

DAMG-LiquidIDS: A Drift-Aware Feature-Gated Liquid Neural Network for Intrusion Detection in Dynamic IoT and Mobile Networks

Nasreddine Hamdi^{1,2}, Akram Belazi^{1,2,3}, Safya Belghith¹, Héctor Migallón³

¹Laboratory RISC-ENIT (LR-16-ES07), Tunis El Manar University, Tunis 1002, Tunisia

²Higher Institute of Computer Science and Multimedia of Gabes, University of Gabes, Gabes 6029, Tunisia

³Department of Computer Engineering, Miguel Hernández University, Elche 03202, Spain

Email: {nasreddine.hamdi, akram.belazi, safya.belghith}@enit.utm.tn, hmgallon@umh.es

Abstract—Wireless and IoT networks operate under continuously evolving traffic conditions that induce distribution shifts and degrade the performance of conventional intrusion detection systems. Existing approaches generally rely on static feature representations, limiting their adaptability to non-stationary environments. This paper proposes DAMG-LiquidIDS, a drift-aware adaptive feature-gated Liquid Neural Network that jointly performs dynamic feature selection and temporal traffic modeling. A PSO-driven gating mechanism continuously updates the active feature subset according to validation performance, sparsity constraints, and drift indicators, while the Liquid Neural Network captures temporal network dynamics. Extensive experiments on the CICIOT2023 and FiveG-NIDD benchmarks under temporal, prior, covariate, and unknown-attack protocols demonstrate consistent robustness across diverse distribution shifts. DAMG-LiquidIDS achieves Macro-F1 scores up to 0.914 on CICIOT2023 and 0.980 on FiveG-NIDD while reducing the number of active features, and statistical analyses confirm its effectiveness over several conventional feature-selection baselines.

Index Terms—Intrusion detection, IoT networks, concept drift, Liquid Neural Networks, particle swarm optimization, feature selection, non-stationary learning.

I. INTRODUCTION

Wireless, mobile, and Internet of Things (IoT) infrastructures are becoming increasingly dynamic with the emergence of 5G/6G and edge-enabled services. Although these technologies enable massive connectivity and low-latency applications, they also expand the attack surface exposed to cyber threats [1], [2]. In such environments, traffic distributions evolve continuously due to mobility, device heterogeneity, application changes, and emerging attacks, making the stationarity assumptions of conventional intrusion detection systems (IDSs) difficult to satisfy in practice [3], [4].

Most existing IDS approaches rely on fixed feature representations learned under static training conditions. While effective under random train-test splits, their performance may degrade when feature relevance changes over time [5], [6]. Similarly, metaheuristic feature-selection methods are usually applied offline, producing fixed subsets that cannot adapt to evolving traffic. Continuous-time models such as Liquid Neural Networks (LNNs) can capture temporal dynamics, but

they typically operate on unchanged input representations [9]. Therefore, adaptive IDS models should jointly address feature evolution and temporal decision dynamics.

To this end, we propose *DAMG-LiquidIDS*, a drift-aware feature-gated LNN for intrusion detection in dynamic IoT and mobile networks. The proposed framework periodically updates a continuous feature gate using a drift-aware Particle Swarm Optimization (PSO) procedure driven by validation performance, sparsity, computational cost, temporal stability, and estimated drift intensity. The selected representation is then processed by an LNN to model temporal traffic dynamics [10], [11], [9]. This alternating strategy allows the feature gate and the temporal classifier to co-adapt during training instead of being optimized as two disconnected stages.

The main contributions are as follows:

- We formulate intrusion detection under evolving traffic as a drift-aware bi-level optimization problem coupling adaptive feature gating and LNN-based temporal modeling.
- We introduce a PSO-driven dynamic gate that adapts feature relevance according to validation performance, sparsity, cost, stability, and drift intensity.
- We evaluate DAMG-LiquidIDS on CICIOT2023 and FiveG-NIDD under temporal, reordered label-prior, covariate-drift, and unseen-attack protocols, with ablation and statistical analyses.

The remainder of this paper is organized as follows: Section II reviews related work, Section III presents DAMG-LiquidIDS, Sections IV and V describe the experimental setup and results, and Section VI concludes the paper.

II. RELATED WORK

Machine learning and deep learning are widely used for intrusion detection in IoT, mobile, and wireless networks. Classical models such as Random Forests, Support Vector Machines, and boosting methods have shown competitive performance on benchmarks such as CICIDS2017, UNSW-NB15, and 5G-oriented datasets [14], [15], [13], while deep architectures improve detection through automatic representation learning. However, many IDS studies still rely on random

train–test splits and assume stationary data distributions. This assumption is often violated in mobile and IoT deployments, where traffic evolves due to device heterogeneity, mobility, network reconfiguration, and adaptive adversaries [3], [4]. Recent studies also emphasize the need to evaluate IDS robustness under realistic dynamic conditions rather than static benchmarks alone [7], [1].

Feature optimization is commonly used to improve IDS efficiency by removing redundant or unstable attributes. Meta-heuristic algorithms such as Particle Swarm Optimization (PSO), Genetic Algorithms, and Grey Wolf Optimization are effective in high-dimensional non-convex search spaces [10], [16], [17], and their use in IDS can improve accuracy while reducing the feature set [18]. Nevertheless, most metaheuristic IDS frameworks perform feature selection offline. The selected subset then remains fixed during training and deployment, although feature relevance may change under distributional drift [6].

Deep IDS models such as CNNs, recurrent networks, LSTMs, GRUs, and Transformers have also shown strong representation-learning capability for traffic analysis [19], [20], [21]. Continuous-time models provide another direction for evolving systems: Neural ODEs learn adaptive latent dynamics [11], while Liquid Neural Networks (LNNs) use input-dependent time constants to improve expressiveness under changing conditions [9]. However, these models typically operate on fixed input representations and do not explicitly adapt feature relevance over time. Overall, existing work addresses either feature optimization or temporal modeling, but rarely couples both in a unified adaptive IDS framework. DAMG-LiquidIDS addresses this gap by combining drift-aware dynamic feature gating with LNN-based temporal modeling through an alternating bi-level optimization strategy, enabling the feature representation and temporal classifier to co-adapt during training.

III. PROPOSED METHOD

A. Problem Definition

We consider an intrusion detection setting in which network traffic is processed through successive windows

$$\mathcal{W}_w = \{(\mathbf{x}_{w,i}, y_{w,i})\}_{i=1}^{n_w}, \quad \mathbf{x}_{w,i} \in \mathbb{R}^d, \quad y_{w,i} \in \{1, \dots, C\}, \quad (1)$$

where w denotes the traffic window index.

Each window is assumed to originate from a non-stationary distribution $\mathcal{P}_w$ such that $\mathcal{P}_w \neq \mathcal{P}_{w-1}$. The objective is to learn an adaptive classifier whose representation and decision dynamics evolve with changing traffic conditions.

B. Drift Estimation

A lightweight drift indicator is computed using the first- and second-order statistics of standardized numerical features:

$$D_w = \|\boldsymbol{\mu}_w - \boldsymbol{\mu}_{w-1}\|_2 + \|\boldsymbol{\Sigma}_w - \boldsymbol{\Sigma}_{w-1}\|_F, \quad (2)$$

where $\boldsymbol{\mu}_w$ and $\boldsymbol{\Sigma}_w$ denote the empirical mean vector and covariance matrix, respectively.

The drift intensity is normalized as

$$\hat{D}_w = \sigma(\alpha_D(D_w - \delta_D)), \quad \hat{D}_w \in [0, 1], \quad (3)$$

where $\sigma(\cdot)$ is the sigmoid function.

C. Dynamic Feature Gating

A continuous gate vector

$$\mathbf{z}_e \in [0, 1]^d \quad (4)$$

controls feature relevance during training epoch e .

The gated representation is obtained as

$$\tilde{\mathbf{x}}_i = \mathbf{z}_e \odot \mathbf{x}_i, \quad (5)$$

where $\odot$ denotes element-wise multiplication.

Unlike conventional feature-selection approaches, the gate is not fixed after preprocessing. Instead, it is periodically updated during training and co-adapts with the classifier.

D. Drift-Aware Gate Optimization

At scheduled update epochs, a population of candidate gates $\mathcal{Z} = \{\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(N_p)}\}$ is evaluated using the validation objective

$$J_e(\mathbf{z}) = \mathcal{L}_{\text{val}}(\theta_e, \mathbf{z}) + \lambda_{\text{sp}} \|\mathbf{z}\|_1 + \lambda_{\text{cost}} C(\mathbf{z}) + \lambda_{\text{st}} e^{-\kappa \hat{D}_w} \|\mathbf{z} - \mathbf{z}_{e-1}^*\|_2^2. \quad (6)$$

The optimal gate is

$$\mathbf{z}_e^* = \arg \min_{\mathbf{z} \in \mathcal{Z}} J_e(\mathbf{z}). \quad (7)$$

The first term evaluates validation performance, $\|\mathbf{z}\|_1$ promotes sparsity, and the stability penalty discourages abrupt gate changes. The factor $e^{-\kappa \hat{D}_w}$ relaxes this constraint under strong drift.

The computational-cost regularizer is defined as

$$C(\mathbf{z}) = \frac{1}{d} \sum_{j=1}^d \mathbb{I}(z_j > \eta), \quad (8)$$

where $\eta = 0.5$ is the gate activation threshold.

E. PSO-Based Gate Search

Particle Swarm Optimization (PSO) is used to optimize the gate.

Each particle encodes a candidate feature gate. Its velocity and position are updated according to

$$\begin{aligned} \mathbf{v}^{(i,r+1)} &= \omega_w \mathbf{v}^{(i,r)} + c_1 \mathbf{r}_1 \odot (\mathbf{p}^{(i)} - \mathbf{z}^{(i,r)}) \\ &\quad + c_2 \mathbf{r}_2 \odot (\mathbf{g} - \mathbf{z}^{(i,r)}), \end{aligned} \quad (9)$$

$$\mathbf{z}^{(i,r+1)} = \Pi_{[0,1]^d} \left(\mathbf{z}^{(i,r)} + \mathbf{v}^{(i,r+1)} \right), \quad (10)$$

where $\mathbf{p}^{(i)}$ and $\mathbf{g}$ denote the personal and global best solutions.

The inertia coefficient adapts to drift intensity:

$$\omega_w = \omega_{\min} + (\omega_{\max} - \omega_{\min}) \hat{D}_w. \quad (11)$$

F. Liquid Neural Intrusion Detection

The gated representation is processed by a Liquid Neural Network (LNN) characterized by continuous-time hidden-state dynamics:

$$\dot{\mathbf{h}}(t) = -\boldsymbol{\tau}^{-1} \odot \mathbf{h}(t) + \phi_{\theta}(\mathbf{h}(t), \tilde{\mathbf{x}}_i, t), \quad (12)$$

where $\boldsymbol{\tau}$ denotes learnable time constants.

The final hidden state is passed to a linear classifier:

$$\hat{y}_{i,c} = \text{softmax}(\mathbf{W}_o \mathbf{h}(T) + \mathbf{b}_o). \quad (13)$$

G. Joint Alternating Training Strategy

DAMG-LiquidIDS is trained through an alternating bi-level strategy. During standard epochs, the feature gate is fixed and the LNN parameters are updated by minimizing the training loss. After a warm-up period and every K epochs, the LNN is temporarily frozen while PSO updates the gate using the validation objective in (6). The updated gate is then reinjected into the training loop. This procedure allows feature adaptation and temporal learning to co-evolve within a single training process rather than being optimized in two separate stages.

H. Bi-Level Formulation

The proposed framework can be summarized as

Inner problem (LNN learning):

$$\theta^*(\mathbf{z}) = \arg \min_{\theta} \mathcal{L}_{\text{train}}(\theta, \mathbf{z}), \quad (14)$$

Outer problem (gate adaptation):

$$\mathbf{z}^* = \arg \min_{\mathbf{z} \in [0,1]^d} \left[\mathcal{L}_{\text{val}}(\theta^*(\mathbf{z}), \mathbf{z}) + \Omega(\mathbf{z}, \hat{D}_w) \right]. \quad (15)$$

a) *Computational complexity.*: The extra cost of DAMG-LiquidIDS comes from drift estimation and periodic PSO gate updates. For a window of size n_w and feature dimension d , drift estimation costs $O(n_w d^2)$ due to the covariance term. Each PSO update costs $O(N_p RC_{\text{val}})$, where C_{val} denotes one validation pass. Since PSO is executed only every K epochs after warm-up and d remains moderate in our datasets, the overhead mainly affects training and remains bounded.

IV. EXPERIMENTAL SETUP

A. Datasets

Experiments are conducted on CICIoT2023 [12] and FiveG-NIDD [13], two recent IDS benchmarks covering heterogeneous IoT traffic and realistic 5G network traffic, respectively. CICIoT2023 includes benign flows and multiple attack families such as DDoS, DoS, Mirai, reconnaissance, and web-based attacks, while FiveG-NIDD contains attacks targeting next-generation mobile infrastructures.

For both datasets, the task is formulated as binary classification (benign vs. malicious). Attack-family identifiers and metadata are removed from the input features to avoid information leakage and are used only to define the unseen-attack protocol in Section IV-C.

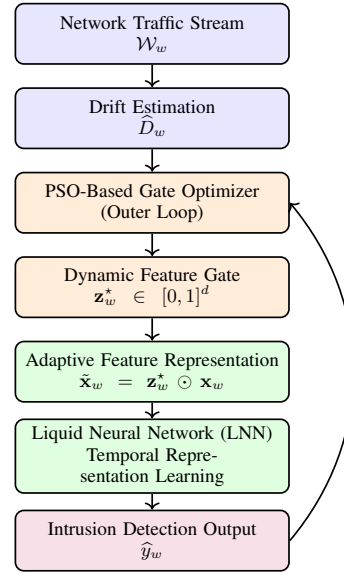


Fig. 1: DAMG-LiquidIDS architecture: drift-aware PSO-based feature gating coupled with a Liquid Neural Network for adaptive intrusion detection.

B. Preprocessing

A unified preprocessing pipeline is applied to all compared methods. Samples containing missing, invalid, or infinite values are discarded. Numerical features are standardized using statistics computed exclusively from the training split. Categorical variables, when present, are one-hot encoded. The fitted preprocessing transformations are then applied without refitting to the validation and test splits.

To avoid data leakage, all dimensionality-reduction and filter-based feature-selection methods are fitted only on the training split. Specifically, PCA, Chi-square, and Mutual Information selectors are trained exclusively on the training data and then applied to the validation and test data. Attack-family labels are never used as input features and are reserved only for defining the unseen-attack evaluation scenario.

C. Evaluation Protocols

All methods are evaluated under four inference-only protocols, with no test-time retraining or label feedback. The **temporal** protocol uses chronological train/validation/test splits to assess generalization to future traffic. The **prior-drift** protocol divides the test stream into $T = 10$ chunks and reorders them to induce a progressive class-prior shift, with a KL divergence greater than 0.4 between the first and last chunk label distributions. Since the same test samples are reused, global Macro-F1 can remain unchanged after reordering; thus, this protocol is interpreted as an offline robustness indicator under reordered label-prior conditions, not as online chunk-wise adaptation.

The **covariate-drift** protocol perturbs a random 20% of test features using Gaussian noise with severity 0.15 while preserving labels. The **unseen-attack** protocol holds out attack

families from training/validation and evaluates them only at test time: Mirai-greeth_flood and Recon-PortScan for CICIOT2023, and SYNScan and UDPScan for FiveG-NIDD. All held-out attacks are mapped to the malicious class.

D. Compared Methods

All methods use the same LNN backbone, optimizer, and training hyperparameters, and differ only in the feature-selection strategy. This controlled design isolates the effect of feature selection and gating from the classifier architecture. We compare **LNN-Full**, which uses all features; **PCA-LNN**, **Chi²-LNN**, and **MI-LNN**, where PCA, Chi-square, and Mutual Information are fitted on the training split; **StaticPSO-LNN**, where PSO is run once before training to obtain a fixed feature subset; and **DAMG-LiquidIDS**, where the feature gate is dynamically updated during training through the proposed drift-aware PSO outer loop.

E. Liquid Neural Network Implementation

Flows are chronologically sorted and grouped into sliding causal windows of length $S = 8$, where each window is labeled by its last flow. The time index is normalized to $[0, 1]$, and hidden states are zero-initialized for each window without being carried across windows.

Given the gated input $\tilde{x}_k = z \odot x_k$, the LNN hidden state is modeled by

$$\dot{h}(t) = -\tau^{-1} \odot h(t) + \phi_\theta(h(t), \tilde{x}(t), t), \quad (16)$$

where τ contains learnable positive time constants and ϕ_θ is implemented as a one-hidden-layer MLP with hyperbolic tangent activation. Positivity of τ is enforced using a softplus parametrization.

The dynamics are integrated using an explicit Euler scheme with $\Delta t = 0.1$:

$$h_{k+1} = h_k + \Delta t (-\tau^{-1} \odot h_k + \phi_\theta(h_k, \tilde{x}_k, t_k)). \quad (17)$$

The last hidden state h_S is passed to a linear layer followed by a softmax classifier to produce the intrusion-detection output.

F. Training Configuration

Table I summarizes the main training and optimization settings. All models are trained using weighted cross-entropy to mitigate class imbalance. Early stopping is applied based on validation Macro-F1 with a patience of 10 epochs.

For DAMG-LiquidIDS, the LNN parameters and the feature gate are updated in an alternating manner. During ordinary epochs, the gate is frozen and the LNN parameters are optimized using backpropagation. After a warm-up period of 5 epochs, the LNN parameters are temporarily frozen every $K = 3$ epochs, and PSO is used to update the feature gate according to the validation objective. The updated gate is then injected back into the LNN training loop.

TABLE I: Training and optimization settings.

Category	Configuration
LNN	Batch=256, $S = 8$, $H = 256$, $\Delta t = 0.1$, LR= 8×10^{-4} , weight decay= 5×10^{-5} , 50 epochs, patience=10, weighted cross-entropy, zero hidden-state initialization.
PSO	20 particles, 20 iterations, $\omega \in [0.4, 0.9]$, $c_1 = c_2 = 1.5$, warm-up=5 epochs, update period $K = 3$.
Regularization	$\lambda_{sp} = 5 \times 10^{-3}$, $\lambda_{cost} = 10^{-2}$, $\lambda_{st} = 10^{-2}$.
Drift	Window=2048, $\alpha_D = 2.5$, $\delta_D = 0.4$, prior-drift KL threshold=0.4, $T = 10$ chunks, covariate severity=0.15, perturbed features=20%.
Evaluation	10 runs with seeds 42–51, mean $\pm$ std, Wilcoxon signed-rank test with Holm–Bonferroni correction.

G. Performance Metrics

We use Macro-F1 as the primary metric because it gives equal importance to benign and malicious classes in imbalanced IDS settings. Precision, Recall, and False Alarm Rate (FAR) are also reported, with

$$\text{FAR} = \frac{FP}{FP + TN}, \quad (18)$$

where FP and TN denote false positives and true negatives.

For the prior-drift protocol, global Macro-F1 is reported for cross-method comparability. Since this protocol reorders the same test samples, the global score is interpreted as an offline robustness indicator under reordered label-prior conditions, not as evidence of online chunk-wise adaptation.

Statistical significance is assessed over 10 independent runs using a paired one-sided Wilcoxon signed-rank test, with DAMG-LiquidIDS compared against each baseline. Holm–Bonferroni correction is applied independently for each dataset and protocol over the five baseline comparisons.

V. EXPERIMENTAL RESULTS

A. Overall Performance Comparison

Figure 2 and Table II summarize the global Macro-F1 performance across the two datasets and the four evaluation protocols. Overall, DAMG-LiquidIDS achieves the best average rank among the compared methods. On CICIOT2023, the proposed method remains highly competitive with LNN-Full and StaticPSO-LNN across all protocols, achieving Macro-F1 values in the range of 0.881–0.914. On FiveG-NIDD, the benefits of the proposed dynamic feature-gating mechanism are more pronounced, reaching Macro-F1 values up to 0.980 ± 0.042 under temporal and prior-drift settings.

It is worth noting that the global Macro-F1 scores of the temporal and prior-drift protocols can be identical when prior drift is simulated by reordering the same test samples and the model is evaluated offline. Therefore, the prior-drift results in Table II should be interpreted as global robustness indicators under reordered label-prior conditions, rather than as evidence of online chunk-wise adaptation.

These results highlight that adaptive feature selection provides marginal benefits in relatively stable environments, while

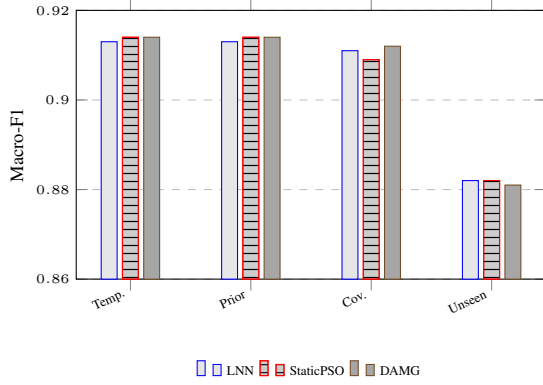


Fig. 2: Macro-F1 comparison on CICIoT2023.

becoming highly effective under stronger distribution shifts typical of real-world 5G IoT traffic.

B. Ablation Study

We evaluate three ablated variants to assess the contribution of the main components of DAMG-LiquidIDS: **DAMG-NoDrift**, which removes the drift-conditioned term; **DAMG-NoPSO**, which disables the PSO-based dynamic gate search; and **DAMG-NoSparsity**, which removes the sparsity regularization. All other settings are kept unchanged.

Table III shows that the full DAMG-LiquidIDS consistently obtains the best Macro-F1. On CICIoT2023, disabling the PSO-based gate search causes the strongest degradation, especially under prior drift, covariate drift, and unseen attacks. On FiveG-NIDD, removing sparsity is the most harmful, reducing the average Macro-F1 from about 0.976 to 0.945. These results confirm that PSO-based adaptation, drift conditioning, and sparsity regularization all contribute to the final robustness, with dataset-dependent effects.

C. Drift Robustness and Feature Adaptation

Table II shows that DAMG-LiquidIDS is especially robust on FiveG-NIDD, achieving Macro-F1 scores of 0.980, 0.980, and 0.975 under temporal, prior-drift, and covariate-drift protocols, respectively, compared with 0.654 for LNN-Full. This corresponds to gains above 32 percentage points in the most drift-sensitive settings. On CICIoT2023, where shifts are milder, DAMG-LiquidIDS remains stable with Macro-F1 values around 0.912–0.914.

Figure 3 illustrates representative gate dynamics under covariate drift. On FiveG-NIDD, the gate reduces the active features from 91 to 32, i.e., about 64.8% reduction. On CICIoT2023, it stabilizes at 37 active features out of 46, corresponding to about 20% reduction. These results show that DAMG-LiquidIDS adapts feature usage according to dataset complexity and drift intensity while preserving detection performance.

D. Statistical Analysis

Statistical significance is evaluated over 10 runs using paired one-sided Wilcoxon signed-rank tests with Holm–Bonferroni

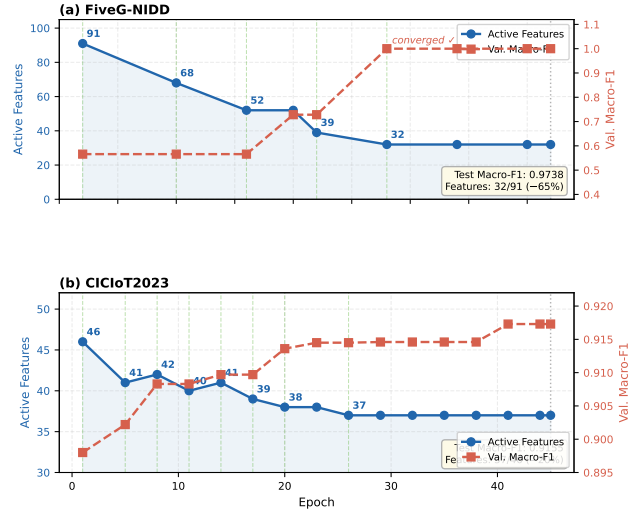


Fig. 3: Feature-gate dynamics and validation Macro-F1.

correction. As shown in Fig. 4, superiority is claimed only when $p < 0.05$ and DAMG-LiquidIDS also obtains a higher mean Macro-F1 in Table II. DAMG-LiquidIDS significantly improves over most feature-selection baselines, especially under temporal, prior-drift, and covariate-drift settings on FiveG-NIDD. Differences with LNN-Full and StaticPSO-LNN are generally not significant on CICIoT2023, where all strong LNN-based methods achieve close performance.

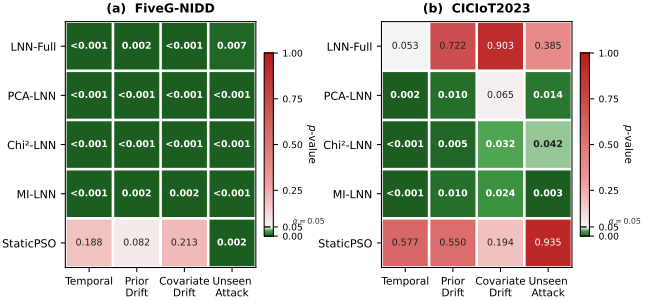


Fig. 4: Wilcoxon signed-rank p-values comparing DAMG-LiquidIDS against each baseline across datasets and evaluation protocols.

VI. CONCLUSION

This paper presented DAMG-LiquidIDS, a drift-aware IDS framework that combines continuous feature gating, PSO-based adaptation, and Liquid Neural Networks. The proposed alternating strategy enables the feature representation and temporal classifier to co-adapt during training under evolving network conditions. Experiments on CICIoT2023 and FiveG-NIDD show that DAMG-LiquidIDS achieves strong robustness under temporal, reordered prior-drift, covariate-drift, and unseen-attack protocols. The gains are particularly clear on FiveG-NIDD, while on CICIoT2023 the method remains competitive with strong LNN-based baselines. Moreover, the

TABLE II: Global Macro-F1 performance comparison over 10 independent runs (mean $\pm$ std).

Dataset	Protocol	LNN-Full	PCA-LNN	Chi ² -LNN	MI-LNN	StaticPSO-LNN	DAMG-LiquidIDS
CICIoT2023	Temporal	0.913 $\pm$ 4e ⁻³	0.896 $\pm$ 3e ⁻³	0.903 $\pm$ 2e ⁻³	0.860 $\pm$ 6e ⁻³	0.914$\pm$2e⁻³	0.914$\pm$3e⁻³
	Prior drift	0.913 $\pm$ 4e ⁻³	0.896 $\pm$ 3e ⁻³	0.903 $\pm$ 2e ⁻³	0.860 $\pm$ 6e ⁻³	0.914$\pm$2e⁻³	0.914$\pm$3e⁻³
	Covariate drift	0.911 $\pm$ 4e ⁻³	0.896 $\pm$ 3e ⁻³	0.889 $\pm$ 1.6e ⁻²	0.860 $\pm$ 6e ⁻³	0.909 $\pm$ 8e ⁻³	0.912$\pm$5e⁻³
	Unseen attack	0.882$\pm$3e⁻³	0.868 $\pm$ 4e ⁻³	0.871 $\pm$ 2e ⁻³	0.831 $\pm$ 3e ⁻³	0.882$\pm$3e⁻³	0.881 $\pm$ 4e ⁻³
FiveG-NIDD	Temporal	0.654 $\pm$ 1.8e ⁻²	0.649 $\pm$ 2.0e ⁻²	0.648 $\pm$ 2.0e ⁻²	0.614 $\pm$ 4e ⁻³	0.878 $\pm$ 1.49e ⁻¹	0.980$\pm$4.2e⁻²
	Prior drift	0.654 $\pm$ 1.8e ⁻²	0.649 $\pm$ 2.0e ⁻²	0.648 $\pm$ 2.0e ⁻²	0.614 $\pm$ 4e ⁻³	0.878 $\pm$ 1.49e ⁻¹	0.980$\pm$4.2e⁻²
	Covariate drift	0.654 $\pm$ 1.9e ⁻²	0.648 $\pm$ 1.9e ⁻²	0.645 $\pm$ 1.8e ⁻²	0.531 $\pm$ 6.3e ⁻²	0.877 $\pm$ 1.48e ⁻¹	0.975$\pm$5.1e⁻²
	Unseen attack	0.968 $\pm$ 2.2e ⁻²	0.889 $\pm$ 4.7e ⁻²	0.832 $\pm$ 6e ⁻³	0.829 $\pm$ 3.6e ⁻²	0.970$\pm$1.6e⁻²	0.968 $\pm$ 2.2e ⁻²
Avg. Rank		3.83	5.00	4.33	6.17	2.33	1.33

TABLE III: Ablation study in terms of Macro-F1.

Dataset	Variant	Temp.	Prior	Cov.	Unseen
CICIoT2023	NoDrift	0.8945	0.8902	0.8834	0.8712
	NoPSO	0.8837	0.8424	0.8412	0.8061
	NoSparsity	0.8621	0.8531	0.8607	0.8363
	DAMG	0.9140	0.9140	0.9120	0.8810
FiveG-NIDD	NoDrift	0.9702	0.9698	0.9630	0.9584
	NoPSO	0.9725	0.9761	0.9703	0.9460
	NoSparsity	0.9473	0.9480	0.9461	0.9382
	DAMG	0.9800	0.9800	0.9750	0.9680

adaptive gate reduces the number of active features, improving feature efficiency without degrading detection performance.

ACKNOWLEDGMENT

This research was partially supported by grant PID2021-123627OB-C55 and by grant PID2024-158682OB-C33 funded by MCIN/AEI/10.13039/501100011033 and by "ERDF A way of making Europe".

REFERENCES

- [1] F. Adjewa, M. Essegghir, and L. Merghem-Boulahia, "Efficient federated intrusion detection in 5g ecosystem using optimized BERT-based model," in *Proceedings of the 20th IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*. Paris, France: IEEE, 2024, pp. 62–67.
- [2] A. Mohammadi Bagha, I. Woungang, I. Traore, and D. B. Rawat, "A graph clustering-based network anomaly detection system," in *Proceedings of the 20th IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*. Paris, France: IEEE, 2024.
- [3] O. A. Wahab, "Intrusion detection in the IoT under data and concept drifts: Online deep learning approach," *IEEE Internet of Things Journal*, vol. 9, no. 20, pp. 19706–19716, 2022.
- [4] R. Chu, P. Jin, H. Qiao, and Q. Feng, "Intrusion detection in IoT data streams using concept drift localization," *AIMS Mathematics*, vol. 9, no. 1, pp. 1–22, 2024.
- [5] M. H. Bhuyan, D. Bhattacharyya, and J. Kalita, "A survey on intrusion detection and prevention systems in IoT," *IEEE Communications Surveys & Tutorials*, vol. 25, no. 2, pp. 1048–1091, 2023.
- [6] G. Andresini, A. Appice, and D. Malerba, "Evolving cybersecurity frontiers: A comprehensive survey on concept drift and feature dynamics aware machine and deep learning in intrusion detection systems," *Engineering Applications of Artificial Intelligence*, vol. 137, p. 109143, 2024.
- [7] G. Quadrio, R. Pompa, E. Bassetti, C. D'Elia, G. Scacco, and M. Conti, "they will adapt": Improving anomaly detection on IoT networks through continuous learning," in *Proceedings of the 20th IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*. Paris, France: IEEE, 2024.
- [8] R. Gao, H. Lu, H. Zhou, C. Zheng, Z. Tian, and X. Du, "Active and passive attack detection methods for malicious encrypted traffic," in *Proceedings of the 20th IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*. Paris, France: IEEE, 2024.
- [9] R. Hasani, M. Lechner, A. Amini, D. Rus, and R. Grosu, "Liquid time-constant networks," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 9, pp. 7657–7666, 2021.
- [10] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*. IEEE, 1995, pp. 1942–1948.
- [11] R. T. Chen, Y. Rubanova, J. Bettencourt, and D. Duvenaud, "Neural ordinary differential equations," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 31, 2018.
- [12] E. C. P. Neto, S. Dadkhah, R. Ferreira, A. Zohourian, R. Lu, and A. A. Ghorbani, "CICIoT2023: A real-time dataset and benchmark for large-scale attacks in IoT environment," *Sensors*, vol. 23, no. 13, p. 5941, 2023.
- [13] S. Samarakoon, L. Laaksonen, T. Saarinen, and M. Yli-Ojanperä, "5G-NIDD: A comprehensive network intrusion detection dataset generated over 5g wireless network," 2022, arXiv preprint arXiv:2212.01298.
- [14] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *Proceedings of the 4th International Conference on Information Systems Security and Privacy (ICISSP)*, 2018, pp. 108–116.
- [15] N. Moustafa and J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems," in *2015 Military Communications and Information Systems Conference (MilCIS)*. IEEE, 2015, pp. 1–6.
- [16] J. H. Holland, *Adaptation in Natural and Artificial Systems*. University of Michigan Press, 1975.
- [17] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Advances in Engineering Software*, vol. 69, pp. 46–61, 2014.
- [18] E. Selem, M. Frikha, and M. Hamdi, "Metaheuristic-based intrusion detection systems: A survey and taxonomy," *Journal of Network and Computer Applications*, vol. 220, p. 103760, 2023.
- [19] J. Kim, J. Kim, H. L. T. Thu, and H. Kim, "CNN-based network intrusion detection against denial-of-service attacks," *Electronics*, vol. 9, no. 6, p. 916, 2020.
- [20] M. M. Lopez Perez, F. Adamsky, R. Souza, and T. Engel, "Network intrusion detection system based on recurrent neural networks," in *2017 IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA)*. IEEE, 2017, pp. 1–8.
- [21] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, 2017.
- [22] H. Liu and R. Setiono, "Chi2: Feature selection and discretization of numeric attributes," in *Proceedings of the 7th IEEE International Conference on Tools with Artificial Intelligence*. IEEE, 1995, pp. 388–391.
- [23] A. Kraskov, H. Stögbauer, and P. Grassberger, "Estimating mutual information," *Physical Review E*, vol. 69, no. 6, p. 066138, 2004.
- [24] I. T. Jolliffe, *Principal Component Analysis*, 2nd ed. New York: Springer, 2002.