

Q-IDS Bench: Reproducible and Resource-Aware Evaluation of Quantum-Inspired Intrusion Detection Across Five Network and IoT Datasets

Fuad Ibne Jashim Farhad

f.farhad@cquemail.com

Central Queensland University

Dr MD Mamunur Rashid

Central Queensland University

Ali Shahidinejad

Central Queensland University

Research Article

Keywords: intrusion detection, IoT security, quantuminspired learning, benchmark, QCNN, reproducibility

Posted Date: October 23rd, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-11273376/v1>

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

[Read Full License](#)

Additional Declarations: No competing interests reported.

Q-IDS Bench: Reproducible and Resource-Aware Evaluation of Quantum-Inspired Intrusion Detection Across Five Network and IoT Datasets

Fuad Ibne Jashim Farhad* MD Mamunur Rashid Ali Shahidinejad

School of Engineering and Technology, CQUniversity Australia, Melbourne, VIC, Australia

*Corresponding author: f.farhad@cquemail.com

m.rashid@cqu.edu.au a.shahidinejad@cqu.edu.au

Abstract

Very high IDS scores are frequently reported, but one accuracy measure does not by itself show whether a detector will remain reliable under a different split, preprocessing pipeline, or deployment budget. Q-IDS Bench examines this issue by benchmarking nine quantum-inspired intrusion-detection system (IDS) configurations on five datasets, all implemented on classical hardware. The data are divided into training, validation, and test sets before fitted preprocessing, and each experiment is repeated with five random seeds. Performance is assessed with weighted F1, AUPRC, MCC, false-positive rate, attack recall, and model-side inference latency. The best weighted-F1 values range from 98.86% to 99.95%, but the leading architecture depends on the dataset. The key result is therefore not a universal winner but the reproducibility of the comparison: model rankings are dataset-specific, and hybridisation alone does not guarantee better performance. The study provides a baseline for future comparison with genuine quantum machine learning (QML) implementations.

Keywords: intrusion detection, IoT security, quantum-inspired learning, benchmark, QCNN, reproducibility

1 Introduction

Modern intrusion detection has to satisfy two requirements that often conflict: statistical performance and operational reliability. A detector can score highly on a benchmark and still be difficult to deploy if that success depends on information leakage, a favourable split, a narrow metric, or latency behaviour that does not match the target environment. This is especially relevant in Internet-of-Things (IoT) and network-security settings, where traffic is heterogeneous, attacks are unevenly distributed, and both missed attacks and false alarms carry practical costs [1, 2, 3]. The same concern extends to distributed IoT, edge, and cloud deployments, where detector latency and computational cost can influence model placement and alert handling.

Quantum machine learning (QML) has introduced an-

other set of terms into an already complex area. Quantum kernels, variational circuits, quantum convolution, and hybrid quantum-classical architectures provide a further family of nonlinear representations that can be considered for security data [4, 5, 26, 27]. The terminology, however, can easily imply more than the computation actually performs. A number of practical security prototypes run entirely on classical hardware while borrowing trigonometric mappings, phase modulation, complex-valued transformations, or circuit-inspired layer structures. These models can still be scientifically useful, but the benefit being tested is representational design rather than a hardware-level quantum advantage.

The benchmark was built around this distinction. It asks a deliberately stricter question than whether a single quantum-inspired model can achieve a high score: *how stable are the conclusions when a family of models is evaluated across several IDS datasets under the same leakage-aware protocol and with both predictive and operational metrics?* The benchmark compares nine configurations across CIC-IDS-2017, NSL-KDD, UNSW-NB15, CICIoT2023, and CSE-CIC-IDS2018. The experiments use five random seeds, validation-based model selection, training-only preprocessing and dimensionality reduction, and final evaluation on held-out test data.

Three research questions guide the study. **RQ1** asks whether the relative ranking of quantum-inspired IDS configurations is stable across heterogeneous datasets. **RQ2** examines whether the proposed dual-branch phase-modulated CNN plus variational-feature fusion offers a consistent advantage over simpler single-branch or learned-feature routes. **RQ3** asks whether the strongest predictive model is also the most attractive from an operational perspective when false-positive behaviour and inference latency are considered.

The contributions are fivefold. First, the paper provides a harmonised five-dataset benchmark rather than a single-dataset demonstration. Second, it evaluates a broad model family under one protocol, including standalone, recurrent-style, attention-fusion, support-vector, and learned-feature tree routes. Third, it presents and evaluates an original

dual-branch architecture in which phase-modulated convolutional features and variational-circuit-inspired features are fused by attention. Fourth, it makes preprocessing discipline explicit by splitting before fitting scalers, feature selectors, or PCA. Fifth, it draws a clear claim boundary: the reported experiments are quantum-inspired classical implementations and are intended as a reproducible baseline for later hardware-executed QML comparisons.

2 Related Work and Claim Boundary

Classical IDS research has long shown that representation, data quality, class balance, and evaluation design can be as important as the classifier itself [1, 2, 3]. Convolutional neural networks (CNNs), recurrent neural networks, attention mechanisms, support vector machines (SVMs), Random Forests, and gradient-boosted trees have become common reference methods for learning nonlinear traffic structure and supporting resource-aware classification decisions [9, 10, 11, 24, 25].

Recent work extends this landscape with QML and hybrid quantum-classical designs. QML-IDS evaluates VQC, QSVM, and QCNN families on modern intrusion datasets; quantum autoencoder studies explore anomaly representation; variational quantum convolution has been applied to network intrusion; and noise-aware QML benchmarking has started to expose the sensitivity of results to backend assumptions [12, 13, 14, 15, 16]. These studies motivate Q-IDS Bench, but they also reinforce a methodological problem: results are difficult to compare when datasets, feature budgets, split logic, metrics, and computational assumptions differ.

The present study therefore uses the term *quantum-inspired* in a strict computational sense. The feature mapping in this benchmark is classically implemented; it uses no qubits, shots, quantum backend, or hardware-executed circuit. Names such as QCNN-, VQC-, or QSVM-inspired describe the structural or representational intuition of the experimental models, not the physical execution substrate. This distinction is central to the paper rather than a disclaimer added after the results.

A second difference is the unit of evidence. Many IDS papers are read through their best reported score, but a benchmark intended for security research should expose how conclusions change when the dataset, seed, and operational metric change. Q-IDS Bench therefore keeps the model names and evaluation logic fixed across five datasets and reports both predictive quality and model-side latency. This design makes it possible to distinguish a model that is consistently strong from one that wins only under a particular data geometry. Table 1 summarises this positioning. It also provides a common baseline for future hardware-executed QML studies, which can reuse the same split discipline and feature budgets instead of introducing a new evaluation protocol at the same time as

a new computational substrate.

3 Benchmark Design and Data Governance

3.1 Datasets and Binary Task Harmonisation

The benchmark uses five labelled datasets with different collection histories, feature schemas, class distributions, and intended environments. CIC-IDS-2017 provides benign and malicious traffic from a controlled enterprise-style testbed [18]; NSL-KDD is a curated derivative of KDD Cup 1999 and retains its official train/test separation [19]; UNSW-NB15 provides modern synthetic traffic with official training and testing files [20]; CICIOT2023 focuses on large-scale IoT attack traffic [21]; and CSE-CIC-IDS2018 adds a further modern flow-based benchmark [22]. Original attack taxonomies are mapped to a common binary task—benign/normal versus attack—so that every model faces the same first-stage IDS objective.

This harmonisation should not be mistaken for distributional equivalence. The datasets remain heterogeneous by design. For this reason, the benchmark is *protocol-harmonised rather than distribution-homogenised*: the same evaluation logic is retained while allowing different traffic environments to favour different representations.

3.2 Leakage-Aware Processing

Figure 1 shows the authors’ original benchmark pipeline. Splitting is performed before data-dependent transformations. Missing-value handling, categorical encoding, scaling, feature selection, and PCA are fitted on the training partition only and then applied unchanged to validation and test data. This order matters because fitting PCA or scaling on the full dataset can allow test-distribution information to shape the learned representation, artificially tightening the boundary between training and evaluation data.

4 Quantum-Inspired Model Family

4.1 Feature Mapping and Terminology

The VQC-inspired branch maps a scaled PCA vector $\mathbf{x}$ to a nonlinear phase-based representation. The construction is motivated by quantum feature-map intuition [26, 27], but the computation is classical. The initial transformation is

$$\mathbf{u}^{(0)}(\mathbf{x}) = \sin(\mathbf{x} \odot \mathbf{w}_{RY}^{(0)}) \odot \cos(\mathbf{x} \odot \mathbf{w}_{RZ}^{(0)}), \quad (1)$$

Table 1: Position of Q-IDS Bench relative to representative IDS and QML-oriented work

Study	Method / focus	Security setting	Strength	Gap addressed by Q-IDS Bench
Buczak and Guven [1]	ML/data-mining survey	General IDS	Establishes the importance of data, features, and evaluation	No experimental quantum-inspired benchmark
Zarpelao et al. [2]	IoT IDS survey	IoT security	Highlights resource and deployment constraints	No common multi-model benchmark
Vinayakumar et al. [3]	Deep-learning IDS	Multiple IDS datasets	Strong classical neural baseline literature	No quantum-inspired representation comparison
de Abreu et al. [12]	VQC, QSVM, QCNN	Network IDS	Direct QML-oriented architecture comparison	Different circuit, backend, and feature assumptions
Hdaib et al. [13]	QAE-based anomaly detection	Network/IoT anomaly detection	Motivates quantum-inspired anomaly representations	Different tasks and deployment assumptions
Gong et al. [14]	Variational quantum convolution	Network IDS	Motivates QCNN-style detection	Hardware scalability and protocol comparability remain open
Cirillo et al. [15]	Noisy QML benchmark	ToN-IoT, NSL-KDD	Explicitly studies noise and circuit complexity	Different datasets and genuine-circuit assumptions
Danasabe et al. [16]	Lightweight hybrid tabular model	Low-latency IDS	Makes latency an explicit design objective	Different hybrid structure
Khan et al. [17]	Hybrid quantum-classical neural network	Incident detection	Shows hybrid models can be strong in low-data regimes	Incident detection rather than network-flow IDS
Q-IDS Bench	Nine quantum-inspired configurations; five seeds	Five IDS/IoT datasets	Leakage-aware, resource-aware cross-dataset evaluation	No claim of quantum-hardware advantage; benchmark for future QML comparison

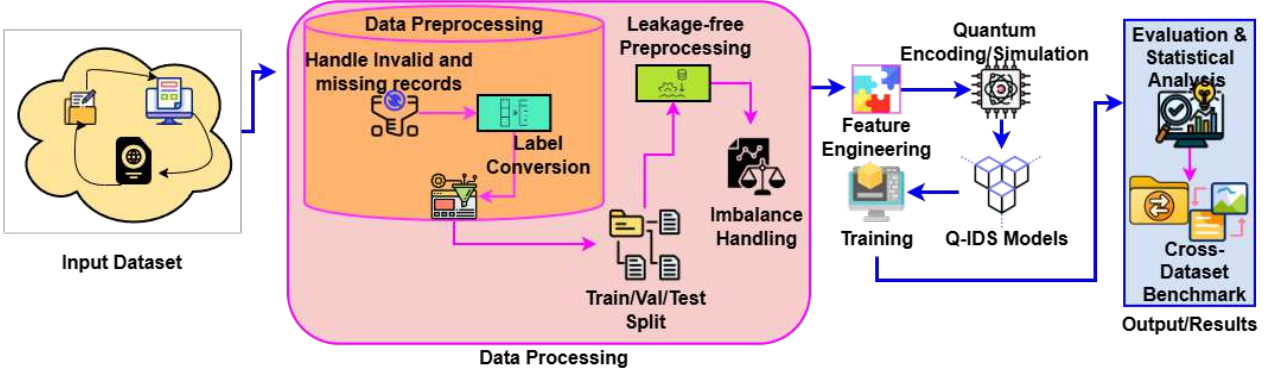


Fig. 1 Authors’ original Q-IDS Bench pipeline. The workflow emphasises train/validation/test separation before leakage-sensitive preprocessing, followed by model training, evaluation, statistical analysis, and cross-dataset benchmarking

followed by trainable phase modulation and cyclic mixing,

$$\begin{aligned}
 \mathbf{v}^{(\ell)} &= \mathbf{u}^{(\ell-1)} \odot e^{i\phi^{(\ell)}}, \\
 \mathbf{u}^{(\ell)} &= \frac{\mathbf{v}^{(\ell)} + 0.5 \text{shift}(\mathbf{v}^{(\ell)})}{1.5}, \\
 \mathbf{q}(\mathbf{x}) &= |\mathbf{u}^{(L)}|^2.
 \end{aligned} \tag{2}$$

The resulting magnitude-squared vector is projected into a hidden representation. Figure 2 is the authors’ original conceptual diagram and explicitly marks the implementation boundary: the benchmark uses no qubits, shots, quantum backend, or hardware execution.

4.2 Phase-Modulated Convolutional Branch

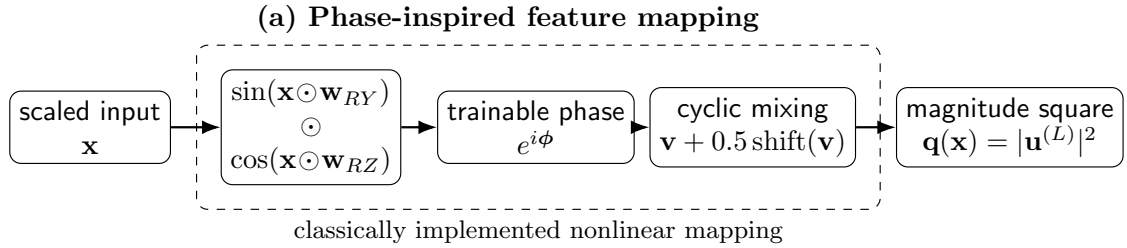
The convolutional branch treats the PCA vector as a one-dimensional signal and learns local interactions with parameter sharing. A learnable phase-like factor modulates each convolutional block,

$$\mathbf{h}^{(\ell)} = \text{ReLU}\left(\text{BN}(\text{Conv1D}(\mathbf{h}^{(\ell-1)})) \odot \sin(\mathbf{a}^{(\ell)})\right). \tag{3}$$

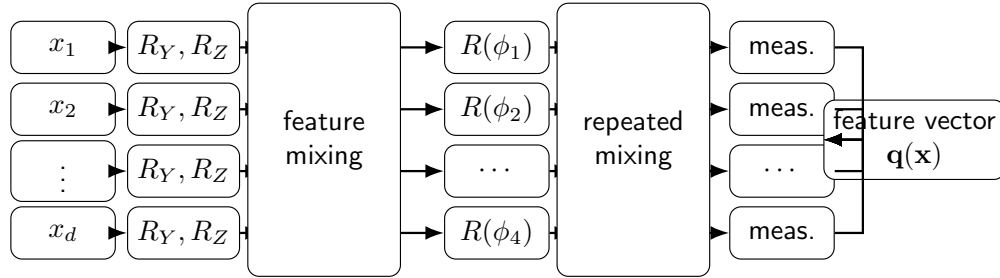
The purpose is not to claim quantum convolution, but to test whether a phase-modulated representation produces features that remain useful across heterogeneous traffic

Table 2: Dataset configuration used in the five-seed experiments

Dataset	Experiment size	Train / validation / test	Feature handling	PCA variance	Protocol	Status
CIC-IDS-2017	368,017 sampled records	220,809 / 73,604 / 73,604	Numeric flow features; class-aware sampling; PCA 20	91.77%	Random 60/20/20; 5 seeds; 10 epochs	Completed
NSL-KDD	Official KD-DTrain+ / KD-DTest+	100,778 / 25,195 / 22,544	Categorical encoding; scaling; PCA 20	90.27%	Official split; 5 seeds; 10 epochs	Completed
UNSW-NB15	Official train/test files	65,865 / 16,467 / 175,341	Categorical encoding; scaling; PCA 20	93.65%	Official split; 5 seeds; 10 epochs	Completed
CICIoT2023	300,000 balanced sampled records	200,000 / 50,000 / 50,000	Numeric IoT-flow features; PCA 20	94.26%	Random 60/20/20; 5 seeds; 10 epochs	Completed
CSE-CIC-IDS2018	90,928 sampled records from 10 files	54,556 / 18,186 / 18,186	50 selected features; PCA 32	99.99%	Random 60/20/20; 5 seeds; 10 epochs	Completed



(b) Variational-circuit-inspired conceptual structure



Conceptual only: no qubits, shots, quantum backend, or hardware execution are used in Q-IDS Bench.

Fig. 2 Authors’ original quantum-inspired feature-mapping schematic. Panel (a) shows the classically implemented phase-inspired map; panel (b) shows the variational-circuit-inspired conceptual analogy. The figure explicitly states that Q-IDS Bench does not execute quantum hardware

datasets and across alternative decision layers.

4.3 Dual-Branch Attention Fusion and Auxiliary Tree Route

The proposed dual-branch model processes the same reduced input through the phase-modulated convolutional branch and the VQC-inspired feature branch. If these representations are denoted $\mathbf{h}_C$ and $\mathbf{h}_Q$, an attention module estimates branch weights,

$$\alpha = \text{softmax}(g([\mathbf{h}_C; \mathbf{h}_Q])), \quad (4)$$

and constructs a fused representation

$$\mathbf{h}_F = [\alpha_C \mathbf{h}_C; \alpha_Q \mathbf{h}_Q; \mathbf{h}_C + \mathbf{h}_Q], \quad (5)$$

which is passed to an MLP classifier. The architecture also exposes a second experimental route in which learned convolutional features feed Random Forest or XGBoost classifiers. This modular architecture makes it possible to examine how much of the observed performance comes from the feature representation, the fusion mechanism, and the final classifier. Figure 3 shows the complete architecture.

4.4 Nine Evaluated Configurations

The benchmark covers nine configurations: QCNN Standalone, VQC Standalone, QCNN+VQC Hybrid, Quantum LSTM+Attention, QRNN, QLSTM, QSVM on VQC-inspired features, QRandomForest on learned QCNN fea-

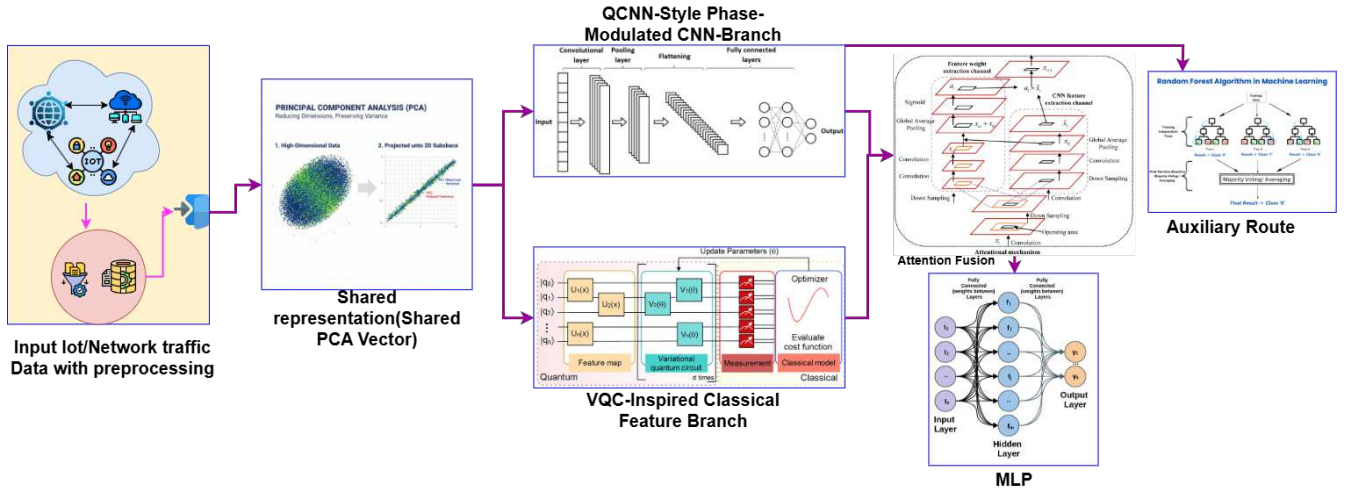


Fig. 3 Authors’ original Q-IDS Bench architecture. The diagram follows the pipeline from PCA-based reduction through parallel convolutional and variational-inspired processing, attention/dense fusion, and the learned-feature tree-classifier route. Labels beginning with “Q” are experiment names only; they do not imply execution on quantum hardware

tures, and QXGBoost on learned QCNN features. The recurrent-style variants use tabular feature vectors and are not treated as full temporal sequence models. The QSVM label is also structural: it refers to an RBF SVM trained on the VQC-inspired representation, not to a quantum SVM executed on hardware. These labels are therefore used to compare representational choices while the computational substrate remains classical.

4.5 Training and Evaluation Algorithm

Algorithm 1 makes the experimental control flow explicit. It shows that label harmonisation and data partitioning precede fitted preprocessing, that validation is used for model or threshold selection, and that the test partition is evaluated only after each model is frozen. The same seed set and metric family are used across all model–dataset combinations before the summary tables and cross-dataset visualisations are generated.

5 Experimental Protocol and Evaluation Metrics

To reduce dependence on a single run, every main experiment is repeated with seeds 42, 123, 2024, 2025, and 777. Neural configurations are trained for ten epochs, with validation data used for checkpoint or operating-threshold selection and the test partition kept for final evaluation. Weighted F1 is used as the main summary measure. AUPRC is reported because precision-recall behaviour is especially informative under class imbalance [7], while MCC provides a balanced correlation-style view of binary classification quality [8]. False-positive rate and attack recall capture different security failure modes, and model-side inference latency adds an initial measure of

Algorithm 1 Q-IDS Bench training and evaluation protocol

Require: Labelled traffic dataset D ; seed set $S = \{42, 123, 2024, 2025, 777\}$.

Ensure: Per-seed test metrics, summary tables, and visual outputs.

- 1: Map dataset labels to benign/normal and attack classes.
- 2: Split data into training, validation, and test partitions according to the dataset protocol.
- 3: Fit imputation, scaling, encoding, feature selection, and PCA using training data only.
- 4: **for** each seed $s \in S$ **do**
- 5: **for** each model m in the nine-model family **do**
- 6: Initialise m and train neural configurations for ten epochs.
- 7: Select the checkpoint or operating threshold using validation data.
- 8: Freeze the selected model and evaluate it once on the held-out test set.
- 9: Record weighted F1, AUPRC, MCC, FPR, attack recall, latency, and confusion outputs.
- 10: **end for**
- 11: **end for**
- 12: Consolidate seed-level outputs and generate benchmark tables, figures, and cross-dataset summaries.

operational cost.

Rather than treating the study as a single leaderboard, the analysis reads cross-dataset averages together with the winner on each individual dataset. A model that dominates one traffic distribution may not retain the same advantage on another, so both views are needed. Formal Friedman/Nemenyi testing is reserved for the final per-seed significance analysis after all seed-level outputs have been consolidated; accordingly, the present manuscript does not claim statistical significance that has not yet been computed.

Table 3: Weighted F1 (%) across the five benchmark datasets

Model	CIC-IDS-2017	NSL-KDD	UNSW-NB15	CICIoT2023	CSE-CIC-IDS2018
QCNN Standalone	96.60	98.31	99.34	99.24	97.07
VQC Standalone	96.79	96.32	97.04	99.18	96.95
QCNN+VQC Hybrid	96.79	96.21	98.45	99.25	97.25
Quantum LSTM+Attention	96.93	99.19	99.20	99.25	96.49
QRNN	97.45	96.42	99.30	99.27	97.17
QLSTM	97.28	99.28	96.95	99.25	97.38
QSVM (VQC features)	97.20	99.95	98.43	99.18	97.17
QRandomForest (QCNN features)	99.01	96.52	97.00	99.29	98.30
QXGBoost (QCNN features)	98.87	96.31	97.30	99.29	98.86

6 Results

6.1 Cross-Dataset Performance

Table 3 and Figures 4–5 show a clear result: the benchmark does not produce a universal winner. QRandomForest leads CIC-IDS-2017, QSVM leads NSL-KDD, QCNN Standalone leads UNSW-NB15, and QXGBoost leads CICIoT2023 and CSE-CIC-IDS2018. The top mean weighted F1 is obtained by QSVM at 98.38%, followed by Quantum LSTM+Attention at 98.21%, QXGBoost at 98.12%, and QCNN Standalone at 98.11%. Only about 1.13 percentage points separate the first- and ninth-ranked configurations by mean weighted F1. Table 4 reports the corresponding cross-dataset ranking. At that scale, false-alarm behaviour, latency, reproducibility, and deployment constraints can realistically change which model is preferable.

Table 4: Average weighted-F1 ranking across the five datasets

Model	Mean W-F1 (%)	Rank
QSVM (VQC features)	98.38	1
Quantum LSTM+Attention	98.21	2
QXGBoost (QCNN features)	98.12	3
QCNN Standalone	98.11	4
QLSTM	98.03	5
QRandomForest (QCNN features)	98.02	6
QRNN	97.92	7
QCNN+VQC Hybrid	97.59	8
VQC Standalone	97.26	9

6.2 Best Configuration by Dataset

The best observed weighted-F1 values are 99.01% on CIC-IDS-2017, 99.95% on NSL-KDD, 99.34% on UNSW-NB15, 99.29% on CICIoT2023, and 98.86% on CSE-CIC-IDS2018. Figure 6 makes the model-switching pattern easy to see, while Table 5 lists the corresponding predictive and operational metrics. Compared with a single pooled average, this view is more revealing because it shows how strongly the preferred architecture depends on the dataset rather than rewarding the same model everywhere.

6.3 Performance and Contribution of the Proposed Hybrid Architecture

The QCNN+VQC Hybrid performs strongly on CICIoT2023 and UNSW-NB15, reaching 99.25% and 98.45% weighted F1, respectively, yet its simple cross-dataset mean places it eighth overall. This should not be read as a failure of the architecture. Instead, the results show that branches and attention can increase representational flexibility without producing the best prediction on every dataset. The dual-branch design remains useful because it separates two nonlinear representations and supports branch-ablation and attention-weight analyses instead of assuming that fusion must always win. Table 6 presents the complete hybrid profile, including AUPRC, MCC, FPR, and model-side latency.

Looking across the wider hybrid family leads to a similar conclusion. On several datasets, learned QCNN features perform best when the final decision layer is Random Forest or XGBoost, which suggests that representation learning and classifier choice should be considered separately. The SVM route on VQC-inspired features is strongest on NSL-KDD, while the attention-based recurrent configuration performs well on UNSW-NB15. Table 7 and Figure 7 therefore show dataset-specific peaks rather than one universal hybrid optimum.

6.4 Comparison with Related Hybrid and QML-Oriented IDS Work

Because the studies considered here differ in their datasets, feature-reduction strategies, split protocols, simulator assumptions, and evaluation metrics, their numerical results are not directly comparable. The comparison is therefore used to position the present evidence rather than to construct a strict cross-study leaderboard. de Abreu et al. evaluate VQC, QSVM, and QCNN models on three datasets that overlap substantially with the present benchmark [12]; Cirillo et al. explicitly study noisy QML behaviour on ToN-IoT and NSL-KDD [15]; Danasabe et al. make low-latency hybrid tabular IDS a first-class objective [16]; and Gong et al. motivate variational convolutional structures for intrusion detection [14]. Q-IDS Bench differs by keeping the present computation classical while

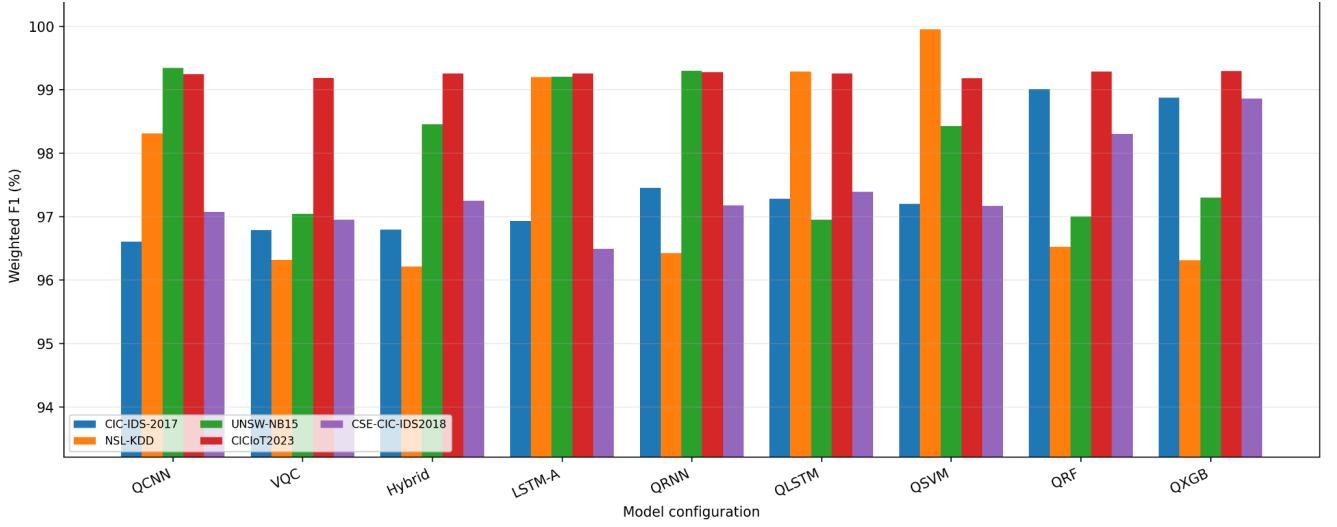


Fig. 4 Authors’ original grouped weighted-F1 comparison across nine model configurations and five datasets. The leading configuration changes with the dataset

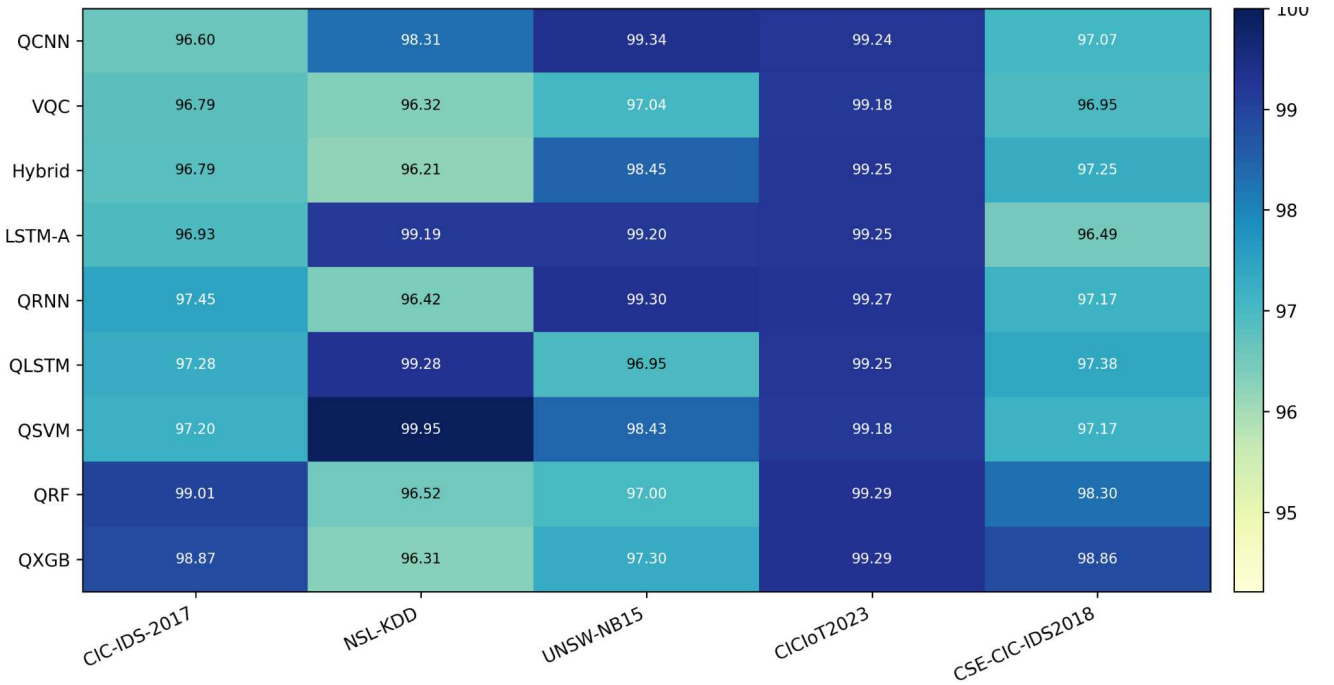


Fig. 5 Authors’ original weighted-F1 heatmap. The dense view highlights both consistently strong regions and the absence of a single row that dominates all five datasets

Table 5: Best observed model for each dataset with selected predictive and operational metrics

Dataset	Best model	Accuracy	W-F1	AUPRC	MCC	FPR	Latency
CIC-IDS-2017	QRandomForest (QCNN features)	99.01	99.01	99.88	0.980	1.03	0.0160
NSL-KDD	QSVM (VQC features)	98.15	99.95	97.42	0.950	0.60	0.0802
UNSW-NB15	QCNN Standalone	99.83	99.34	99.23	0.961	0.75	0.0279
CICIoT2023	QXGBoost (QCNN features)	99.29	99.29	99.81	0.986	0.12	0.0046
CSE-CIC-IDS2018	QXGBoost (QCNN features)	98.73	98.86	99.99	0.967	1.17	0.0112

widening the repeated-seed, multi-dataset, resource-aware son. comparison. Table 8 summarises these points of compari-

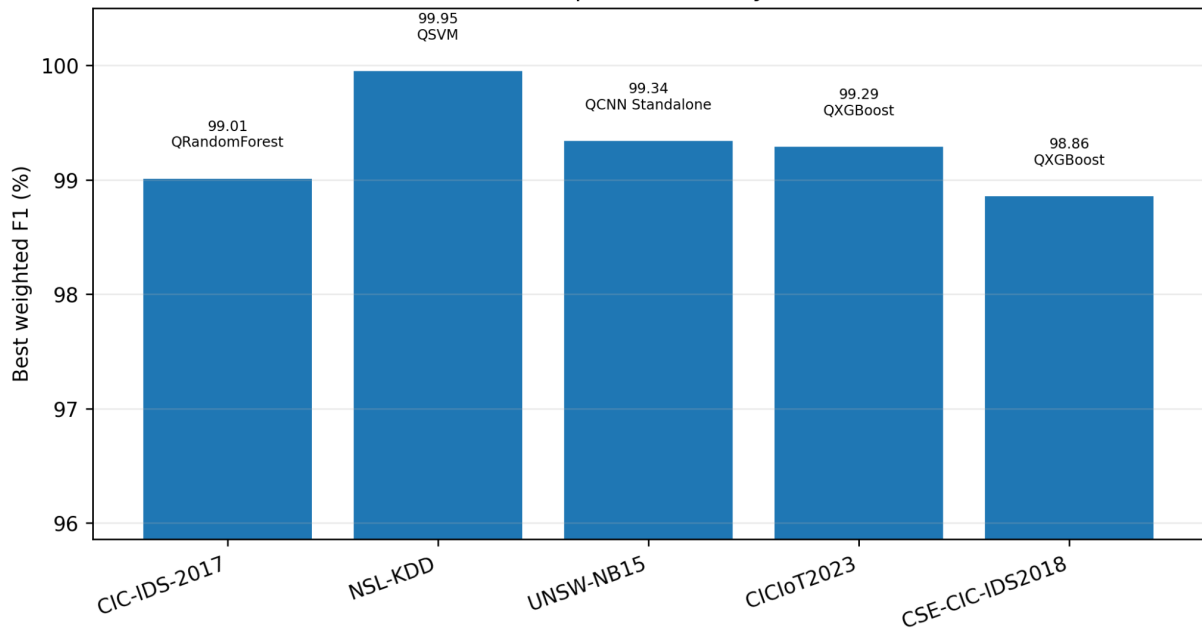


Fig. 6 Authors’ original best-by-dataset weighted-F1 figure. The dataset-specific winners reinforce the central finding that model preference is environment-dependent

Table 6: Performance of the proposed QCNN+VQC Hybrid across the five datasets

Dataset	Accuracy	W-F1	AUPRC	MCC	FPR	Latency
CIC-IDS-2017	96.79	96.79	99.27	0.936	4.28	0.0149
NSL-KDD	96.58	96.21	99.72	0.969	0.86	0.0207
UNSW-NB15	99.03	98.45	99.95	0.952	0.67	0.0294
CICIOT2023	99.25	99.25	99.79	0.985	0.17	0.0219
CSE-CIC-IDS2018	96.07	97.25	99.31	0.936	4.65	0.0179

Table 7: Selected hybrid-family weighted-F1 results (%)

Model	CIC-IDS-2017	NSL-KDD	UNSW-NB15	CICIOT2023	CSE-CIC-IDS2018	Best
QCNN+VQC Hybrid	96.79	96.21	98.45	99.25	97.25	99.25
Quantum LSTM+Attention	96.93	99.19	99.20	99.25	96.49	99.25
QSVM (VQC features)	97.20	99.95	98.43	99.18	97.17	99.95
QRandomForest (QCNN features)	99.01	96.52	97.00	99.29	98.30	99.29
QXGBoost (QCNN features)	98.87	96.31	97.30	99.29	98.86	99.29

Table 8: Positioning against selected hybrid and QML-oriented intrusion-detection studies

Study	Model family	Datasets / task	Evidence focus	Relation to Q-IDS Bench
de Abreu et al. [12]	VQC, QSVM, QCNN	UNSW-NB15, CIC-IDS-2017, CICIOT2023	Binary and multiclass QML IDS	Closest dataset overlap; different circuit and backend assumptions
Cirillo et al. [15]	QSVC, VQC, hybrid QNN	ToN-IoT, NSL-KDD	Ideal and noisy QML comparison	Motivates explicit noise-aware future validation
Danasabe et al. [16]	VQC encoder with tabular head	Flow-based IDS	Low-latency hybrid design	Closely related operational framing
Gong et al. [14]	Variational quantum convolution	Network IDS	QCNN-style detection	Motivates convolutional quantum-inspired representation
Khan et al. [17]	Hybrid quantum-classical neural network	Connected-vehicle incident detection	Strong hybrid performance under limited-data conditions	Related hybrid evidence outside network-flow IDS
Q-IDS Bench	Nine classically executed quantum-inspired configurations	Five datasets; five seeds	Leakage-aware evaluation with F1, AUPRC, MCC, FPR, and latency	Reproducible classical baseline for later genuine QML comparison

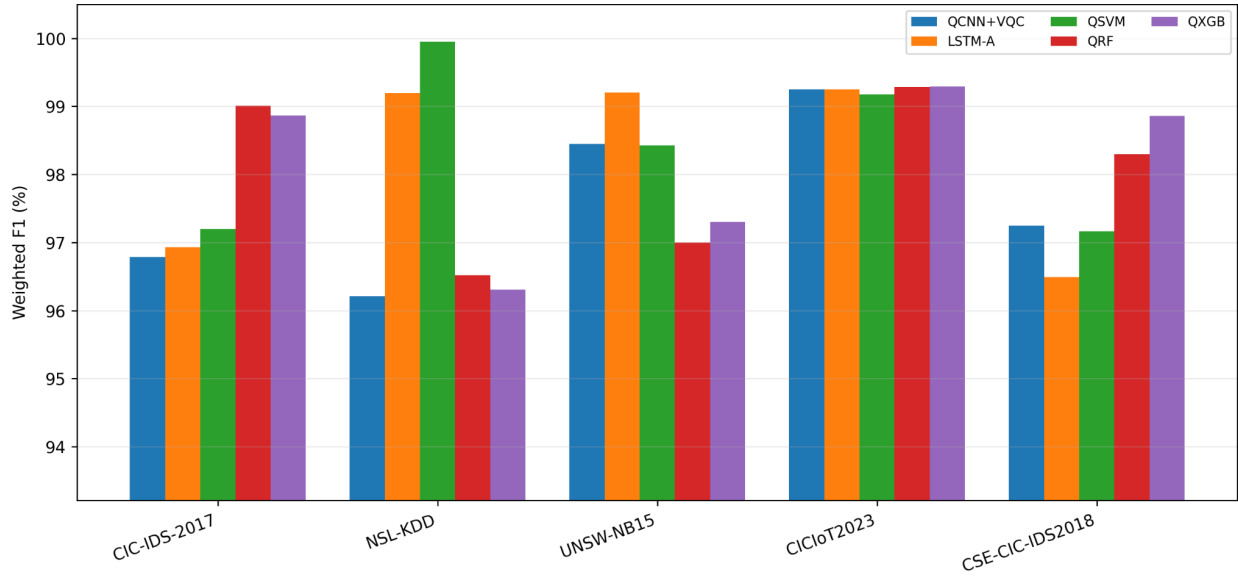


Fig. 7 Authors’ original hybrid-family comparison. Performance varies across QCNN+VQC, LSTM-A, QSVM, QRandomForest, and QXGBoost routes rather than converging on one universal hybrid solution

6.5 Latency and Error Structure

Security deployment is a multi-objective problem: a detector must identify attacks, avoid excessive false alarms, and respond within the budget of the target environment. Figure 8 shows that the highest-F1 point is not synonymous with the lowest-latency point. For example, the best NSL-KDD configuration reaches 99.95% weighted F1 with 0.0802 ms/sample model-side latency, whereas the best CICIoT2023 configuration reaches 99.29% at 0.0046 ms/sample. These values are not end-to-end network response times, but they illustrate why deployment decisions cannot be made from F1 alone.

Figure 9 is included as a descriptive diagnostic rather than as primary evidence for class-specific errors. Its counts were reconstructed from aggregate rates and test-set class supports instead of being exported directly from stored per-sample predictions. The panels are still useful as a consistency check, but the archival version should regenerate each confusion matrix from frozen test predictions for every seed.

7 Discussion

7.1 The Main Result Is Instability of the Winner

The more telling result is not the 99.x% score on any one dataset, but the fact that the winning route changes with the dataset. This changes how the benchmark should be read. If one architecture had dominated all five datasets, a broad modelling advantage would have been a reasonable conclusion. Instead, the results support a conditional view in which representation choice interacts with dataset geometry, class distribution, feature construction, and split

design strongly enough to reorder the ranking.

Across the five datasets, only 1.13 percentage points separate the first- and ninth-ranked means. When predictive averages are this close, model selection becomes as much an engineering decision as a statistical one. A slightly lower F1 may be preferable when it comes with fewer false positives, lower model-side latency, simpler calibration, or easier reproducibility. Conversely, a high F1 has limited value if the deployment path cannot meet the response requirements of the target system.

7.2 Representation and Classifier Should Be Evaluated Separately

The learned-feature tree routes help separate representation quality from classifier choice. QRandomForest and QXGBoost are not simple tree baselines; both use features learned by the phase-modulated convolutional branch. Their strong results on CIC-IDS-2017, CICIoT2023, and CSE-CIC-IDS2018 show that the learned representation remains useful even when the final decision boundary is handed to a strong tabular classifier. This modularity supports controlled ablation studies that can isolate what is contributed by the representation, the fusion mechanism, and the final classifier rather than treating the end-to-end hybrid as a single indivisible model.

7.3 Why the Quantum-Inspired Claim Boundary Strengthens the Paper

Hardware-level quantum claims are deliberately excluded from this benchmark. That boundary strengthens the comparison by keeping the test conditions clear. The current models provide a classical testbed for examining whether phase-based and circuit-inspired transformations are useful

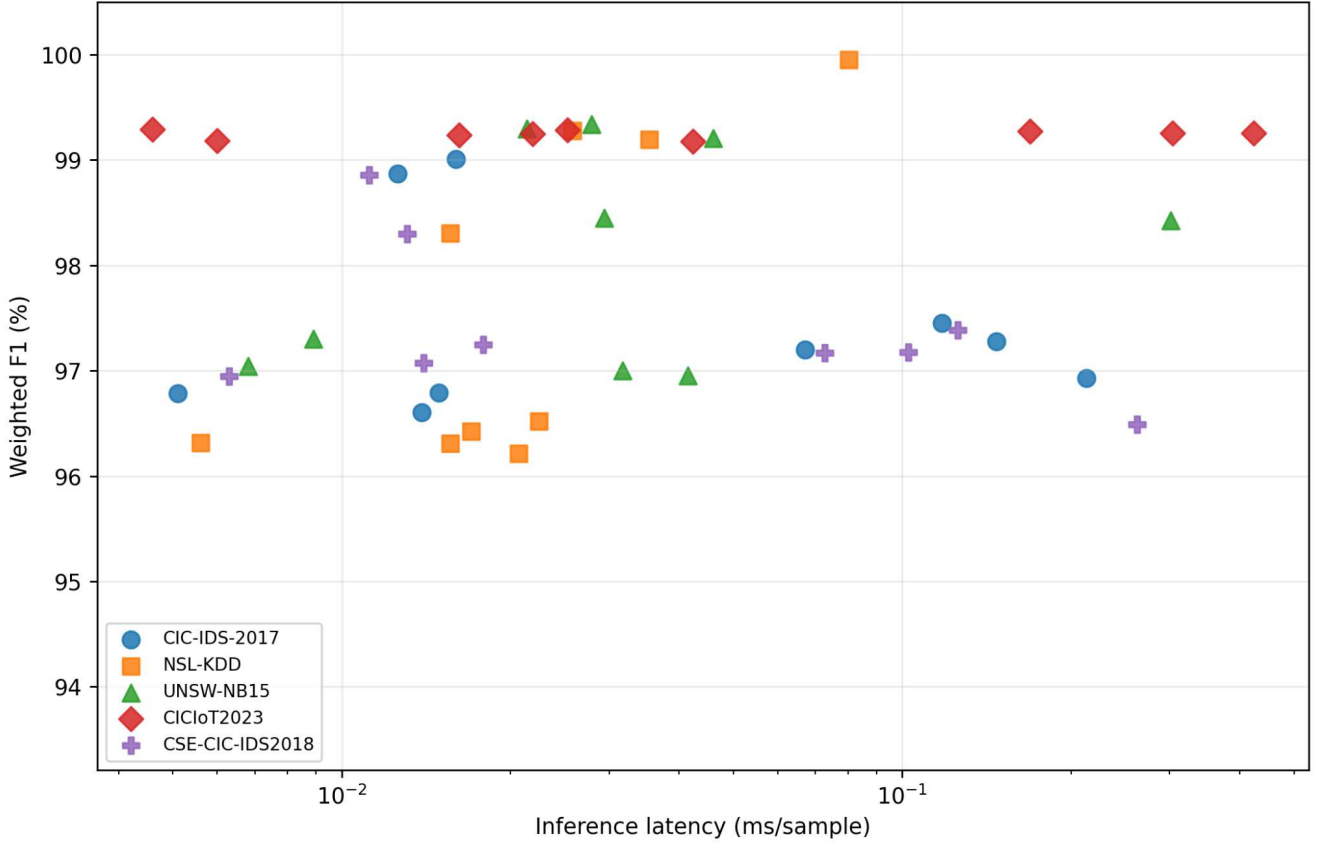


Fig. 8 Authors’ original plot of model-side inference latency versus weighted F1. The benchmark shows a performance–cost frontier rather than one model dominating both axes

before taking on the cost and noise of genuine quantum execution. Future Qiskit or PennyLane implementations can reuse the same data splits, feature budgets, metrics, and latency accounting to compare hardware-executed circuits with the current surrogates under controlled conditions.

7.4 Operational Interpretation

In an operational IDS, false-positive rate, attack recall, latency, and reproducibility all affect deployment decisions. These factors shape analyst workload, the risk of missed attacks, and whether a detector is suitable for inline use, edge deployment, or offline triage. Q-IDS Bench is therefore intended as a resource-aware benchmark rather than a simple leaderboard. The preferred model is the one that meets the security objective within the constraints of the deployment environment, and this need not be the model with the highest isolated F1.

8 Threats to Validity

A number of limitations shape the next stage of this work. The current models are quantum-inspired and classically executed, so genuine QML still needs a separate evaluation that reports qubit count, encoding, ansatz depth, shots, backend, noise model, queue time, and end-to-end runtime.

The five datasets also differ in age, generation process, feature schema, and split strategy; strong within-dataset results therefore do not demonstrate cross-domain transfer. In addition, the recurrent-style models operate on tabular vectors rather than true host- or session-level sequences, which limits temporal interpretation. The reported latency is model-side only and excludes data capture, feature extraction, transport, orchestration, and analyst response. Finally, the confusion-summary visual should be regenerated from stored per-sample predictions instead of reconstructed aggregate rates.

9 Future Work

Future work should move from suggestive model labels toward controlled attribution. Cross-day, source-file, and cross-dataset transfer experiments are a natural first step because they can expose traffic drift and hidden train–test similarity. Second, genuine QML baselines should be implemented in Qiskit or PennyLane with explicit qubit count, encoding method, ansatz depth, shots, backend, noise model, queue time, and full end-to-end runtime. Third, the recurrent-style branch should be redesigned around actual host-, device-, session-, or time-window sequences rather than static tabular vectors. Fourth, formal Friedman/Nemenyi or other suitable repeated-measures

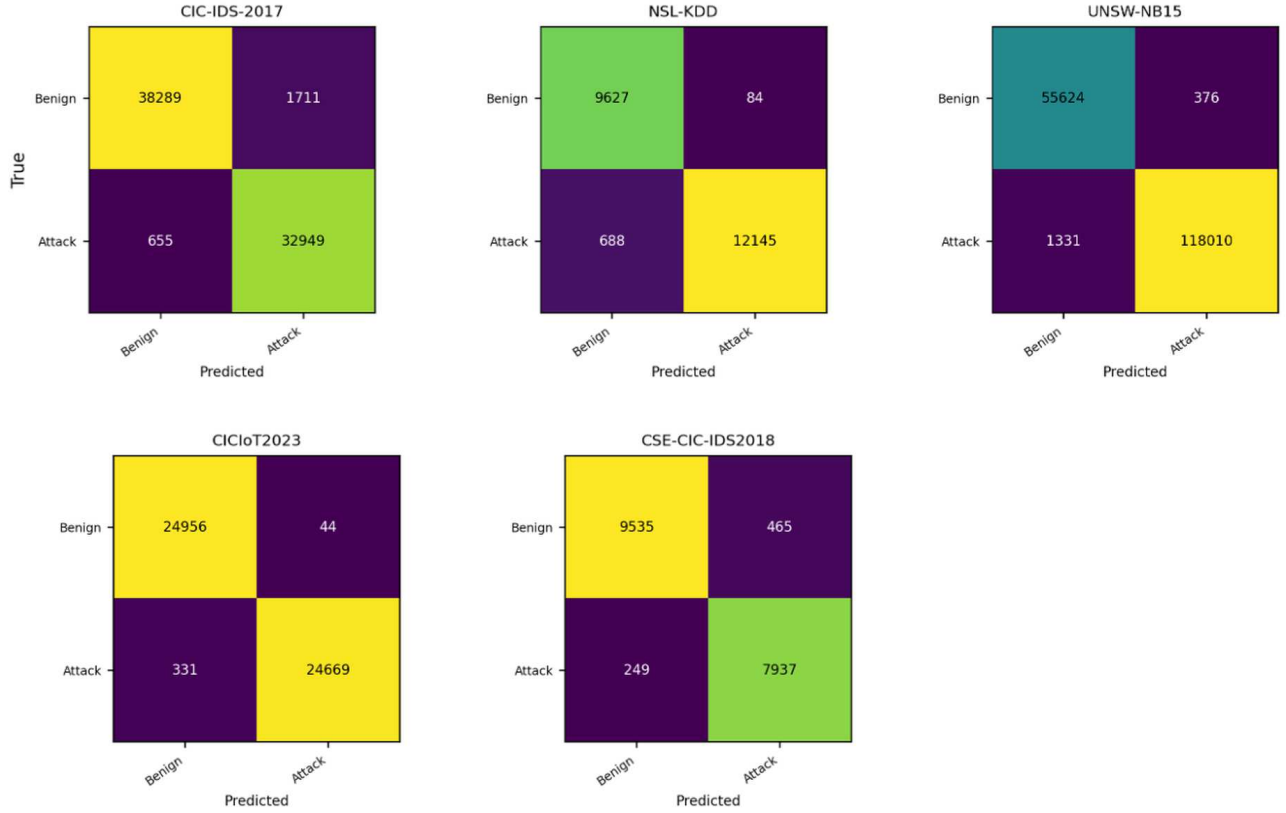


Fig. 9 Authors’ original QCNN+VQC Hybrid confusion-summary figure. Because the counts were reconstructed from aggregate rates and class supports, the panels are treated as descriptive diagnostics rather than raw-prediction confusion matrices

comparisons should be conducted on the final per-seed outputs, accompanied by calibration and threshold analysis. Fifth, resource profiling should be expanded to throughput, memory, energy proxy, and analyst-alert burden.

The strongest extension would be a paired experiment in which the present classical surrogates and hardware-executed QML models receive identical split indices, identical feature budgets, and identical model-selection rules. That design would isolate the value of the computational substrate from the value of preprocessing or representation choice and would move the discussion from suggestive terminology to attributable quantum evidence.

10 Conclusion

Q-IDS Bench provides a leakage-aware, repeated-seed, multi-dataset benchmark for evaluating quantum-inspired intrusion-detection models without overstating what the implementation does. Across five datasets, best weighted-F1 scores range from 98.86% to 99.95%, yet the winning configuration changes with the traffic environment. The QSVM-labelled route on VQC-inspired features achieves the highest mean weighted F1, learned-feature tree models lead several datasets, and the proposed QCNN+VQC Hybrid is competitive but not universally superior. The key

contribution is therefore methodological: strong IDS claims should survive heterogeneous datasets, explicit preprocessing boundaries, multiple metrics, and resource-aware analysis. By preserving the original architecture and result figures while drawing a strict line between quantum-inspired computation and hardware-executed QML, the benchmark creates a credible baseline for the next generation of reproducible quantum-security experiments.

Acknowledgements

The authors acknowledge CQUniversity Australia, School of Engineering and Technology, for providing the research environment and occasional access to university high-performance computing resources used to support computational experiments. During manuscript preparation, OpenAI ChatGPT was used for language editing, LaTeX formatting, and document-structure refinement. The authors reviewed the resulting edits and retain responsibility for the scientific content, analysis, interpretation, and final manuscript.

References

- [1] Buczak AL, Guven E (2016) A survey of data mining and machine learning methods for cyber security intrusion detection. *IEEE Commun Surv Tutor* 18(2):1153–1176.
- [2] Zarpelao BB, Miani RS, Kawakani CT, de Alvarenga SC (2017) A survey of intrusion detection in Internet of Things. *J Netw Comput Appl* 84:25–37.
- [3] Vinayakumar R, Alazab M, Soman KP, Poornachandran P, Al-Nemrat A, Venkatraman S (2019) Deep learning approach for intelligent intrusion detection system. *IEEE Access* 7:41525–41550.
- [4] Benedetti M, Lloyd E, Sack S, Fiorentini M (2019) Parameterized quantum circuits as machine learning models. *Quantum Sci Technol* 4(4):043001.
- [5] Preskill J (2018) Quantum computing in the NISQ era and beyond. *Quantum* 2:79.
- [6] Paszke A et al (2019) PyTorch: an imperative style, high-performance deep learning library. In: *Advances in Neural Information Processing Systems*.
- [7] Saito T, Rehmsmeier M (2015) The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. *PLoS ONE* 10(3):e0118432.
- [8] Chicco D, Jurman G (2020) The advantages of the Matthews correlation coefficient over F1 score and accuracy in binary classification evaluation. *BMC Genomics* 21:6.
- [9] LeCun Y, Bottou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition. *Proc IEEE* 86(11):2278–2324.
- [10] Hochreiter S, Schmidhuber J (1997) Long short-term memory. *Neural Comput* 9(8):1735–1780.
- [11] Vaswani A et al (2017) Attention is all you need. In: *Advances in Neural Information Processing Systems*.
- [12] de Abreu DM, Rothenberg CE, Abelém AJG (2024) QML-IDS: Quantum Machine Learning Intrusion Detection System. In: *2024 IEEE Symposium on Computers and Communications (ISCC)*. <https://doi.org/10.1109/ISCC61673.2024.10733655>.
- [13] Hdaib M, Rajasegarar S, Pan L (2024) Quantum deep learning-based anomaly detection for enhanced network security. *Quantum Mach Intell* 6:26. <https://doi.org/10.1007/s42484-024-00163-2>.
- [14] Gong C, Guan W, Zhu H, Gani A, Qi H (2024) Network intrusion detection based on variational quantum convolution neural network. *J Supercomput* 80:12743–12770. <https://doi.org/10.1007/s11227-024-05919-y>.
- [15] Cirillo F, Esposito C, Seo JT (2026) Benchmarking quantum machine learning methods for intrusion detection on noisy quantum computers. *Quantum Mach Intell* 8:27. <https://doi.org/10.1007/s42484-026-00379-4>.
- [16] Danasabe A, Afaq M, El-Maleh A, Khan A, El Ferik S, Kaleem Z (2026) QATNet: a lightweight quantum-classical tabular network for low-latency intrusion detection. *IEEE Trans Quantum Eng*. <https://doi.org/10.1109/TQE.2026.3674396>.
- [17] Khan Z et al (2023) Hybrid quantum-classical neural network for incident detection. In: *Proceedings of the International Conference on Information Fusion (FUSION)*, pp 1–8.
- [18] Sharafaldin I, Lashkari AH, Ghorbani AA (2018) Toward generating a new intrusion detection dataset and intrusion traffic characterization. In: *Proceedings of ICISP*, pp 108–116.
- [19] Tavallaee M, Bagheri E, Lu W, Ghorbani AA (2009) A detailed analysis of the KDD CUP 99 data set. In: *Proceedings of IEEE CISDA*, pp 1–6.
- [20] Moustafa N, Slay J (2015) UNSW-NB15: a comprehensive data set for network intrusion detection systems. In: *Proceedings of MilCIS*, pp 1–8.
- [21] Canadian Institute for Cybersecurity (2023) CIC IoT dataset 2023. University of New Brunswick. <https://www.unb.ca/cic/datasets/iotdataset-2023.html>. Accessed 14 Aug 2026.
- [22] Canadian Institute for Cybersecurity (2018) CSE-CIC-IDS2018 on AWS. University of New Brunswick. <https://www.unb.ca/cic/datasets/ids-2018.html>. Accessed 14 Aug 2026.
- [23] Jolliffe IT, Cadima J (2016) Principal component analysis: a review and recent developments. *Philos Trans R Soc A* 374:20150202.
- [24] Breiman L (2001) Random forests. *Mach Learn* 45(1):5–32.
- [25] Chen T, Guestrin C (2016) XGBoost: a scalable tree boosting system. In: *Proceedings of KDD*, pp 785–794.
- [26] Havlíček V et al (2019) Supervised learning with quantum-enhanced feature spaces. *Nature* 567:209–212.
- [27] Schuld M, Killoran N (2019) Quantum machine learning in feature Hilbert spaces. *Phys Rev Lett* 122:040504.

Statements and Declarations

Funding: This research received no external grant, commercial sponsorship, or other third-party funding. The project was self-funded. CQUniversity high-performance computing resources were used occasionally as computational infrastructure and did not constitute external research funding.

Competing interests: The authors declare no financial, personal, or other competing interests that could have influenced the work reported in this manuscript.

Author contributions: Fuad Ibne Jashim Farhad: conceptualization, methodology, software, investigation, formal analysis, data curation, visualization, validation, and writing—original draft. MD Mamunur Rashid: primary supervision, conceptual guidance, methodology oversight, validation, project administration, and writing—review and editing. Ali Shahidinejad: co-supervision, methodological guidance, validation, critical review, and writing—review and editing. All authors contributed to interpretation and refinement of the study.

Data Availability: The raw benchmark datasets are publicly available from the sources cited in the manuscript. Experiment scripts and configurations, processed split indices, per-seed outputs, and generated evaluation artefacts are intended to be shared with the submission/reproducibility package. Raw datasets remain subject to the access and licensing conditions of their original providers.