Evaluating the Efficacy of Quantum Machine Learning at Detecting Advance-Fee Fraud Attacks

***Abstract*—Advance-Fee Fraud (AFF) is a time-critical financial scam that exploits behavioral urgency, often causing victims to initiate payments within minutes of deceptive communication. Traditional fraud detection systems, which rely on historical transaction patterns, are frequently unable to intervene before the first payment occurs. To address this, this paper investigates the feasibility of Quantum Machine Learning (QML) for rapid AFF detection under Noisy Intermediate-Scale Quantum (NISQ) constraints. For this, we designed a QML framework for AFF detection using behavior-driven temporal features. For evaluation, we compared QNN performance against classical machine-learning baselines using performance metrics and inference latency. Results indicated that even though the QNN achieved efficient performance at detecting AFF attacks with Accuracy, Precision, Recall, and F-1 scores of 0.820, 0.858, 0.793, and 0.824, respectively, it is still currently underperforming in comparison to classical ML models. Additionally, the results also indicate that QML provides slower inference latency compared to traditional ML models, with $2.92e-03$ seconds. These observed results indicate that current QML approaches are best positioned as complementary tools within hybrid detection pipelines, in conjunction with classical ML models.**

***Index Terms*—Quantum Machine Learning, Quantum Neural Networks, Fraud Detection, Cybersecurity, Artificial Intelligence**

I. Introduction

Financial fraud has emerged as one of the fastest-growing global cyber risks, with attackers increasingly exploiting psychological manipulation and time pressure to coerce victims into making rapid, irreversible payments [1]. Over the past five years, the Internet Crime Complaint Center (IC3) of the Federal Bureau of Investigation has received an average of more than 836,000 cybercrime complaints annually, reflecting the growing scale and sophistication of online financial fraud [2]. Among these schemes, Advance-Fee Fraud (AFF) is particularly effective due to its reliance on urgency: victims are often persuaded to transfer funds within minutes of receiving a deceptive message [3]. In 2024 alone, AFF accounted for 7,097 complaints while generating losses exceeding $102 million [4].

One of the informative behavioral indicators in AFF scenarios is the Inter-Payment Interval (IPI), defined as the time elapsed between the scam message and the first outgoing payment [5]. Short IPIs are strongly associated with attacker-induced urgency, as victims are pressured to act before verifying legitimacy. Unlike historical spending patterns, IPI captures temporal behavior at the earliest stage of fraud execution, making it particularly suitable for early detection. Due to the compressed IPI in AFF, a critical vulnerability exists in many traditional fraud-detection systems, specifically those that use machine learning (ML), which are not designed to intervene before the first payment occurs. Consequently, modeling rapid behavioral responses has become essential for developing early-warning mechanisms capable of preventing financial loss [6].

Towards this, Quantum Machine Learning (QML) [7] can be an attractive investigative opportunity to protect against AFF attacks. This is due to leveraging quantum computing characteristics like feature encoding, where even small temporal features can be mapped into a high-dimensional Hilbert space [8], which can lead to non-linear decision boundaries. This raises the possibility that QNN can amplify the subtle timing differences associated with AFF IPI and effectively detect them. However, a limitation for QNNs remains that under Noisy Intermediate-Scale Quantum (NISQ) constraints [9], QNNs may not be effectively deployed in everyday classical computing systems. In this paper, we will investigate whether the usage of QNNs can be a viable option for detecting AFF attacks. We will evaluate whether QNNs can achieve competitive performance using compact behavioral features, and whether increasing qubit counts meaningfully improves detection accuracy or inference speed. To the best of our knowledge, this is the first work that investigates temporal urgency features like IPI using QNNs. Our main contributions are:

- Designing a QML framework for AFF detection using behavior-driven temporal features.
- Empirically evaluating the impact of qubit scaling on QNN performance under realistic NISQ constraints.
- Comparing QNN performance against classical machine-learning baselines using performance metrics and inference latency.

The rest of the paper is structured as follows: Section II provides our literature review; Section III details the methodology of the research; Section IV presents evaluation results for our project; and Section V provides concluding remarks and future work.

II. Related Work

Numerous papers have investigated the idea for QML in various machine learning applications. For instance, the research by [10] established that near-term quantum models can be competitive without requiring large-scale quantum hardware, where systematic comparison of QML algorithms and classical machine-learning models using high-performance simulators. Similarly, the work in [8] provides a foundational theoretical framework connecting quantum computing and classical

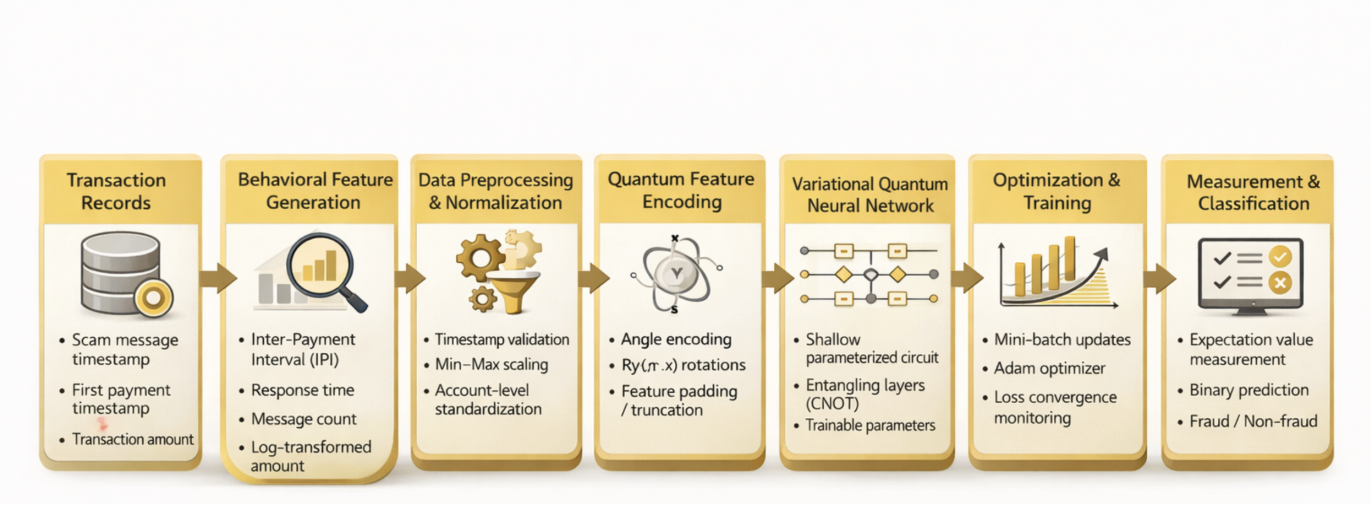

Fig. 1. End-to-end pipeline of the proposed QNN-based AFF detection framework.

kernel methods by interpreting quantum feature encoding as a nonlinear feature map in a high-dimensional Hilbert space [11]. Researchers in [12] conduct a systematic comparison of QML algorithms and classical machine-learning models using high-performance simulators, including Fujitsu Qulacs. These works indicated the promise of QML towards accelerating solutions for classical ML problems.

QML has also been investigated within the cybersecurity domain. For instance, [13] investigates the application of QML for cybersecurity threat prediction, demonstrating that quantum-enhanced models can analyze high-dimensional network traffic more efficiently than classical machine-learning systems. Research in [14] explores the application of Quantum Support Vector Machines (QSVMs) for malware detection, reporting detection accuracies of up to 95%. The work in [15] proposed QuantumNetSec, a novel intrusion detection system that combined quantum and classical computing techniques. While these efforts have actively investigated classical cybersecurity domains, the field of financial fraud detection using QML has not received as much attention.

Some recent works have started investigating QML for financial fraud detection. Research by [16] examines QML approaches for cybercrime and fraud detection with an emphasis on behavioral and transaction-level features. However, the proposed solutions rely on conventional learning architectures and centralized processing pipelines, which limit their suitability for low-latency decision-making. Researchers in [17] developed unsupervised quantum protocols for anomaly detection in credit card fraud detection. The work in [18] proposed a Quantum Graph Neural Networks-based mechanism to perform anomaly detection in financial fraud. The limitations of these works are that they only investigated general credit card and financial fraud anomaly detection, and did not focus on specific types of credit card fraud, like AFFs.

Overall, while existing literature demonstrates that QML can be beneficial in solving many cybersecurity problems, like financial fraud, no work investigates the usage of QML towards attacks like AFF. In particular, there remains a gap in understanding how optimization-centric QNNs can leverage minimal temporal features to enable rapid inference for AFF attacks under practical NISQ limitations.

III. Methodology

This section describes the end-to-end pipeline of the proposed QNN-based AFF detection framework, shown in Fig. 1.

A. Transaction Records

The transaction data used in this study are customized to reflect patterns relevant to advance-fee fraud detection. The dataset consists of the following key attributes:

- **Scam message timestamp:** Represents the exact time when the fraudulent message was sent to the victim. This feature helps analyze how quickly victims respond to scam attempts and identify abnormal behavioral patterns. This is denoted by t_s.
- **First payment timestamp:** Indicates the time when the victim made their first financial transaction after receiving the scam message. The time gap between this and the scam message timestamp provides insights into user decision-making behavior. This is denoted by t_p.
- **Transaction amount:** Captures the monetary value transferred during the transaction. Fraudulent transactions often show unusual or irregular payment patterns compared to legitimate ones.

Together, these attributes enable the modeling of temporal behavior and financial anomalies, which are critical indicators for detecting advance-fee fraud.

B. Behavioral Feature Generation

From raw data, behavioral features are extracted to model user interaction patterns:

- **Inter-Payment Interval (IPI):** Time difference between consecutive payments

- **Response Time:** Delay between scam message and payment
- **Message Count:** Number of interactions before payment
- **Log-transformed Amount:** Stabilizes large transaction values

These features reflect temporal dependencies and abnormal spending behaviors. Given the scam message time t_s and the first payment time t_p, we compute the Inter-Payment Interval (IPI) as:

$$\text{IPI} = \max\left(0, \frac{t_p - t_s}{60}\right) \tag{1}$$

This measures how quickly a victim responds after receiving a scam message. Shorter IPIs indicate urgency, a common trait in fraud behavior.

C. Data Preprocessing and Normalization

To ensure numerical stability, the following preprocessing steps are applied:

- Timestamp validation and missing value handling
- Min-Max scaling to normalize features to $[0, 1]$
- Account-level standardization to reduce user bias

The normalized feature x_{norm} is computed as:

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

To ensure stable training and compatibility with quantum circuits, all features are scaled to the range $[0, 1]$:

$$x_j^{\text{norm}} \in [0, 1] \tag{3}$$

This prevents large values from dominating model learning.

D. Quantum Feature Encoding

Classical features are mapped into quantum states using **angle encoding**. First, each feature x_i is encoded using a rotation gate:

$$R_y(\pi x_i) \tag{4}$$

Feature truncation is applied to match the number of available qubits. This transforms classical inputs into quantum superposition states. Next, each normalized feature is mapped into a quantum state using angle encoding:

$$|\psi_j\rangle = R_y(\pi x_j^{\text{norm}})\,|0\rangle \tag{5}$$

This converts classical data into quantum superposition states.

E. Quantum Neural Network (VQNN)

The encoded quantum states are processed by a parameterized quantum circuit consisting of:

- Shallow circuit layers (NISQ compatible)
- Entangling layers using CNOT gates
- Trainable rotation parameters

This layer learns non-linear feature relationships through quantum interference and entanglement. Using the QNN, the model outputs a fraud probability:

$$\hat{y} = \langle \psi(\boldsymbol{\theta}) | \hat{O} | \psi(\boldsymbol{\theta}) \rangle \tag{6}$$

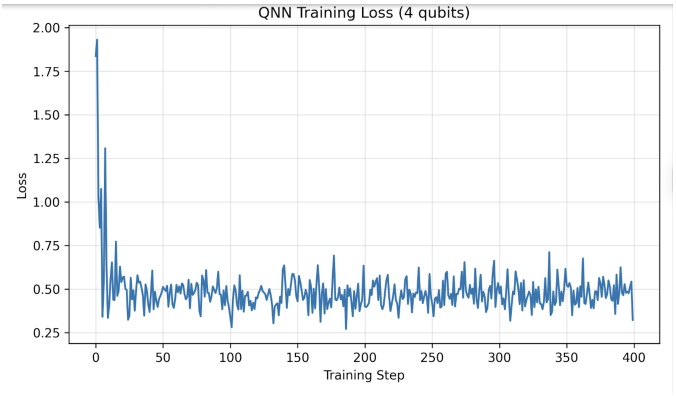

Fig. 2. Training loss trajectory for the 4-qubit QNN, showing stable convergence under shallow circuit depth.

Fig. 3. Training loss for the 5-qubit QNN, maintaining convergence with mild increases in loss variability.

F. Optimization and Training

Model parameters are trained using a hybrid classical–quantum learning approach. During training, the dataset is divided into small batches (mini-batches), and model parameters are updated iteratively.

- **Mini-batch training:** Instead of using the full dataset at once, smaller batches are used to make training faster and more stable.
- **Adam optimizer:** A popular optimization algorithm that automatically adjusts learning rates to improve convergence speed.
- **Loss monitoring:** Training progress is tracked by observing how the loss value decreases over iterations.

The training goal is to minimize the **binary cross-entropy loss**, which measures how well the model predictions match the true labels:

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)] \tag{7}$$

Here, y_i represents the true label, $\hat{y}_i$ is the predicted probability, and N is the total number of samples. Lower loss values indicate better model performance.

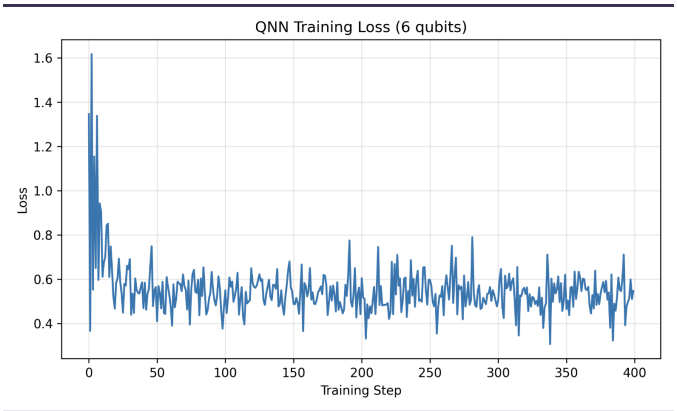

Fig. 4. Training loss for the 6-qubit QNN, showing increased oscillations and sensitivity to parameter updates.

Fig. 5. Training loss for the 7-qubit QNN, illustrating reduced convergence reliability under aggressive qubit scaling.

G. Measurement and Classification

Quantum measurements are performed to obtain expectation values. The output $\langle Z \rangle$ is mapped to binary predictions:

- $\langle Z \rangle = 1 \rightarrow$ Fraud
- $\langle Z \rangle = 0 \rightarrow$ Non-Fraud

IV. Experimentation Setup and Results

A. Setup

To evaluate the performance of the QML at AFF detection, we are evaluating the proposed QNN with classical ML models like Logistic Regression, Support Vector Machine, and Classical Neural Network (NN). Training settings for the QNN and NN use the Adam optimizer with mini-batches. Finally, performance is evaluated using Accuracy (A), Precision (P), Recall (R), and F1-score (F). Additionally, inference latency is also evaluated to assess real-time usability. All experiments for QML use the quantum simulator Qiskit [19].

B. Results

First, we observe the training convergence for the proposed QNN framework by varying the number of qubits. The convergence of a 4-qubit QNN is illustrated in Fig. 2. We note that the QNN converges rapidly using 4 qubits, and full convergence can be observed starting at Epoch 5. Similarly, the convergence of a 5-qubit QNN is illustrated in Fig. 3. We note that the QNN converges faster than using 5 qubits, and full convergence can be observed starting at Epoch 2. Next, the convergence of a 6-qubit QNN is illustrated in Fig. 4. We note that the QNN converges more slowly than using 4 qubits, and full convergence can be observed starting at Epoch 20. Finally, the convergence of a 7-bit QNN is illustrated in Fig. 4. We note that the QNN converges more slowly than using 6 qubits, and full convergence can be observed starting at Epoch 50. Through this, we note that the 5-qubit QNN is performing the best, and adding qubits to the QNN is slowing down the speed of convergence. This seems to be consistent with NISQ-induced gradient noise and optimization sensitivity.

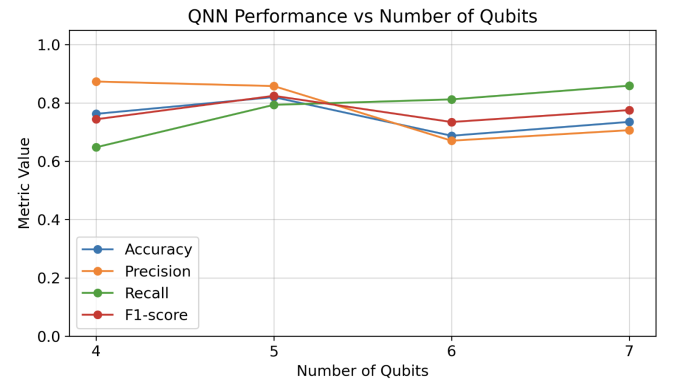

Fig. 6. Performance Metrics upon Qubit Scaling

Next, we observe the impact of qubit scaling on detection performance for AFF attacks, illustrated in Fig. 6. We observe that when 4 qubits are used, the QNN achieves A, P, R, and F scores of 0.78, 0.86, 0.75, and 0.65, respectively. When 5 qubits are used, the QNN achieves A, P, R, and F scores increase to 0.83, 0.85, 0.80, and 0.83, respectively. However, when we increase qubits to 6, the QNN performance decreases with A, P, R, and F scores of 0.67, 0.66, 0.81, and 0.75, respectively. Finally, when 7 qubits are used, the QNN achieves A, P, R, and F scores of 0.72, 0.70, 0.83, and 0.78, respectively. Through this, we note that the 5-qubit QNN is performing the best, and adding qubits to the QNN is decreasing ML performance at detecting AFF attacks.

Next, we observe the performance of the QNN in comparison to our classical ML models in detecting AFF attacks, as illustrated in Figure 7. We observe that Logistic Regression performs the least effectively with A, P, R, and F scores of 0.550, 0.604, 0.451, and 0.516, respectively. The next lowest performance is from the 5-qubit QNN, which is our most efficient QML model. The QNN recorded A, P, R, and F scores of 0.820, 0.858, 0.793, and 0.824, respectively. Next, comes the Support Vector Machine with A, P, R, and F scores of 0.915, 0.950, 0.887, and 0.917, respectively. The NN achieved the best performance in detecting AFF attacks

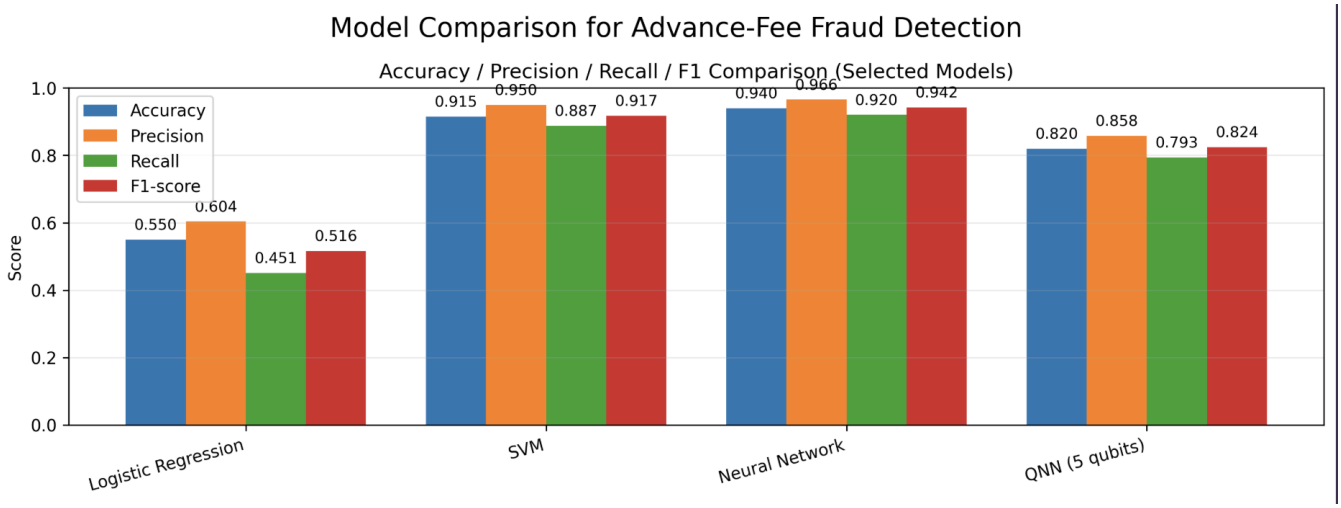

Fig. 7. Model Performance Comparison

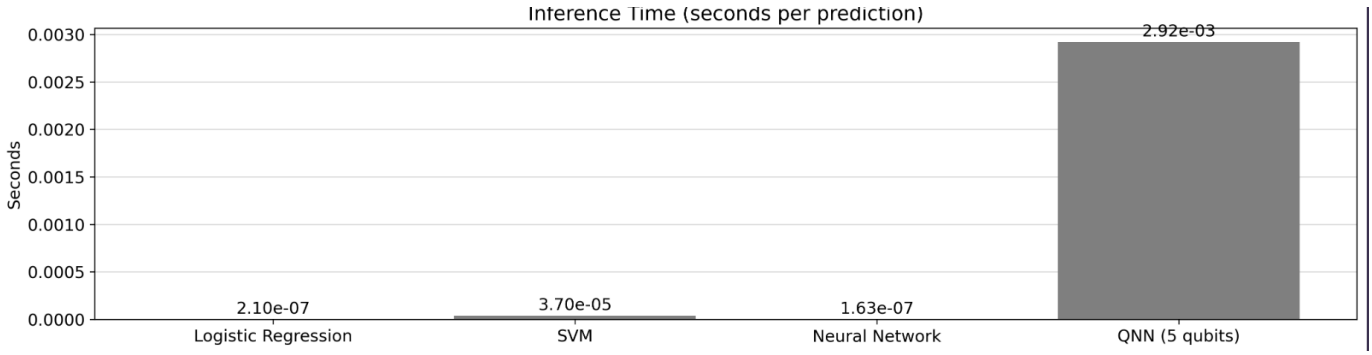

Fig. 8. Model Inference Latency Comparison

with A, P, R, and F scores of 0.940, 0.966, 0.920, and 0.942, respectively. This highlights that even though our QNN achieves comparatively efficient performance at detecting AFF attacks, it is not yet comparable to classical ML models like NNs in performance. These observations suggest that current QML approaches are best positioned as complementary tools within hybrid detection pipelines, in conjunction with classical ML models.

Finally, we observe the performance of the QNN in comparison to our classical ML models in detecting AFF attacks, as illustrated in Figure 8. We observe that the classical NN has the quickest inference time at detecting AFF attacks with $1.63e-07$ seconds. The Logistic Regression came next with an inference time of $2.10e-07$ seconds. The Support Vector Machine came next with $3.70e-05$ seconds. Finally, the QNN achieved the slowest inference time with $2.92e-03$ seconds. This highlights the level of influence optimization dynamics and circuit design have upon inference times in QML [20]. Through these experiments, we have observed that even though QML is slowly becoming a major computing paradigm in the future, it is still **NOT** yet an effective mechanism to detect financial fraud like AFF attacks today compared to classical ML models.

V. Conclusion

In this paper, we evaluated the feasibility of QML for rapid AFF detection under NISQ constraints. Towards this, we designed a QML framework for AFF detection using behavior-driven temporal features. We empirically evaluated the impact of qubit scaling on QNN performance under realistic NISQ constraints. Following this, we compared QNN performance against classical machine-learning baselines using performance metrics and inference latency. Results indicated that even though the QNN achieved efficient performance at detecting AFF attacks with A, P, R, and F scores of 0.820, 0.858, 0.793, and 0.824, respectively, it is still currently underperforming in comparison to classical ML models. Additionally, the results also indicate that QML provides slower inference latency compared to traditional ML models, with $2.92e{-}03$ seconds. These observed results indicate that current QML approaches are best positioned as complementary tools within hybrid detection pipelines, in conjunction with classical ML models. Future work will explore increasing the number of qubits to evaluate the solution further. Additionally, we will also explore QML for other types of financial fraud cyberattacks, like Account Takeover and Business Email Compromise.

REFERENCES

[1] A. A. Kolapo *et al.*, "Machine learning techniques for cybercrime and financial fraud detection," *International Journal of Information Security*, vol. 19, no. 4, pp. 389–402, 2020.

[2] S. R. Biedron, "Cybercrime in the digital age," Ph.D. dissertation, University of Oxford, 2024.

[3] A. Tambe Ebot, "Advance fee fraud scammers' criminal expertise and deceptive strategies: a qualitative case study," *Information & Computer Security*, vol. 31, no. 4, pp. 478–503, 2023.

[4] T. J. Holt and D. C. Graves, "A qualitative analysis of advance fee fraud e-mail schemes," *International Journal of Cyber Criminology*, vol. 1, no. 1, pp. 137–154, 2007.

[5] J. J. Chang, "An analysis of advance fee fraud on the internet," *Journal of Financial Crime*, vol. 15, no. 1, pp. 71–81, 2008.

[6] M. Schuld, I. Sinayskiy, and F. Petruccione, "An introduction to quantum machine learning," *Contemporary Physics*, vol. 56, no. 2, pp. 172–185, 2015.

[7] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd, "Quantum machine learning," *Nature*, vol. 549, no. 7671, pp. 195–202, 2017.

[8] 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.

[9] J. Preskill, "Quantum computing in the NISQ era and beyond," *Quantum*, vol. 2, p. 79, 2018, arXiv:1801.00862.

[10] P. Lamichhane and D. B. Rawat, "Quantum machine learning: Recent advances, challenges and perspectives," *IEEE Access*, 2025.

[11] M. Schuld and N. Killoran, "Quantum machine learning in feature hilbert spaces," *Physical Review Letters*, vol. 122, no. 4, p. 040504, 2019.

[12] M. Adnan *et al.*, "Comparative evaluation of quantum and classical machine learning models," *arXiv preprint arXiv:2509.13353*, 2025.

[13] A. Awasthi, "The role of quantum machine learning in cybersecurity."

[14] Ganapathy and Venkatasubramanian, "Quantum support vector machines for malware classification," *Journal of Cybersecurity*, vol. 6, no. 1, pp. 1–12, 2020.

[15] D. Abreu, D. Moura, C. Esteve Rothenberg, and A. Abelém, "Quantum-netsec: Quantum machine learning for network security," *International Journal of Network Management*, vol. 35, no. 4, p. e70018, 2025.

[16] S. S. Kamble, S. S. Pawar, T. B. Veer, and D. M. Padulkar, "Fraud detection using quantum machine learning (qml)," in *2024 Global Conference on Communications and Information Technologies (GCCIT)*. IEEE, 2024, pp. 1–6.

[17] O. Kyriienko and E. B. Magnusson, "Unsupervised quantum machine learning for fraud detection," *arXiv preprint arXiv:2208.01203*, 2022.

[18] N. Innan, A. Sawaika, A. Dhor, S. Dutta, S. Thota, H. Gokal, N. Patel, M. A.-Z. Khan, I. Theodonis, and M. Bennai, "Financial fraud detection using quantum graph neural networks," *Quantum Machine Intelligence*, vol. 6, no. 1, p. 7, 2024.

[19] A. Javadi-Abhari, M. Treinish, K. Krsulich, C. J. Wood, J. Lishman, J. Gacon, S. Martiel, P. D. Nation, L. S. Bishop, A. W. Cross *et al.*, "Quantum computing with qiskit," *arXiv preprint arXiv:2405.08810*, 2024.

[20] A. Paler, L. Sasu, A.-C. Florea, and R. Andonie, "Machine learning optimization of quantum circuit layouts," *ACM Transactions on Quantum Computing*, vol. 4, no. 2, pp. 1–25, 2023.