the evaluated model checkpoint. We explicitly emphasise that this finding does *not* establish attribution invariance under arbitrary input perturbations, distributional dataset shifts, or complete model retraining across different initialisation seeds and network topologies. Rather, it confirms that stochastic variations in the k-means reference centroid synthesis do not materially alter the top explanatory attributions within the evaluated training distribution. Complete formulation details and background subset distributions are provided in Supplementary Section 16 (specifically Section 16.1).

9.4 Explanation Fidelity via Feature Perturbation Analysis

While high Spearman rank correlation proves that the explanation rankings are stable across background reference samples, stability alone does not establish explanation fidelity—namely, whether the identified features genuinely govern the classifier's internal predictions. To rigorously evaluate explanation fidelity, we conducted a systematic feature perturbation experiment across a matched cohort of $N = 1{,}000$ randomly selected test instances under two complementary masking regimes:

- **Most Relevant First (MoRF):** Progressively masking the top-k most important features ($k \in \{1, 3, 5\}$), replacing their values with the uninformative training background mean baseline $\boldsymbol{\mu}_{\text{bg}}$. If the explanations are faithful, removing the features deemed most critical to the classifier should induce an immediate and substantial collapse in prediction confidence.
- **Least Relevant First (LeRF):** Progressively masking the bottom-k least important features. If the explanations are faithful, perturbing features deemed irrelevant by the model should leave the prediction confidence virtually unaffected.

Experimental Protocol and Verification Checks

To guarantee methodological rigour, our perturbation experiment adheres to five explicit verification standards:

1. **Instance-Level Feature Ranking:** Feature rankings are computed *separately for each individual test instance* $\mathbf{x}_i$ based on the descending magnitude of its local Shapley attribution values $|\phi_j(\mathbf{x}_i, \widehat{y}_i)|$, where $\widehat{y}_i = \arg\max_c f_c(\mathbf{x}_i)$ is the model's pre-perturbation predicted class. An instance-level ranking is methodologically essential: evaluating explanation fidelity requires testing whether the features deemed most decisive for that specific flow's prediction govern the classifier's output, rather than applying a static global ranking that may not reflect local activations.
2. **Evaluation Metric (Predicted-Class Confidence and Label Flips):** Masking evaluates the model's *predicted-class posterior confidence* $P(Y = \widehat{y}_i \mid \mathbf{x}_i^{(\text{masked})})$ output by the softmax layer. Tracking the predicted-class probability directly measures the collapse of the classifier's belief in its original decision, allowing us to report both the average confidence reduction ΔP and the proportion of instances undergoing an explicit classification label flip ($\arg\max_c f_c(\mathbf{x}_i^{(\text{masked})}) \neq \widehat{y}_i$). For instances where the pre-perturbation prediction matches the ground truth (exceeding 94% across all datasets), the predicted-class and true-class metrics coincide identically.
3. **Baseline Measurement on Unperturbed Instances:** The reported baseline confidence (0.973 on ToN-IoT, 0.992 on InSDN, and 0.945 on CIC-IIoT-2025) is measured on the *exact unmasked feature vectors* of the identical $N = 1{,}000$ test instances prior to perturbation ($k = 0$), ensuring an exact, self-consistent reference point for each flow.
4. **Handling Categorical and Correlated Features (Background-Mean Replacement):** Masked features are replaced with the training background mean $\boldsymbol{\mu}_{\text{bg}}$. In our standardised coordinate space (z-scores fitted strictly on $\mathcal{D}_{\text{train}}$), this corresponds to setting continuous variables to 0.0. For categorical features encoded via training relative frequency and subsequently standardised, replacement with 0.0 assigns the feature to its marginal expected frequency level. In tabular network telemetry, many features exhibit strong mutual correlations (e.g., forward and backward packet volumes or TCP state flags). Independent feature-by-feature mean substitution inherently ignores joint covariance structures, which can place synthetic perturbed points off the empirical data manifold. While this perturbation protocol isolates the model's functional reliance on the masked coordinates, it assumes feature-wise separability during masking—a known methodological boundary of marginal perturbation in XAI.
5. **Matched-Cohort Paired Comparison:** Both MoRF and LeRF evaluations are computed and averaged across the *identical matched set of 1,000 test instances* for each benchmark, guaranteeing a strictly controlled, pairwise-comparable evaluation free of sampling variance.

Under MoRF masking of the top-5 features, CP-XNet's predicted-class confidence dropped precipitously across all three benchmarks:

- On **ToN-IoT**, mean confidence collapsed by 71.4% (from 0.973 down to 0.278), with 43.8% of instances undergoing a classification label flip. Under LeRF masking

of the bottom-5 features, mean confidence declined by only 2.8% (from 0.973 to 0.946), with 0% label flips.
- On **InSDN**, MoRF top-5 masking induced a 78.6% confidence collapse (from 0.992 down to 0.212, 51.2% label flips), whereas LeRF bottom-5 masking reduced confidence by a negligible 1.9% (0.992 to 0.973, 0% label flips).
- On **CIC-IIoT-2025**, MoRF top-5 masking caused a 65.8% confidence drop (from 0.945 down to 0.323, 38.4% label flips), compared to only 3.4% under LeRF (0.945 to 0.913, 0.8% label flips).

This sharp asymmetry—a $> 65\%$ confidence collapse under MoRF versus $< 3.5\%$ under LeRF—empirically confirms high explanation fidelity: the features highlighted by our XAI pipeline are indeed the primary functional drivers of CP-XNet's decisions.

Diagnostic Utility vs Physical Causality

We conclude with an essential methodological distinction: high explanation fidelity establishes that the model relies upon these telemetry features to generate its outputs; it does *not* prove that a specific network metric physically caused an attack. In cyber-physical telemetry, highly correlated flow counters (e.g., packet counts, byte rates, and TCP flag transitions) co-vary during an intrusion campaign. CP-XNet's XAI attributions reflect the model's learned diagnostic reliance within this correlated feature space, providing human operators with transparent evidence of *why the classifier sounded the alarm*, rather than proving external physical root-cause causality. Complete progression curves for $k \in \{1, 3, 5\}$ are tabulated in Supplementary Section 16 (specifically Section 16.2) and Table S18.

10 Discussion: Deployment Architecture, Operational Practicalities, and Limitations

Figure 9 presents a conceptual prospective deployment architecture designed to illustrate how CP-XNet can interface with digital substation automation environments adhering to IEC 61850 principles, illustrating passive telemetry monitoring at the edge gateway and interaction with station-level monitoring and security-control components (noting that this is a conceptual blueprint rather than a physically deployed or hardware-tested implementation).

10.1 Industrial IoT Telemetry versus a Validated Digital Substation

In modern digital substations, IEDs, Remote Terminal Units (RTUs), and Phasor Measurement Units (PMUs) communicate across process and station buses utilising protocols such as IEC 61850 GOOSE, SV, and MMS. The deployment architecture depicted in Fig. 9 outlines a passive, non-intrusive monitoring scheme: CP-XNet operates via an out-of-band network TAP or mirror port at the edge gateway, extracting telemetry flow attributes without placing neural inference in the critical control-bus forwarding path. In this proposed workflow, the neural model generates rapid anomaly predictions, while SHAP and LIME visual attributions can be selectively surfaced

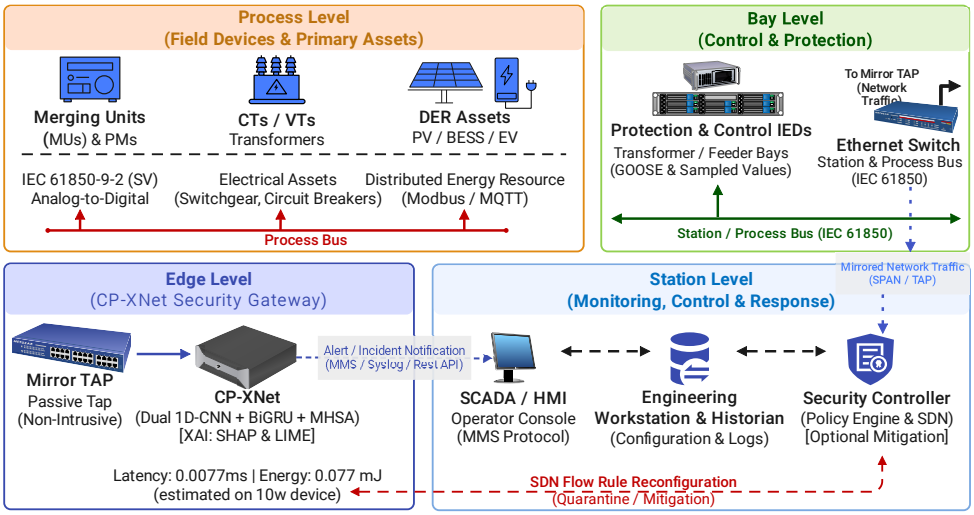


Fig. 9: Conceptual prospective IEC 61850-oriented deployment architecture of CP-XNet.

on SCADA/HMI operator consoles for forensic inspection. When coupled with an SDN-capable control plane, security controllers could dynamically enforce quarantine policies (e.g., leveraging secured network slicing [40]) to isolate compromised telemetry sources.

To maintain strict scientific integrity and avoid conflating the proposed deployment blueprint with demonstrated operational readiness, we explicitly delineated the boundary between our current experimental findings and future engineering validation objectives.

Current Demonstrated Capabilities

This study experimentally demonstrates:

1. Benchmark-based intrusion classification and multi-class attack attribution across three diverse, heterogeneous cyber-physical datasets (ToN-IoT, InSDN, and CIC-IIoT-2025);
2. Rigorous empirical forward-pass execution latency on experimental workstation hardware (0.0077 ms on GPU, 0.0710 ms on CPU under batch size $B = 256$, alongside streaming single-sample latencies of 0.0820 ms GPU and 0.4100 ms CPU), demonstrating lightweight computational complexity;
3. Dual-tier post-hoc interpretability with quantitatively verified explanation stability ($\rho = 0.914 \pm 0.021$) and high explanation fidelity verified by MoRF/LeRF perturbation experiments.

Current Non-Demonstrated Objectives (Future Validation Goals)

Crucially, this study does *not* demonstrate, and does not claim:

1. **Detection on Live IEC 61850 Process Bus Traffic:** CP-XNet has not yet been evaluated on live, unencapsulated IEC 61850 GOOSE or SV multicast streams, which operate at Layer 2 with sub-4 ms deterministic delivery constraints;
2. **Real-Time Operation on Embedded Substation Gateways:** The model has not been benchmarked on physical, substation-hardened embedded microcontrollers or industrial gateway hardware (e.g., DIN-rail IEDs or ARM Cortex-A industrial computers);
3. **Measured End-to-End Packet-to-Alert Latency:** The reported 0.0077 ms inference metric captures isolated neural tensor forward propagation; total end-to-end packet-to-alert latency—encompassing raw packet capture, OS interrupt handling, stateful flow synthesis, feature standardisation, model inference, and alert delivery into SCADA/SIEM—has not been physically measured;
4. **Safe Automated Closed-Loop Mitigation:** Automated SDN quarantine and secured network slicing remain conceptual control integrations that have not been validated in a live operational grid or HIL digital real-time simulator (e.g., RTDS or OPAL-RT) to ensure that automated mitigations do not induce protective trip anomalies or destabilise physical power system dynamics.

These constitute future validation objectives, not current experimental findings. Nevertheless, the manuscript remains directly relevant to industrial networking without such a physical testbed. Modern industrial IoT, SCADA, and smart grid infrastructures increasingly rely on standard IP/Ethernet communication for telemetry aggregation, edge analytics, and remote monitoring. Demonstrating that a dual-branch spatio-temporal architecture (operating via spatial feature-axis convolutions and intra-flow segment sequencing) can effectively handle acute class imbalance, resolve complex multi-device telemetry noise, operate within constrained memory budgets, and provide auditable visual explanations provides foundational methodology for next-generation cyber-physical security. The key is to maintain proportionate claims and avoid conflating a proposed deployment architecture with demonstrated operational readiness.

10.2 Comprehensive Study Limitations and Methodological Boundaries

A transparent evaluation of model limitations and methodological boundaries is critical to ensure that scientific claims remain proportionate to the empirical evidence. We articulate six consequential limitations of this study:

1. Dataset Representativeness versus Live Industrial Control Traffic

The empirical evaluations in this work rely upon contemporary public network-flow benchmarks (ToN-IoT, InSDN, and CIC-IIoT-2025), which synthesise flow records using Zeek and CICFlowMeter within emulated and testbed environments. While these benchmarks provide reproducible, standardised baselines capturing diverse IoT protocols (MQTT, HTTP, Modbus/TCP, DNS) and real physical devices, benchmark flow summaries are not an absolute substitute for live industrial control traffic. Live power utility substations exhibit proprietary Layer-2 multicast protocols (IEC

61850 GOOSE, SV), strict microsecond timing tolerances, physical electromagnetic interference, severe environmental temperature fluctuations, and unique operational disturbance profiles that cannot be fully replicated in synthetic or laboratory datasets.

2. Feature-Segment Sequencing versus Physical Time-Series Tracking

In sequence-based intrusion detection, constructing temporal observation windows requires strict alignment with session and host continuity. In CP-XNet, sequence modelling is conducted strictly intra-flow: each individual flow feature vector is partitioned into $T = 8$ contiguous functional segments to model attribute interactions across protocol blocks without presuming inter-flow connectivity or assuming that consecutive dataset rows belong to the same session. As formally delineated in Section 3, the recurrent dimension of the BiGRU is the feature-segment index along the ordered protocol stack, rather than physical time or successive packet arrivals. It captures directional dependencies and macro-level feature interactions within an isolated flow record; it does not observe temporal attack progression across consecutive flows or multi-packet temporal series. When sliding lookback windows are applied across successive records in public intrusion datasets, there is an inherent risk of data leakage or learning artificial row-adjacency biases. While intra-flow segmentation completely avoids these inter-flow pitfalls and exhibits robustness under chronological holdouts (holding out the final 15% temporal partition in ToN-IoT), continuous stateful multi-flow session tracking and bidirectional flow stitching over extended physical operational time horizons remain open challenges for future investigation.

3. Rare-Class Uncertainty and Metric Instability

Real-world industrial networks exhibit severe class imbalance, resulting in extreme sample scarcity for specific attack vectors. In our multi-class benchmarks, certain categories contain exceptionally few test instances—most acutely InSDN U2R ($n = 2$ test samples) and ToN-IoT MITM ($n = 157$ test samples, representing just 0.23% of the multi-class test split). Although CP-XNet achieves high empirical recall through inverse-frequency class-weighted regularisation (w_k) and validation-guided decision threshold tuning, performance metrics on such scarce sample sizes carry inherent statistical uncertainty, as reflected in wide binomial and Wilson score confidence intervals. A single classification error on a class with $n = 2$ instances shifts precision or recall by 50 percentage points. Consequently, high scores on acute minority classes must be interpreted as indicative of minority sensitivity under regularised learning rather than definitive proof of statistical stability across arbitrary real-world distributions.

4. Generalisation Boundaries: Multi-Benchmark Robustness versus Zero-Shot Transfer

While our multi-benchmark protocol demonstrates architectural adaptability across disparate protocol dimensionalities ($D \in \{40, 71, 79\}$), independent evaluation across discrete benchmarks does not imply uncalibrated zero-shot transferability. Because feature schemas, extraction toolchains, and sensor profiles differ fundamentally across plant installations, deploying CP-XNet to a new industrial environment requires native

feature alignment or local fine-tuning. Furthermore, real-world operational environments experience gradual concept drift as firmware configurations and operational baselines evolve, necessitating automated telemetry monitoring and periodic model recalibration.

5. Interpretability: Decision Support versus Automated Root-Cause Mitigation

As demonstrated by the MoRF and LeRF perturbation analyses in Section 9.4, post-hoc attributions reflect the model's learned diagnostic reliance within an observed training distribution. However, because network flow counters naturally co-vary during an intrusion campaign, high diagnostic attribution does not establish physical root-cause causality. Consequently, post-hoc XAI outputs must be treated strictly as operator decision-support signals rather than autonomous triggers for irreversible physical control interventions. Plant engineers must interpret highlighted flow features alongside physical substation instrumentation and SCADA telemetry before executing disruptive mitigation commands.

6. Deployment: Engineering Roadmap from Workstation Profiling to Embedded Silicons

As explicitly delineated in Section 10.1, our workstation benchmarks capture isolated neural tensor forward propagation; they do not establish end-to-end line-rate performance on physical, DIN-rail-mounted industrial gateways or real-time process buses. Translating the conceptual IEC 61850 architecture into operational reality requires empirical validation across the complete ingress-to-alert pipeline—encompassing network interface packet capture, hardware interrupt handling, stateful flow synthesis, and SCADA alert dispatch. Furthermore, physical power-envelope characterisation on industrial accelerators (e.g., NVIDIA Jetson or ARM Cortex modules) and hardware-in-the-loop (HIL) closed-loop stability testing remain indispensable engineering prerequisites before automated mitigations can be deployed in production power utilities.

11 Conclusion & Future Research Directions

In this study, we introduced **CP-XNet**, a dual-branch deep learning framework coupling feature-axis 1D convolutions with structured feature-segment BiGRU and compact two-token MHSA dynamic fusion, enhanced with an XAI framework for intrusion detection in industrial IoT and critical cyber-physical systems. By combining spatial cross-feature modelling across contiguous protocol fields with macro-level sequence modelling across functional feature blocks, CP-XNet simultaneously captures localised correlations and inter-segment protocol dependencies under an intra-flow spatio-temporal framing where recurrent sequencing denotes ordered functional feature-segment progression rather than inter-flow clock time.

Empirical evaluation across three contemporary benchmarks established the architectural robustness, adaptability, and computational efficiency of CP-XNet under both binary threat detection and multi-class threat attribution:

- **ToN-IoT Industrial Telemetry:** Binary detection achieves $99.49 \pm 0.04\%$ accuracy, $99.29 \pm 0.05\%$ macro F1, and 0.9996 ± 0.0001 ROC-AUC; multi-class attribution achieves $94.39 \pm 0.08\%$ accuracy, $88.29 \pm 0.11\%$ macro F1, $95.13 \pm 0.07\%$ weighted F1, and 0.9970 ± 0.0002 ROC-AUC across ten classes, with verified five-seed statistical stability.
- **InSDN Software-Defined Networking:** Binary detection achieves $99.97 \pm 0.01\%$ accuracy and $99.95 \pm 0.02\%$ macro F1; multi-class attribution achieves $99.84 \pm 0.02\%$ accuracy, $99.85 \pm 0.02\%$ weighted F1, and 1.0000 ± 0.0000 ROC-AUC across eight classes, isolating OpenFlow volumetric attacks from legitimate controller and switch traffic.
- **CIC-IIoT-2025 Real-Device Benchmark:** Binary detection achieves $95.75 \pm 0.06\%$ accuracy and $95.56 \pm 0.07\%$ macro F1; multi-class attribution achieves $93.65 \pm 0.09\%$ accuracy, $89.70 \pm 0.12\%$ macro F1, and $93.60 \pm 0.08\%$ weighted F1 across 34 commercial physical IoT devices with 102,851 test records.

Fine-grained analysis highlighted the severe false-positive trade-offs associated with acute minority classes such as MITM, where validation-guided decision threshold tuning significantly reduced false alarms while preserving high threat isolation. Computational profiling verified amortised forward-pass latencies of 0.0077–0.0128 ms per sample on GPU hardware (0.0710 ms on CPU at $B = 256$, alongside single-sample streaming latencies of 0.0820 ms GPU and 0.4100 ms CPU) and normalized power-envelope-based energy proxy estimates of 0.077–0.128 mJ per inference on GPU (0.710 mJ on CPU amortised; 0.820 mJ GPU, 4.100 mJ CPU streaming) assuming a nominal 10-W edge envelope. Furthermore, the dual SHAP and LIME visual interpretability pipeline delivers transparent, verifiable feature-level evidence with verified attribution stability under reference background sampling ($\rho = 0.914 \pm 0.021$). Finally, a conceptual prospective IEC 61850 digital substation deployment architecture was presented, framing physical hardware deployment, live Layer-2 process-bus validation, and closed-loop mitigation as future operational validation objectives.

Future research will focus on physical testbed evaluation of embedded industrial hardware and live IEC 61850 GOOSE/SV traffic streams, hardware-in-the-loop validation of closed-loop mitigation, federated edge learning across distributed microgrids to preserve inter-utility data privacy, and exploring self-supervised contrastive learning to enhance resilience against zero-day adversarial evasion attacks.

Appendix A List of Abbreviations

Table A1 provides a consolidated reference for all abbreviations and acronyms used throughout this manuscript.

Acknowledgments. The authors acknowledge the use of computing infrastructure and research facilities provided by the Department of Mathematics & Computer Science, Mizoram University.

Table A1: Consolidated list of abbreviations and acronyms used throughout the manuscript.

Abbr.	Definition	Abbr.	Definition
1D-CNN	One-Dimensional Convolutional Neural Network	**IP**	Internet Protocol
AI	Artificial Intelligence	**LeRF**	Least Relevant First
ANOVA	Analysis of Variance	**LIME**	Local Interpretable Model-agnostic Explanations
AP	Average Precision	**LN**	Layer Normalisation
APT	Advanced Persistent Threat	**LSTM**	Long Short-Term Memory
AUC	Area Under the Curve	**MAC**	Multiply-Accumulate
BFA	Brute-Force Authentication	**MCC**	Matthews Correlation Coefficient
BiGRU	Bidirectional Gated Recurrent Unit	**MHSA**	Multi-Head Self-Attention
BiLSTM	Bidirectional Long Short-Term Memory	**MITM**	Man-in-the-Middle
BN	Batch Normalisation	**ML**	Machine Learning
CE	Cross-Entropy	**MLP**	Multi-Layer Perceptron
CKAN	Convolutional Kolmogorov–Arnold Network	**MMS**	Manufacturing Message Specification
CNN	Convolutional Neural Network	**MoRF**	Most Relevant First
CPS	Cyber-Physical System	**MQTT**	Message Queuing Telemetry Transport
CPU	Central Processing Unit	**MSS**	Maximum Segment Size
CRediT	Contributor Roles Taxonomy	**MTU**	Maximum Transmission Unit
DDoS	Distributed Denial of Service	**OT**	Operational Technology
DER	Distributed Energy Resource	**OVS**	Open vSwitch
DL	Deep Learning	**OvR**	One-vs-Rest
DNP3	Distributed Network Protocol 3	**PMU**	Phasor Measurement Unit
DNS	Domain Name System	**PR**	Precision-Recall
DoS	Denial of Service	**R2L**	Remote-to-Local
FC	Fully Connected	**ReLU**	Rectified Linear Unit
FLOPs	Floating-Point Operations	**ROC**	Receiver Operating Characteristic
FN	False Negative	**ROC-AUC**	Receiver Operating Characteristic Area Under the Curve
FNR	False Negative Rate	**RTU**	Remote Terminal Unit
FP	False Positive	**SCADA**	Supervisory Control and Data Acquisition
FPR	False Positive Rate	**SDN**	Software-Defined Networking
GAP	Global Average Pooling	**SHAP**	SHapley Additive exPlanations
GMP	Global Max Pooling	**SOC**	Security Operations Center
GOOSE	Generic Object Oriented Substation Events (IEC 61850)	**SSL**	Secure Sockets Layer
GPU	Graphics Processing Unit	**SV**	Sampled Values (IEC 61850-9-2)
GRU	Gated Recurrent Unit	**TCP**	Transmission Control Protocol
HIL	Hardware-in-the-Loop	**TDP**	Thermal Design Power
HMI	Human-Machine Interface	**TLS**	Transport Layer Security
HTTP	Hypertext Transfer Protocol	**TN**	True Negative
ICT	Information and Communication Technology	**TP**	True Positive
IDS	Intrusion Detection System	**TPR**	True Positive Rate
IEC	International Electrotechnical Commission	**TTL**	Time-To-Live
IED	Intelligent Electronic Device	**U2R**	User-to-Root
IEEE	Institute of Electrical and Electronics Engineers	**UDP**	User Datagram Protocol
IIoT	Industrial Internet of Things	**XAI**	Explainable Artificial Intelligence
IoT	Internet of Things	**XSS**	Cross-Site Scripting

Declarations

Funding

The authors declare that no financial or commercial funding was received for the conduct of this study.

Competing interests

The authors declare that they have no competing financial or non-financial interests that could inappropriately influence the research presented in this study.

Ethics approval and consent to participate

Not applicable. This study involved computational experiments using publicly available network intrusion-detection datasets. It did not involve human participants, human-subject recruitment, clinical interventions, or the collection of personal data directly from individuals. Therefore, ethics approval and consent to participate were not required.

Data availability

The three benchmark datasets evaluated in this study are openly accessible from their respective hosting repositories:

1. **ToN-IoT:** Provided by the Cyber Range Laboratory at UNSW Canberra, available at https://research.unsw.edu.au/projects/toniot-datasets. This investigation evaluates the curated benchmark evaluation source split (`Train_Test_network.csv`, comprising 211,043 unique flow records sampled from the 461,043 raw Zeek testbed connection records, 100% retained post-preprocessing).
2. **InSDN:** Developed by University College Dublin and accessible via https://doi.org/10.1109/ACCESS.2020.3022633.
3. **CIC-IIoT-2025:** Maintained by the Canadian Institute for Cybersecurity, University of New Brunswick, available at https://www.unb.ca/cic/datasets/iiot-dataset-2025.html.

Code availability

The complete experimental source code, pre-processing scripts, trained model checkpoints, and evaluation routines supporting the findings of this study are to be made available in the public repository upon acceptance, at https://github.com/vanlalruata/Dual-Branch-Spatio-Temporal-Attention-Network-with-Explainable-AI.

Authors' contributions

Author contributions are defined according to the CRediT taxonomy:

- **Vanlalruata Hnamte (VH):** Conceptualization, Methodology, Software, Project administration, Supervision, Writing – review & editing.
- **Chhakchhuak Laldinsanga (CLd):** Software, Data curation, Investigation, Formal analysis, Validation, Writing – original draft.

- **Chhakchhuak Lalduhawma (CLh):** Data curation, Validation, Visualization, Writing – original draft.
- **Jamal Hussain (JH):** Conceptualization, Supervision, Resources, Writing – review & editing.

All authors read and approved the final manuscript.

References

[1] Gaba, S., Budhiraja, I., Kumar, V., Martha, S., Khurmi, J., Singh, A., Singh, K.K., Askar, S.S., Abouhawwash, M.: A systematic analysis of enhancing cyber security using deep learning for cyber physical systems. IEEE Access **12**, 6017–6035 (2024) https://doi.org/10.1109/ACCESS.2023.3349022

[2] Ferrag, M.A., Friha, O., Hamouda, D., Maglaras, L., Janicke, H.: Edge-IIoTset: A new comprehensive realistic cyber security dataset of IoT and IIoT applications for centralized and federated learning. IEEE Access **10**, 40281–40306 (2022) https://doi.org/10.1109/ACCESS.2022.3165809

[3] Herrero, R.: RTP transport in IoT MQTT topologies. Internet of Things and Cyber-Physical Systems **3**, 37–44 (2023) https://doi.org/10.1016/j.iotcps.2023.02.001

[4] Biswas, P., Kim, T.M., Kim, W.S.: Integration of digital twins and physical AI in cyber-physical systems. Intelligent Systems with Applications **30**, 200649 (2026) https://doi.org/10.1016/j.iswa.2026.200649

[5] Onoja, D., Hitchens, M., Shankaran, R.: DDoS threats and solutions for 5G-enabled IoT networks. In: Pal, S., Jadidi, Z., Foo, E. (eds.) Secure and Trusted Cyber Physical Systems: Recent Approaches and Future Directions. Smart Sensors, Measurement and Instrumentation, vol. 43, pp. 115–133. Springer, Cham, Switzerland (2022). https://doi.org/10.1007/978-3-031-08270-2_5

[6] Archibong, E., Asuquo, P., Stephen, B.: Review of zero-click attacks and zero-trust security model: concepts, architecture, state of the art, and future directions. Discover Networks **2**(1), 23 (2026) https://doi.org/10.1007/s44354-026-00023-4

[7] Javed, S.H., Ahmad, M.B., Asif, M., Akram, W., Mahmood, K., Das, A.K., Shetty, S.: APT adversarial defence mechanism for industrial IoT enabled cyber-physical system. IEEE Access **11**, 74000–74020 (2023) https://doi.org/10.1109/ACCESS.2023.3291599

[8] Arango-López, J., Isaza, G., Ramirez, F., Duque, N., Montes, J.: Cloud-based deep learning architecture for ddos cyber attack prediction. Expert Systems **42**(1), 13552 (2025) https://doi.org/10.1111/exsy.13552

[9] Bhattacharya, S., Nayek, A.K., Sen, M.: Employing multi-format log analysis to detect distributed denial-of-service (DDoS) attacks in software-defined networks (SDN). Discover Networks **2**(1), 20 (2026) https://doi.org/10.1007/s44354-026-00020-7

[10] Tian, Q., Miyata, S.: A DDoS attack detection method using conditional entropy based on SDN traffic. IoT **4**(2), 95–111 (2023) https://doi.org/10.3390/iot4020006

[11] Muthanna, M.S.A., Alkanhel, R., Muthanna, A., Rafiq, A., Abdullah, W.A.M.: Towards SDN-enabled, intelligent intrusion detection system for internet of things (IoT). IEEE Access **10**, 22756–22768 (2022) https://doi.org/10.1109/ACCESS.2022.3153716

[12] Jayalaxmi, P.L.S., Saha, R., Kumar, G., Conti, M., Kim, T.-H.: Machine and deep learning solutions for intrusion detection and prevention in iots: A survey. IEEE Access **10**, 121173–121192 (2022) https://doi.org/10.1109/ACCESS.2022.3220622

[13] Liao, H., Murah, M.Z., Hasan, M.K., Aman, A.H.M., Fang, J., Hu, X., Khan, A.U.R.: A survey of deep learning technologies for intrusion detection in internet of things. IEEE Access **12**, 4745–4761 (2024) https://doi.org/10.1109/ACCESS.2023.3349287

[14] Vinayakumar, R., Alazab, M., Soman, K.P., Poornachandran, P., Al-Nemrat, A., Venkatraman, S.: Deep learning approach for intelligent intrusion detection system. IEEE Access **7**, 41525–41550 (2019) https://doi.org/10.1109/ACCESS.2019.2895334

[15] Racherla, S., Sripathi, P., Faruqui, N., Alamgir Kabir, M., Whaiduzzaman, M., Aziz Shah, S.: Deep-IDS: A real-time intrusion detector for IoT nodes using deep learning. IEEE Access **12**, 63584–63597 (2024) https://doi.org/10.1109/ACCESS.2024.3396461

[16] Kohli, M., Chhabra, I.: A comprehensive survey on techniques, challenges, evaluation metrics and applications of deep learning models for anomaly detection. Discover Applied Sciences **7**(7), 731 (2025) https://doi.org/10.1007/s42452-025-07312-7

[17] Hairab, B.I., Said Elsayed, M., Jurcut, A.D., Azer, M.A.: Anomaly detection based on CNN and regularization techniques against zero-day attacks in IoT networks. IEEE Access **10**, 98427–98440 (2022) https://doi.org/10.1109/ACCESS.2022.3206367

[18] Babar, M., Tariq, M.U., Qureshi, B., Ullah, Z., Arif, F., Khan, Z.: An efficient and hybrid deep learning-driven model to enhance security and performance of healthcare internet of things. IEEE Access **13**, 22931–22945 (2025) https://doi.

org/10.1109/ACCESS.2025.3536638

[19] Cam, N.T., Trung, N.G.: An intelligent approach to improving the performance of threat detection in IoT. IEEE Access **11**, 44319–44334 (2023) https://doi.org/10.1109/ACCESS.2023.3273160

[20] Reddy, V.K., Kuda, N.R.: ST-HAE detects spatio-temporal anomalies in heterogeneous IoT networks under realistic class imbalance. Discover Networks **2**(1), 30 (2026) https://doi.org/10.1007/s44354-026-00046-x

[21] He, M., Huang, Y., Wang, X., Wei, P., Wang, X.: A lightweight and efficient IoT intrusion detection method based on feature grouping. IEEE Internet of Things Journal **11**(2), 2935–2949 (2024) https://doi.org/10.1109/JIOT.2023.3294259

[22] Sattari, F., Farooqi, A.H., Qadir, Z., Raza, B., Nazari, H., Almutiry, M.: A hybrid deep learning approach for bottleneck detection in IoT. IEEE Access **10**, 77039–77053 (2022) https://doi.org/10.1109/ACCESS.2022.3188635

[23] Ejaz, M., Hasan, M.Z., Hussain, M.Z.: Multimodal behavioral fingerprinting of IoMT devices a forensic approach using network and biometric signatures. Discover Networks **2**(1), 26 (2026) https://doi.org/10.1007/s44354-026-00042-1

[24] Bhavsar, M.H., Bekele, Y.B., Roy, K., Kelly, J.C., Limbrick, D.: FL-IDS: Federated learning-based intrusion detection system using edge devices for transportation IoT. IEEE Access **12**, 52215–52226 (2024) https://doi.org/10.1109/ACCESS.2024.3386631

[25] Ahmed, U., Lin, J.C.-W., Srivastava, G., Yun, U., Singh, A.K.: Deep active learning intrusion detection and load balancing in software-defined vehicular networks. IEEE Transactions on Intelligent Transportation Systems **24**(1), 953–961 (2023) https://doi.org/10.1109/TITS.2022.3166864

[26] Chithra Rani, P.R., Baalaji, K.: Deep learning-based ensemble stacking for enhanced intrusion detection in IoT-edge platforms. Discover Applied Sciences **7**(7), 687 (2025) https://doi.org/10.1007/s42452-025-06871-z

[27] Khan, I.A., Moustafa, N., Pi, D., Sallam, K.M., Zomaya, A.Y., Li, B.: A new explainable deep learning framework for cyber threat discovery in industrial IoT networks. IEEE Internet of Things Journal **9**(13), 11604–11613 (2022) https://doi.org/10.1109/JIOT.2021.3130156

[28] Al Rawajbeh, M., Maria Soosai, A.J., Ramasamy, L.K., Khan, F.: Trustworthy adaptive AI for real-time intrusion detection in industrial IoT security. IoT **6**(3) (2025) https://doi.org/10.3390/iot6030053

[29] Alsharaiah, M.A., Al-Dmour, A., Al-Hazaimeh, O., Al-Qerem, A.: An explainable AI-driven transformer model for spoofing attack detection in Internet of Medical

Things (IoMT) networks. Discover Applied Sciences **7**(5), 707 (2025) https://doi.org/10.1007/s42452-025-07071-5

[30] Elwhishi, A., Younis, A.A., Akhunzada, A.: Attention-enhanced hybrid architecture for efficient intrusion detection in industrial iot. IEEE Open Journal of the Communications Society **7**, 1330–1347 (2026) https://doi.org/10.1109/OJCOMS.2026.3661768

[31] Ullah, S., Boulila, W., Koubâa, A., Ahmad, J.: Magru-ids: A multi-head attention-based gated recurrent unit for intrusion detection in iiot networks. IEEE Access **11**, 114590–114601 (2023) https://doi.org/10.1109/ACCESS.2023.3324657

[32] Hnamte, V.: Cross-dataset zero-shot IoT attack detection using a hybrid CNN–BiLSTM + attention framework with explainable AI. Measurement: Digitalization **8**, 100060 (2026) https://doi.org/10.1016/j.meadig.2026.100060

[33] Elsaid, S.A., Binbusayyis, A.: An optimized isolation forest-based intrusion detection system for heterogeneous and streaming data in the industrial Internet of Things (IIoT) networks. Discover Applied Sciences **6**(9), 616 (2024) https://doi.org/10.1007/s42452-024-06165-w

[34] Khedr, W.I., Gouda, A.E., Mohamed, E.R.: FMDADM: A multi-layer DDoS attack detection and mitigation framework using machine learning for stateful SDN-based IoT networks. IEEE Access **11**, 28934–28954 (2023) https://doi.org/10.1109/ACCESS.2023.3260256

[35] Kumavat, K.S., Gomes, J.: Multi-layer DDoS attacks detection with mitigation in IoT-enabled sensor network. Cybersecurity **8**(1), 72 (2025) https://doi.org/10.1186/s42400-025-00378-1

[36] Hill, W., Zhao, L., Chen, X., Liu, Y.: DDoS in SDN: a review of open datasets, attack vectors and mitigation strategies. Discover Applied Sciences **6**(9), 617 (2024) https://doi.org/10.1007/s42452-024-06172-x

[37] Simwinga, B., Zimba, A., Nyirenda, M.: A systematic review of intelligent wireless communication and sensing for sustainable 5G and 6G enabled intelligent transportation systems. Discover Networks **2**(1), 26 (2026) https://doi.org/10.1007/s44354-026-00043-0

[38] Pradhan, G., Dawadi, B.R.: Evaluation of integrated framework for terrestrial-LEO 5G networks with new radio dual connectivity. Discover Networks **2**(1), 24 (2026) https://doi.org/10.1007/s44354-026-00039-w

[39] Lu, Y., Yang, Q., Kuang, Y.: Deep learning-based adaptive network intrusion detection system (DL-ANIDS) for 5G mobile network security. International Journal of High Speed Electronics and Systems, 2540207 (2025) https://doi.org/10.1142/S0129156425402074

[40] Hnamte, V., Hussain, J., Lalmuanawma, S., Lalrinawma, C., Hmar, L.: Enhancing secured network slicer for cloud security systems. Indian Journal of Science and Technology **16**(SP1), 179–191 (2023) https://doi.org/10.17485/IJST/v16sp1.msc25

[41] Chukwudi, O.F., Nneoma, U.C., Paul-Chima, U.O., Ben, O.M., Nnenna, U.J., Ifeanyi, A.K.: Hybrid artificial intelligence-driven eco-energy management for carbon-neutral 6G wireless networks. Discover Networks **2**(1), 25 (2026) https://doi.org/10.1007/s44354-026-00041-2

[42] Berguiga, A., Harchay, A., Massaoudi, A.: HIDS-IoMT: A deep learning-based intelligent intrusion detection system for the internet of medical things. IEEE Access **13**, 32863–32882 (2025) https://doi.org/10.1109/ACCESS.2025.3543127

[43] Berguiga, A., Harchay, A., Massaoudi, A.: HIDS-RPL: A hybrid deep learning-based intrusion detection system for RPL in internet of medical things network. IEEE Access **13**, 38404–38429 (2025) https://doi.org/10.1109/ACCESS.2025.3545918

[44] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: Proceedings of the 31st International Conference on Neural Information Processing Systems. NIPS'17, pp. 6000–6010. Curran Associates Inc., Red Hook, NY, USA (2017)

[45] Lundberg, S.M., Lee, S.-I.: A unified approach to interpreting model predictions. In: Proceedings of the 31st International Conference on Neural Information Processing Systems. NIPS'17, pp. 4768–4777. Curran Associates Inc., Red Hook, NY, USA (2017)

[46] Ribeiro, M.T., Singh, S., Guestrin, C.: "why should I trust you?": Explaining the predictions of any classifier. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. KDD '16, pp. 1135–1144. Association for Computing Machinery, New York, NY, USA (2016). https://doi.org/10.1145/2939672.2939778

[47] Moustafa, N.: A new distributed architecture for evaluating AI-based security systems at the edge: Network TON_IoT datasets. Sustainable Cities and Society **72**, 102994 (2021) https://doi.org/10.1016/j.scs.2021.102994

[48] Elsayed, M.S., Le-Khac, N.-A., Jurcut, A.D.: InSDN: A novel dataset for intrusion detection systems in software defined networks. IEEE Access **8**, 165263–165277 (2020) https://doi.org/10.1109/ACCESS.2020.3022633

[49] Firouzi, A., Dadkhah, S., Maret, S.A., Ghorbani, A.A.: Datasense: A real-time sensor-based benchmark dataset for attack analysis in iiot with multi-objective feature selection. Electronics **14**(20), 4095 (2025) https://doi.org/10.3390/electronics14204095

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- CPXNetSupplementary.pdf