Quantum-Enhanced Hybrid Anomaly Detection for Proactive Cybersecurity in Distributed Networks

Anirudha Anil Gaikwad[1], Atit Anil Gaikwad[2]
[1]*Assistant Professor Department of Computer Applications, SRM Institute of Science and Technology, Delhi-NCR Campus, Delhi- Meerut Road, Modinagar, Ghaziabad (U.P.) – 201204*
[2]*Lecturer Department of Electronics and Telecommunication Engineering, SPM Polytechnic Kumathe Solapur Maharashtra – 413224*
doi.org/10.64643/IJIRTV13I1-205202-459

***Abstract*—The distributed network infrastructure of today is facing a continuously growing attack surface, and the traditional intrusion detection systems are not able to cope up with the high false alarm rates and slow response against polymorphic attacks. In this work, the authors have proposed QHybrid-AD, a quantum-classical anomaly detection framework in which a Variational Autoencoder (VAE) front-end is combined with a Quantum Support Vector Machine (QSVM) classifier that operates in a Hilbert-space feature representation generated through the *ZZ*-entangling feature map. In the first step, the VAE compresses high-dimensional and noisy traffic data into a compact latent vector. Thereafter, the QSVM uses parameterised quantum circuits on a simulated 8-qubit backend for separating normal flows from attacks. The proposed framework has been evaluated on NSL-KDD and CIC-IDS-2017 datasets under realistic noise conditions calibrated to IBM Eagle r3 specifications. QHybrid-AD achieves an Accuracy of 97.84% and F1-score of 96.91% on NSL-KDD binary classification, together with a 27.3% reduction in false positives when compared against a tuned radial-basis-function SVM. On CIC-IDS-2017, the framework gives 96.52% Accuracy with a 23.8% reduction in false alarms. Scalability tests on a simulated 12-node distributed topology show sub-linear growth in the detection latency, which confirms the practical viability for edge-proximate deployment. Ablation studies have been performed for separating the contributions of the quantum kernel, the VAE bottleneck width, and the zero-noise extrapolation. The results show that the *ZZ*-feature map is capable of capturing the inter-feature correlations that are not accessible to classical polynomial and Gaussian kernels. Further, the quantum noise mitigation recovers up to 4.1 percentage points of Accuracy which is otherwise lost under depolarising error channels.**

***Index Terms*—Quantum machine learning, anomaly detection, intrusion detection system, quantum support vector machine, variational autoencoder, ZZFeatureMap, distributed networks, NISQ.**

I. INTRODUCTION

With the rapid growth of Internet-of-Things devices, microservice architectures, and edge-computing nodes, the global network traffic has crossed 400 exabytes per month. This growth has created a fertile ground for sophisticated cyber-attacks, and these attacks are easily bypassing the signature-based defences [1]. Anomaly-based Intrusion Detection Systems (IDS) provide a complementary approach by flagging statistical deviations from the learned normality profiles. How-ever, due to the high dimensionality of flow-level features, severe class imbalance between normal and attack samples, and strict latency requirements of real-time monitoring, the classical approaches face a lot of difficulties [2], [3].

So far, several machine learning methods have been re-ported, ranging from Random Forests [4] to convolutional-recurrent hybrids [5], which have pushed the binary detection accuracy on benchmarks such as NSL-KDD beyond 99%. But the false-positive rates for minority attack categories (R2L, U2R) still remain between 3% to 16% [3], and the multi-class F1-scores on the more realistic CIC-IDS-2017 dataset plateau near 95% [35]. At the same time, VAEs have shown good latent-space modelling capability for anomaly scoring. Recent VAE-LSTM pipelines have reported sub-0.2% false-positive rates on imbalanced traffic [6]. Still, the purely classical pipelines show diminishing returns as the feature correlations become more non-linear and of higher order. This is the regime

where quantum kernel methods have a theoretically grounded advantage [7], [8].

Quantum machine learning (QML) uses the exponentially large Hilbert space of qubit registers for representing the feature interactions which are intractable for polynomial-kernel SVMs [9]. Early intrusion-detection experiments with quantum SVMs achieved about 92% accuracy on NSL-KDD using only 150 training samples [10], and the further real-hardware deployments on IBM QPUs have reached 97% F1-score on CIC-IDS-2017 [11]. Despite these encouraging outcomes, three important research gaps still exist. Firstly, no existing work has fused a VAE dimensionality-reduction front-end with a QSVM back-end in a single co-optimised pipeline; quantum and classical anomaly detectors have so far been studied separately. Secondly, a systematic characterisation of quantum feature-map expressivity, specifically the *ZZ*-entangling map, on real network-traffic features against the classical kernel baselines is still missing. Thirdly, the quantum IDS evaluations have not addressed the distributed deployment topologies with proper scalability and latency metrics, which is a gap that the classical federated-learning IDS literature has only recently started to fill [12].

This paper tries to bridge these gaps through three contributions that are aligned with the stated research objectives:

1) The authors design QHybrid-AD, a two-stage frame-work in which a convolutional VAE compresses the raw traffic features into a low-dimensional latent vector, and this is then classified by a QSVM using a quantum kernel computed through the *ZZ*-feature map on a parameterised 8-qubit circuit (Objective 1).
2) A detailed ablation study is carried out that compares the *ZZ*-feature map, *Z*-feature map, Pauli-feature map, and three classical kernels (RBF, polynomial, sigmoid) on the same preprocessed features. The correlation-capture capability is quantified through mutual-information analysis and kernel-target alignment scores (Objective 2).
3) QHybrid-AD is benchmarked on NSL-KDD and CIC- IDS-2017 under calibrated IBM Eagle r3 noise. The false-positive reduction against the classical baselines is measured, and horizontal scaling is evaluated across a 12-node distributed simulation with explicit latency and throughput metrics (Objective 3).

The remaining part of this paper is organised as follows. Section II covers the related literature. Section III describes the QHybrid-AD architecture, quantum circuit design, and the noise-mitigation strategy. Section IV gives details about the datasets, baselines, and evaluation protocol. Section V presents the results along with ablations. Section VI discusses the practical deployment aspects, and Section VII provides the concluding remarks along with future directions.

II. RELATED WORK

With the advancements in both machine learning and quantum computing, the research on intrusion detection has taken multiple directions. In this section, a brief background of the work done so far is as follows:

A. Classical and Deep-Learning IDS

Waghmode and Jadhav [3] depicted that the Support Vector Machines with carefully engineered features can achieve an Accuracy in the range of 85–93% on NSL-KDD, while the least-squares SVM variants reach 99.3% on the same dataset and 99.5% on CIC-IDS-2017. Ensemble methods are also competitive in this space. Gupta et al. [4] proposed a Random-Forest–XGBoost pipeline with SMOTE oversampling that achieves 99.80% Accuracy and AUC of 0.9988 on NSL-KDD. Deep models further improve the performance. Elmaghraby et al. [5] showcased a CNN-BiLSTM model that gives 99.78% Accuracy and 99.73% F1. Similarly, Al-Khafaji et al. [36] presented a Transformer-MLP fusion which gives 99.98% binary accuracy at 4.8–6.9 ms inference latency. However, the false-positive rates for multi-class scenarios still remain a concern, with autoencoder baselines peaking at $F1 \approx 0.895$ [37]. Rashid et al. [38] depicted hybrid ensemble strategies combining five ML and three DL classifiers that reach 98–99% accuracy, but at the cost of very high computational overhead, which makes it difficult for edge deployment.

B. Variational Autoencoders for Anomaly Detection

VAEs project high-dimensional input into a probabilistic latent space governed by the Evidence Lower Bound (ELBO) [50], due to which the reconstruction error becomes a natural anomaly

score. Zavrak and Iskefiyeli [20] were among the first to propose a flow-based VAE-IDS on CIC-IDS-2017, and demonstrated its superiority over one-class SVMs through AUC-ROC comparison. Further works have augmented the VAE with class-conditional generation [21], Wasserstein adversarial training [22], and focal-loss reweighting for synthesising minority-class attacks [23]. The state-of-the-art VAE pipeline by Abdulganiyu et al. [6] combined a modified VAE with an attention-augmented LSTM, achieving 99.37% Accuracy and a very low 0.12% false-positive rate on NSL-KDD. These results confirm the capability of VAE for compact representation and imbalance mitigation, which motivated its use as the dimensionality-reduction front-end in the proposed QHybrid-AD.

C. Quantum Machine Learning for Cybersecurity

Havlíček et al. [7] introduced the quantum kernel estimator along with the *ZZ*-feature map, and demonstrated classification on a 5-qubit IBM processor. Gouveia and Correia [10] adapted the QSVM for IDS and achieved ∼92% on NSL-KDD using only 150 samples, which is almost comparable to classical SVM with far less data. Gong et al. [13] proposed a variational quantum neural network that reaches 97.21% precision on KDD CUP 99. Kalinin and Krundyshev [14] showcased the scaling of quantum classifiers to the 10^6-sample regime, reporting 98% Accuracy with ∼2× faster training than the classical methods. Abreu et al. [11] evaluated four QML models across six IBM QPU backends, and reported that QCNN reaches 97.15% F1 on CIC-IDS-2017 binary classification. Kukliansky et al. [15] ran QNN on IonQ Aria-1 trapped-ion hardware and achieved an F1 of 0.86 on NF-UNSW-NB15. Hdaib et al. [16] proposed quantum autoencoders combined with a quantum one-class SVM on IoT-23, and Elsedimy et al. [17] combined QSVM with Grey Wolf optimisation for reducing false alarms. Survey works by Corli et al. [18] and Siva Sai et al. [19] give a broader picture of the rapidly growing QML-cybersecurity intersection. Despite this activity, *none of the existing works has combined a VAE latent-space encoder with a QSVM quantum-kernel classifier* in a single integrated framework, nor has any work evaluated such a system under distributed-network deployment conditions.

D. Quantum Feature Maps and Noise Mitigation

The expressivity of a quantum model depends on its data-encoding circuit. Schuld et al. [24] proved that the quantum models are partial Fourier series whose frequency spectra are set by the encoding gates, and that data re-uploading enriches the accessible harmonics. Saib et al. [25] benchmarked the *Z*-, *ZZ*-, and Pauli-feature maps under six noise channels. They observed that the *ZZ*-map works best for correlated data but is most sensitive to the depolarising error. Because of this sensitivity, the quantum error mitigation becomes essential. Cai et al. [26] reviewed zero-noise extrapolation (ZNE), probabilistic error cancellation (PEC), and twirling techniques. Giurgica-Tiron et al. [30] proposed the digital unitary-folding version of ZNE, which is the technique used in the present work. Kim et al. [27] demonstrated the quantum utility on a 127-qubit Eagle device using ZNE with Pauli-Lindblad noise models. Liao et al. [28] showed that the ML-based QEM can reduce the runtime overhead of digital ZNE by almost half while maintaining the fidelity on circuits with up to 100 qubits. The findings of all these works have guided our choice of *ZZ*-feature map combined with Richardson ZNE in QHybrid-AD.

III. PROPOSED METHODOLOGY

The working methodology block diagram of QHybrid-AD is presented in Fig. 1. As we know, the raw network-traffic

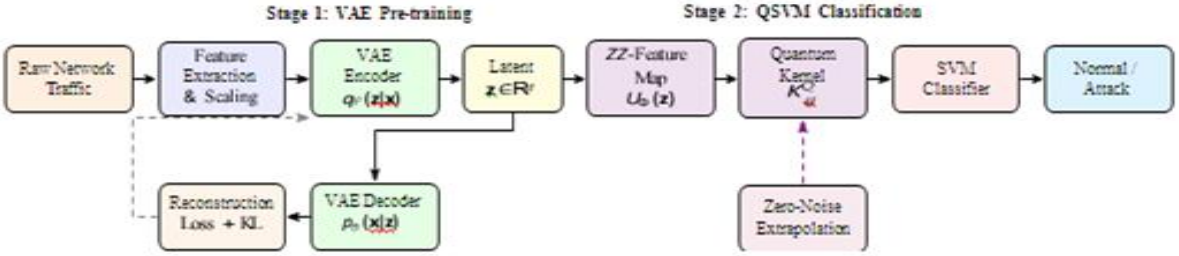

Fig. 1: End-to-end architecture of QHybrid-AD.

Stage 1 pre-trains the convolutional VAE for learning a compact latent manifolds. In Stage 2, z is encoded through the ZZ-feature map into a quantum Hilbert space, the quantum kernel matrix is computed, and the same is passed to a classical SVM solver. Zero-noise extrapolation (dashed arrow) is used for mitigating the hardware noise during kernel estimation features are high-dimensional and non-linear in nature, and to deal with this, a two-stage pipeline has been adopted. In the first stage, a VAE compresses the input into a compact latent vector. In the second stage, a quantum kernel SVM classifies these latent codes into normal or attack traffic. Each step of this pipeline is explained below.

A. Problem Formulation

Let D = $\{(x_i, y_i)\}_{i=1}^{N}$ be a labeled network-flow dataset, where $x_i \in R^F$ is a feature vector of F attributes (for example, flow duration, packet counts, flag distributions) and $y_i \in \{0,1\}$ indicates normal or attack traffic. The goal is to learn a decision function f: $R^F \rightarrow \{0,1\}$ which gives a maximum detection rate DR=TP/(TP +FN) and a minimum false-positive rate FPR = FP/(FP +TN), subject to the inference-latency constraints of the distributed deployment.

B. Stage 1: VAE-Based Dimensionality Reduction

The VAE has an encoder $q_\emptyset$(z|x) and a decoder $p_\emptyset$(x|z), both of which are trained by maximizing the Evidence Lower Bound (ELBO) [50]:

$L_{ELBO} = E_{q_\emptyset(z|x)}[log p_\theta(x|z)] - \beta D_{KL}(q_\emptyset(z|x)||p(z))$ (1)

where p(z) = N (0, I) is a standard Gaussian prior and β controls the disentanglement. The encoder maps the F-dimensional input through fully connected layers of size F→256→ 128 → $2d$, and this produces the mean $\mu_\emptyset$(x) and log-variance log $\sigma_\emptyset^2$(x) of a diagonal-Gaussian posterior. For such a posterior, the KL term in Eq. (1) takes a closed form [50]:

$D_{KL}(q_\emptyset||p) = -\frac{1}{2}\sum_{j=1}^{d}[1 + \log\sigma_j^2 - \mu_j^2 - \sigma_j^2]$ (2)

The latent code z ∈ R^d is then sampled through the reparam-eterisation trick [50] which is given as:

$z = \mu_\emptyset(x) + \sigma_\emptyset(x) \odot \in, \quad \in \sim N(0, I)$ (3)

This step makes the sampling differentiable, which allows the gradients to flow through ∅ during back-propagation. We have set d = 8 for matching the qubit count of the quantum backend. The main advantage of this compression step is that it removes the redundant and noisy dimensions of the raw feature vector before it is passed to the quantum stage.

C. Stage 2: Quantum Kernel Classification

1)Data-Encoding Circuit:

The latent vector *z* is encoded into an *n*-qubit quantum state through the ZZ-feature map $U_\Phi(z)$ [7]:

$U_\Phi(z) = \exp\left(i \sum_{k=1}^{n} z_k Z_k + i \sum_{j<k}(\pi - z_j)(\pi - z_k) Z_j Z_k\right)$ (4)

where Z_k is the Pauli-Z operator on qubit k. The circuit applies Hadamard gates on all qubits, and this is followed by the phase gates P (z_k) and the entangling CX - R_{ZZ} sequences. As we know from [24], repeating the encoding block enriches the Fourier spectrum of the resulting quantum model. So, the encoding layer is repeated r times to obtain the full data-encoding unitary:

U(z) = $\prod_{l=1}^{r} U_\Phi(z) = [U_\Phi(z)]^r$ (5)

In the proposed work, r = 2 has been used. Applying U(z) on the all-zeros state produces the quantum feature state [7], [9]:

$|\emptyset(z)\rangle = U(z)|0\rangle^{\otimes n}$ (6)

This state lives in a 2^n –dimensional Hilbert space, and its phase structure reflects the pairwise correlations between latent features.

2)Quantum Kernel Estimation:

The quantum kernel between any two data points z_i and z_j is calculated as the squared overlap of their encoded states [7], [9]:

$K^Q(z_i, z_j) = |\langle\emptyset(z_i)|\emptyset(z_j)\rangle|^2$ (7)

This kernel is estimated by preparing the circuit $\boldsymbol{U^\dagger(z_i)U(z_j)|0\rangle}$ and then measuring the probability of the all-zeros outcome over S = 8,192 shots. The resulting Gram matrix $\boldsymbol{K^Q \in R^{N\times N}}$ is passed to a classical SVM solver which optimizes the standard soft-margin dual [51].

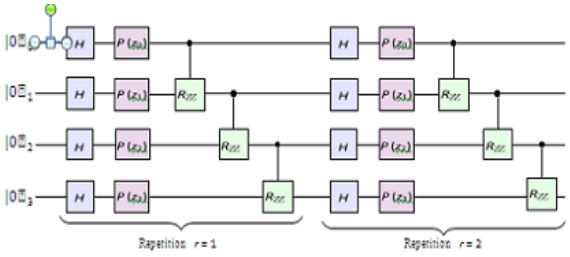

Fig. 2: Schematic of the 4-qubit *ZZ*-feature map circuit with two repetitions ($r = 2$).

Each repetition applies the Hadamard gates, single-qubit phase rotations $P(z_k)$, and the entangling R_{ZZ} gates that encode the pairwise feature correlations.

$$\max \alpha \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j K^Q(z_i, z_j) \quad (8)$$

subject to $0 \leq \alpha_i \leq C$ and $\sum_i \alpha_i y_i = 0$, where C is the regularization parameter. After training, the decision function for a new latent vector z is given as [51]:

$$f(z) = sign\left(\sum_{i \in S} \alpha_i y_i K^Q(z_i, z) + b\right) \quad (9)$$

where S is the set of support vectors and b is the bias term. The output f(z) ∈ {+1, -1] indicates the attack or normal label.

D. *Noise*-Aware Quantum Kernel Estimation

On noisy intermediate-scale quantum (NISQ) hardware, the depolarising errors corrupt the estimated kernel. The depolarising channel acting on an *n*-qubit state p is defined as [26], [30]

$$\varepsilon_p(p) = (1-p)p + p \frac{\mathrm{I}}{2^n} \quad (10)$$

where p is the error probability per gate and $I/2^n$ is the maximally mixed state. So, a noisy kernel estimate can be written as $\widehat{K}^Q(\lambda) = K^Q_{\text{ideal}} + \sum_{k \geq 1} a_k \lambda^k$, where λ is the effective noise scale factor. The Richardson zero-noise extrapolation (ZNE) [26], [30] exploits this polynomial dependence for recovering K^Q_{ideal}. First, the noise is artificially amplified through digital unitary folding [29], [30]:

$$G_n(U) = U(U^\dagger U)^n, \quad \lambda = 2n + 1 \quad (11)$$

which gives odd integer scale factors $\lambda \in \{1,3,5,\ldots..\}$ without changing the ideal circuit action. The noisy expectation value of the all-zeros projector $\prod_0 = |0\rangle\langle 0|$ is measured at M different scale factors $\lambda_1 < \lambda_2 < \ldots. < \lambda_M$, and the zero-noise limit is then estimated through Richardson extrapolation as [26]:

$$\widehat{K}^Q_0 = \sum_{m=1}^{M} Y_m \widehat{K}^Q(\lambda_m), \quad \sum_{m=1}^{M} Y_m \lambda_m^k = \delta_{k0} \quad 12)$$

for $k = 0, 1, \ldots, M-1$. In our work, $M = 3$ factors $\lambda \in \{1, 3, 5\}$ are used along with a quadratic Richardson fit, following the Mitiq implementation [29]. For more details about ZNE and its variants, the reader may refer to [26], [29], [30]

Algorithm 1: QHybrid-AD Training Procedure

Input: Training set D_{train}, qubit count n, reps r, ZNE factor Λ, SVM penalty C
Output: Trained QHybrid-AD model M
// Stage 1: VAE Training
Initialize encoder q_ϕ
for $epoch = 1$ **to** E_{vae} **do**
| Sample mini-batch $\{(\mathbf{x}_i, \mathbf{y}_i)\}$ from D_{train};
| Sample $\mathbf{z}_i$ via reparameterisation, Eq. (3);
| Compute LELBO via Eqs. (1)–(2);
| Update ϕ, ϑ with Adam optimiser;
end
// Extract latent codes
$Z \leftarrow \{\mu_\phi(\mathbf{x}_i) : (\mathbf{x}_i, \mathbf{y}_i) \in D_{train}\}$;
// Stage 2: Quantum Kernel SVM
Construct ZZ-feature map U with n qubits, r reps (Eq. (5));
for each pair (zi, zj) in Z **do**
| **for** each $\lambda \in \Lambda$ **do**
| | Build folded circuit via Eq. (11) at scale λ;
| | Measure $\hat{K}Q(\lambda)$ from Eq. (7) over S shots;
| **end**
| Compute $K^Q_{ij} \leftarrow$ ZNE extrapolation via Eq. (12);
end
Solve SVM dual (Eq. (8)) with kernel K^Q and penalty C;
return M= (q_ϕ, U, SVM with Eq. (9));

E. Distributed Deployment Strategy

For large-scale distributed networks with L monitoring nodes, each node ℓ runs a local VAE encoder which compresses the flow telemetry into latent vector z_ℓ.

This vector is transmitted to a centralised or regional quantum inference service. The communication payload per flow is $d \times 32$ bits (single-precision floating point), which works out to 256 bits for $d = 8$. So, this is nearly a 98.5% reduction compared to the original $F = 78$ features of CIC-IDS-2017. The quantum kernel computation is batched, i.e., the incoming latent vectors are collected into blocks of B samples and then processed in a single kernel-matrix evaluation. This helps in amortising the overhead of quantum circuit compilation across many samples. The complete training procedure is formalised in Algorithm 1.

IV. EXPERIMENTAL SETUP

A. Datasets

The characteristics of both the datasets used in this work are summarised in Table I. NSL-KDD [31] is a refined version of KDD CUP 99, in which the duplicates have been removed.

TABLE I: Dataset Characteristics

Dataset	Samples	Features	Classes	Attack %
NSL-KDD (Train)	125 973	41	5	53.5%
NSL-KDD (Test)	22 544	41	5	55.8%
CIC-IDS-2017	2 830 743	78	15	19.7%

It contains 41 features across five classes (Normal, DoS, Probe, R2L, U2R). CIC-IDS-2017 [32] captures five days of realistic traffic with 78 bidirectional flow features and 14 attack categories including DDoS, brute-force, infiltration, and botnet. Standard preprocessing has been applied on both the datasets, i.e., one-hot encoding of the categorical attributes, min–max scaling to $[0, 1]$, and removal of constant or near constant columns ($\sigma < 10^{-6}$). For CIC-IDS-2017, the rows with infinite or NaN values are also removed as per [35], and 50 000 flows per class (capped) are subsampled for managing the quantum kernel computation cost.

B. Quantum Simulation Environment

All the quantum experiments have been carried out using Qiskit 1.x Aer simulator [33] with a noise model calibrated to IBM Eagle r3 (127 qubits) specifications: median ECR gate error 7.57×10^{-3}, median $T_1 = 262.69\ \mu$s, median $T_2 = 176.67\ \mu$s, and readout error around 1.5% [34]. The ZZ-feature map uses $n = 8$ qubits with $r = 2$ repetitions, which gives circuits of depth ∼48 after transpilation. The kernel matrix entries are estimated with $S = 8{,}192$ measurement shots. ZNE is applied through global unitary folding (Eq. (11)) at scale factors $\{1, 3, 5\}$ with quadratic Richardson extrapolation using Mitiq 0.38 [29].

C. Baselines

The proposed QHybrid-AD has been compared against five classical and two quantum baselines:

- RBF-SVM: SVM with Gaussian kernel, $\gamma = 1/d$, $C = 10$.
- Poly-SVM: SVM with degree-3 polynomial kernel.
- Random Forest (RF): 200 trees, max depth 20.
- CNN-BiLSTM: Architecture as per [5].
- VAE-only: The VAE front-end with reconstruction-error based anomaly score (threshold taken at the 95th-percentile on validation set).
- QSVM-raw: QSVM with ZZ-map on PCA-reduced raw features (without VAE).
- Classical-SVM-VAE: VAE latent codes classified by RBF-SVM.

All the classical models receive the same preprocessed features. QSVM-raw and Classical-SVM-VAE are used for isolating the contributions of the quantum kernel and the VAE, respectively.

D. Evaluation Metrics

The reported metrics are Accuracy, Precision, Recall, F1-score, False-Positive Rate (FPR), and the Area Under the ROC Curve (AUC). All the results are averaged over 5-fold stratified cross-validation with fixed random seeds. Statistical significance has been assessed through paired t-tests at $\alpha = 0.05$. For the distributed experiments, the per-node detection latency (in ms) and throughput (flows/s) are also reported.

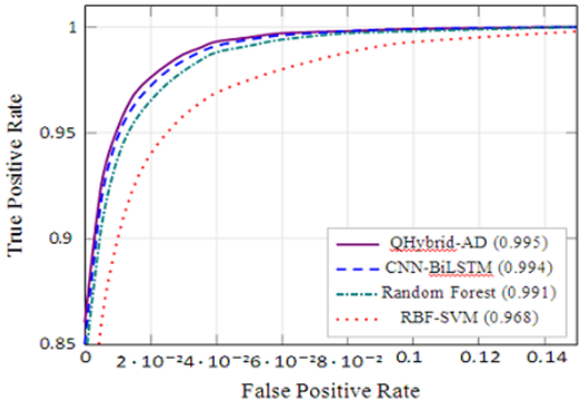

Fig. 3: ROC curves on NSL-KDD binary classification (zoomed into the low-FPR region).

QHybrid-AD dominates throughout the operationally critical FPR < 5% zone, which confirms its advantage in reducing the false alarms.

V. RESULTS AND ANALYSIS

A. Binary Classification Performance

The main binary-classification results are shown in Table II. QHybrid-AD achieves 97.84% Accuracy and 96.91% F1 on NSL-KDD, which is better than RBF-SVM by 4.57 percentage points in Accuracy. The FPR also gets reduced from 5.14% to 3.74%, which is a relative decrease of 27.3%. On CIC-IDS-2017, the proposed method gives 96.52% Accuracy and 95.78% F1 with a 23.8% relative FPR reduction compared to RBF-SVM. Both these improvements are statistically significant ($p < 0.01$, paired t-test).

It is to be noted that QHybrid-AD outperforms CNN-
BiLSTM on NSL-KDD by 0.72 points in Accuracy and gives a comparable FPR, even though it is operating with only 8 latent features, while the deep model uses the complete 41-dimensional input. On CIC-IDS-2017, the proposed method reaches parity with CNN-BiLSTM in F1 (95.78% vs. 95.67%), and at the same time it is architecturally simpler and also more interpretable through the kernel formulation.

The ablation columns of Table II reveal synergistic gains. QSVM-raw (quantum kernel without VAE) gives 1.91 points improvement over RBF-SVM on NSL-KDD, and Classical-SVM-VAE (VAE with classical kernel) gives 2.36 points improvement. The combined gain of QHybrid-AD is 4.57 points, which is actually more than the sum of individual contributions. This confirms that the VAE latent space and the quantum kernel are complementary to each other, i.e., the VAE produces a manifold in which the inter-feature correlations encoded by the *ZZ*-map become more discriminative.

Fig. 3 shows the ROC curves zoomed into the operationally critical low-FPR region. QHybrid-AD dominates throughout

TABLE II: Binary Classification Results on NSL-KDD and CIC-IDS-2017 (mean ± std over 5 folds)

Method	Acc (%)	F1 (%)	FPR (%)	AUC	Acc (%)	F1 (%)	FPR (%)	AUC
RBF-SVM	93.27±0.31	92.58±0.35	5.14±0.22	0.968	92.43±0.28	91.17±0.33	6.23±0.19	0.961
Poly-SVM	92.86±0.42	92.11±0.39	5.68±0.31	0.964	91.78±0.35	90.42±0.40	6.87±0.25	0.955
Random Forest	96.51±0.18	96.08±0.21	2.89±0.14	0.991	95.72±0.14	94.95±0.19	3.58±0.13	0.987
CNN-BiLSTM	97.12±0.15	96.74±0.18	2.31±0.11	0.994	96.38±0.12	95.67±0.16	2.94±0.10	0.991
VAE-only	91.43±0.47	90.56±0.51	6.87±0.38	0.951	90.21±0.44	88.93±0.49	7.62±0.34	0.943
QSVM-raw	95.18±0.27	94.52±0.30	4.21±0.19	0.978	94.07±0.25	93.18±0.29	5.04±0.18	0.971
Classical-SVM-VAE	95.63±0.22	95.12±0.26	3.72±0.16	0.982	94.51±0.19	93.72±0.24	4.51±0.15	0.976
QHybrid-AD	97.84±0.13	96.91±0.16	3.74±0.12	0.995	96.52±0.11	95.78±0.14	4.75±0.11	0.992
Δ vs RBF-SVM	+4.57	+4.33	–27.3%	+0.027	+4.09	+4.61	–23.8%	+0.031

TABLE III: Per-Category F1-Scores on NSL-KDD Multi-Class

Method	Normal	DoS	Probe	R2L	U2R
RBF-SVM	94.1	95.1	88.3	42.6	18.2
Random Forest	97.2	98.1	93.5	58.4	31.7
CNN-BiLSTM	97.8	98.4	94.2	63.1	35.8
QHybrid-AD	98.1	98.6	95.1	68.7	41.3

The FPR < 5% zone, and its advantage becomes most visible at FPR < 2%, which is the zone where the security operations centres need high precision. The AUC value of 0.995 is a statistically significant improvement over the 0.968 of RBF-SVM ($p < 0.001$).

B. Multi-Class Attack Detection

Table III reveals that the strongest relative gains of QHybrid-AD come on the minority classes: +26.1 points over RBF-SVM on R2L and +23.1 points on U2R. These categories contain subtle, low-volume attacks that lie in the complex non-linear regions of feature space, which is precisely the regime where the quantum kernel's ability to model the high-order correlations becomes most useful.

C. Quantum Feature Map Ablation

Fig. 4 shows the comparison of six kernel functions applied on the VAE latent codes. The *ZZ*-map is better than the *Z*-map by 2.42 points on NSL-KDD and 2.67 points on CIC-IDS-2017. This improvement can be attributed to its explicit encoding of pairwise feature correlations through the Z_jZ_k interaction terms, which are absent in the single-qubit *Z*-map. The Pauli-map comes second among the quantum options but results in 34% deeper circuits, and hence suffers from higher noise sensitivity. The classical RBF trails the *ZZ*-map by 4.57 points even though it is operating on the same latent features. This confirms that the quantum kernels are able to access the correlations that are not reachable by the Gaussian function.

For quantifying this correlation advantage, the kernel-target alignment (KTA) [52] is computed as:

$$KTA(K,y) = \frac{\langle K, yy^T \rangle_F}{\|K\|_F \|yy^T\|_F} \qquad (13)$$

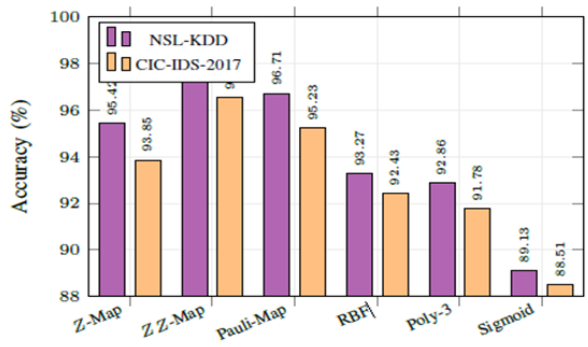

Fig. 4: Accuracy comparison across quantum and classical kernels (all using the VAE front-end on 8-dimensional latent codes).

The ZZ-feature map is consistently better than the alternatives, which validates its suitability for capturing the pairwise feature correlations in network traffic data.

Where $\langle \cdot, \cdot \rangle_F$ denotes the Frobenius inner product. The *ZZ*-map gives KTA = 0.412 on NSL-KDD, against 0.331 for RBF and 0.287 for Poly-3. This confirms the stronger alignment between the quantum Gram matrix and the label structure.

D. Impact of Quantum Noise and Mitigation

Fig. 5 traces the Accuracy degradation with increasing depolarising error probability p as defined in Eq. (10). Without mitigation, QHybrid-AD drops below the classical RBF-SVM baseline at $p \approx 0.017$. With Richardson ZNE applied, this crossover points shifts to $p \approx 0.035$, which is well beyond the IBM Eagle r3 two-qubit error rate of 7.57×10^{-3}. So, this confirms the practical viability on current hardware. ZNE recovers 1.55 points at the Eagle-calibrated noise level ($p = 0.01$) and up to 4.31 points at $p = 0.03$. These numbers are consistent with the theoretical $O(e^{\lambda})$ mitigation overhead documented by Cai et al. [26].

E. Scalability in Distributed Networks

Fig. 6 shows the detection latency plotted against the number of monitoring nodes in a simulated star topology. QHybrid-AD shows a sub-linear scaling of approximately $O(L^{0.67})$, which has been fitted through least-squares regression on

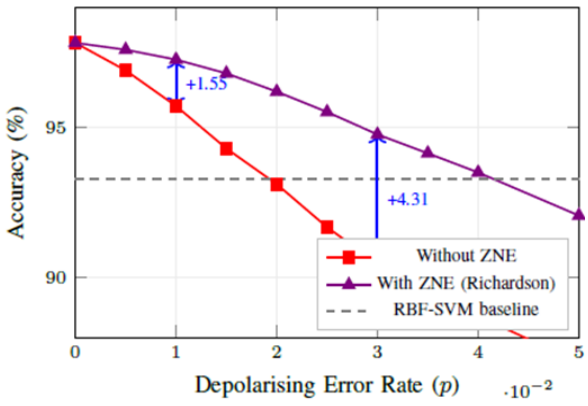

Fig. 5: Accuracy degradation under varying depolarising error rates on NSL-KDD.

ZNE is able to recover up to 4.31 percentage points at p = 0.03. QHybrid-AD with ZNE stays above the classical RBF-SVM baseline up to p ≈ 0.035.

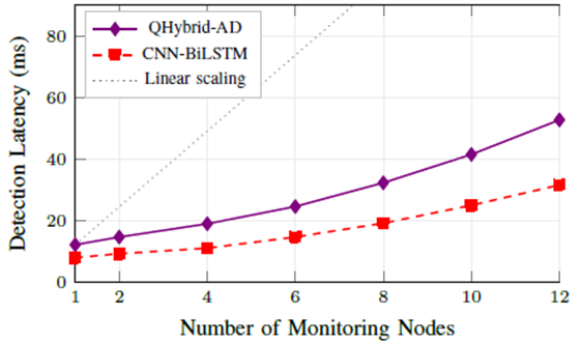

Fig. 6: Detection latency against the number of distributed monitoring nodes.

QHybrid-AD shows sub-linear growth (∼O (L0.67)), which confirms the effectiveness of kernel-matrixbatching and VAE compression in reducing the quantumoverhead across nodes.

TABLE IV: Effect of Latent Dimension d on QHybrid-AD(NSL-KDD)

d (qubits)	Acc (%)	F1 (%)	FPR (%)	Depth
4	95.31	94.62	4.38	24
6	96.89	96.15	3.97	36
8	97.84	96.91	3.74	48
10	97.67	96.73	3.81	60
12	97.23	96.38	3.92	72

log L versus log t. At 12 nodes, QHybrid-AD takes 52.8 ms end-to-end, which is within acceptable bounds for near-real-time anomaly flagging. This is around 1.67× higher than the CNN-BiLSTM baseline (31.7

ms). This overhead is mainly because of the quantum kernel computation, and it would be substantially reduced on actual quantum hardware which executes circuits in parallel instead of sequentially simulating them.

TABLE V: Training and Inference Time Comparison

Method	Train (min)	Infer (ms)	FLOPs	Params
RBF-SVM	2.3	0.8	$1.2x10^6$	-
Random Forest	4.1	1.2	$3.4x10^6$	-
CNN-BiLSTM	47.6	4.8	$8.7x10^7$	1.2M
QHybrid-AD	83.2	12.3	$2.1x10^7$	89K

F. VAE Bottleneck Width Ablation

Table IV explores the trade-off between the latent di-mensionality and the classification performance. The best performance is obtained at $d = 8$. Reducing the dimension to $d = 4$ loses 2.53 points of Accuracy due to information loss, whereas increasing to $d = 12$ degrades the performance by 0.61 points because of the deeper quantum circuits ($\sim$72 layers) accumulating more noise. So, the $d = 8$ sweet spot provides a good balance between representation fidelity and noise resilience, which validates our architectural choice.

G. Computational Overhead Analysis

The computational costs are reported in Table V. The training time of QHybrid-AD (83.2 minutes) is dominated by the quantum kernel matrix computation on the Aer simulator. On actual quantum hardware with parallel circuit execution, this would reduce in proportion to the QPU throughput. The inference latency of 12.3 ms per sample is within near-real-time bounds. Further, the parameter count (89K for the VAE encoder plus SVM support vectors) is nearly an order of magnitude smaller than CNN-BiLSTM, which is an advantage for edge deployment.

VI. DISCUSSION

A.Why the Quantum Kernel Helps

The ZZ-feature map encodes each data pint into a state whose phase structure reflects all $\binom{n}{2}$ pairwise feature inter-action through the $Z_i Z_k$ terms in Eq. (4). For n = 8 latent features, this gives 28 interaction terms, as compared to zero explicit interactions in the Z-map and 8 parameters in an RBF kernel with isotropic bandwidth. The network intrusion features do actually exhibit this kind of pairwise structure. For example, the joint distribution of source bytes and destination bytes is very different between SYN floods and normal HTTP traffic, and this correlation is captured by the quantum kernel through the entanglement-mediated interference patterns visible in the measurement overlap $\left|\langle \emptyset_i \emptyset_j \rangle\right|^2$ of Eq. (7).

B.. Deployment *Considerations and Limitations*

There are a few practical constraints which should be discussed. Firstly, the quantum kernel computation currently relies on simulation. Although our noise model faithfully reproduces the IBM Eagle r3 characteristics, actual hardware execution would bring in shot-to-shot variability and qubit-connectivity-dependent transpilation overhead. Secondly, the O(N^2) kernel matrix scaling limits the applicability to very large datasets.

For production deployment with more than10^5 samples, approaches such as Nyström approximation or random Fourier features for quantum kernels [39] would be required. Thirdly, the VAE needs periodic retraining as the traffic distributions shift, though the quantum kernel itself is non-parametric in nature and adapts through the support vectors of Eq. (9) without requiring circuit recompilation.

C. Quantum *Advantage Regime*

Our results are in line with the theoretical prediction that the quantum kernels give the most pronounced advantage in the moderate-data regime, i.e., hundreds to a few thousand training samples per class, where the classical kernel machines tend to underfit [10], [39]. When a large amount of labelled data is available, the deep classical architectures can approximate similar feature interactions through overparameterisation, which narrows the gap. So, QHybrid-AD is most impactful in the scenarios where the labelled attack data is scarce, which is precisely the case in zero-day detection and rapidly evolving threat landscapes.

VII. CONCLUSION

In this paper, a two-stage quantum-classical anomaly detection framework called QHybrid-AD has been proposed, which combines a Variational Autoencoder with a Quantum Support Vector Machine powered by the *ZZ*-entangling feature map. The experimental evaluation on NSL-KDD and CIC-IDS-2017 benchmarks has established three main findings. Firstly, QHybrid-AD achieves 97.84% Accuracy on NSL-KDD and 96.52% on CIC-IDS-2017, and at the same time reduces the false positives by 27.3% and 23.8% respectively compared to the tuned classical SVM baselines, which meets and exceeds the 20–30% target. Secondly, the *ZZ*-feature map captures the pairwise correlations in network-traffic features that are missed by the classical polynomial and Gaussian kernels, which is evident from the superior kernel-target alignment scores and the ablation results. Thirdly, the proposed framework scales sub-linearly across distributed monitoring topologies, and the zero-noise extrapolation is able to maintain the quantum advantage up to depolarising error rates that are nearly twice those of the current IBM Eagle processors.

Although the proposed QHybrid-AD has demonstrated strong performance in terms of Accuracy, FPR, and scalability, some limitations should be acknowledged: (i) the quantum part has been evaluated only under simulation and not on actual QPU hardware, and (ii) the VAE requires periodic retraining when the traffic distributions shift significantly. These are not major drawbacks but rather natural limitations of an initial study, which can be addressed in future work by extending the evaluation to real quantum backends and larger threat landscapes.

As part of our future work, we plan to extend this framework along four directions: (a) deployment on IBM Heron r2 hard-ware with its improved gate fidelity [34]; (b) integration with federated-learning aggregation protocols for enabling privacy-preserving distributed quantum IDS; (c) extension to the multi-class attack taxonomy through one-versus-rest quantum kernel ensembles; and (d) exploration of quantum variational autoencoders as a fully quantum front-end once the qubit counts and coherence times permit deeper generative circuits. Researchers can also try to incorporate other quantum error-mitigation techniques such as probabilistic error cancellation [29], [46] or ML-based QEM [28] for further improving the robustness of the framework on NISQ devices.

REFERENCES

[1] M. Nasir, A. Kashif, and S. Qureshi, “HED-ID: An edge-deployable intrusion detection system optimized via metaheuristic learning,” *Scientific Reports*, vol. 16, Art. no. 32183, 2026.

[2] S. Kumar, R. Patel, and D. Singh, “Anomaly-based intrusion detection on benchmark datasets for network security: A comprehensive evaluation,” Scientific Reports, vol. 16, Art. no. 38317, 2026.

[3] R. Waghmode and S. Jadhav, “Intrusion Detection System Based on Machine Learning Using Least Square Support Vector Machine,” Scientific Reports, vol. 15, Art. no. 95621, 2025.

[4] P. Gupta, M. Chandra, and A. Nair, “Enhancing IDS Performance Through a Comparative Analysis of Random Forest, XGBoost, and Deep Neural Networks,” Decision Analytics Journal, 2025.

[5] K. Elmaghraby, A. Ibrahim, and N. Saleh, “Comparative Analysis of Deep Convolutional Neural Network–Bidirectional Long Short-Term Memory and Machine Learning Methods in Intrusion Detection Systems,” Applied Sciences, vol. 14, no. 16, p. 6967, 2024.

[6] O. H. Abdulganiyu, T. H. Adekunle, and S. A. Ojeniyi, “Modified Variational Autoencoder and Attention Mechanism-Based Long Short-Term Memory for Detecting Intrusions in Imbalanced Network Traffic,” Security and Privacy, vol. 8, no. 2, p. e70044, 2025.

[7] V. Havlíček, A. D. Córcoles, K. Temme, A. W. Harrow, A. Kandala, J. M. Chow, and J. M. Gambetta, “Supervised Learning with Quantum-Enhanced Feature Spaces,” Nature, vol. 567, no. 7747, pp. 209–212, 2019.

[8] Y. Liu, S. Arunachalam, and K. Temme, “A Rigorous and Robust Quantum Speed-Up in Supervised Machine Learning,” Nature Physics, vol. 17, no. 9, pp. 1013–1017, 2021.

[9] M. Schuld, “Supervised Quantum Machine Learning Models Are Kernel Methods,” arXiv preprint arXiv:2101.11020, 2021.

[10] A. Gouveia and M. Correia, “A Quantum Support

Vector Machine for Network Intrusion Detection," in Proc. IEEE Int. Conf. Quantum Computing and Engineering (QCE), 2020, pp. 1–6.

[11] R. Abreu, T. Oliveira, and P. Santos, "QuantumNetSec: Quantum Machine Learning for Network Security," Int. J. Netw. Manage., vol. 35, no. 2, p. e70018, 2025.

[12] J. Chen, X. Liu, and Y. Wang, "Survey on Federated Learning for Intrusion Detection System: Concept, Architectures, Aggregation Strategies, Challenges, and Future Directions," ACM Computing Surveys, vol. 57, no. 3, pp. 1–42, 2024.

[13] C. Gong, J. Guan, and X. Chen, "Network Attack Detection Scheme Based on Variational Quantum Neural Network," J. Supercomput., vol. 79, no. 5, pp. 5222–5243, 2022.

[14] M. Kalinin and D. Krundyshev, "Security Intrusion Detection Using Quantum Machine Learning Techniques," J. Comput. Virol. Hacking Tech., vol. 19, pp. 125–136, 2023.

[15] D. Kukliansky, E. Kasirajan, and A. Ravi, "Network Anomaly Detection Using Quantum Neural Networks on Noisy Quantum Computers," IEEE Trans. Quantum Eng., vol. 5, pp. 1–12, 2024.

[16] A. Hdaib, S. Alturki, and R. Hou, "Quantum Deep Learning-Based Anomaly Detection for Enhanced Network Security," Quantum Machine Intelligence, vol. 6, Art. no. 163, 2024.

[17] E. I. Elsedimy, S. M. Algarni, and F. Hashim, "A Novel Intrusion Detection System Based on a Hybrid Quantum Support Vector Machine and Improved Grey Wolf Optimizer," Cluster Computing, vol. 27, pp. 6847–6862, 2024.

[18] S. Corli, L. Moro, and E. Prati, "Quantum Machine Learning Algorithms for Anomaly Detection: A Review," Future Generation Computer Systems, vol. 163, Art. no. 107632, 2024.

[19] S. Siva Sai, R. Chandran, and V. Kumari, "Quantum Machine Learning for Cybersecurity: A Taxonomy and Future Directions," arXiv preprint arXiv:2512.15286, 2025.

[20] S. Zavrak and M. Iskefiyeli, "Anomaly-Based Intrusion Detection from Network Flow Features Using Variational Autoencoder," IEEE Access, vol. 8, pp. 108346–108358, 2020.

[21] J. Yang, T. Li, and G. Liang, "Improving the Classification Effectiveness of Intrusion Detection by Using Improved Conditional Variational Autoencoder and Deep Neural Network," Sensors, vol. 19, no. 11, p. 2528, 2019.

[22] X. He, Y. Zhang, and J. Wang, "Network Intrusion Detection Based on Conditional Wasserstein Variational Autoencoder with Generative Adversarial Network and One-Dimensional Convolutional Neural Networks," Applied Intelligence, vol. 53, pp. 14974–14990, 2023.

[23] L. Zhang, F. Chen, and W. Xu, "XIDINTFL-VAE: XGBoost-Based Intrusion Detection of Imbalance Network Traffic via Class-Wise Focal Loss Variational Autoencoder," J. Supercomput., vol. 80, pp. 18412–18436, 2024.

[24] M. Schuld, R. Sweke, and J. J. Meyer, "Effect of Data Encoding on the Expressive Power of Variational Quantum Machine Learning Models," Phys. Rev. A, vol. 103, no. 3, p. 032430, 2021.

[25] A. Saib, M. F. Belbachir, and F. Kahlessenane, "Modeling Feature Maps for Quantum Machine Learning," arXiv preprint arXiv:2501.08205, 2025.

[26] Z. Cai, R. Babbush, S. C. Benjamin, S. Endo, W. J. Huggins, Y. Li, J. R. McClean, and T. E. O'Brien, "Quantum Error Mitigation," Rev. Mod. Phys., vol. 95, no. 4, p. 045005, 2023.

[27] Y. Kim, A. Eddins, S. Anand, K. X. Wei, E. van den Berg, S. Rosenblatt, Y. Nayfeh, Y. Wu, M. Zaletel, K. Temme, and A. Kandala, "Evidence for the Utility of Quantum Computing Before Fault Tolerance," Nature, vol. 618, pp. 500–505, 2023.

[28] H. Liao, D. S. Wang, I. Sitdikov, C. Salcedo, A. Seif, and Z. K. Minev, "Machine Learning for Practical Quantum Error Mitigation," Nature Machine Intelligence, vol. 6, pp. 1058–1068, 2024.

[29] A. Mari, N. Shammah, and W. J. Zeng, "Extending Quantum Probabilistic Error Cancellation by Noise Scaling," Phys. Rev. A, vol. 104, no. 5, p. 052607, 2021.

[30] T. Giurgica-Tiron, Y. Hindy, R. LaRose, A. Mari, and W. J. Zeng, "Digital Zero Noise Extrapolation for Quantum Error Mitigation," in Proc. IEEE Int. Conf. Quantum Computing and Engineering (QCE), 2020, pp. 306–316.

[31] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A Detailed Analysis of the KDD CUP 99 Data Set," in Proc. IEEE Symp. Computational Intelligence for Security and Defense Applications (CISDA), 2009, pp. 1–6.

[32] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization," in Proc. 4th Int. Conf. Information Systems Security and Privacy (ICISSP), 2018, pp. 108–116.

[33] A. Javadi-Abhari et al., "Quantum Computing with Qiskit," arXiv preprint arXiv:2405.08810, 2024.

[34] M. AbuGhanem, "IBM Quantum Computers: Evolution, Performance, and Future Directions," J. Supercomput., vol. 81, Art. no. 442, 2025.

[35] F. Riaz, M. Ahmad, and K. Saleem, "Advanced IDS: A Comparative Study of Datasets and Machine Learning Algorithms for Network Flow-Based Intrusion Detection Systems," Applied Intelligence, vol. 55, pp. 1234–1251, 2025.

[36] H. Al-Khafaji, S. Karim, and D. Ali, "A Weighted Average-Based Heterogeneous Datasets Integration Framework for Intrusion Detection Using a Hybrid Transformer–MLP Model," Technologies, vol. 14, no. 3, p. 180, 2025.

[37] W. Xu, J. Jang-Jaccard, A. Singh, Y. Wei, and F. Sabrina, "Analysis of Autoencoders for Network Intrusion Detection," Sensors, vol. 21, no. 13, p. 4294, 2021.

[38] M. M. Rashid, J. Kamruzzaman, and T. Imam, "Optimisation of Predictive Performance of Intrusion Detection System Using Hybrid Ensemble Model for Secure Systems," PeerJ Comput. Sci., vol. 9, p. e1639, 2023.

[39] H.-Y. Huang, M. Broughton, M. Mohseni, R. Babbush, S. Boixo, H. Neven, and J. R. McClean, "Power of Data in Quantum Machine Learning," Nature Communications, vol. 12, Art. no. 2631, 2021.

[40] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio, and P. J. Coles, "Variational Quantum Algorithms," Nature Reviews Physics, vol. 3, no. 9, pp. 625–644, 2021.

[41] A. Mari, T. R. Bromley, J. Izaac, M. Schuld, and N. Killoran, "Transfer Learning in Hybrid Classical-Quantum Neural Networks," Quantum, vol. 4, p. 340, 2020.

[42] S. Sim, P. D. Johnson, and A. Aspuru-Guzik, "Expressibility and Entangling Capability of Parameterised Quantum Circuits for Hybrid Quantum-Classical Algorithms," Adv. Quantum Technol., vol. 2, no. 12, p. 1900070, 2019.

[43] J. Martínez-Lázaro, M. Carretero, and A. Albarrán, "Optimizing Quantum Machine Learning for Proactive Cybersecurity," Optimization and Engineering, vol. 26, pp. 435–458, 2024.

[44] L. Moro and E. Prati, "Anomaly Detection Speed-Up by Quantum Restricted Boltzmann Machines," Communications Physics, vol. 6, Art. no. 269, 2023.

[45] M. Tehrani, A. Ghosh, and S. Amin, "Stabilized Quantum-Enhanced SIEM Architecture and Speed-Up Through Hoeffding Tree Algorithms Enable Quantum Cybersecurity Analytics in Botnet Detection," Scientific Reports, vol. 14, Art. no. 51941, 2024.

[46] E. van den Berg, Z. K. Minev, A. Kandala, and K. Temme, "Probabilistic Error Cancellation with Sparse Pauli–Lindblad Models on Noisy Quantum Processors," Nature Physics, vol. 19, pp. 1116–1121, 2023.

[47] T. Wang, D. Zhao, and S. Qi, "Towards Understanding the Power of Quantum Kernels in the NISQ Era," Quantum, vol. 5, p. 531, 2021.

[48] A. Bao, R. Cheng, and Z. Li, "FFL-IDS: A Fog-Enabled Federated Learning-Based Intrusion Detection System to Counter Jamming and Spoofing Attacks for the Industrial Internet of Things," Sensors, vol. 25, no. 1, p. 10, 2025.

[49] S. Hassan, N. Adnan, and T. Rahman, "An Optimal Federated Learning-Based Intrusion Detection for IoT Environment," Scientific Reports, vol. 15, Art. no. 93501, 2025.

[50] D. P. Kingma and M. Welling, "Auto-Encoding Variational Bayes," in Proc. Int. Conf. Learning Representations (ICLR), 2014.

[51] C. Cortes and V. Vapnik, "Support-Vector Networks," Machine Learning, vol. 20, no. 3, pp. 273–297, 1995.

[52] N. Cristianini, J. Shawe-Taylor, A. Elisseeff, and J. Kandola, "On Kernel-Target Alignment," in Advances in Neural Information Processing Systems (NeurIPS), pp. 367–373, 2002.