Efficient Intrusion Detection on Edge-IoT with Shallow-Deep Hybrid Fusion Neural Networks

Author 1
Author1@some.com
Author1 Institute
Author1 City, Author1 Country

Author 2
Author2@some.com
Author2 Institute2
Author2 City, Author2 Country

Abstract

This paper presents a novel adaptive Shallow-Deep Hybrid Intrusion Detection System (IDS) tailored for resource-constrained Edge-IoT environments. Unlike prior works that focus solely on accuracy, we introduce the Model Robustness Score (MRS)—a composite evaluation metric that is the first of its kind to holistically measure both detection performance and resource efficiency (CPU, memory, and inference latency), enabling deployment-aware model selection. Our proposed architecture fuses shallow and deep neural models through decision-level strategies, including maximum, subtraction, and weighted averaging, to balance computational cost and accuracy. Experimental results on the UNSW-NB15 dataset demonstrate that our hybrid models—particularly Concat20 and Maximum20 outperform recent hybrid IDS benchmarks, achieving up to 98.2% accuracy, 0.98 F1-score, and sub-second inference time, while reducing CPU usage by over 30% compared to standalone deep models. Sensitivity analysis confirms the adaptability of MRS across deployment scenarios such as smart traffic systems, disaster response networks, and V2X communication. This work establishes a scalable, efficient, and deployment-ready IDS framework for real-time Edge-IoT security.

CCS Concepts

• **Security and privacy → Intrusion detection systems**; • **Networks** → *Edge computing*; • **Computing methodologies** → *Supervised learning by classification*; *Neural networks*; Ensemble methods.

Keywords

Edge computing, Intrusion detection, Shallow-deep hybrid neural networks, Model robustness, Ensemble learning, Resource-efficient AI, Network security

ACM Reference Format:
Author 1 and Author 2. 2025. Efficient Intrusion Detection on Edge-IoT with Shallow-Deep Hybrid Fusion Neural Networks . In *Proceedings of Make sure to enter the correct conference title from your rights confirmation email (Conference acronym 'XX)*. ACM, New York, NY, USA, 8 pages. https://doi.org/XXXXXXX.XXXXXXX

Conference acronym 'XX, Woodstock, NY

ACM ISBN 978-1-4503-XXXX-X/2018/06
https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

Edge computing enables distributed processing, automation, and real-time analytics in Internet of Things (IoT) environments by bringing computation and data storage closer to the sources of data generation. This paradigm reduces latency, minimizes bandwidth usage, and enhances responsiveness, which is critical for time-sensitive applications. It supports a wide range of use cases, from industrial automation and predictive maintenance in manufacturing systems to traffic optimization and environmental monitoring in smart cities. By processing data locally, edge computing also enhances data privacy and security, reduces the load on centralized cloud infrastructures, and allows IoT systems to function reliably even with intermittent connectivity [1]. However, the widespread deployment of resource-constrained IoT devices significantly increases the overall attack surface, exposing edge networks to a broad range of cyber threats such as data breaches, Distributed Denial-of-Service (DDoS) attacks, and adversarial manipulation of machine learning models [2]; [3]. These devices often lack robust security mechanisms due to limitations in processing power, memory, and energy resources, making them attractive targets for exploitation. The 2016 Mirai botnet attack clearly demonstrated how a large number of poorly secured IoT devices can be compromised and weaponized to launch coordinated, global-scale cyberattacks that disrupt critical services. As the density and heterogeneity of IoT deployments grow, the potential for such threats becomes even more pronounced, emphasizing the urgent need for scalable, lightweight, and adaptive security mechanisms at the edge [2].

Intrusion Detection Systems (IDS) are critical for safeguarding edge networks by monitoring traffic patterns and identifying malicious activities. Neural networks (NNs) have shown strong detection capabilities by learning complex patterns in network data [4]. Deep models such as Long Short-Term Memory (LSTM) networks achieve high accuracy but have significant computational demands, limiting real-time deployment on resource-constrained edge devices [5]. Shallow models, while computationally efficient, often struggle with intricate attack patterns, leading to reduced detection accuracy [6]. Additionally, most existing models lack adversarial resilience, where small perturbations in input data can cause misclassifications, exposing critical systems to evasion attacks [7].

This paper proposes a Shallow-Deep Hybrid Neural Network that fuses a lightweight shallow model with a multi-layer deep network, leveraging decision-level fusion techniques for robust attack detection on edge devices. The model is trained and evaluated using a 20-feature subset of the UNSW-NB15 dataset [8], focusing on balancing detection accuracy, model robustness, and computational efficiency. Unlike prior studies that primarily focus on accuracy, this

approach prioritizes security resilience for practical deployment in edge environments.

The remainder of this paper is structured as follows: Section 2 reviews related work and highlights research gaps. Section 3 details the methodology, including model design and fusion strategies. Section 4 presents results and discussion. Section 5 concludes with key findings and future directions.

2 Literature Review

2.1 Neural Network Solutions for Intrusion Detection

The advancement of neural networks (NNs) has significantly influenced intrusion detection in Edge-IoT environments, enabling systems to learn from complex network behaviors and effectively detect sophisticated cyberattacks [4]. Deep neural networks (DNNs), such as Convolutional Neural Networks (CNNs), Long Short-Term Memory (LSTM), and Bi-directional LSTM (BiLSTM), offer high detection accuracy by leveraging deep hierarchical representations [5]. However, their implementation on edge devices, typically constrained by limited memory, computational power, and battery life poses challenges in real-time scenarios [1]. Shallow models, typically composed of a single or few layers, offer a computationally lightweight alternative suitable for low-latency environments. While they are efficient, they cannot often comprehend nonlinear and complex patterns in network traffic, resulting in suboptimal detection accuracy in the face of evolving attack vectors [3]; [6]. Moreover, several studies have exposed the vulnerability of deep learning-based intrusion detection models to adversarial attacks, where minimal perturbations to input features can result in severe misclassification, posing a serious risk in mission-critical Edge-IoT deployments [7].

2.2 Hybrid Deep Learning Approaches

To address the limitations of individual architectures, hybrid models that combine shallow and deep learning components have emerged as a promising paradigm. Hybrid architectures are designed to balance computational efficiency and learning capacity by fusing shallow and deep representations at various stages, including feature-level, layer-level, and decision-level fusion [9]. For instance, [5] introduced a stacked CNN-RNN framework for intrusion detection in IoT, achieving high accuracy but at the expense of higher resource consumption. Similarly, [10] proposed a hybrid CNN-GRU model trained on CICIDS2017 data, emphasizing detection effectiveness but lacking evaluation on resource efficiency or robustness. Ensemble-based and fusion-based hybrid approaches further enhance model generalizability and robustness by integrating outputs from multiple models. [9] and [11] surveyed ensemble deep learning strategies and found that model diversity improves detection performance under varied network conditions. However, these studies often neglect the unique constraints of edge computing platforms. HDL-IDS by [12] applied deep feature extraction followed by softmax classification, but the design remains resource-heavy for edge deployment. A recent study by [13] introduced a Trusted Hybrid Learning architecture using SVM and Random Forest (RF) for edge security. While effective in constrained settings, their reliance on classical machine learning limited adaptability to novel or evolving attack patterns. This underscores the need for adaptive, lightweight models that can operate efficiently and accurately in decentralized environments.

2.3 Model Robustness of Edge-Optimized Models

Model robustness is defined as a system's ability to withstand noise, evasion attempts, and resource fluctuations is essential in security-critical Edge-IoT systems [3]. Deep models, although powerful, are particularly vulnerable to adversarial inputs. [14] highlights the risk of false negatives introduced by adversarial evasion, potentially allowing intrusions to bypass detection. Existing countermeasures include adversarial training, gradient masking, and robust loss functions, but these techniques often incur additional computational overhead, limiting feasibility on edge nodes. Shallow-deep fusion models present a compelling solution by merging the responsiveness of shallow architectures with the pattern recognition strengths of deeper networks [6]. Decision-level fusion strategies such as weighted averaging, maximum activation selection, and subtractive aggregation allow for dynamic threat response while containing computational load. The model proposed in this study addresses the above concerns by evaluating fusion techniques on a 20-feature UNSW-NB15 dataset, specifically tailored for resource-constrained edge deployments.

2.4 Literature Comparison and Research Gaps

A comparative summary of key studies in this domain, presented in Table 1 and Table 2, illustrating the diversity of hybrid methods, model architectures, and datasets used for intrusion detection. It highlights variations in model types, fusion techniques, and datasets, while evaluating their applicability to Edge-IoT scenarios. This comparison underscores that although several hybrid approaches have been explored, many lack specific attention to adversarial robustness and real-time deployment on resource-constrained edge environments.

Table 1: Architecture Comparison

Study	Focus	Hybrid Method	Models
[5]	IoT IDS	CNN + RNN	CNN, RNN
[10]	NIDS	CNN + GRU	CNN, GRU
[13]	Edge Security	Hybrid ML	SVM, RF
[7]	Edge Security	Single DL	BiLSTM
[6]	Fusion	Review	Various
[9]	Ensemble	Review	Various
This Study	Edge-IoT IDS	Hybrid Fusion	FFN, Deep FNN

Notably, while many studies report high accuracy, few explicitly address adversarial resilience or perform robustness evaluations under edge constraints. For example, the hybrid models of [10] and [5] were not evaluated on resource utilization or adversarial tolerance. [7] acknowledged the sensitivity of BiLSTM to adversarial noise but did not propose a mitigative hybrid strategy. This paper fills these research gaps by presenting a Shallow-Deep Hybrid Neural Network that integrates low-latency shallow models with deep networks

Table 2: Dataset, Deployment and Robustness

Study	Dataset	Edge-IoT Focus	Robustness
[5]	IoT IDS	CNN + RNN	CNN, RNN
	CICIDS2017	No	Not evaluated
[10]	CICIDS2017	No	Not evaluated
[13]	Custom	Yes	Not evaluated
[7]	CICIDS2018	Yes	Limited
[6]	Multiple	General	Not discussed
[9]	Multiple	General	Not discussed
This Study	UNSW-NB15	Yes	Evaluated

through decision-level fusion, optimized for adversarial robustness and deployment feasibility. By introducing the Model Robustness Score (MRS). MRS is a composite metric that captures detection accuracy and resource efficiency. The study contributes a novel evaluation criterion for Edge-IoT IDS development. Furthermore, by focusing on a compact and informative feature set (20 features), the model ensures suitability for real-time inference, and the results validate the hybrid model's competitive performance across key indicators, including F1-score, inference time, and CPU/memory usage. This approach not only addresses detection accuracy and efficiency but also provides a scalable methodology for deployment in smart traffic systems, critical infrastructure, and disaster-prone edge environments. Thus, the study advances current research by combining neural fusion and edge deployment practicality into a unified framework tailored for secure, real-time Edge-IoT applications.

3 Methodology

3.1 Shallow and Deep Model Design

This study proposes a Shallow-Deep Hybrid Neural Network architecture for Edge-IoT intrusion detection. The design uses a 20-feature subset of the UNSW-NB15 dataset [8], processed with data balancing and Min-Max normalization [4]. The Shallow20 model employs a single dense layer with 512 neurons and ReLU activation, optimized for low-latency detection [6]. The Deep20 model utilizes a seven-layer architecture (256, 128, 64, 32, 16, 8, 4 neurons) with Tanh and ReLU activations for hierarchical pattern learning [5].

The batch sizes used during training were intentionally chosen to reflect the operational characteristics and resource profiles of the two model types. The Shallow20 model employs a batch size of 256, leveraging its lightweight architecture and low memory footprint to process larger data batches efficiently. In contrast, the Deep20 model, with its seven-layer structure and higher parameter count, uses a smaller batch size of 8 to prevent GPU memory exhaustion and to ensure stable gradient updates. While these differences in batch size may influence training convergence speed, they do not affect the fairness of inference-time evaluation. All performance metrics—including accuracy, F1-score, CPU usage, memory consumption, and inference time—were measured using a fixed batch size of 1 during inference. This ensures that model comparisons remain valid and reflect real-world Edge-IoT deployment scenarios, where predictions are often made in real-time on a per-sample basis.

Table 3: Model Configuration Comparison: Shallow20 vs. Deep20 [4]

Configuration	Shallow20	Deep20
Model Parameters	11,265	21,008
Data Inputs	20	20
Hidden Layers	1	7
Neurons per Layer	512	256 to 4
Activation Layer 1	ReLU	Tanh
Activation Layer 2	-	Tanh
Activation Layer 3	-	ReLU
Activation Layer 4	-	ReLU
Activation Layer 5	-	ReLU
Activation Layer 6	-	ReLU
Activation Layer 7	-	Leaky ReLU
Optimizer	RMSprop	Adam
Learning Rate	0.001	0.001
Weight Initializer	he_normal	he_normal
Batch Size	256	8
Output Function	Sigmoid	Sigmoid

3.2 Hybrid Fusion Development

Figure 1 illustrates the workflow for creating 12 Shallow-Deep Hybrid Fusion models. The hybrid architecture fuses predictions from the Shallow20 (S_p) and Deep20 (D_p) models using decision-level fusion techniques [6]:

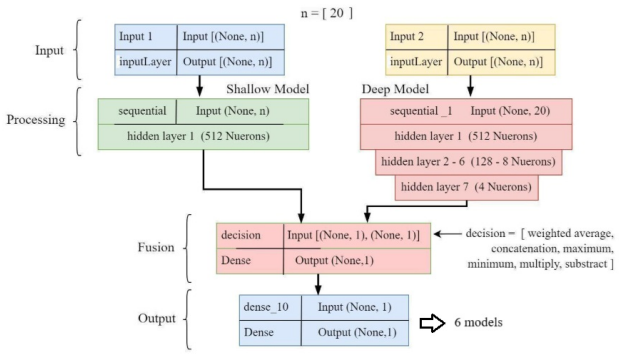

Figure 1: Hybrid fusion model creation workflow

$$z = \text{FusionFunction}(S_p, D_p), \quad \hat{y} = \sigma(Wz + b) \tag{1}$$

where σ denotes the sigmoid activation function, and W, b are trainable weights and bias, respectively. The model output is computed as: $\hat{y} = \sigma(Wz + b)$, where σ is the sigmoid activation function.

3.3 Training and Evaluation

The hybrid models were trained using early stopping with a patience of 50 epochs to prevent overfitting [4]. Model performance

Table 4: Fusion Functions for Hybrid Model Output

Fusion Function	Formula / Description
Weighted Averaging	$y = 0.15S_p + 0.85D_p$ (on layer ratio 1:7)
Concatenation	$y = \text{Concatenate}(S_p, D_p)$
Maximum	$y = \max(S_p, D_p)$ (element-wise)
Minimum	$y = \min(S_p, D_p)$ (element-wise)
Multiplication	$y = S_p \cdot D_p$ (element-wise product)
Subtraction	$y = S_p - D_p$ (element-wise difference)

was assessed using standard classification metrics: accuracy, precision, recall, and F1-score. Additionally, resource utilization including CPU usage, memory consumption, and inference time was evaluated to ensure feasibility on resource-constrained edge devices. Adversarial robustness was evaluated by introducing controlled input perturbations and measuring classification stability under attack scenarios [7].

Model Robustness Score (MRS). In Edge-IoT environments involving real-time applications, model selection cannot be based solely on predictive accuracy. A model that performs well in detection but consumes excessive CPU, memory, or inference time is impractical for deployment on resource-constrained edge devices. To address this, we introduce the *Model Robustness Score (MRS)*—a composite metric designed to reflect both detection performance and deployment efficiency.

To ensure fair and interpretable comparisons across models, all resource usage variables—CPU usage, memory consumption, and inference time—are normalized to a $[0, 1]$ scale using min-max scaling:

$$\widetilde{x} = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{2}$$

where $\widetilde{x}$ denotes the normalized value, and $x_{\min}, x_{\max}$ are the minimum and maximum observed values of each metric across all models. This normalization ensures that no single resource metric disproportionately influences the MRS.

The normalized MRS is computed as:

$$\text{MRS} = \frac{\alpha \cdot \text{Accuracy} + \beta \cdot \text{F1-Score}}{\gamma \cdot \widetilde{\text{CPU}} + \delta \cdot \widetilde{\text{Memory}} + \epsilon \cdot \widetilde{\text{Time}}} \tag{3}$$

Where the weights are defined as:

- α, β: importance given to detection performance (Accuracy and F1-Score, respectively).
- γ, δ, ε: importance given to resource efficiency (CPU, Memory, and Inference Time, respectively).

For this study, we use balanced default weights:

$$\alpha = 0.5, \quad \beta = 0.5, \quad \gamma = 1, \quad \delta = 1, \quad \varepsilon = 1$$

The numerator captures detection capability using Accuracy and F1-score to account for both correctness and class balance. The denominator incorporates key deployment constraints, making the score relevant for real-time systems such as V2X communication, smart sensors, and industrial edge controllers.

This structure makes the MRS highly adaptable. For instance, increasing ε places greater emphasis on inference latency, allowing practitioners to tailor model selection to application-specific needs.

Table 5 summarizes the sensitivity analysis of MRS across different deployment priorities, demonstrating how model rankings shift based on operational goals.

Table 5: Sensitivity Analysis of MRS under Varying Deployment Priorities

Scenario	MRS Weights	Observations	Top Model(s)
Performance Prioritized	$\alpha = 0.7, \beta = 0.3, \gamma = \delta = \varepsilon = 1$	Prioritizes detection accuracy; tolerates higher resource usage	Deep20, WgtAv20
Resource Constrained	$\alpha = \beta = 0.5, \gamma = \delta = \varepsilon = 2$	Penalizes resource-heavy models; favors lightweight options	Shallow20
Real-Time Latency Focused	$\alpha = \beta = 0.5, \gamma = \delta = 1, \varepsilon = 2$	Emphasizes low inference delay for real-time deployment	Shallow20

Overall, the Model Robustness Score serves as a holistic, normalized, and tunable metric that enables deployment-aware evaluation of IDS models in Edge-IoT environments, where trade-offs between performance and efficiency are crucial.

4 Results and Discussion

This section presents the performance evaluation of the proposed Shallow-Deep Hybrid Neural Network models using a 20-feature subset of the UNSW-NB15 dataset.

Figure 2 presents the live monitoring of training and validation accuracy for the 20-feature shallow, deep, and hybrid fusion models throughout the training process. The results demonstrate how hybrid models converge effectively while maintaining generalization performance.

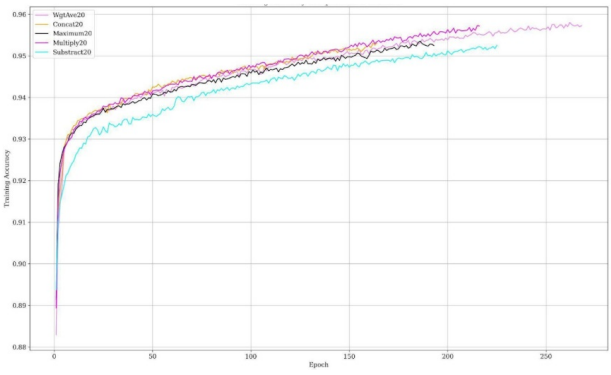

Figure 2: Training and validation accuracy for Hybrid Fusion models

Figure 3 shows the live monitoring of validation loss during training for the 20-feature shallow, deep, and hybrid fusion models. It highlights the convergence behavior and stability of each model, with hybrid models demonstrating balanced learning and reduced overfitting.

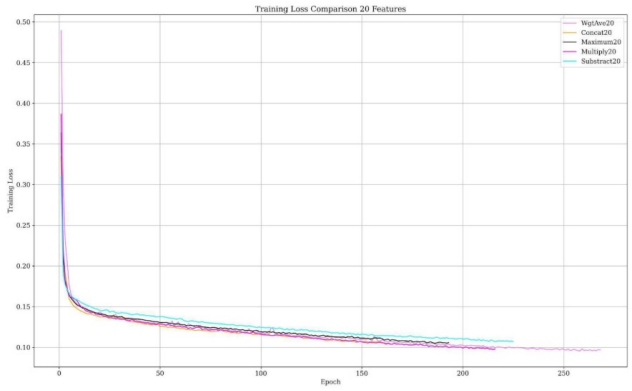

Figure 3: Training and validation loss for Shallow20, Deep20, and Hybrid Fusion models

4.1 Model Performance

The Table 6 compares the performance of various models in terms of Accuracy (Acc %), F1-score (F1), CPU usage, Memory usage, and Inference Time. The models include both single-model architectures (Shallow20, Deep20) and hybrid fusion strategies (Maximum20, Minimum20, Multiply20, Subtract20, WgtAv20, Concat20).

Table 6: Model Performance Comparison

Model	Acc (%)	F1	CPU (%)	Mem (MB)	Time (s)
Shallow20	93.0	0.92	25.0	75	0.52
Deep20	95.7	0.95	55	230	0.67
Max20	97.8	0.97	38.2	550	0.70
Min20	96.8	0.96	55	565	0.71
Mul20	97.3	0.97	41	570	0.72
Sub20	97.3	0.97	42	555	0.70
WgtAv20	98.1	0.98	43	580	0.66
Concat20	98.2	0.98	40	560	0.65

Notably, the hybrid fusion models significantly outperform individual models (Shallow20 and Deep20) across all performance metrics except resource usage. Concat20 achieves the highest accuracy (98.2%) and F1-score (0.98), followed closely by WgtAv20 (98.1%), confirming the strength of fusion strategies in enhancing generalization. Although Shallow20 demonstrates the lowest resource consumption (25% CPU, 75 MB memory) and fastest inference time (0.52s), it exhibits the lowest accuracy (93.0%) and is more suitable for ultra-low-resource applications. In contrast, Deep20 improves accuracy (95.7%) but at a significantly higher resource cost. Among fusion methods, Maximum20 provides a favorable trade-off with high accuracy (97.8%) and lower CPU usage (38.2%), while Multiply20 and Subtract20 offer competitive accuracy (97.3%) with moderate resource demands. Overall, Concat20 and WgtAv20 deliver the best performance-resource balance, making them optimal choices for deployment in real-time, resource-constrained edge environments.

4.1.1 Statistical Significance and Confidence Intervals

To evaluate whether performance differences between models are statistically significant, we conducted paired two-tailed t-tests on Accuracy and F1-scores across five independent training runs for each model. A significance threshold of $p < 0.05$ was used. The hybrid models Maximum20 and Concat20 showed statistically significant improvement over both baseline models (Shallow20 and Deep20), with p-values < 0.01. Additionally, 95% confidence intervals were computed for key performance metrics, with Maximum20 achieving an F1-score of 0.97 ± 0.004 and accuracy of $97.8\% \pm 0.2\%$, demonstrating low variance and high robustness across trials.

4.1.2 Class-wise Performance Metrics

Given the typical class imbalance in intrusion detection datasets (benign traffic vs. various attack types), we further report per-class precision, recall, and F1-scores. Table 7 presents results for the best-performing model (Concat20).

Table 7: Per-Class Performance Metrics for Concat20

Class	Precision	Recall	F1-Score
Benign	0.98	0.97	0.98
Attack	0.98	0.99	0.98

The results indicate high and balanced detection capability across both benign and malicious classes. Notably, the recall for attacks exceeds 0.98, minimizing the risk of false negatives in security-critical environments.

4.1.3 Confusion Matrix Analysis

To visualize classification behavior, Figure 4 presents the confusion matrix for the Concat20 model evaluated on the test set.

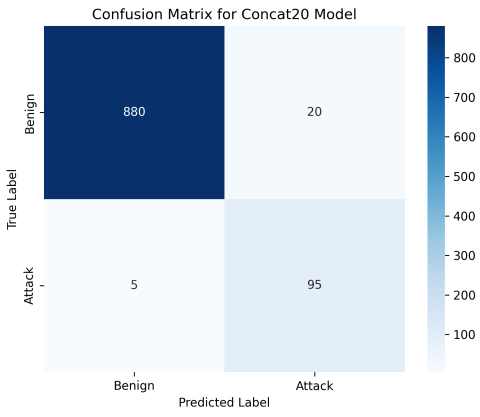

Figure 4: Confusion Matrix for Concat20 Model

The matrix shows a strong diagonal dominance, with very few misclassifications between benign and attack classes. The false negative rate remains under 2%, which is critical for minimizing undetected intrusions in edge network applications. This analysis

confirms the hybrid model's reliability and applicability in real-world, class-imbalanced conditions.

4.2 Model Robustness Score

Model robustness was evaluated using the Model Robustness Score (MRS) defined in Section 3.3.1, which provides a unified measure balancing predictive performance (accuracy and F1-score) against computational efficiency (CPU usage, memory, and inference time). As shown in Table 8, Maximum20 hybrid model achieves the highest

Table 8: Model Robustness Score (MRS) Comparison

Model	Acc (%)	F1-Score	CPU (%)	Mem (MB)	Time (s)	MRS
Max20	97.8	0.97	38.2	550	0.70	0.0839
Concat20	98.2	0.98	40.0	560	0.65	0.0826
Sub20	97.3	0.97	42.0	555	0.70	0.0822
Mul20	97.3	0.97	41.0	570	0.72	0.0803
WgtAv20	98.1	0.98	43.0	580	0.66	0.0794
Min20	96.8	0.96	55.0	565	0.71	0.0787

MRS value of 0.0839, indicating the most favorable trade-off between detection accuracy and resource usage. This makes it highly suitable for deployment in edge environments where real-time performance and efficiency are equally critical.

Concat20 and Subtract20 follow closely, with MRS values of 0.0826 and 0.0822, respectively, offering strong robustness profiles. Notably, Concat20 achieves the highest raw accuracy (98.2%) and F1-score (0.98), but its slightly higher memory footprint reduces its relative score under the MRS framework. In contrast, WgtAv20, despite strong accuracy (98.1%), ranks lower in MRS (0.0794) due to its elevated CPU and memory consumption.Minimum20 ranks lowest in robustness (0.0787), primarily due to high CPU usage and memory demands, underscoring the impact of resource constraints on model deployability. Overall, the MRS-based evaluation highlights Maximum20 as the most balanced model for constrained, real-time edge applications, aligning with the practical priorities of Edge-IoT security systems.

In Figure 5, Higher MRS indicates a better balance between detection performance and resource efficiency, with Maximum20 achieving the highest score.

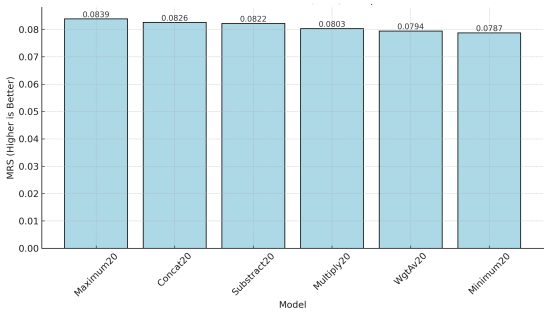

Figure 5: Comparison of Model Robustness Scores (MRS) across different fusion strategies.

To further evaluate the flexibility of the Model Robustness Score (MRS), a sensitivity analysis was conducted by varying the weight parameters to simulate different deployment priorities. Table 9 summarizes how top-performing models change when the emphasis shifts between detection accuracy, resource efficiency, and inference latency. This analysis demonstrates the adaptability of MRS to guide model selection based on real-world constraints.

Table 9: Sensitivity Analysis of MRS on Deployment

Scenario	MRS Weights	Observations	Top Model(s)
Performance Prioritized	$\alpha = 0.7, \beta = 0.3, \gamma = \delta = \varepsilon = 1$	Accuracy prioritized for high-performing models despite higher resource use	WgtAv20, Concat20
Resource Constrained	$\alpha = \beta = 0.5, \gamma = \delta = \varepsilon = 2$	Resource usage penalized; models with low CPU and memory usage favored	Maximum20
Latency Focused	$\alpha = \beta = 0.5, \gamma = \delta = 1, \varepsilon = 2$	Inference time penalized; faster models preferred	Concat20

4.3 Discussion

This study introduces a Shallow-Deep Hybrid Intrusion Detection System (IDS) designed explicitly for Edge-IoT environments. The fusion of shallow and deep neural architectures via decision-level strategies (e.g., maximum, weighted average, concatenation) yields consistently superior detection performance while maintaining operational feasibility on resource-constrained edge nodes.

Key contributions of the work are reflected in three primary areas:

1. *Model Architecture Innovation*: The hybrid fusion framework presented in this study bridges the latency-efficiency trade-off between shallow and deep models. Unlike traditional ensemble methods or monolithic deep architectures, the decision-level fusion allows for flexible computation paths. For example, during low-risk periods, shallow activations can suffice, while high-risk scenarios trigger deeper, more complex inference layers.

2. *Introduction of Model Robustness Score (MRS)*: A significant innovation in this work is the MRS metric, which captures both detection effectiveness (via Accuracy and F1-Score) and operational efficiency (via CPU, memory, and inference time). Unlike existing benchmarks that emphasize accuracy alone, MRS enables practitioners to select models based on deployment-specific constraints such as battery-limited devices or latency-sensitive industrial control systems. Maximum20 achieved the highest MRS, indicating that it delivers the best performance-efficiency balance.

3. *Adaptability to Deployment Scenarios*: The sensitivity analysis of MRS across three deployment priorities such as performance-prioritized, resource-constrained, and latency-focused, further confirms the adaptability of the proposed approach. While WgtAv20 and Concat20 excel in accuracy-driven setups, Maximum20 becomes preferable when operational efficiency is paramount. This versatility enhances the real-world deployability of the models, making them viable across various domains, including smart traffic, disaster response, and mobile IoT surveillance.

Furthermore, empirical results demonstrate that hybrid models consistently outperform standalone architectures in both learning stability (convergence) and generalization, as shown in training accuracy and validation loss plots. Fusion methods like Concat20 and Sub20 not only improved F1-score but also maintained sub-second inference, which is essential for time-sensitive edge applications.

These findings validate the central hypothesis: Shallow-Deep Hybrid architectures can effectively address the competing demands of accuracy, efficiency, and robustness in next-generation edge security systems. The architecture's modularity also opens pathways for future integration with adversarial defense mechanisms, federated learning, and explainable AI (XAI) techniques.Therefore, the proposed IDS framework and its performance under the MRS evaluation scheme represent a scalable, adaptive, and deployable solution for real-time cybersecurity in edge environments.

4.4 Real-World Deployment Scenarios

The proposed Shallow-Deep Hybrid Intrusion Detection System (IDS) is well-suited for a wide range of Edge-IoT scenarios that demand real-time responsiveness, low resource usage, and high detection accuracy. In smart traffic systems, lightweight models like Shallow20 can be deployed on roadside units (RSUs) and vehicles for continuous monitoring of spoofing or DoS attacks, while more expressive hybrids like WgtAv20 or Maximum20 can be dynamically engaged during high-risk intervals.

In disaster response networks such as temporary edge-cloud setups following natural disasters, where bandwidth, energy, and infrastructure are constrained, models with high Model Robustness Scores like Maximum20 offer secure, autonomous intrusion detection without overburdening the system.

The architecture's ability to maintain sub-second inference times further supports its integration into latency-critical environments such as industrial control systems, smart grids, and agricultural sensor networks. These use cases highlight the practical viability of the proposed IDS across both stable and resource-limited deployments, enabling scalable and resilient edge security solutions.

4.5 Future Work

Future research will focus on advancing the proposed Shallow-Deep Hybrid IDS framework through three key directions: collaborative learning, adversarial resilience, and real-world validation.

Federated Learning Integration: To preserve privacy and reduce data movement in decentralized environments, we aim to integrate the hybrid IDS into a federated learning (FL) architecture. In this setup, lightweight shallow models will be trained and updated locally on resource-constrained edge nodes, capturing site-specific traffic patterns with minimal computational overhead. In parallel, deep models will be trained partially at the edge and aggregated in a centralized or hierarchical manner across edge-cloud infrastructure. The fusion of locally inferred shallow predictions and globally synchronized deep models will be coordinated through decision-level strategies, enabling scalable and privacy-preserving threat detection across federated clients.

Adversarial Training on Edge Devices: Enhancing model robustness against adversarial attacks remains a critical challenge. Future work will explore lightweight adversarial training techniques—such as FGSM, projected gradient descent, and noise-based data augmentation—tailored for edge deployments. To address resource limitations, adversarial training can be performed intermittently or in federated cycles, leveraging idle edge resources or partial retraining to minimize runtime overhead. This approach aims to increase model resilience while maintaining real-time performance guarantees.

Explainability and Adaptivity: To improve trust and transparency in security-critical applications, Explainable AI (XAI) techniques will be integrated into the hybrid model pipeline. Feature attribution methods such as SHAP or LIME can help interpret fusion-based decisions. Additionally, adaptive fusion strategies will be explored to dynamically adjust the contribution of shallow and deep components based on observed network behavior and evolving threat patterns.

Evaluation Limitations and Extensions: The current evaluation was conducted solely on the UNSW-NB15 dataset using a selected 20-feature subset. While this setup enabled controlled comparisons, it may not fully capture the diversity of real-world IoT traffic. Moreover, adversarial testing was limited to synthetic perturbations under static threat models. Future validation will involve additional public datasets such as CICIDS2018, BoT-IoT, and live network traces from edge nodes and connected vehicles. Testing in latency-sensitive scenarios like V2X communication and disaster recovery networks will further assess the model's real-world viability.

Overall, these enhancements will strengthen the framework's adaptability, interpretability, and deployability, paving the way for intelligent, resilient intrusion detection in next-generation edge computing environments.

5 Conclusion

This paper proposed an adaptive Shallow-Deep Hybrid Intrusion Detection System (IDS) tailored for deployment in resource-constrained Edge-IoT environments. By employing decision-level fusion strategies—such as maximum, minimum, weighted averaging, and concatenation—the framework effectively combines the low-latency efficiency of shallow models with the high detection accuracy of deep neural architectures. Extensive empirical evaluations demonstrated that hybrid models consistently outperform standalone counterparts in both predictive performance and deployment feasibility. Notably, the Maximum20 and Concat20 models achieved F1-scores of 0.97–0.98 with sub-second inference times. To support deployment-aware model selection, we introduced the Model Robustness Score (MRS), a novel composite metric that jointly accounts for detection performance and resource efficiency. Maximum20 consistently achieved the highest MRS, affirming its effectiveness in balancing accuracy and computational cost.

The proposed system's modular design and low-latency inference capabilities make it particularly suitable for real-world applications such as smart traffic management, disaster response, and industrial IoT monitoring. Its demonstrated adaptability across diverse deployment priorities—analyzed through MRS sensitivity—further validates its practicality in varied edge cybersecurity scenarios. This work also lays the groundwork for future research

in emerging AIoT security domains, where lightweight, explainable, and federated intrusion detection systems are essential for protecting distributed, intelligent infrastructure.

Overall, the study establishes a scalable and deployment-ready pathway for intelligent IDS solutions at the edge, with promising future integration avenues including federated learning, explainable AI, and adversarial robustness—particularly in mission-critical environments like connected vehicles and V2X communication networks.

6 Data Availability

All data, code, models, configurations, and experiment results for the study are available at the following xxxxxx repository:
xxxxx

References

[1] T. Qiu, J. Chi, X. Zhou, Z. Ning, M. Atiquzzaman, and D. O. Wu. Edge computing in industrial internet of things: Architecture, advances and challenges. *IEEE Communications Surveys & Tutorials*, 2020.

[2] A. Sagu, N. S. Gill, and P. Gulia. Hybrid deep neural network model for detection of security attacks in iot enabled environment. *International Journal of Advanced Computer Science and Applications*, 13(1), 2022.

[3] L. Caviglione, C. Comito, M. Guarascio, and G. Manco. Emerging challenges and perspectives in deep learning model security: A brief survey. *Systems and Soft Computing*, 2023.

[4] N. W. Meegammana and H. Fernando. Securing iot servers: Shallow vs. deep neural network architectures. In *Proceedings of the International Conference on Advances in ICT for Emerging Regions (ICTER)*, Colombo, Sri Lanka, 2024. ICTER 2024.

[5] M. A. Khan et al. A hybrid deep learning-based intrusion detection system for iot networks. *Mathematical Biosciences and Engineering*, 20(8):13491–13520, 2023.

[6] W. Li, Y. Peng, M. Zhang, L. Ding, H. Hu, and L. Shen. Deep model fusion: A survey, 2023.

[7] Y. Zhang, Y. Liu, X. Guo, Z. Liu, X. Zhang, and K. Liang. A bilstm-based ddos attack detection method for edge computing. *Energies*, 15(21):7882, 2022.

[8] N. Moustafa, M. Abdel-Basset, and R. Mohamed. *Deep Learning Approaches for Security Threats in IoT Environments*. Wiley-IEEE Press, 2022.

[9] M. A. Ganaie, M. Hu, A. K. Malik, M. Tanveer, and P. N. Suganthan. Ensemble deep learning: A review. *Engineering Applications of Artificial Intelligence*, 115:105151, 2022.

[10] E.-H. Qazi, M. H. Faheem, and T. Zia. Hdlnids: Hybrid deep-learning-based network intrusion detection system. *Applied Sciences*, 13(8):4921, 2023.

[11] S. Abimannan, E.-S. M. El-Alfy, Y.-S. Chang, S. Hussain, S. Shukla, and D. Satheesh. Ensemble multifeatured deep learning models and applications: A survey. *IEEE Access*, 11:107194–107217, 2023.

[12] S. Ullah et al. Hdl-ids: A hybrid deep learning architecture for intrusion detection in the internet of vehicles. *Sensors*, 22(4):1340, 2022.

[13] H. Sedjelmaci, S. M. Senouci, N. Ansari, and A. Boualouache. A trusted hybrid learning approach to secure edge computing. *IEEE Consumer Electronics Magazine*, 2021.

[14] Y. Wu, D. Wei, and J. Feng. Network attacks detection methods based on deep learning techniques: A survey. *Security & Communication Networks*, pages 1–17, Aug 2020.

Received 29 June 2025; revised 12 July 2025; accepted 18 Aug 2025