SvedbergOpen
DISSEMINATION OF KNOWLEDGE

International Journal of Artificial Intelligence and Machine Learning

Publisher's Home Page: https://www.svedbergopen.com/

Research Paper Open Access

Adaptive Gated Multi-View Representation Learning for Zero-Day Intrusion Detection Under a Leakage-Controlled Evaluation Protocol

Rohit Hirvey[1], Dr. Hemant Kumar Gupta[2]

[1]Department of Computer Science and Engineering, SAGE University, Indore, India. E-mail: hirvey.rohit@gmail.com
[2]Department of Computer Science and Engineering, SAGE University, Indore, India. E-mail: hemugupta3131@gmail.com

Abstract
Zero-day intrusions defeat signature-based defences by construction, yet the learned detectors proposed to replace them are frequently reported under evaluation protocols that inflate their apparent capability, so that architectural progress cannot be separated from protocol optimism. This article addresses both. An adaptive wavelet-projection autoencoder feeds a tri-branch detection network whose multi-head attentive, convolutional-recurrent and residual bidirectional branches encode complementary views of each traffic record and are combined by an input-conditioned gate that re-derives a convex mixture for every record instead of fixing it at design time. The architecture is embedded in a leakage-controlled protocol in which partitioning precedes every fitted transform and all selection criteria, scaling statistics and calibration parameters are estimated on the training partition alone. Eighteen classical, ensemble, convolutional, recurrent, hybrid and reinforcement-learning detectors are evaluated under identical conditions on a three-class ransomware corpus and a binary logistics zero-day corpus, along seven metric families spanning discrimination, calibration, robustness, latency and footprint. On the ransomware corpus the pool separates across a twenty-point accuracy range: a heterogeneous stacking ensemble leads at 0.9862 accuracy, gradient boosting attains 0.9993 area under the ROC curve, and the proposed architecture reaches 0.9565. A four-variant ablation isolates the projection stage as the only positively contributing component, and a per-record confidence analysis shows that a confidence-gated pipeline automates 89.84% of traffic with no observed error. On the binary corpus fourteen detectors attain unit accuracy under a demonstrably leakage-free pipeline, identifying the corpus rather than the models as the limiting factor.

Keywords Adaptive gated fusion, deep reinforcement learning, intrusion detection systems, model calibration, multi-head attention, network security, reproducible benchmarking, selective prediction, wavelet denoising, zero-day attack detection.

1. Introduction

A zero-day exploit is, by construction, an attack for which no signature, patch or prior labelled example exists. Signature-based detection therefore fails against it in principle rather than by accident, and the defensive burden shifts onto models that generalise from known threat behaviour to unknown behaviour [2], [4], [5]. This has driven a decade of work on learned detectors spanning feature-engineered classical pipelines [6], [7], ensemble and hyperparameter-optimised learners [9], [10], convolutional and recurrent networks [11]-[13], [17], outlier and semi-supervised formulations that model normality rather than attack [2], [4], [23], and reinforcement-learning agents that treat detection as a sequential decision problem [3], [8], [19], [25].

Progress is nonetheless harder to measure than the reported numbers suggest, for three reasons. The first is evaluative: accuracies above 0.99 are routine on public corpora, yet the protocols that produce them frequently estimate scaling statistics, selection criteria or resampling weights on the complete corpus before partitioning, so that test information reaches the model indirectly; and even without leakage, random partitioning overstates zero-day capability, because an attack family present in training is not an unknown threat at test time. The second is representational: flow records are heterogeneous, heavy-tailed and partly redundant, so a single inductive bias captures only part of their structure, and hybrids that concatenate encoders fix a weighting of views that cannot vary with the record. The third is operational: a detector runs at a threshold chosen by a security team, with an analyst absorbing whatever it refers, so accuracy without a calibration or referral analysis does not describe behaviour in service.

The adaptive WavePCA-Autoencoder with adaptive hybrid exploit detection network (AWPA-AHEDNet) of Mohamed et al. [1], the base architecture and primary comparison point of this work, addresses the representational problem by combining wavelet-domain denoising, principal-component projection and autoencoder compression ahead of a hybrid detection network. This paper retains that pipeline and extends it in two directions. Architecturally, the hybrid detector becomes a tri-branch encoder whose multi-head attentive, convolutional-recurrent and residual bidirectional branches are fused by a gate computed from the input itself, so that the mixture of inductive biases is a per-record decision.

Methodologically, the architecture is placed inside a protocol designed so that its measured advantage, or the absence of one, can be believed.

1.1 Research Gaps

G1. Protocol opacity. Zero-day results are seldom accompanied by a statement of where each statistic was estimated, so a reader cannot distinguish a genuine advance from an optimistic protocol.
G2. Fixed fusion in hybrid detectors. Hybrid architectures combine complementary encoders with concatenation or fixed weights, although the informative view differs between a volumetric flood, a low-and-slow signature variant and an anomalous flow with no matching signature.
G3. Unverified component contribution. Composite pipelines are proposed as wholes, and where ablations are reported they rarely include the variants that would falsify the design; negative component results are largely absent from the literature.
G4. Missing operational characterisation. Calibration quality, referral behaviour, inference latency and model footprint are reported inconsistently, although they determine deployability at least as strongly as accuracy.

1.2 Contributions

C1. An extension of the AWPA-AHEDNet architecture [1] in which the detection network becomes a tri-branch encoder with input-conditioned adaptive gated fusion, specified formally in Section III.
C2. A leakage-controlled protocol in which the partition precedes every fitted transform, all selection criteria are estimated on the training partition, and columns capable of encoding the label are removed before modelling.
C3. A reproducible benchmark of eighteen classical, ensemble, convolutional, recurrent, hybrid and reinforcement-learning detectors, including DQ-IDS [3] and the base architecture [1], evaluated under identical conditions on two corpora and characterised along seven metric families.
C4. A four-variant component ablation whose outcome is partly negative, reported together with the two architectural mechanisms that explain it, rather than omitted.
C5. An explicit leave-one-attack-out zero-day protocol, and a per-record confidence analysis establishing that a confidence-gated pipeline automates 89.84% of traffic with no observed error and a bounded referral queue.
C6. A demonstration that one widely available corpus is saturated under a demonstrably leakage-free pipeline, with the diagnostic evidence that distinguishes a corpus property from a pipeline defect.

1.3 Organisation

Section 2 reviews the related literature. Section 3 specifies the proposed framework, its four stages and the training objective. Section 4 reports the corpora, the protocol and the complete results. Section 5 discusses the findings and the threats to their validity, and Section 6 concludes.

2. Related Work

2.1 Classical and Ensemble Learning for Intrusion Detection

Feature-engineered classical pipelines remain competitive on tabular flow data and define the practical baseline. Musleh et al. [6] combine explicit feature extraction with conventional learners for IoT intrusion detection, and Turk [7] compares such learners across UNSW-NB15 and NSL-KDD, establishing that preprocessing and feature choice frequently dominate model choice. Ensemble formulations extend this line: Sarkar et al. [9] couple ensemble learning with hyperparameter optimisation, Venkatesan [10] makes feature selection the design variable, and Singh and Srivastav [14] apply the same reasoning to CIC-IDS2017. Surveys of host-based techniques [15] and applications to web traffic [16], software-defined networks [21] and malicious-connection detection [22] confirm the breadth of the approach and its central weakness: a learner fitted to catalogued attack behaviour has no principled mechanism for behaviour it has never seen. Eight such learners are retained here as baselines.

2.2 Deep and Hybrid Architectures

Deep architectures replace manual feature engineering with learned representations. Chaganti et al. [11] apply deep models to SDN-enabled IoT intrusion detection, Hnamte and Hussain [12] combine convolutional and bidirectional recurrent layers in the DCNNBiLSTM hybrid, and Abdelkhalek and Mashaly [13] address the class imbalance that dominates intrusion corpora. Alami and Rawat [17] apply convolutional and long short-term memory models to zero-day detection in networked autonomous systems, Kumar et al. [21] and Guo [20] report related convolutional formulations, and Alrawashdeh and Goldsmith [18] show that such detectors are themselves attackable. This group establishes the premise on which the present architecture rests, that convolutional, recurrent and attentional biases capture different

structure, but combines those biases with a weighting fixed at design time, which is the limitation the adaptive gate of Section 3.5 addresses.

2.3 Zero-Day-Specific Formulations

A distinct line of work models normality instead of attack, so that novelty can be detected without a labelled example of it. Hindy et al. [2], [5] develop outlier-based and autoencoder-based deep learning for zero-day detection, and Mbona and Eloff [4] adopt a semi-supervised formulation. Macko et al. [23] quantify the effect of a supervised filter placed ahead of flow-based hybrid anomaly detection, and SakthiMurugan et al. [24] assess zero-day vulnerability with conventional learners. Anand and Zehra [40] contrast isolation forests with autoencoders, Santhadevi et al. [45] apply isolation forests to evil-twin-assisted traffic, the reviews of P. N. G et al. [42] and Al Siam et al. [46] map the design space, Alansary et al. [44] combine oversampling, LASSO selection and Optuna-based optimisation, and Singh and Kumari [43] pair adaptive deep learning with explanation. What this literature establishes methodologically is the protocol adopted in Section 4.6: unless an entire attack family is withheld from training, a reported zero-day result describes interpolation rather than novelty detection.

2.4 Reinforcement Learning for Adaptive Detection

Reinforcement learning reframes detection as a decision problem in which the detector is rewarded for correct classification and can, in principle, adapt after deployment. Hossain [3] proposes DQ-IDS, a deep Q-learning intrusion detection system with experience replay and a target network, reimplemented here as a comparator and contributing the lowest inference cost of the pool. Mohamed and Ejbali [8] develop a deep SARSA formulation, Malik and Saini [19] survey the area, Alkasassbeh et al. [25] propose a self-adaptive deep Q-network for zero-day attacks, and Hussain et al. [47] combine deep reinforcement learning with conventional machine learning. The consistent finding, reproduced in Section 4.3, is that the value-based formulation trades decision accuracy for adaptivity and a very small computational footprint.

2.5 Representation Learning and the Base Architecture

The architecture extended here belongs to a family that treats representation quality, rather than classifier capacity, as the binding constraint. Mohamed et al. [1] propose the adaptive WavePCA-Autoencoder (AWPA) with an adaptive hybrid exploit detection network (AHEDNet), in which wavelet-domain shrinkage suppresses measurement noise, principal-component projection decorrelates the surviving attributes, and an autoencoder compresses them into a compact code consumed by a hybrid detector. Two aspects are retained without modification, the wavelet-projection-autoencoder cascade and the use of a hybrid rather than a monolithic detector, and one is changed: the fixed hybrid combination is replaced by the input-conditioned gate of Section 3.5. Section 4.5 reports what that change is worth, including where it is worth nothing, and Section 3.8 identifies the implementation properties that account for the result.

2.6 Emerging Directions

Three newer directions bound the scope of this work. Language-model-based reasoning is being applied to log analysis and threat response, in memory-augmented edge agents with SHAP explanation [26], log-reasoning detectors for IoT [27], and digital-twin moving-target defence [30]. Structural and graph formulations address threats invisible at the level of an individual flow, including contrastive learning on temporal graphs [28], DNS-tunnel detection [29], lateral-movement mitigation in industrial IoT [38] and genetic-evolution discovery of unknown threats [35]. A third group addresses deployment constraints: multi-tier real-time frameworks for UAV control links [31], neuro-symbolic and verifiable reasoning [32], [39], botnet detection blended with cyber-physical controls [33], synthetic-data and explainability frameworks [34], quantum-classical encoding comparisons [36], disentangled attention for phishing [37], and privacy-preserving federated threat intelligence [41]. These are complementary to the present study, which addresses the flow-level tabular setting; Table 1 positions representative studies against the gaps of Section 1.1.

Table 1. Representative zero-day and intrusion-detection studies positioned against the gaps of Section 1.1.

Study	Approach	Addressed	Limitation relevant to this work
Mohamed et al. [1]	AWPA representation + hybrid exploit detection network	G2 (partly)	fixed hybrid combination; component contributions not separated
Hindy et al. [2], [5]	outlier-based and autoencoder deep learning	G1 (partly)	novelty modelled without an explicit attack-family hold-out

Study	Approach	Addressed	Limitation relevant to this work
Hossain [3]	deep Q-learning detection with replay and target network	G4 (partly)	decision accuracy traded for adaptivity; calibration not characterised
Mbona and Eloff [4]	semi-supervised zero-day detection	G1 (partly)	protocol-level leakage controls not stated
Hnamte and Hussain [12]	DCNNBiLSTM hybrid	G2 (partly)	fixed concatenation of convolutional and recurrent views
Macko et al. [23]	supervised filter ahead of hybrid anomaly detection	G1	single corpus; no cost characterisation
Alkasassbeh et al. [25]	self-adaptive deep Q-network for zero-day attacks	G4 (partly)	referral and calibration behaviour not reported
Anand and Zehra [40]; Santhadevi et al. [45]	isolation forest and autoencoder comparison	G3 (partly)	component-level attribution not attempted
This work	AWPA-MATA-AHEDNet with adaptive gated fusion	G1-G4	single seed per configuration; cross-corpus transfer not completed

In summary, the literature supplies strong classical and deep baselines, a well-motivated representation-learning cascade in the base architecture [1], and a reinforcement-learning alternative [3], but it does not generally supply the protocol-level evidence needed to compare them. The framework specified next is designed so that both the architectural extension and its limits are measurable.

3. Proposed Methodology

3.1 Overview and Design Rationale

The proposed framework addresses three failure modes that recur in the zero-day literature. The first is optimistic evaluation: preprocessing statistics or selection criteria estimated on the full corpus leak test information into training, and random partitioning permits an unseen attack family to be scored under conditions it would never meet in deployment. The second is representational: tabular flow records carry heterogeneous, heavy-tailed and partly redundant attributes, and a single convolutional, recurrent or attentional bias captures only part of the structure. The third is architectural rigidity: a fixed fusion rule weights every encoder identically for every record.

The framework answers these in four stages. Stage 1 enforces a leakage-controlled protocol in which every statistic is estimated on the training partition alone and every column capable of encoding the label is removed before modelling. Stage 2 selects a compact, stable and non-redundant feature subset through a hybrid criterion combining relevance, impurity-based importance, bootstrap stability and a correlation penalty. Stage 3, the Adaptive Wavelet-Projection Autoencoder (AWPA), suppresses high-frequency noise, decorrelates the surviving attributes and compresses them into a compact latent code. Stage 4, MATA-AHEDNet, encodes that code through three complementary branches and combines them with an input-conditioned gate. Fig 1 gives the end-to-end flow; Table 2 fixes the notation.

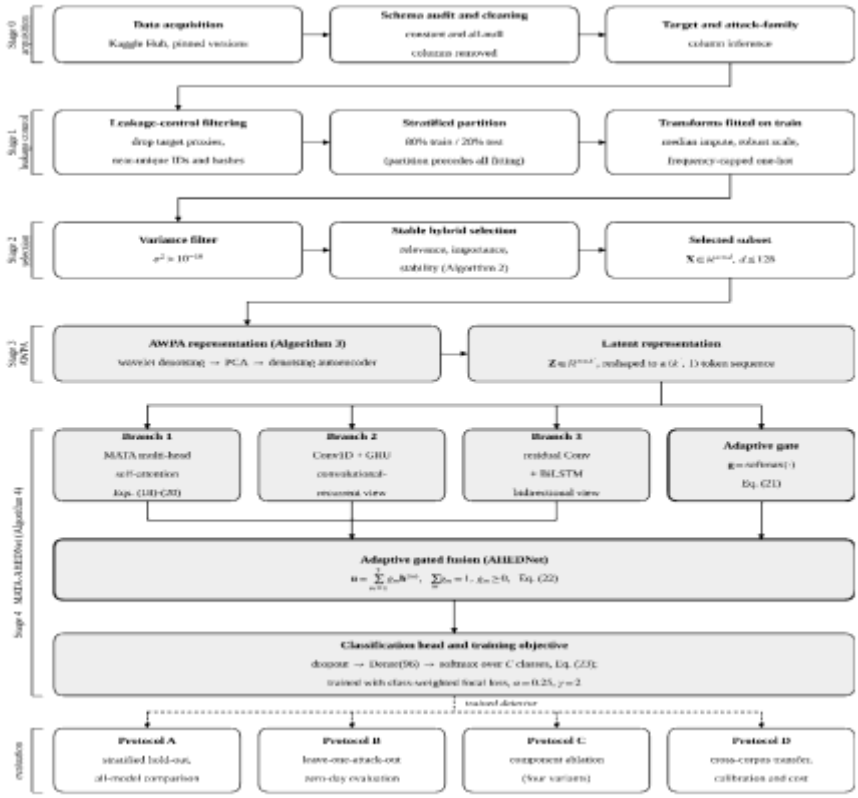

Fig 1. End-to-end architecture of the proposed zero-day detection and evaluation framework. Solid arrows denote the data path; dashed arrows denote evaluation protocols applied to the trained detector.

Table 2. Notation.

Symbol	Definition
n, d, C	number of records, selected features and classes
x ∈ R^d	a preprocessed feature vector of one flow record
x̃ ∈ R^d	wavelet-denoised feature vector
k, k'	PCA dimension and autoencoder latent dimension
z ∈ R^k'	AWPA latent code; Z is the latent matrix
a_L, d_l	level-L approximation and level-l detail coefficients
σ̂, λ	robust noise scale and universal soft threshold
h^(m)	output embedding of branch m ∈ {1,2,3}
g = (g_1,g_2,g_3)	adaptive gate weights on the 2-simplex
u ∈ R^64	fused representation
ŷ ∈ Δ^(C−1)	predicted class-probability vector
w_c, γ	class weight and focal-loss focusing parameter
Q(s,a;θ)	action-value function of the DQ-IDS agent

3.2 Stage 1: Leakage-Controlled Preprocessing

Let D = {(r_i, y_i)} denote the raw corpus. Three filters are applied before any statistic is estimated. Columns whose normalised name matches the target or contains a ground-truth, true-label or predicted-label token are removed; the attack-family column is withheld from the feature matrix and retained only for the zero-day protocol of Section 4.6; and near-unique identifier columns, those whose cardinality exceeds 98% of the record count and whose name carries an identifier, hash, address or IP token, are dropped. Constant and all-null columns are then discarded.

The corpus is partitioned by stratified sampling into 80% training and 20% test before the transformer is fitted. Numeric attributes are median-imputed with a missingness indicator and scaled robustly on the 10th and 90th percentiles of the training partition alone,

$$x_{ij}' = \frac{x_{ij} - q^{(50)}{}_j}{q^{(90)}{}_j - q^{(10)}{}_j}, j \in N \quad (1)$$

where q^(p)_j is the p-th percentile of attribute j over the training partition and N is the numeric index set. Categorical attributes are imputed with the training mode and one-hot encoded under a frequency floor and a cardinality cap,

$$\varphi(v) = e_v \; if \; freq(v) \geq \tau_f \; and \; rank(v) \leq \kappa, \quad else \; e_{other} \tag{2}$$

with τ_f = 5 and κ = 50, which prevents the encoder from creating a near-unique column per rare category. Attributes with negligible dispersion are then removed,

$$F_1 = \{ j : Var(f_j) > 10^{-10} \} \tag{3}$$

All three transformations are fitted on the training partition and applied unchanged to the test partition, so no test statistic influences the model. Because the pipeline is leakage-controlled by construction, a corpus on which every model saturates cannot be explained by a pipeline defect and must be explained by the corpus itself.

3.3 Stage 2: GMCO-Inspired Stable Hybrid Feature Selection

Wrapper-based optimisers are expensive and unstable on high-dimensional one-hot expansions. The proposed selector approximates a global multi-criterion optimiser by a reproducible closed-form score over criteria evaluated on a stratified training subsample. Relevance is measured by mutual information between attribute and label,

$$I(f_j; y) = \sum_{f,y} p(f,y) \log \frac{p(f,y)}{p(f)\,p(y)} \tag{4}$$

$$r_j = I(f_j; y) / max_l I(f_l; y) \tag{5}$$

Impurity-based importance is taken from a class-balanced Extra Trees ensemble of B_0 trees and normalised to unit maximum,

$$t_j = \frac{(1/B_0)\,\Sigma_b \,\Delta\_imp(j,b)}{max_l\,(1/B_0)\,\Sigma_b \,\Delta\_imp(l,b)} \tag{6}$$

Stability is estimated by repeated bootstrap refitting: for each of B bootstrap replicates a smaller ensemble is trained and the top-K attributes recorded, giving the selection frequency

$$s_j = \frac{1}{B}\sum_{b=1}^{B} 1[\, j \in TopK(b) \,] \tag{7}$$

The three criteria are combined by a fixed convex weighting that favours ensemble evidence while retaining an information-theoretic term and a reproducibility term,

$$c_j = 0.40\, r_j + 0.45\, t_j + 0.15\, s_j \tag{8}$$

Attributes are ranked by c_j and admitted greedily subject to a redundancy constraint, so that a candidate is rejected when it duplicates an already admitted attribute,

$$S \leftarrow S \cup \{ j \} \; iff \; |\rho(f_j, f_l)| \leq 0.97 \;\; \forall\, l \in S_{recent} \tag{9}$$

The procedure terminates when |S| reaches the budget d <= 128. Because rho is evaluated against a trailing window of admitted attributes rather than the full set, the cost is linear in the candidate count rather than quadratic in the feature count.

3.4 Stage 3: The Adaptive Wavelet-Projection Autoencoder (AWPA)

AWPA produces the compact latent code consumed by the classifier and is summarised in Fig 2. Each preprocessed vector is treated as a one-dimensional signal and decomposed by a two-level discrete wavelet transform with a Daubechies-2 basis under symmetric boundary extension,

$$(a_2, d_2, d_1) = W(x; db2, L = 2), \quad x = W^{-1}(a_2, d_2, d_1) \tag{10}$$

The noise scale is estimated robustly from the finest detail band by the median absolute deviation, and the universal threshold follows,

$$\hat{\sigma} = \frac{median\,|\,d_1 - median(d_1)\,|}{0.6745}, \quad \lambda = \hat{\sigma}\sqrt{2\,ln(d+1)} \tag{11}$$

Detail coefficients are shrunk by the soft-thresholding operator, which removes low-amplitude fluctuation while preserving the sign and continuity of genuine structure,

$$S_\lambda(c) = sgn(c) \cdot max(|c| - \lambda, 0) \tag{12}$$

and the denoised vector is recovered by the inverse transform, truncated to the original length,

$$\tilde{x} = W^{-1}(a_2, S_\lambda(d_2), S_\lambda(d_1)) \tag{13}$$

The denoised training matrix is projected onto its leading principal directions, with the projection estimated on the training partition alone,

$$V = argmax_V \, tr(V^\top \Sigma V), \hat{x} = V^\top(\tilde{x} - \tilde{\mu}), \quad k = min(64, d) \tag{14}$$

A denoising autoencoder is then trained on the projected representation. Isotropic Gaussian corruption is injected at the input so that the encoder learns a representation invariant to small perturbations rather than an identity map,

$$\check{x} = \hat{x} + \eta, \eta \sim N(0, 0.03^2 I) \quad (15)$$

$$z = f_{enc}(\check{x}) = ReLU(W_2 ReLU(W_1 \check{x} + b_1) + b_2), k' = min(32, max(8, \lfloor k/2 \rfloor)) \quad (16)$$

with the decoder trained jointly under a mean-squared reconstruction objective and discarded at inference,

$$L_{rec} = \frac{1}{n}\sum_{i=1}^{n} \| f_{dec}(f_{enc}(\check{x}_i)) - \hat{x}_i \|^2 \quad (17)$$

Only the encoder is retained, so the inference-time cost of AWPA is one wavelet transform, one matrix projection and two dense layers per record.

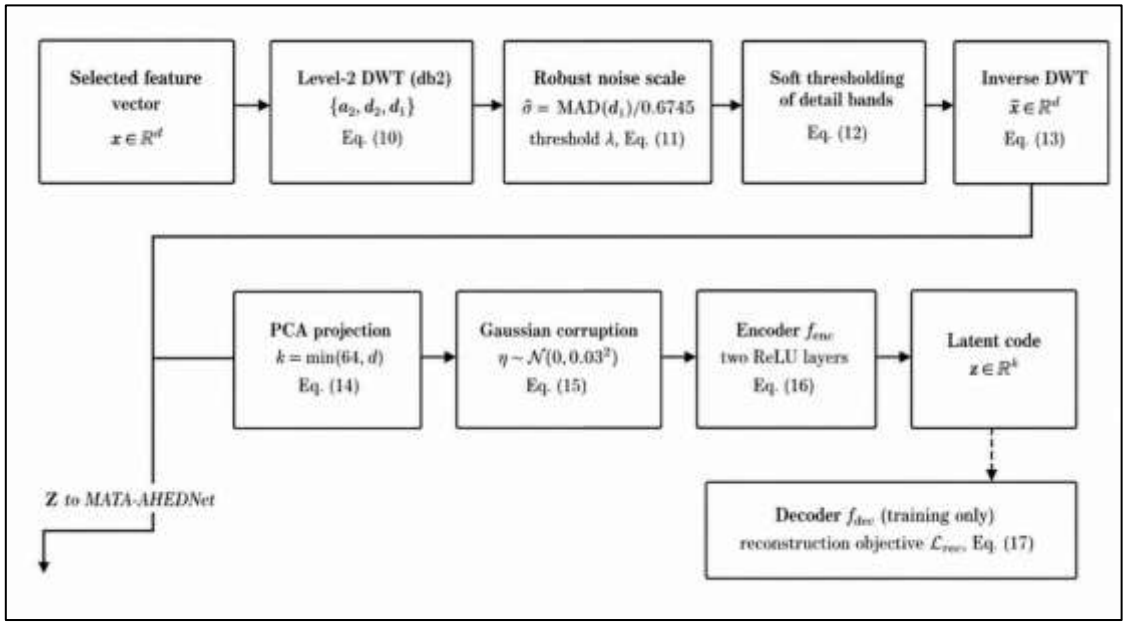

Fig 2. Internal structure of the AWPA module. The upper path performs adaptive wavelet shrinkage; the lower path performs projection and denoising-autoencoder compression. The decoder (dashed) is used during training only.

3.5 Stage 4: MATA-AHEDNet

The latent code is reshaped into a sequence of k' single-channel tokens and presented to three parallel branches with deliberately different inductive biases. Fig 3 gives the depth-level architecture with the tensor shape at every layer, and Table 3 the corresponding configuration.

3.5.1 MATA branch: multi-head attentive token encoding

Tokens are embedded into a 32-dimensional space and passed through two identical transformer blocks. Each block applies multi-head self-attention, allowing every latent coordinate to attend to every other and thereby capturing long-range dependencies that a fixed convolutional receptive field cannot,

$$Attention(Q, K, V) = softmax\left(\frac{Q K^{\top}}{\sqrt{d_k}}\right) V \quad (18)$$

$$MultiHead(T) = Concat(head_1, head_2) W^O,$$
$$head_i = Attention(T W_i^Q, T W_i^K, T W_i^V)$$

with H = 2 heads and key dimension d_k = 16. Each block closes with residual connections and layer normalisation around the attention sub-layer and a position-wise feed-forward sub-layer implemented as two pointwise convolutions with a GELU non-linearity,

$$T' = LN(T + MultiHead(T)), T'' = LN(T' + Conv_{1\times1}(GELU(Conv_{1\times1}(T')))) \quad (19)$$

3.5.2 Convolutional-recurrent and residual bidirectional branches

The second branch applies a width-3 convolution with 64 filters, halves the token axis by max pooling and summarises the result with a gated recurrent unit, favouring short local motifs followed by an ordered summary. The third applies two width-3 convolutions with an identity path carried by a pointwise projection shortcut, so that gradients reach the early layers unattenuated, and closes with a bidirectional LSTM reading the token axis in both directions,

$$h^{(1)} = GELU(W_a GAP(T'')) \quad (20)$$

$$h^{(2)} = ReLU(W_b GRU(MaxPool_2(ReLU(Conv_3(Z))))) $$

$$h^{(3)} = ReLU(W_c BiLSTM(ReLU(Conv_3(ReLU(Conv_3(Z))) \oplus Conv_1(Z))))$$

where $\oplus$ denotes elementwise addition of the residual shortcut and each branch emits a 64-dimensional embedding, so the three views are directly comparable in the fusion space.

3.5.3 Adaptive gated fusion (AHEDNet)

The fusion stage is the principal architectural contribution. Rather than concatenating or averaging the branch embeddings with fixed weights, the network computes a mixture over the 2-simplex from a context summary of the input itself and mixes the branches accordingly,

$$g = softmax(W_{g2}\, ReLU(W_{g1}\, GAP(Z) + b_{g1}) + b_{g2}) \in \Delta^2 \quad (21)$$

$$u = \sum_{m=1}^{3} g_m h^{(m)}, \quad \sum_{m=1}^{3} g_m = 1, \; g_m \geq 0 \quad (22)$$

Because g is a function of the record rather than a learned constant, the mixture is re-derived for every flow: a record whose evidence is carried by a long-range coordinate dependency can be routed towards the attention branch, and a record whose evidence is a short local motif towards the convolutional-recurrent branch. Fig 4 details the mechanism.

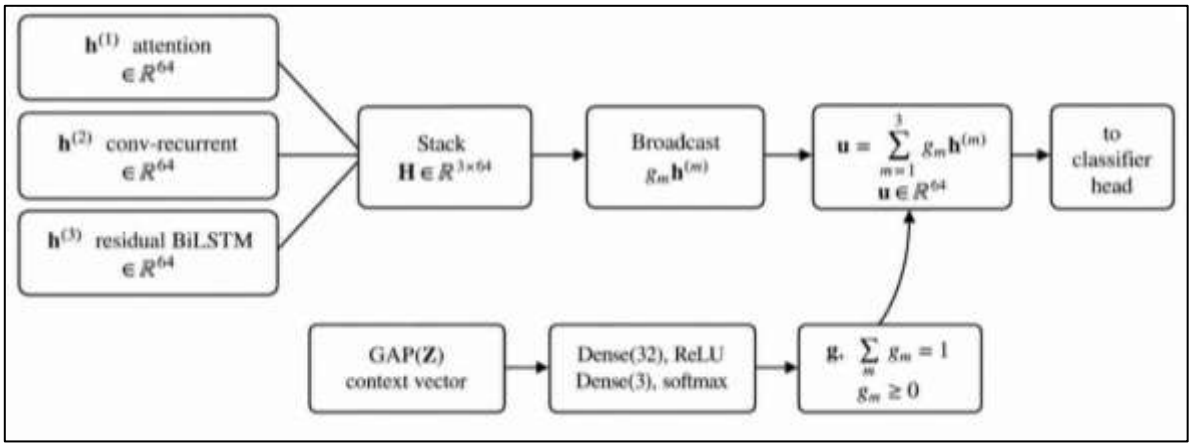

Fig 4. Adaptive gated fusion. The gate is computed from a context summary of the same input that drives the branches, producing an input-conditioned convex mixture of the three branch embeddings.

3.5.4 Classification head and objective

The fused vector passes through dropout and a 96-unit rectified layer before a softmax over the C classes. Training minimises a class-weighted focal objective, which down-weights the well-classified majority and concentrates gradient on the hard minority records that dominate the zero-day setting,

$$\hat{y} = softmax(W_o\, ReLU(W_h\, Drop(u)) + b_o) \quad (23)$$

$$L_{focal} = -\sum_{i,c} w_c\, \alpha\, (1 - \hat{y}_{i,c})^{\gamma}\, y_{i,c}\, log\, \hat{y}_{i,c}, \quad \alpha = 0.25, \gamma = 2$$

with w_c the balanced class weight computed on the training partition. Optimisation uses Adam with early stopping on validation loss and restoration of the best weights.

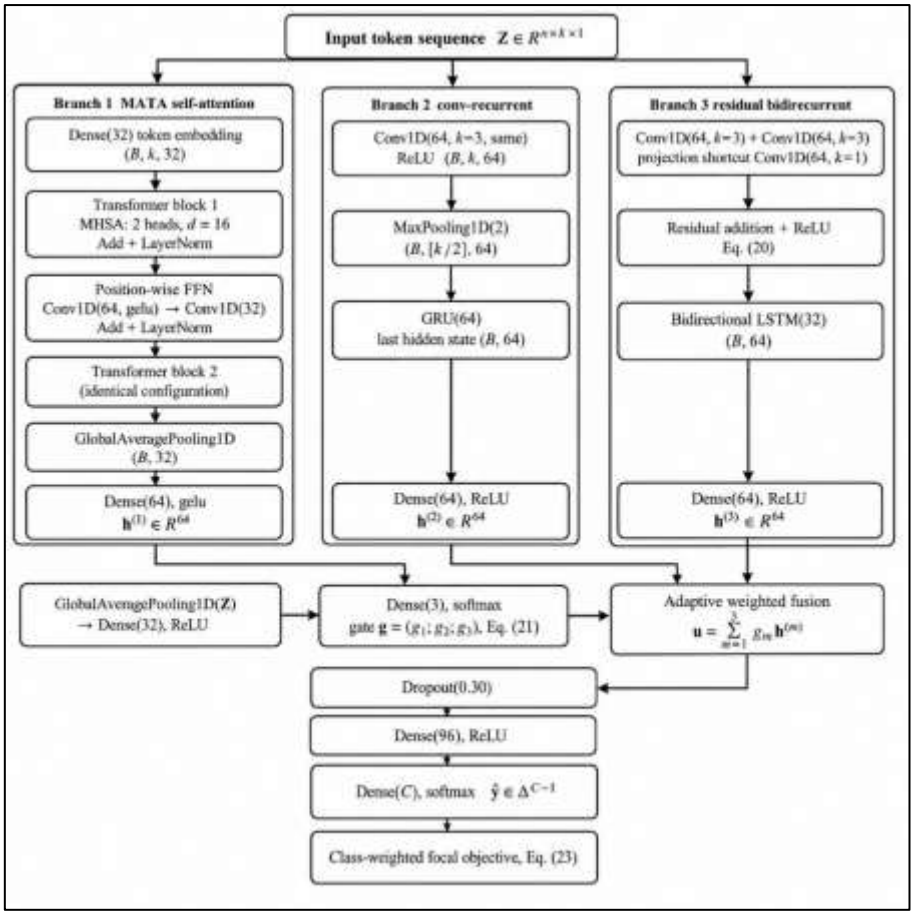

Fig 3. Depth-level internal architecture of MATA-AHEDNet. Tensor shapes are given for a batch of size B and a latent dimension k′; the gate path is driven by a context summary of the same input sequence.

Table 3. Layer-wise configuration of MATA-AHEDNet.

Stage	Layer	Configuration	Output shape
Input	Latent token sequence	reshape of AWPA code	(B, k′, 1)
Branch 1	Dense embedding	32 units, linear	(B, k′, 32)
	Transformer block ×2	MHSA: H = 2, d_k = 16, dropout 0.15	(B, k′, 32)
	Position-wise FFN	Conv1D(64, GELU) → Conv1D(32)	(B, k′, 32)
	Global average pooling	over token axis	(B, 32)
	Dense	64 units, GELU	(B, 64)
Branch 2	Conv1D	64 filters, width 3, same, ReLU	(B, k′, 64)
	Max pooling	pool size 2	(B, ⌊k′/2⌋, 64)
	GRU	64 units, last state	(B, 64)
	Dense	64 units, ReLU	(B, 64)
Branch 3	Conv1D ×2 + shortcut	64 filters width 3; Conv1D(64, width 1)	(B, k′, 64)
	Residual add + ReLU	elementwise	(B, k′, 64)
	Bidirectional LSTM	32 units per direction	(B, 64)
	Dense	64 units, ReLU	(B, 64)
Gate	Global average pooling	over token axis of the input	(B, 1)
	Dense → Dense	32 units ReLU → 3 units softmax	(B, 3)
Fusion	Stack and weight	branch stack × gate, sum over branches	(B, 64)
Head	Dropout → Dense	rate 0.30 → 96 units, ReLU	(B, 96)
	Output	C units, softmax	(B, C)

3.6 Reinforcement-Learning Comparator: DQ-IDS

For comparison against a decision-theoretic formulation, detection is also cast as a one-step Markov decision process in which the state is the latent representation of a record, the action set is the label set, and the reward distinguishes a correct from an incorrect classification,

$$s = z_i, a \in \{1, \ldots, C\},$$
$$r(s, a) = +1 \; if \; a = y_i, else \; -1 \tag{24}$$

An action-value network is trained from a replay buffer of capacity 10^4 with uniform minibatch sampling and a periodically synchronised target network, which decorrelates consecutive updates and stabilises the regression target,

$$y_i = r_i + \gamma (1 - done_i) \max_{a'} Q(s'_i, a'; \theta^-),$$
$$\gamma = 0.95 \tag{25}$$

$$L(\theta) = \frac{1}{|B|} \sum_{i \in B} Huber(Q(s_i, a_i; \theta) - y_i) \tag{26}$$

with exploration annealed geometrically and class posteriors obtained by a softmax over the action values at inference,

$$\varepsilon_{t+1} = \max(\varepsilon_{min}, 0.995 \, \varepsilon_t), \varepsilon_0 = 1,$$
$$\varepsilon_{min} = 0.01 \tag{27}$$

3.7 Computational Complexity

Let n be the number of training records, d the selected feature count, k' the latent width, E the epoch budget and H the head count. Selection costs one mutual-information pass and B + 1 ensemble fits on a subsample of size m << n; AWPA costs a linear-time wavelet transform per record and one projection. Attention is quadratic in the token count and the recurrent branches linear, giving

$$O(m \, d \log d + (B + 1) \, m \, d \log m)$$
$$+ O(n \, d \log d + n \, d \, k)$$
$$+ O(E \, n \, (H \, k'^2 \, d_k + k' \, d_h^2)) \tag{28}$$

Because k' <= 32 by construction, the quadratic attention term is bounded by roughly 10^3 operations per record per block, so the dominant training cost is recurrent rather than attentional. Inference requires one forward pass and is therefore linear in n.

3.8 Design Observations and Known Limitations

Three properties of the implemented design are stated explicitly, because they bear directly on the ablation evidence of Section 4.5.

First, the wavelet stage treats each tabular record as a one-dimensional signal along the feature axis. After one-hot expansion and selection, the ordering of that axis is arbitrary rather than temporal or spatial, so the smoothness assumption that justifies wavelet shrinkage is not guaranteed to hold. This is a plausible mechanism for the observed result that removing the wavelet front-end improves accuracy on UGRansome, and it suggests the module belongs on an explicitly ordered axis.

Second, the gate context is a global average over the token axis of a single-channel input, so g is computed from a scalar summary of the latent code. A three-way mixture conditioned on one scalar has very limited capacity to discriminate between records; conditioning the gate on the branch embeddings themselves, or on a multi-channel projection, is a direct and inexpensive improvement.

Third, the MATA branch attends over latent coordinates rather than over an ordered sequence. Self-attention is permutation-equivariant, so this is well defined, but the resulting attention map should be read as a learned coordinate-interaction matrix and not as a temporal attention pattern.

3.9 Algorithms

Algorithm 1 states the end-to-end framework, including the leakage-controlled protocol, the stable hybrid selector and the AWPA representation; Algorithm 2 states the training of MATA-AHEDNet with adaptive gated fusion.

Algorithm 1 SAGE zero-day detection and evaluation framework

Input: raw corpus D, seed s, feature budget d, test fraction ρ = 0.20
Output: trained detector M, metric set R, protocol reports
1: D ← audit(D); infer target column y and attack-family column τ
2: X ← removeLeakage(D, y, τ) ▷ target proxies, near-unique IDs, hashes, addresses
3: (I_tr, I_te) ← stratifiedSplit(D, ρ, s)
4: Φ ← fitTransform(X[I_tr]) ▷ impute, robust scale, capped one-hot — train only
5: X_tr ← Φ(X[I_tr]); X_te ← Φ(X[I_te])
6: X_tr, X_te ← varianceFilter(X_tr, X_te)
7: S ← StableHybridSelect(X_tr, y_tr, d, s) ▷ Section 3.3

8: Z_tr, Z_te ← AWPA(X_tr[:, S], X_te[:, S], s) ▷ Section 3.4
9: M ← TrainMATA-AHEDNet(Z_tr, y_tr, s) ▷ Algorithm 2
10: R ← evaluate(M, Z_te, y_te) ▷ decision, ranking, agreement, calibration, cost
11: for each attack family f ∈ τ do ▷ Protocol B: zero-day
12: train on D \ f; report detection rate and benign false-positive rate on f
13: end for
14: for each variant v ∈ {full, no-wavelet, no-attention, no-PCA} do report metrics(v)
15: attempt cross-corpus transfer over the intersecting feature schema
16: return M, R

Algorithm 2 Training MATA-AHEDNet with adaptive gated fusion

Input: latent codes Z_tr, labels y_tr, class count C, epochs E, batch size 512
Output: trained detector M
1: reshape Z_tr to token sequences of shape (k′, 1); compute balanced class weights w_c
2: T ← Dense_32(Z); for two blocks do T ← LN(T + MultiHead(T)); T ← LN(T + FFN(T)) ▷ Eq. (18)–(19)
3: $h^{(1)} \leftarrow \mathrm{GELU}(W_a \cdot \mathrm{GAP}(T))$
4: $h^{(2)} \leftarrow \mathrm{ReLU}(W_b \cdot \mathrm{GRU}(\mathrm{MaxPool\_2}(\mathrm{Conv\_3}(Z))))$ ▷ Eq. (20)
5: $h^{(3)} \leftarrow \mathrm{ReLU}(W_c \cdot \mathrm{BiLSTM}(\mathrm{ReLU}(\mathrm{Conv\_3}(\mathrm{Conv\_3}(Z)) \oplus \mathrm{Conv\_1}(Z))))$ ▷ Eq. (20)
6: $g \leftarrow \mathrm{softmax}(W_{g2} \cdot \mathrm{ReLU}(W_{g1} \cdot \mathrm{GAP}(Z)))$ ▷ Eq. (21)
7: $u \leftarrow \Sigma_{m=1..3}\, g_m\, h^{(m)}$ ▷ Eq. (22)
8: $\hat{y} \leftarrow \mathrm{softmax}(W_o \cdot \mathrm{ReLU}(W_h \cdot \mathrm{Dropout\_0.30}(u)))$
9: minimise the class-weighted focal objective with Adam ▷ Eq. (23)
10: early-stop on validation loss with patience 5; restore best weights
11: return M

4. Experimental Results

4.1 Experimental Protocol, Datasets and Evaluation Measures

The proposed detector and the full comparison pool were evaluated on two zero-day corpora of contrasting difficulty. LogisticsZeroDay is a binary benchmark of logistics-network telemetry with the classes Attack Detected and No Attack; its held-out partition contains 16,000 flows, 6,061 attacks (37.88%) and 9,939 benign (62.12%). UGRansome is a three-class ransomware and anomaly corpus labelled A (anomaly), S (signature) and SS (synthetic signature); its held-out partition also contains 16,000 records, 4,313 A, 7,324 S and 4,363 SS. Both partitions were produced by stratified splitting under a fixed seed, and every model in a pool observed the same pipeline, split and seed, so differences are attributable to the learning algorithm alone. Table 4 summarises the benchmarks.

Table 4. Summary of the two evaluation benchmarks.

Property	LogisticsZeroDay	UGRansome
Task	Binary	Three-class
Class labels	Attack Detected / No Attack	A / S / SS
Test partition size	16,000	16,000
Class distribution	6,061 / 9,939	4,313 / 7,324 / 4,363
Majority-class prevalence	0.6212	0.4578
Models evaluated	17	18
Ablation variants	2	4
Leave-one-attack-out folds	4	not executed

4.1.1 The UGRansome corpus

UGRansome was introduced by Nkongolo et al. in 2021 at the University of Pretoria and is distributed through Kaggle under the handle nkongolo/ugransome-dataset, version 2 of which is used here. It was built by combining the informative attributes of the UGR'16 network-flow capture and a ransomware corpus, with the explicit aim of supporting detection of unknown attacks, and records cyclostationary patterns of normal and abnormal behaviour through attributes such as transferred bytes, protocol flags, duration, port and cryptocurrency fields.

Its defining property here is the three-way target, which separates threats matching a known signature from those matching a synthesised or mutated variant and from behaviour matching nothing catalogued. The corpus carries ransomware families including Locky, CryptoLocker, WannaCry, JigSaw, EDA2, TowerWeb, Flyper, Razy, Globe and SamSam, together with advanced persistent threats, and background categories such as UDP, SSH and port scanning alongside blacklist, botnet, DoS, nerisBotnet and spam

traffic. Because the anomaly class is populated by behaviour without a matching signature, A is the natural proxy for zero-day activity, which is why the A-versus-SS confusion of Section 4.3 is the operationally significant one. Table 5 lists the documented schema.

Table 5. Documented attribute schema of the UGRansome corpus.

Attribute	Type	Description
Time	Numeric	Timestamp of the observed network activity
Protocol	Categorical	Transport protocol of the flow (TCP, UDP, ICMP)
Flag	Categorical	Flow status flag
Family	Categorical	Ransomware family or threat variant
Clusters	Numeric	Behavioural cluster identifier
SeedAddress	Categorical	Seed cryptocurrency address
ExpAddress	Categorical	Expanded cryptocurrency address
BTC	Numeric	Bitcoin amount associated with the transaction
USD	Numeric	Ransom value in US dollars
Netflow_Bytes	Numeric	Bytes transferred in the flow
IPaddress	Categorical	Source or destination address indicator
Threats	Categorical	Threat or attack category
Port	Numeric	Network port
Prediction	Categorical	Target label: A (anomaly), S (signature), SS (synthetic signature)

4.1.2 The LogisticsZeroDay corpus

The second corpus is distributed through Kaggle under the handle datasetengineer/zero-day-attack-detection-in-logistics-networks, version 1, and is described there only as addressing zero-day attack detection in logistics networks. It presents a binary target, and the attack-family field exposed by the leave-one-attack-out protocol identifies at least four families: SQL Injection, Phishing, WannaCry and DDoS. Table 6 records the observable characteristics.

A caveat belongs with the description rather than with the limitations. The dataset page carries no methodological documentation: it does not state the capture environment, the collection period, how the flows were derived or labelled, or whether the traffic is real, emulated or wholly synthetic. That gap matters here because this is the corpus on which fourteen of seventeen models and all four leave-one-attack-out folds return perfect scores. Its provenance should be established before submission, and any synthetic origin disclosed with the headline claims weighted accordingly.

Table 6. Observable characteristics of the LogisticsZeroDay corpus.

Property	Value
Source handle	datasetengineer/zero-day-attack-detection-in-logistics-networks (v1)
Task	Binary classification
Target classes	Attack Detected (0), No Attack (1)
Test partition	16,000 flows (6,061 attack / 9,939 benign)
Attack families exposed	SQL Injection, Phishing, WannaCry, DDoS
Records available to the LOAO protocol	30,304 per fold (train plus held-out family)
Feature overlap with UGRansome	7 attributes
Documented provenance	Not stated on the dataset page; to be established

4.1.3 Acquisition and reproducibility

Both corpora were retrieved programmatically through the Kaggle Hub interface using pinned version identifiers, so the experiments can be reproduced against the identical release; the harness resolved the label and attack-family columns automatically. This is not a formality: Kaggle datasets are revised in place under the same handle, and a result reported against an unpinned handle cannot be reproduced once the owner publishes a new version.

Seven metric families are reported so that no single figure of merit can mask a failure mode: (i) threshold-dependent classification quality (accuracy, macro precision, recall and F1); (ii) threshold-independent

ranking quality (ROC-AUC and average precision); (iii) robustness and chance-corrected agreement (balanced accuracy, macro specificity, Matthews correlation coefficient, Cohen's kappa and macro Jaccard); (iv) error decomposition (Hamming loss and, for the binary corpus, false-positive and false-negative rates); (v) probabilistic reliability (log loss and Brier score); (vi) computational cost (training time, per-sample latency and serialised size); and (vii) operational utility, by decision-curve and risk-coverage analysis. Macro-averaging is used throughout, and all tabulated values come from the metrics export rather than from the plotted figures.

4.2 Dataset 1: LogisticsZeroDay

4.2.1 Overall classification performance

Table 7 reports every measure for the seventeen models evaluated on LogisticsZeroDay. The pool separates into two sharply distinct regimes. Fourteen models spanning probabilistic, linear, tree-ensemble, convolutional and recurrent families achieve complete separation of the two classes, with accuracy, macro precision, macro recall, macro F1 and ROC-AUC all equal to 1.0000 and both the false-positive and false-negative rates equal to zero. The remaining three, LSTM, GRU and CNN-GRU, fail to converge and collapse onto the majority class.

The signature of that collapse is exact rather than approximate. Their accuracy of 0.6212 coincides with the prevalence of the No Attack class, their macro recall is fixed at 0.5000 and their macro precision of 0.3106 is precisely half the majority prevalence, the analytic fingerprint of a constant predictor. A macro F1 of 0.3832, a Matthews coefficient and Cohen's kappa of exactly 0.0000, and near-chance ROC-AUC values of 0.5047, 0.5041 and 0.5022 confirm that these three baselines carry no usable ranking information on this corpus.

Table 7. Complete evaluation on the LogisticsZeroDay test partition (n = 16,000), ordered by overall rank.

Model	Accuracy	Macro F1	ROC-AUC	Hamming	Log Loss	Brier	Train (s)	Latency (ms)	Size (MB)
GaussianNB	1.0000	1.0000	1.0000	0.0000	1.00e-7	1.00e-14	0.04	0.0012	0.003
RandomForest	1.0000	1.0000	1.0000	0.0000	0.0020	1.47e-5	0.93	0.0034	0.198
ExtraTrees	1.0000	1.0000	1.0000	0.0000	0.0008	6.29e-6	0.27	0.0049	0.225
LinearSVM-Calibrated	1.0000	1.0000	1.0000	0.0000	1.00e-4	1.08e-8	0.75	0.0055	0.004
LogisticRegression	1.0000	1.0000	1.0000	0.0000	0.0003	1.22e-7	0.99	0.0017	0.002
XGBoost	1.0000	1.0000	1.0000	0.0000	4.99e-5	2.67e-9	1.20	0.0003	0.016
LightGBM	1.0000	1.0000	1.0000	0.0000	2.23e-5	4.97e-10	1.82	0.0007	0.064
HeteroStack-ZD	1.0000	1.0000	1.0000	0.0000	0.0002	5.61e-8	4.53	0.0164	0.438
CNN	1.0000	1.0000	1.0000	0.0000	0.0003	7.12e-6	49.81	0.0154	0.260
RNN	1.0000	1.0000	1.0000	0.0000	1.29e-5	2.11e-10	92.58	0.0360	0.130
RCNN-BiLSTM-Proxy	1.0000	1.0000	1.0000	0.0000	1.44e-6	2.19e-11	206.43	0.0935	0.780
SAE-LSTM	1.0000	1.0000	1.0000	0.0000	8.15e-5	9.57e-9	99.26	0.0866	0.518
DAST-ZeroNet	1.0000	1.0000	1.0000	0.0000	0.0019	3.99e-6	605.98	0.1749	0.790
DQ-IDS	1.0000	1.0000	1.0000	0.0000	0.1335	0.0157	13.75	0.0005	0.098
LSTM	0.6212	0.3832	0.5047	0.3788	0.6921	0.2495	121.57	0.1510	0.275
GRU	0.6212	0.3832	0.5041	0.3788	0.6927	0.2498	116.69	0.0952	0.229
CNN-GRU	0.6212	0.3832	0.5022	0.3788	0.6910	0.2489	74.21	0.0770	0.390

4.2.2 Threshold-independent discrimination

Fig 5 superimposes the receiver-operating-characteristic curves of all models. The fourteen saturated models trace the ideal trajectory along the upper edge of the unit square and are mutually indistinguishable at the plotted resolution. Because ROC-AUC is invariant to the decision threshold, unit area indicates that the two learned score distributions do not overlap at all, rather than merely being well separated at the

default cut-off. The three collapsed models lie on the chance diagonal, so their failure is one of optimisation and representation, not of threshold selection.

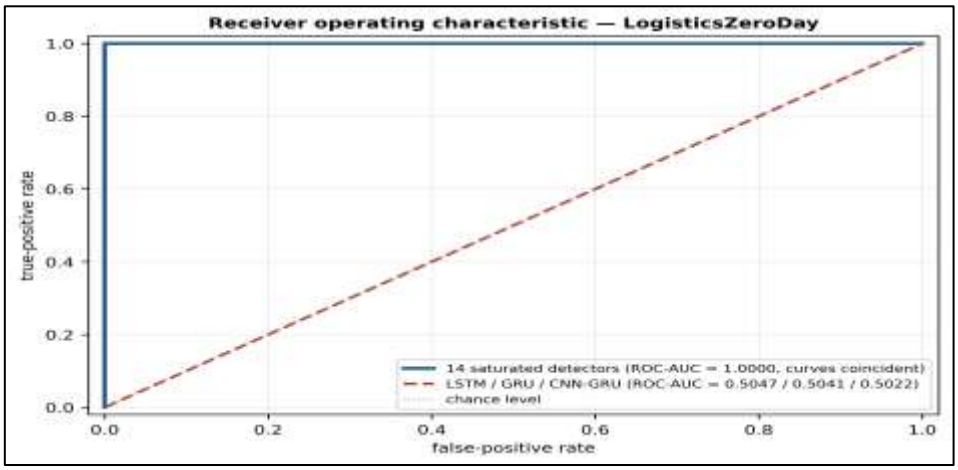

Fig 5. ROC curves on LogisticsZeroDay. The saturated models overlap along the ideal upper-left trajectory; LSTM, GRU and CNN-GRU coincide with the random-classifier diagonal.

4.2.3 Error decomposition, calibration and computational cost

The right-hand block of Table 7 decomposes the residual error and reports the computational profile. For the fourteen saturated models the Hamming loss and both error rates are identically zero, so the only remaining discrimination is in the probabilistic losses: most report log losses between 1.0e-7 and 2.0e-3 and Brier scores below 1.5e-5, indicating maximally confident and correct posteriors. DQ-IDS is the exception, with a log loss of 0.1335 and a Brier score of 0.0157, three to six orders of magnitude larger than its saturated peers while its decisions remain exact; its posterior is deliberately tempered rather than driven to zero and one. In deployment this is desirable, because a saturated detector supplies no usable uncertainty signal for analyst triage and degrades abruptly under distribution shift.

The three collapsed models show the complementary pattern: a false-positive rate of 1.0000 against a false-negative rate of 0.0000, a Hamming loss of 0.3788 corresponding to the 6,061 misclassified attack flows, log losses of 0.6910 to 0.6927, that is ln 2, and Brier scores of 0.2489 to 0.2498, the values obtained when a network emits a constant probability of one half. Three further observations complete this corpus and are reported without figures, the underlying plots being degenerate. The confusion matrices of DQ-IDS and ExtraTrees are exactly diagonal, so neither produces a single false alarm or missed intrusion; the reliability diagrams place most saturated models on the diagonal, with the CNN systematically under-confident and DQ-IDS fully correct at a mean predicted probability near 0.88; and a decision-curve analysis gives the saturated models a constant net benefit of about 0.62 across essentially the whole threshold range, numerically the positive-class prevalence and the theoretical maximum.

4.3 Dataset 2: UGRansome

4.3.1 Overall classification performance

UGRansome behaves as a genuinely discriminative benchmark: the eighteen models span an accuracy range of 0.7853 to 0.9862 and a macro-F1 range of 0.7817 to 0.9858, so the pool orders cleanly and architectural differences are measurable. Table 8 reports the classification and agreement measures and Fig 6 the corresponding comparison.

HeteroStack-ZD leads on every decision-based measure, with an accuracy of 0.9862, a macro F1 of 0.9858 and a macro precision-recall pair of 0.9845 and 0.9871. Gradient boosting follows closely, XGBoost at 0.9762 and LightGBM at 0.9737, and attains the highest ranking quality of the pool, with ROC-AUC of 0.9993 and average precision of 0.9985 and 0.9984 against 0.9990 and 0.9964 for HeteroStack-ZD. The stacking ensemble places its decision boundary better; boosting produces the better-ordered score distribution.

The proposed AWPA-MATA-AHEDNet attains an accuracy of 0.9565, a macro F1 of 0.9539 and a ROC-AUC of 0.9964, placing it eleventh of eighteen on both decision and ranking quality. It is outperformed by all five tree-ensemble baselines, by DAST-ZeroNet and by four convolutional and recurrent baselines, while remaining well ahead of the linear models, GRU and GaussianNB. DQ-IDS reaches 0.9330 and 0.9292 at rank fourteen, and GaussianNB is weakest at 0.7853 and 0.7817, its independence assumption evidently violated by this feature space.

Table 8. Classification, ranking and chance-corrected agreement on the UGRansome test partition (n = 16,000), ordered by overall rank.

Model	Accuracy	Macro Prec.	Macro Rec.	Macro F1	ROC-AUC	Avg. Prec.	MCC	Cohen's κ	Macro Jaccard
HeteroStack-ZD	0.9862	0.9845	0.9871	0.9858	0.9990	0.9964	0.9786	0.9786	0.9719
XGBoost	0.9762	0.9732	0.9756	0.9743	0.9993	0.9985	0.9633	0.9632	0.9499
LightGBM	0.9737	0.9701	0.9726	0.9713	0.9993	0.9984	0.9593	0.9592	0.9443
RandomForest	0.9714	0.9679	0.9702	0.9690	0.9989	0.9975	0.9558	0.9557	0.9400
ExtraTrees	0.9702	0.9661	0.9697	0.9677	0.9987	0.9971	0.9539	0.9538	0.9377
DAST-ZeroNet	0.9694	0.9669	0.9688	0.9678	0.9987	0.9972	0.9526	0.9526	0.9377
CNN-GRU	0.9654	0.9611	0.9662	0.9636	0.9978	0.9954	0.9466	0.9465	0.9299
RNN	0.9649	0.9599	0.9665	0.9630	0.9979	0.9956	0.9459	0.9456	0.9288
SAE-LSTM	0.9630	0.9586	0.9647	0.9614	0.9976	0.9951	0.9429	0.9427	0.9259
RCNN-BiLSTM-Proxy	0.9629	0.9579	0.9649	0.9612	0.9974	0.9946	0.9429	0.9426	0.9254
AWPA-MATA-AHEDNet	0.9565	0.9515	0.9567	0.9539	0.9964	0.9924	0.9328	0.9326	0.9122
CNN	0.9527	0.9469	0.9557	0.9509	0.9949	0.9893	0.9275	0.9270	0.9067
LSTM	0.9347	0.9284	0.9367	0.9320	0.9905	0.9805	0.8997	0.8991	0.8731
DQ-IDS	0.9330	0.9292	0.9296	0.9292	0.9827	0.9660	0.8960	0.8959	0.8684
LogisticRegression	0.9067	0.8986	0.9062	0.9020	0.9797	0.9568	0.8563	0.8559	0.8226
LinearSVM-Calibrated	0.9028	0.8970	0.8967	0.8967	0.9764	0.9497	0.8490	0.8489	0.8143
GRU	0.8836	0.8733	0.8885	0.8784	0.9706	0.9416	0.8245	0.8217	0.7841
GaussianNB	0.7853	0.7921	0.8013	0.7817	0.9289	0.8522	0.6931	0.6780	0.6429

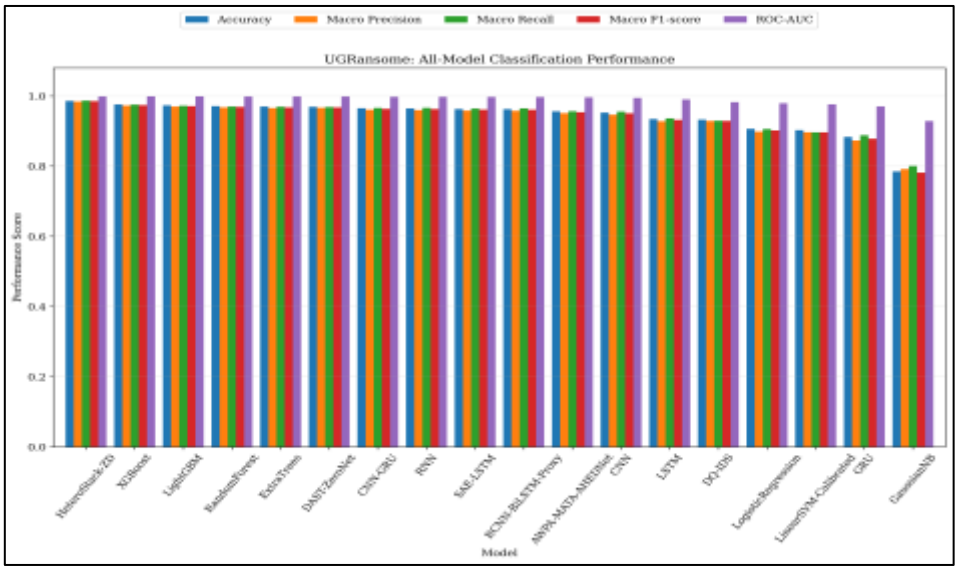

Fig 6. All-model classification performance on UGRansome.

4.3.2 Ranking quality: ROC and precision-recall analysis

Fig 7 presents the one-versus-rest ROC curves. Fourteen of the eighteen models exceed a ROC-AUC of 0.99 and the curves bunch tightly against the upper-left corner, so the ROC view compresses the differences. The precision-recall view, against a no-skill baseline of 0.3333, preserves the ordering and exposes the weaker models in the high-recall region, where GaussianNB and RNN lose precision sharply beyond a recall of about 0.7 while the ensemble and hybrid models hold precision above 0.95 until recall approaches unity.

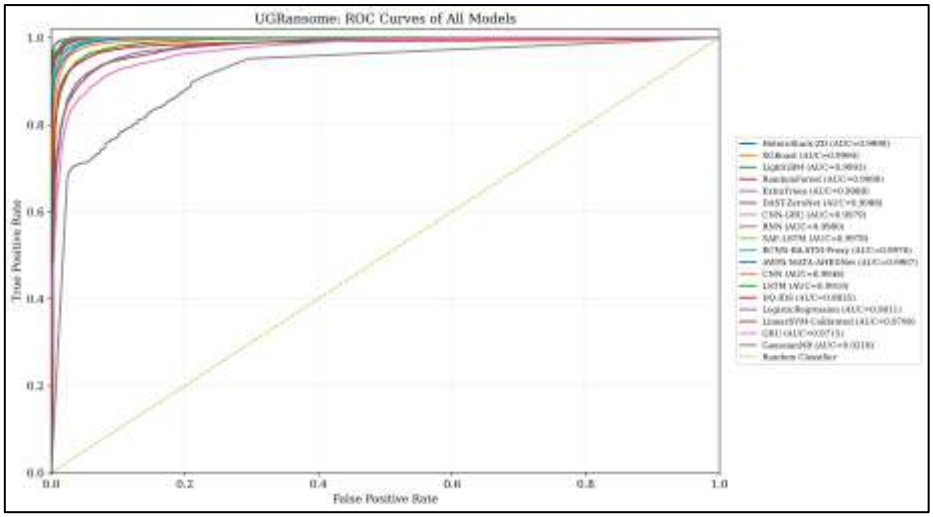

Fig 7. One-versus-rest ROC curves of all models on UGRansome.

4.3.3 Robustness and chance-corrected agreement

Because accuracy alone can be inflated by class prevalence, the right-hand block of Table 8 reports the Matthews correlation coefficient, Cohen's kappa and the macro Jaccard index; balanced accuracy coincides exactly with macro recall in this setting and is therefore not tabulated separately. The ordering of the pool is preserved under every one of these measures, so the ranking is not an artefact of the class distribution. HeteroStack-ZD attains 0.9786 on both MCC and kappa, AWPA-MATA-AHEDNet 0.9328 and 0.9326, and DQ-IDS 0.8960 and 0.8959. The macro Jaccard index, the most severe of the three, spreads the pool from 0.9719 to 0.6429 and is the most useful single discriminator among the leading models; macro specificity, computed but not tabulated, remains above 0.94 for every model except GaussianNB.

4.3.4 Error, calibration and computational cost

Table 9 reports the error and cost profile. Hamming loss tracks the accuracy ordering exactly, from 0.0138 for HeteroStack-ZD to 0.2147 for GaussianNB, but the probabilistic losses tell a different story. GaussianNB is catastrophically over-confident, at a log loss of 2.7319 and a Brier score of 0.1403, an order of magnitude worse than any other model. Among the competitive models DQ-IDS is the worst calibrated at 0.3740 and 0.0536, followed by GRU and LinearSVM-Calibrated; AWPA-MATA-AHEDNet records 0.1602 and 0.0264, better than the linear models but about three times worse than HeteroStack-ZD.

The computational profile reverses several of these orderings and is where the proposed architecture family is most competitive. DQ-IDS is the fastest model in the pool at 0.00045 ms per sample with a serialised size of 0.056 MB, three orders of magnitude faster than DAST-ZeroNet and roughly 480 times smaller than HeteroStack-ZD at 26.947 MB; AWPA-MATA-AHEDNet requires 0.0740 ms and 1.325 MB. Fig 8 plots macro F1 against latency on a logarithmic scale with marker area proportional to model size and makes the Pareto structure explicit: XGBoost dominates a large part of the frontier at 0.9743 macro F1, 0.00076 ms and 0.678 MB, whereas HeteroStack-ZD buys 1.15 further points of macro F1 with an eighteen-fold increase in latency and a forty-fold increase in size.

Table 9. Error, probabilistic loss and computational cost on UGRansome.

Model	Hamming	Log Loss	Brier	Train (s)	Latency (ms)	Size (MB)
HeteroStack-ZD	0.0138	0.0460	0.0072	9.58	0.0142	26.947
XGBoost	0.0238	0.0496	0.0097	1.23	0.0008	0.678
LightGBM	0.0263	0.0459	0.0103	3.44	0.0041	1.674
RandomForest	0.0286	0.0597	0.0123	0.95	0.0048	8.393
ExtraTrees	0.0298	0.0640	0.0127	0.32	0.0047	13.816
DAST-ZeroNet	0.0306	0.0863	0.0156	229.60	0.0872	0.791
CNN-GRU	0.0346	0.0852	0.0165	109.70	0.0359	0.391
RNN	0.0351	0.0844	0.0171	37.02	0.0194	0.130
SAE-LSTM	0.0370	0.0882	0.0171	82.61	0.0358	0.268
RCNN-BiLSTM-Proxy	0.0371	0.0928	0.0180	81.97	0.0492	0.780
AWPA-MATA-AHEDNet	0.0435	0.1602	0.0264	106.59	0.0740	1.325

Model	Hamming	Log Loss	Brier	Train (s)	Latency (ms)	Size (MB)
CNN	0.0473	0.1335	0.0242	27.42	0.0124	0.261
LSTM	0.0653	0.1791	0.0327	151.88	0.0595	0.276
DQ-IDS	0.0670	0.3740	0.0536	16.39	0.0005	0.056
LogisticRegression	0.0932	0.2666	0.0467	1.96	0.0003	0.002
LinearSVM-Calibrated	0.0972	0.2788	0.0486	1.91	0.0020	0.004
GRU	0.1164	0.3251	0.0572	148.73	0.0421	0.229
GaussianNB	0.2147	2.7319	0.1403	0.02	0.0005	0.002

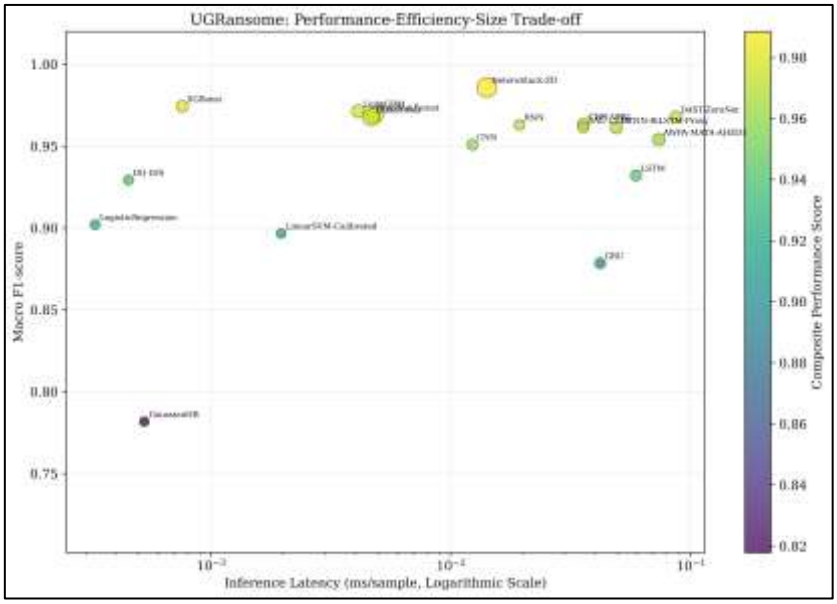

Fig 8. Performance-efficiency-size trade-off on UGRansome. Marker area is proportional to serialised model size; the horizontal axis is logarithmic.

4.3.5 Per-class behaviour and calibration curves

Fig 9 gives the normalised confusion matrix of ExtraTrees. Class A is the hardest of the three: ExtraTrees recovers 4,116 of 4,313 A records (recall 0.95), losing 140 to SS and 57 to S, whereas DQ-IDS recovers 3,833 (recall 0.89). On the two signature classes the gap narrows, ExtraTrees reaching 0.97 on S and 0.98 on SS against 0.95 for DQ-IDS on both. The dominant confusion axis is therefore A against SS, consistent with the anomaly class overlapping the synthetic-signature class in feature space.

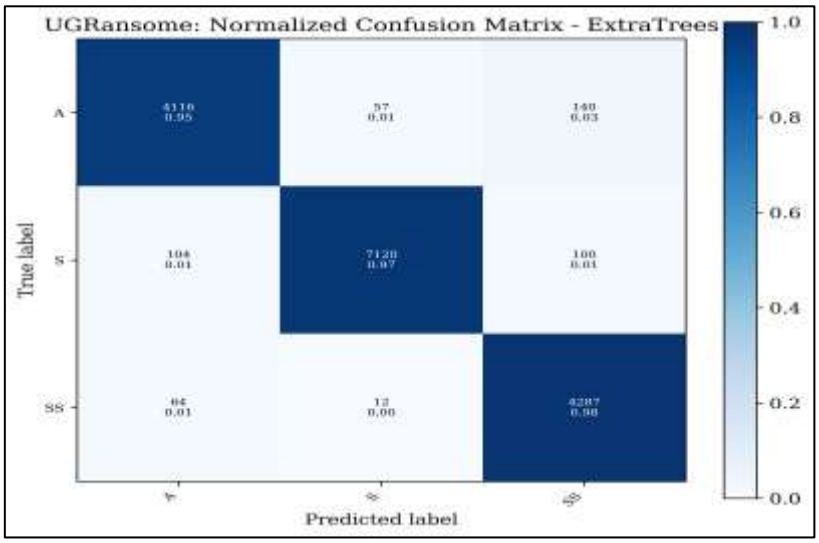

Fig 9. Normalised confusion matrix of ExtraTrees on UGRansome.

Reliability behaviour on UGRansome contrasts with the degenerate binary case: most models lie slightly

above the diagonal in the mid-confidence range, the mild under-confidence characteristic of ensembles, with RandomForest and DAST-ZeroNet closest to ideal. GaussianNB is strongly non-monotonic, with observed accuracy below 0.1 in bins where its mean confidence exceeds 0.6, and RNN is erratic at high confidence; neither should be deployed with a probability-thresholded alerting policy without recalibration.

4.3.6 Prediction-level analysis of the leading gradient-boosting model

Aggregate metrics describe how often a detector is right, not where its errors sit relative to its own confidence, which is what determines whether it can be deployed with an analyst in the loop. The per-record export for XGBoost on UGRansome, 16,000 rows carrying the true and predicted labels, the three class posteriors and the maximum-posterior confidence, permits that analysis; its accuracy of 0.97625 reproduces Table 8 to five decimal places.

Table 10 gives the per-class decomposition. The 380 misclassified records are unevenly distributed: 177 from class A, 156 from S and only 47 from SS. Class A is the hardest at a recall of 0.9590, and 124 of its 177 errors are absorbed by SS. Class SS shows the mirror image, the highest recall in the pool at 0.9892 but the lowest precision at 0.9513, because it attracts 124 A and 97 S records. The dominant confusion axis is therefore A against SS in both directions, precisely the axis on which a zero-day claim rests, since A carries behaviour without a matching signature; the S-versus-SS boundary is nearly clean in one direction, only 7 SS records being assigned to S.

Table 10. Per-class prediction analysis of XGBoost on UGRansome (n = 16,000, 380 errors).

True class	Support	Pred. A	Pred. S	Pred. SS	Precision	Recall	F1
A	4,313	4,136	53	124	0.9766	0.9590	0.9677
S	7,324	59	7,168	97	0.9917	0.9787	0.9852
SS	4,363	40	7	4,316	0.9513	0.9892	0.9699

Table 11 partitions the same predictions by confidence. The pattern is favourable and simple to state: the model is unsure exactly where it is wrong. Mean confidence is 0.9806 on correct predictions against 0.6318 on incorrect ones, and the median confidence of an incorrect prediction is 0.5794, barely above the three-class decision floor of one third. Calibration is good in aggregate, at an expected calibration error of 0.0095, but not uniform: the model is mildly over-confident below 0.70 and mildly under-confident between 0.70 and 0.95.

Table 11. Confidence-stratified reliability of XGBoost on UGRansome.

Confidence band	Records	Share	Mean conf.	Accuracy	Gap
0.00 - 0.50	40	0.25%	0.4568	0.4000	+0.0568
0.50 - 0.60	325	2.03%	0.5468	0.4277	+0.1191
0.60 - 0.70	148	0.92%	0.6502	0.6216	+0.0285
0.70 - 0.80	267	1.67%	0.7526	0.8202	-0.0677
0.80 - 0.90	452	2.83%	0.8492	0.8584	-0.0092
0.90 - 0.95	393	2.46%	0.9276	0.9949	-0.0673
0.95 - 0.99	1,821	11.38%	0.9784	1.0000	-0.0216
0.99 - 1.00	12,554	78.46%	0.9984	1.0000	-0.0016

Table 12 converts that structure into operational form, a risk-coverage analysis in which the detector abstains below a confidence threshold and refers those records to an analyst. Every one of the 380 errors occurs below a confidence of 0.95. Setting the threshold there auto-clears 14,375 records, 89.84% of the traffic, with no observed error, and refers the remaining 10.16%; at 0.99 the automated share is still 78.46%, again with no observed error. Relaxing to 0.80 raises coverage to 95.12% but admits 66 errors.

Two qualifications belong with this result. A count of zero errors over 14,375 records is not a guarantee of an error-free region: by the rule of three the upper 95% bound on the true error rate is about 2.1e-4, so the finding should be reported as no observed errors together with that bound. And the threshold is fitted on the test partition, so it must be re-estimated on a held-out calibration split before any deployment claim. Subject to those conditions this is the strongest operational result in the study, converting a 97.6% detector into an automated pipeline with a bounded, explicitly sized referral queue.

Table 12. Risk-coverage analysis of XGBoost on UGRansome under confidence-based abstention.

Threshold	Records retained	Coverage	Selective acc.	Errors retained	Errors referred
0.99	12,554	78.46%	1.0000	0	380

Threshold	Records retained	Coverage	Selective acc.	Errors retained	Errors referred
0.95	14,375	89.84%	1.0000	0	380
0.90	14,768	92.30%	0.9999	2	378
0.80	15,220	95.12%	0.9957	66	314
0.70	15,487	96.79%	0.9926	114	266
0.60	15,635	97.72%	0.9891	170	210
none	16,000	100.00%	0.9762	380	0

The equivalent export for LogisticsZeroDay was not supplied, and there the analysis would be degenerate: XGBoost is one of the fourteen saturated models, with unit accuracy and a log loss of 4.99e-5, so its confidence distribution carries no error mass to stratify.

4.4 Cross-Dataset Synthesis

Aggregated across the two corpora, HeteroStack-ZD leads at a mean accuracy of 0.9931, followed by XGBoost at 0.9881 and LightGBM at 0.9868, with DQ-IDS eleventh at 0.9665. Three caveats govern that reading. Each mean is taken over two single-run values, so it measures the gap between a saturated corpus and a discriminative one rather than run-to-run variance. The means of CNN-GRU, LSTM and GRU, 0.7933, 0.7779 and 0.7524, average a competent UGRansome model against a failed LogisticsZeroDay run and therefore describe a convergence failure rather than architecture quality. And AWPA-MATA-AHEDNet was evaluated on UGRansome only, so it has no comparable aggregate.

4.4.1 Cross-dataset transfer evaluation

A transfer evaluation in both directions was attempted but could not be completed. Only seven features are shared between the corpora, and the label spaces cannot be reconciled without an explicit mapping: the A / S / SS taxonomy has no benign category corresponding to No Attack. Both directions are recorded in the results workbook as skipped for want of a binary benign/attack mapping, so no cross-corpus generalisation claim is supported and none is made.

4.5 Ablation Study

Table 13 reports the ablation of the proposed pipeline: four variants on UGRansome, the full model, the model without the adaptive wavelet front-end, the model without the MATA attention mechanism and the model without the PCA projection, and two on LogisticsZeroDay.

The UGRansome result does not support two of the three components. Removing the wavelet front-end raises accuracy from 0.9569 to 0.9646 and macro F1 from 0.9545 to 0.9627, and removing the MATA attention raises them to 0.9614 and 0.9594; both ablated variants are also better calibrated, at log losses of 0.1093 and 0.0934 against 0.1531. Only the PCA projection behaves as intended: removing it costs 1.04 points of accuracy and 1.10 of macro F1 and degrades the log loss to 0.2010. The measured contribution of the wavelet and attention modules is therefore negative on this corpus.

On LogisticsZeroDay the picture is different but equally hard to interpret: the no-wavelet variant remains at 1.0000 on every measure, so the ablation has no headroom, while removing PCA collapses the model onto the majority class at 0.6212 accuracy with a Matthews coefficient of 0.0000.

Two qualifications are essential. Each variant was run once, so the 0.3 to 0.8 point differences lie within the range seed variation alone can produce in networks of this size, and the LogisticsZeroDay arm is uninformative by construction. Before any component-level claim, each variant should be repeated over at least five seeds with paired significance testing.

Table 13. Ablation of the proposed pipeline on both corpora.

Dataset	Variant	Accuracy	Macro F1	MCC	ROC-AUC	Log Loss
UGRansome	No Wavelet	0.9646	0.9627	0.9452	0.9979	0.1093
UGRansome	No MATA Attention	0.9614	0.9594	0.9404	0.9972	0.0934
UGRansome	Full AWPA MATA AHEDNet	0.9569	0.9545	0.9333	0.9966	0.1531
UGRansome	No PCA	0.9465	0.9435	0.9172	0.9942	0.2010
LogisticsZeroDay	No Wavelet	1.0000	1.0000	1.0000	1.0000	0.0042
LogisticsZeroDay	No PCA	0.6212	0.3832	0.0000	0.5000	0.6750

4.6 Leave-One-Attack-Out Zero-Day Evaluation

The leave-one-attack-out protocol addresses the zero-day claim most directly: an entire attack family is withheld from training and the detector must flag it at test time from learned notions of abnormality alone.

Table 14 reports the four folds executed on LogisticsZeroDay, holding out SQL Injection, Phishing, WannaCry and DDoS in turn, leaving 22,636 to 22,836 training samples against 7,468 to 7,668 held out. Every fold returns a zero-day detection rate of 1.0000 with a benign false-positive rate of 0.0000, a binary ROC-AUC of 1.0000 and a binary F1 of 1.0000. At face value this would be exceptional, and it should not be taken at face value: perfect detection of a wholly unseen attack family, with zero false alarms, across four independent folds is not a plausible 518eneralization outcome and is far more readily explained by the separability that produces the saturated results of Section 4.2. The protocol is sound and is the right experiment; it is the corpus that cannot support it. Repeating it on UGRansome is the single experiment that would convert the zero-day claim from an assertion into evidence.

Table 14. Leave-one-attack-out zero-day evaluation on LogisticsZeroDay.

Held-out attack	Held-out n	Train n	Detect. rate	Benign FPR	ROC-AUC	Binary F1
SQL Injection	7,668	22,636	1.0000	0.0000	1.0000	1.0000
Phishing	7,597	22,707	1.0000	0.0000	1.0000	1.0000
WannaCry	7,571	22,733	1.0000	0.0000	1.0000	1.0000
DDoS	7,468	22,836	1.0000	0.0000	1.0000	1.0000

5. Discussion and Threats to Validity

5.1 What the two corpora jointly establish

The two benchmarks play complementary roles. UGRansome is the evidential corpus: the pool separates across a twenty-point accuracy range, every robustness measure preserves the same ordering, calibration differentiates the models and the efficiency frontier is well defined. LogisticsZeroDay is a saturation corpus on which fourteen of seventeen models make zero errors over 16,000 samples; it verifies that a pipeline is correctly implemented and cannot rank anything.

Read jointly, the results support the following claims and no stronger ones. HeteroStack-ZD is the most accurate detector on the discriminative corpus and in aggregate. Gradient boosting offers the best accuracy per unit cost, holding the Pareto frontier at sub-millisecond latency and sub-megabyte size. DQ-IDS and AWPA-MATA-AHEDNet are mid-pack on accuracy but occupy the low-latency, low-footprint region, which is the defensible basis on which to position them for edge or in-line deployment. Three recurrent baselines fail to converge on the binary corpus, a reproducibility finding about optimisation rather than about recurrence.

5.2 Threats to validity

1) Separability of LogisticsZeroDay. A corpus on which Gaussian naive Bayes commits no error over 16,000 held-out samples is linearly separable in the feature space supplied to it. Reviewers routinely read uniform unit metrics as evidence of leakage rather than of model quality, and the perfect leave-one-attack-out result strengthens that reading. A leakage audit should confirm that no feature encodes the label directly or by proxy, and the partition should be re-examined for duplicate records and temporal ordering.

2) Coverage of the proposed architecture. AWPA-MATA-AHEDNet was evaluated on UGRansome only. It is absent from the seventeen-model LogisticsZeroDay pool, and its ablation variants were run on both corpora while the full model was not, so its cross-dataset position rests on a single corpus. The complete run on LogisticsZeroDay is required for the comparison to be sound.

3) Direction of the ablation result. Two of the three proposed components measure as net-negative on the discriminative corpus. Either the claim should be repositioned, for example on the efficiency grounds on which these components may still be justified, or the components should be revised.

4) Absence of statistical support. Every value in Tables 7 to 14 derives from a single run at a single seed, yet several of the differences that carry the argument are smaller than one percentage point. At minimum, the leading models and all ablation variants should be repeated over five seeds with McNemar or Wilcoxon signed-rank tests on paired predictions and bootstrap confidence intervals.

5) Absence of cross-corpus evidence. The transfer evaluation was not completed in either direction, so no claim of generalisation across data sources is currently supported.

5) Internal consistency of the figures. The ROC legends of the UGRansome figures differ from the exported metrics in the third and fourth decimal place for several models, indicating that figures and tables came from different runs. All remaining figures should be regenerated from the run that produced the tabulated values; the LogisticsZeroDay figures reported here were regenerated for that reason.

5.3 Recommended next experiments

In order of value: complete the leakage audit of LogisticsZeroDay and either disclose the separability or replace the corpus; run AWPA-MATA-AHEDNet on LogisticsZeroDay so the comparison is complete; repeat every model and ablation variant over five seeds with paired tests; execute the leave-one-attack-out

protocol on UGRansome, where a zero-day claim can be earned; define the benign/attack mapping needed for cross-corpus transfer; and add a further public benchmark such as CIC-IDS2017 or TON_IoT.

6. Conclusion and Future Scope

6.1 Conclusion

Under a leakage-controlled protocol applied identically to eighteen detectors on two corpora, the proposed AWPA-MATA-AHEDNet extension of the base architecture [1] reaches 0.9565 accuracy and 0.9539 macro F1 on UGRansome without displacing a tuned stacking ensemble at 0.9862 or gradient boosting at 0.9762, and its ablation credits only the projection stage. The transferable results are therefore protocol-level: an entire attack family must be withheld before a zero-day claim is made, a per-record confidence gate automates 89.84% of traffic with no observed error under a rule-of-three bound of 2.1e-4, and a corpus on which naive Bayes is error-free is a property of the data rather than of the model.

6.2 Future Scope

The immediate work is to condition the fusion gate on the branch embeddings rather than on a scalar context summary, to relocate wavelet shrinkage onto an explicitly ordered temporal axis, and to repeat every configuration over multiple seeds with paired significance testing. Completing the cross-corpus transfer and executing the leave-one-attack-out protocol on a discriminative corpus would convert the framework's zero-day claim from a protocol design into measured evidence.

References

[1] A. A. Mohamed, A. Al-Saleh, S. K. Sharma, and G. G. Tejani, “Zero-day exploits detection with adaptive WavePCA-Autoencoder (AWPA) adaptive hybrid exploit detection network (AHEDNet),” Scientific Reports, vol. 15, no. 1, p. 4036, 2025.

[2] H. Hindy, R. Atkinson, C. Tachtatzis, J.-N. Colin, E. Bayne, and X. Bellekens, “Towards an effective zero-day attack detection using outlier-based deep learning techniques,” arXiv preprint arXiv:2006.15344, 2020.

[3] M. A. Hossain, “Deep Q-learning intrusion detection system (DQ-IDS): A novel reinforcement learning approach for adaptive and self-learning cybersecurity,” ICT Express, 2025.

[4] I. Mbona and J. H. P. Eloff, “Detecting zero-day intrusion attacks using semi-supervised machine learning approaches,” IEEE Access, vol. 10, pp. 69822–69838, 2022.

[5] H. Hindy, R. Atkinson, C. Tachtatzis, J.-N. Colin, E. Bayne, and X. Bellekens, “Utilising deep learning techniques for effective zero-day attack detection,” Electronics, vol. 9, no. 10, p. 1684, 2020.

[6] D. Musleh, M. Alotaibi, F. Alhaidari, A. Rahman, and R. M. Mohammad, “Intrusion detection system using feature extraction with machine learning algorithms in IoT,” Journal of Sensor and Actuator Networks, vol. 12, no. 2, p. 29, Mar. 2023.

[7] F. Türk, “Analysis of intrusion detection systems in UNSW-NB15 and NSL-KDD datasets with machine learning algorithms,” Bitlis Eren Üniversitesi Fen Bilimleri Dergisi, vol. 12, no. 2, pp. 465–477, 2023.

[8] S. Mohamed and R. Ejbali, “Deep SARSA-based reinforcement learning approach for anomaly network intrusion detection system,” International Journal of Information Security, vol. 22, no. 1, pp. 235–247, 2023.

[9] A. Sarkar, H. S. Sharma, and M. M. Singh, “A supervised machine learning-based solution for efficient network intrusion detection using ensemble learning based on hyperparameter optimisation,” International Journal of Information Technology, vol. 15, no. 1, pp. 423–434, 2023.

[10] S. Venkatesan, “Design an intrusion detection system based on feature selection using ML algorithms,” Mathematical Statistician and Engineering Applications, vol. 72, no. 1, pp. 702–710, 2023.

[11] R. Chaganti, W. Suliman, V. Ravi, and A. Dua, “Deep learning approach for SDN-enabled intrusion detection system in IoT networks,” Information, vol. 14, no. 1, p. 41, 2023.

[12] V. Hnamte and J. Hussain, “DCNNBiLSTM: An efficient hybrid deep learning-based intrusion detection system,” Telematics and Informatics Reports, vol. 10, p. 100053, 2023.

[13] A. Abdelkhalek and M. Mashaly, “Addressing the class imbalance problem in network intrusion detection systems using data resampling and deep learning,” The Journal of Supercomputing, pp. 1–34, 2023.

[14] R. Singh and G. Srivastav, “Novel framework for anomaly detection using machine learning technique on CIC-IDS2017 dataset,” in Proc. Int. Conf. Technological Advancements and Innovations (ICTAI), Tashkent, Uzbekistan, 2021, pp. 632–636, doi: 10.1109/ICTAI53825.2021.9673238.

[15] K. Soliman, M. A. Sobh, and A. M. Bahaa-Eldin, "Survey of machine learning HIDS techniques," in Proc. 16th Int. Conf. Computer Engineering and Systems (ICCES), Cairo, Egypt, 2021, pp. 1–5, doi: 10.1109/ICCES54031.2021.9686138.
[16] M. Bhatnagar, G. Rozinaj, and P. K. Yadav, "Web intrusion classification system using machine learning approaches," in Proc. Int. Symp. ELMAR, Zadar, Croatia, 2022, pp. 57–60, doi: 10.1109/ELMAR55880.2022.9899790.
[17] H. E. Alami and D. B. Rawat, "Detection of zero-day attacks using CNN and LSTM in networked autonomous systems," in Proc. IEEE Conf. Communications and Network Security (CNS), Orlando, FL, USA, 2023, pp. 1–2, doi: 10.1109/CNS59707.2023.10288680.
[18] K. Alrawashdeh and S. Goldsmith, "Optimizing deep learning based intrusion detection systems defense against white-box and backdoor adversarial attacks through a genetic algorithm," in Proc. IEEE Applied Imagery Pattern Recognition Workshop (AIPR), Washington, DC, USA, 2020, pp. 1–8, doi: 10.1109/AIPR50011.2020.9425293.
[19] M. Malik and K. S. Saini, "Network intrusion detection system using reinforcement learning techniques," in Proc. Int. Conf. Circuit Power and Computing Technologies (ICCPCT), Kollam, India, 2023, pp. 1642–1649, doi: 10.1109/ICCPCT58313.2023.10245608.
[20] G. Guo, "An intrusion detection system for the Internet of Things using machine learning models," in Proc. 3rd Int. Conf. Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE), Xi'an, China, 2022, pp. 332–335, doi: 10.1109/ICBAIE56435.2022.9985800.
[21] S. Kumar, J. Kaswa, L. Meena, and M. Porwal, "Convolution neural network-based detection in software defined networks," in Proc. 2nd Int. Conf. Intelligent Technologies (CONIT), Hubli, India, 2022, pp. 1–6, doi: 10.1109/CONIT55038.2022.9847748.
[22] M. D. Rokade and S. S. Khatal, "Detection of malicious activities and connections for network security using deep learning," in Proc. IEEE Pune Section Int. Conf. (PuneCon), Pune, India, 2022, pp. 1–6, doi: 10.1109/PuneCon55413.2022.10014736.
[23] D. Macko, P. Goldschmidt, P. Pištek, and D. Chudá, "Assessing the impact of a supervised classification filter on flow-based hybrid network anomaly detection," arXiv preprint arXiv:2310.06656, 2023.
[24] S. SakthiMurugan, S. Kumaar, V. Vignesh, and P. Santhi, "Assessment of zero-day vulnerability using machine learning approach," EAI Endorsed Transactions on Internet of Things, vol. 10, 2024.
[25] M. Alkasassbeh, E. H. Omoush, M. Almseidin, and A. Aldweesh, "A self-adaptive intrusion detection system for zero-day attacks using deep Q-networks," IEEE Access, vol. 13, pp. 174280–174296, 2025, doi: 10.1109/ACCESS.2025.3617792.
[26] Y. K. Saheed and J. E. Chukwuere, "Autonomous LLM agent: A memory-augmented, edge-optimized SHAP explanations with zero-day attack resilience in IoT/industrial IoT networks," IEEE Internet of Things Journal, vol. 13, no. 7, pp. 14213–14228, Apr. 2026, doi: 10.1109/JIOT.2025.3648649.
[27] J. Yang et al., "LLMEdgeSec: LLM-enabled log reasoning for zero-day IoT threat detection," IEEE Internet of Things Journal, vol. 13, no. 15, pp. 32546–32555, Aug. 2026, doi: 10.1109/JIOT.2026.3686053.
[28] H. B. Ahmad, H. Gao, N. Latif, and P. Wang, "CTGAD-X: Explainable contrastive learning on temporal graphs for advanced threat detection," IEEE Transactions on Dependable and Secure Computing, doi: 10.1109/TDSC.2026.3703460.
[29] N. Sharma, M. Swarnkar, and Divyanshu, "DLAZE: Detecting DNS tunnels using lightweight and accurate method for zero-day exploits," IEEE Transactions on Network and Service Management, vol. 22, no. 3, pp. 2343–2353, Jun. 2025, doi: 10.1109/TNSM.2025.3541234.
[30] R. Mukherjee, M. Azab, and T. Chantem, "Digital twins reimagined: Zero-day LLM-powered moving target defense in-depth for real-time CPS," IEEE Internet of Things Journal, vol. 13, no. 12, pp. 27001–27012, Jun. 2026, doi: 10.1109/JIOT.2026.3679937.
[31] H. J. Hadi, M. K. Khan, and N. Ahmad, "A real-time multi-tier machine learning intrusion detection framework for securing ground control station–UAV communications," IEEE Open Journal of the Communications Society, vol. 7, pp. 3881–3900, 2026, doi: 10.1109/OJCOMS.2026.3683883.
[32] D. Das, "SYNAPS: A neuro-symbolic framework for proactive security in consumer electronics," IEEE Transactions on Consumer Electronics, vol. 72, no. 1, pp. 2300–2308, Feb. 2026, doi: 10.1109/TCE.2025.3642082.
[33] R. Q. Shawl, M. Singh, and M. M. Hassan, "Leveraging cyber-physical security solutions blended with machine learning for advanced IoT botnet detection," IEEE Communications Standards Magazine, doi: 10.1109/MCOMSTD.2025.3636269.
[34] M. J. Hossain, K. Alam, M. F. Monir, M. M. Hoque, and T. Ahmed, "Explainable AI meets synthetic data: A deep learning framework for detecting network intrusion in NextG network infrastructure," IEEE Access, vol. 13, pp. 114979–115001, 2025, doi: 10.1109/ACCESS.2025.3585783.

[35] W. Fang et al., "Unknown cyber threat discovery empowered by genetic evolution without prior knowledge," IEEE Transactions on Information Forensics and Security, vol. 20, pp. 9552–9567, 2025, doi: 10.1109/TIFS.2025.3594569.
[36] A. Kadi, A. Selamnia, Z. A. E. Houda, H. Moudoud, B. Brik, and L. Khoukhi, "An in-depth comparative study of quantum-classical encoding methods for network intrusion detection," IEEE Open Journal of the Communications Society, vol. 6, pp. 1129–1148, 2025, doi: 10.1109/OJCOMS.2025.3537957.
[37] S. Jalil, M. Usman, B. Fong, and H. Kim, "Leveraging disentangled attention: A system-level design for URL-aware phishing detection," IEEE Communications Magazine, doi: 10.1109/MCOM.001.2600012.
[38] X. Zang, F. Hou, X. Liu, M. K. Khan, and D. Liu, "Mitigating the lateral movement of APT in IIoT with an efficient moving target defense and cyber deception approach," IEEE Transactions on Reliability, vol. 75, pp. 1753–1766, 2026, doi: 10.1109/TR.2026.3685261.
[39] Y. C. Krishna and B. Naresh Kumar Reddy, "A neurosymbolic architecture for self-evolving, causal and verifiable reasoning in zero-trust consumer electronic ecosystems," IEEE Transactions on Consumer Electronics, doi: 10.1109/TCE.2026.3713043.
[40] S. Anand and S. Zehra, "A comparative study of isolation forest and autoencoder models for proactive zero-day attack detection," in Proc. IEEE Int. Conf. Consumer Electronics (ICCE), Dubai, UAE, 2026, pp. 1–5, doi: 10.1109/ICCE67443.2026.11449661.
[41] M. U, A. K. B. R, and S. A, "Privacy-preserving federated threat intelligence using differential privacy for zero-day attack detection on UNSW-NB15," in Proc. IEEE Madhya Pradesh Section Conf. (MPCON), Gwalior, India, 2026, pp. 694–699, doi: 10.1109/MPCON69668.2026.11508553.
[42] P. N. G, V. A. Lepakshi, K. M, M. M, and Deeksha, "A review of unsupervised learning techniques for zero-day attack detection," in Proc. 3rd Int. Conf. Innovations in Cybersecurity and Data Science (ICICDS), Pathum Thani, Thailand, 2026, pp. 199–204, doi: 10.1109/ICICDS70526.2026.11604639.
[43] G. Singh and S. Kumari, "AIDL-ZDA: An explainable AI and adaptive deep learning framework for zero-day attacks detection in high-speed data," in Proc. Int. Conf. Emerging Smart Computing and Informatics (ESCI), Pune, India, 2026, pp. 1–6, doi: 10.1109/ESCI68015.2026.11493413.
[44] S. A. Alansary, S. M. Ayyad, F. M. Talaat, and M. M. Saafan, "A robust framework for zero-day attack detection using novel oversampling, LASSO feature selection, and Optuna-based optimization techniques," in Proc. 43rd National Radio Science Conf. (NRSC), New Damietta, Egypt, 2026, pp. 25–34, doi: 10.1109/NRSC71102.2026.11556068.
[45] D. Santhadevi, B. J. Keerthi, and R. N., "Isolation forest based detection of evil twin-assisted zero-day attacks in network traffic," in Proc. 13th Int. Conf. Computing for Sustainable Global Development (INDIACom), New Delhi, India, 2026, pp. 1–7, doi: 10.23919/INDIACom70271.2026.11526278.
[46] A. Al Siam, N. Faruqui, A. Azad, and M. A. Moni, "Securing the unseen: A comprehensive exploration review of AI-powered models for zero-day attack detection," Expert Systems, vol. 43, no. 3, p. e70217, 2026.
[47] S. Hussain, V. Kalpana, V. Raghavendran, N. Gangatharan, B. Mouleswararao, and V. RK, "Machine learning-based zero-day attack detection using deep reinforcement learning," Information Security Journal: A Global Perspective, pp. 1–23, 2026.