2025 3rd International Conference on Intelligent Systems, Advanced Computing and Communication (ISACC) | 979-8-3315-2389-3/25/$31.00 ©2025 IEEE | DOI: 10.1109/ISACC65211.2025.10969366

PhishVQC: Optimizing Phishing URL Detection with Correlation Based Feature Selection and Variational Quantum Classifier

Md. Farhan Shahriyar
Dept. of Computer Science and Engineering
World University of Bangladesh
Dhaka, Bangladesh
farhanshahriyar.cse1@gmail.com

Gazi Tanbhir
Dept. of Computer Science and Engineering
World University of Bangladesh
Dhaka, Bangladesh
gazitanbhir@gmail.com

Abdullah Md Raihan Chy
Dept. of Computer Science and Engineering
Southern University Bangladesh
Chittagong, Bangladesh
raihanchowdhuryengineer@gmail.com

Mohammed Abdul Al Arafat Tanzin
Dept. of Computer Science and Engineering
Brac University
Dhaka, Bangladesh
tanzinabdul@gmail.com

Md. Jisan Mashrafi
Dept. of Computer Science and Engineering
Brac University
Dhaka, Bangladesh
jisanmashrafi542@gmail.com

***Abstract*—Phishing URL detection is crucial in cybersecurity as malicious websites disguise themselves to steal sensitive information. Traditional machine learning techniques struggle to perform well in complex real-world scenarios due to large datasets and intricate patterns. Motivated by quantum computing, this paper proposes using Variational Quantum Classifiers (VQC) to enhance phishing URL detection. We present PhishVQC, a quantum model that combines quantum feature maps and variational ansatzes such as RealAmplitude and EfficientSU2. The model is evaluated across two experimental setups with varying dataset sizes and feature map repetitions. PhishVQC achieves a maximum macro average F1-score of 0.89, showing a 22% improvement over prior studies. This highlights the potential of quantum machine learning to improve phishing detection accuracy. The study also notes computational challenges, with execution wall times increasing as dataset size grows.**

***Index Terms*—Phishing Detection, Quantum Machine Learning, Variational Quantum Classifier, URL Detection, Feature Selection, Correlation Based Feature Selection, Quantum Computing, Quantum Classical Hybrid Model, Cybersecurity,VQC,QML**

I. Introduction

Phishing attacks typically begin with a redirection or manipulation of the victim into clicking a malicious link leading to efforts aimed at exfiltrating sensitive information such as login credentials or financial details. As phishing techniques evolve over time the threat landscape in cyberspace becomes increasingly sophisticated [1] [2].

This appears more challenging to effectively counter such attacks. While classical machine learning approaches have made significant progress in detecting phishing attempts the rapidly evolving nature of phishing necessitates more advanced solutions.

Quantum machine learning (QML) presents a promising solution to these challenges given its ability to process large computations in a significantly shorter time frame. QML is powered by Quantum Computing extending its area on machine learning. Quantum computers process data through qubit superposition.In fig 1 it is represented by the yellow vector on

the Bloch Sphere. Unlike classical bits (0 or 1), qubits exist as a combination of $|0\rangle$ and $|1\rangle$, defined by angles θ and ϕ .

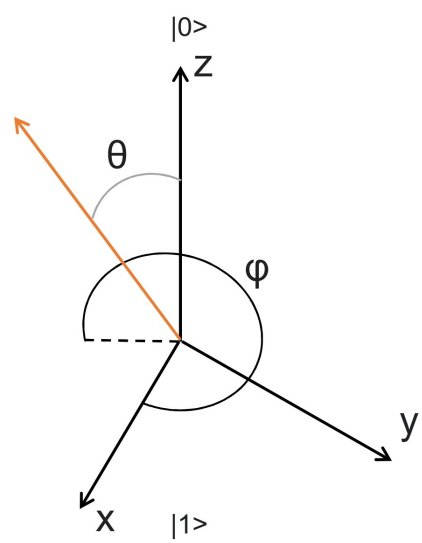

Fig. 1. Visual representation of the Bloch Sphere , the state of a single Qubit.

979-8-3315-2389-3/25/$31.00 ©2025 IEEE

Authorized licensed use limited to: Zhejiang University. Downloaded on April 26,2025 at 01:24:25 UTC from IEEE Xplore. Restrictions apply.

The qubits are being used to encode classical data into methods like amplitude or angle encoding where features determine the qubit's state. Quantum gates manipulate these states in parallel, enabling efficient computations, such as processing phishing dataset features (f1,f2...fn) simultaneously. However, challenges like inefficient encoding, limited qubits, gate errors, and decoherence hinder progress. Optimized algorithms, better encoding, and advanced hardware are crucial for QC's success.

However, its application is currently limited to certain fields. QML also demonstrates a superior capacity for identifying complex patterns compared to classical methods. Recently, the use of quantum machine learning in cybersecurity has emerged as a new frontier [3]. Quantum computing is opening new possibilities for more powerful computational capabilities [4]. Although much of the current research on QML cybersecurity focuses on fraud detection through feature encoding and variational quantum circuits, its application to phishing URL detection remains an underexplored area [5] [6]. Given the growing sophistication of phishing techniques there is an urgent need to explore quantum based solutions for effective phishing detection. This presents a critical opportunity for advancing the field of phishing detection through quantum computing.

Previous studies in quantum machine learning have primarily relied on small datasets (e.g., 200 data points) [7] and there has been limited research on how the scalability of data points affects the quantum classification process. Additionally, most classical machine learning models have incorporated large sets of features, yet their integration with quantum processing has not been thoroughly investigated. This study aims to address these gaps by focusing on the following key contributions:

- Analyzed how the increase in the number of data points affects the performance of quantum algorithms in phishing URL detection.
- Examined the integration of correlation based classical feature selection with quantum models to improve detection accuracy in phishing URL classification.
- Introduced PhishVQC a novel quantum model specifically designed to classify phishing URLs more efficiently.

This paper presents a literature review in Section II and details the methodology used in Section III. Section IV covers the experiments and results while Section V concludes with discussions and future work.

II. LITERATURE REVIEW

The study [8] demonstrated the impact of correlation independency on feature selection and algorithm performance highlighting its significance in enhancing the results of machine learning models.The paper by [9] explores a hybrid classical quantum approach that integrates classical and quantum algorithms in an ensemble model to improve fraud prevention decisions. In this context the Quantum Support Vector Machine (QSVM) offers a complementary exploration of the feature space resulting in improved accuracy for the mixed quantum - classical model in fraud detection when applied to a significantly reduced dataset. Notably, the new algorithm identified features with minimal overlap in the correlation matrix suggesting that these feature choices are viable for improving the detection process.

Reyes-Dorta et al. [7] developed a quantum machine learning approach for detecting malicious URLs using a Variational Quantum Classifier (VQC). The dataset consisted of 145,532 random data points, including 6,965 fraudulent URLs, from which 200 observations were selected for analysis. To map the data into quantum states, the study tested several ansatzes, including RealAmplitudes (RealAmp), EfficientU2, and ExcitingPreserving (ExcitPreserving). Optimizers such as COBYLA, GradientDescent, and SLSQP, each running for 20 iterations, were also evaluated. The results showed that the combination of the RealAmplitudes ansatz (with two repetitions) and the SLSQP optimizer produced the best performance. The VQC model achieved a precision of 0.89 on the training dataset and 0.97 on the test dataset, highlighting the potential of quantum-based methods for effectively detecting malicious URLs.

Ray et al. [10] developed a Quantum Machine Learning (QML) approach for improving phishing detection in the Ethereum transaction network. The proposed method utilizes Variational Quantum Classifier (VQC) and Quantum Support Vector Machine (QSVM) algorithms, applied to a dataset scraped from Etherscan, which contains both labeled phishing accounts and a significantly larger number of unlabeled legitimate accounts. To address the class imbalance, the authors focused on a training set of 300 data points and a test set of 300 data points, extracting seven statistical features such as in-degree, out-degree, and transaction strength. The data was encoded using a second-order Pauli-Z (ZZ) feature map and QRAC embedding. The VQC model, tested on a simulator with 32 qubits and using the ZZ feature map, achieved a macro F1 score of 0.67, phishing F1 score of 0.63, and phishing recall of 0.55. However, when run on quantum hardware, the model's performance was limited by longer training times, which required reducing the dataset size for practical implementation. Despite these challenges, the results demonstrate the potential of QML for detecting phishing in Ethereum networks.

Abreu et al. [11] showed that VQC achieves the highest F1 scores among QML models across datasets outperforming traditional ML methods. For instance, VQC scored 88.10% on the UNSW-NB15 dataset, surpassing SVM's 82.34%. Similarly, in the CIC-IDS-17 dataset, QCNN reached 95.60%. While QML models excel in detecting specific attacks like Analysis, DDoS, and DoS traditional methods still outperform them in categories such as Worms and Shellcode. These results highlight the strengths of QML models but also the continued effectiveness of traditional methods suggesting the need for further research in QML for cybersecurity.

III. METHODOLOGY

A. The Proposed Method

The process begins with dataset preprocessing to scale features (e.g., feature1 to featureN) into numerical values with

Authorized licensed use limited to: Zhejiang University. Downloaded on April 26,2025 at 01:24:25 UTC from IEEE Xplore. Restrictions apply.

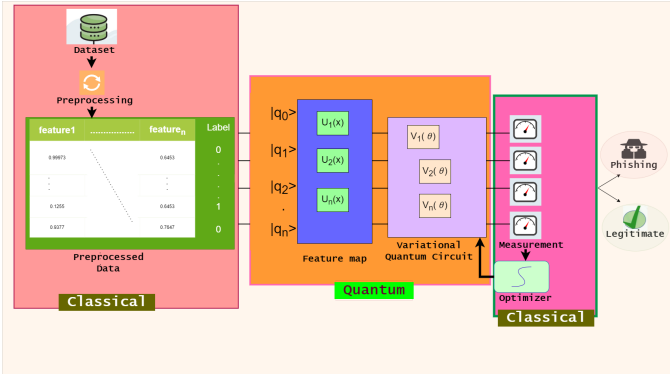

Fig. 2. The PhishVQC model for phishing detection.

associated labels (1 for legitimate, 0 for phishing). These features are encoded into qubits $(q_0, q_2, \ldots, q_n)$ using a quantum feature map. A variational quantum circuit (VQC) with adjustable parameters $V(\theta)$ processes the qubits. Measurements

are taken and a classical optimizer updates the parameters to minimize error. The final measurement classifies outputs as phishing or legitimate.

B. The URL Phishing Dataset

The dataset used in this study is the PhiUSIIL Phishing URL Dataset [12] which contains 134,850 legitimate URLs and 100,945 phishing URLs. Sourced from the UCI Machine Learning Repository, the dataset includes 54 features that describe the structural and content-based characteristics of

URLs. Key features include the number of subdomains ('NoOfSubDomain'), the presence of obfuscation ('HasObfuscation'), the count of obfuscated characters ('NoOfObfuscatedChar'), the letter-to-character ratio in the URL ('LetterRatioInURL'), the number of question marks in the URL ('NoOfQMarkInURL'), and the number of lines of code in the URL ('LineOfCode').In the dataset, Class 1 consists of legitimate URLs while Class 0 includes phishing URLs. These features provide valuable insights for differentiating between legitimate and phishing URLs based on their structural patterns and content characteristics.

C. Data Preprocessing

Missing or null values in the dataset are checked. A good feature set consists of features that are strongly correlated with the target class but uncorrelated with each other . To reduce the complexity of quantum processing, the correlation between features is analyzed, and a threshold value of 0.5 is set to identify the most important features, reducing the feature count from 54 to 32.Finally, normalization is applied to prepare the classical data for quantum processing.

D. Encoding Classical Data into Quantum States

For quantum machine learning, classical data must be represented in a way that can be efficiently processed by quantum circuits.

Amplitude Encoding with Raw Feature Vector: In this context, we use amplitude encoding which requires that the number of features be a power of two. This is because the quantum state corresponding to the data must have a Hilbert space dimension that is a power of two, which is a fundamental requirement for amplitude encoding.

$$n = \log_2 [\text{number of features}] \tag{1}$$

Where n is the number of qubits required for encoding the features. The total dimension of the quantum state then becomes 2^n . If the number of features is not equal to 2^n after determining n, the number of features is padded with zeros to ensure the total number of features equals 2^n. This enables us to encode the data into a quantum state using amplitude encoding.

To encode the classical data into quantum states, we use the Raw Feature Vector circuit. This circuit is a parameterized quantum circuit that initializes a quantum state based on the classical data. In the context of amplitude encoding, the classical data is mapped to the amplitudes of the quantum state. The Raw Feature Vector circuit in Qiskit [13] provides a direct, untransformed encoding of classical features into quantum states. The circuit works by applying an initialization operation, where the classical data defines the amplitudes of the quantum state.

The RFV circuit requires a number of qubits equal to n, where n is the logarithm of the feature dimension, dim. Specifically, $n = \log_2(\text{dim})$. The quantum state is initialized accordingly. The quantum state is initialized as follows:

$$|\psi\rangle = \sum_{i=0}^{2^n-1} c_i |i\rangle \tag{2}$$

The quantum state $|\psi\rangle$ is expressed as a linear combination of the computational basis states $|i\rangle$ with corresponding amplitudes c_i derived from the input classical data.

This RawFeatureVector circuit directly initializes the quantum state based on the classical data, where the input data determines the amplitudes of the quantum state. The Raw Feature Vector does not apply kernel transformations, which means it directly encodes the raw features into the quantum state. This is an essential part of amplitude encoding.

E. Quantum Ansatz

To perform quantum classification, a variational quantum circuit is used. This variational circuit is designed to approximate the target quantum state that represents the classification decision boundary. In this study, we use the RealAmplitude and EfficientSU2 ansätze which are parameterized quantum circuits capable of expressing complex quantum states. These ansatz also function as quantum feature maps. Since they are parameterized circuits, they can be considered ansatz in VQC.

Authorized licensed use limited to: Zhejiang University. Downloaded on April 26,2025 at 01:24:25 UTC from IEEE Xplore. Restrictions apply.

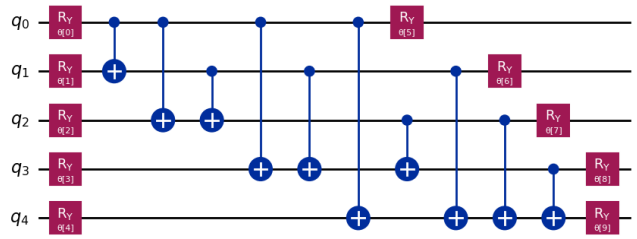

Fig. 3. The RealAmplitude Ansatz.

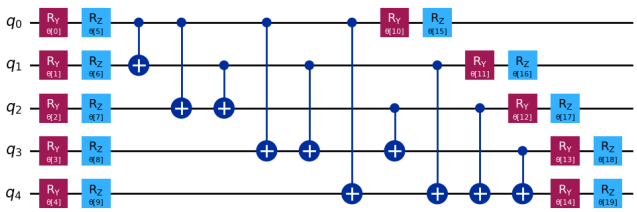

Fig. 4. The EfficientSU2 Ansatz.

1) RealAmplitude Circuit: : The RealAmplitude ansatz [14] typically includes entangling gates such as CNOT (controlled-NOT gates) that introduce correlations between qubits. These entangling gates are necessary to represent non-product states and are crucial for learning complex patterns in the data.

2) EfficientSU2 Circuit: : The EfficientSU2 ansatz applies a sequence of parameterized rotations and entangling gates to prepare the quantum state. The rotation gates manipulate the quantum state, while the entangling gates (e.g., CNOT gates) create correlations between qubits.

The number of qubits is set to n, which corresponds to the number of features after zero padding.The ansatz can be repeated multiple times to increase the circuit depth and express more complex states. In this study, we use 3 and 4 repetitions. The entanglement='full' option means that all qubits are entangled in every layer allowing the quantum circuit to explore a broader state space and create more complex entanglements between qubits.

F. Training the Variational Quantum Classifier (VQC)

The Variational Quantum Classifier (VQC) combines the quantum feature map (RawFeatureVector) with the variational ansatz (RealAmplitude or EfficientSU2) to perform classification. The VQC is trained by adjusting the parameters of the ansatz to minimize the classification error.

Optimizer: We use the COBYLA optimizer with a maximum of 300 iterations to minimize the objective function. COBYLA is a derivative-free optimization algorithm making it suitable for optimizing quantum circuits where gradients are not easily computed.

StatevectorSampler: This sampler [15] is used to compute the state vector, which gives the full quantum state of the system after applying the quantum operations.

Callback: A callback function is used to track the optimization progress by plotting the objective function value at each iteration. This helps to visualize how the optimization is proceeding.

IV. EXPERIMENTS AND RESULTS

In Table I and Table II ASetup and BSetup are two distinct configurations of the experimental environment for running the PhishVQC algorithm each with slightly different parameters like dataset sizes and feature map repetitions.The setup for the PhishVQC experiment involves two configurations: ASetup and BSetup, both utilizing a combination of classical and quantum libraries, including Qiskit, scikit-learn, pandas and others.In ASetup, the training dataset contains 640 samples with 160 samples for testing. The dataset has 32 features and the quantum model uses 5 qubits. The experiment runs for 300 iterations with the Feature Map encoding repeated 3 times. The quantum simulator used is the Statevector simulator.In BSetup, the dataset size increases to 960 training samples and 240 testing samples. The number of repetitions for the Feature Map encoding increases to 4, but other parameters remain unchanged. Both setups use the same quantum simulator and libraries.The key difference between the setups is the larger dataset and additional Feature Map repetition in BSetup aimed at examining the model's performance.

TABLE I
ASETUP: THE SETUP FOR THE FIRST EXPERIMENTAL ENVIRONMENT OF PHISHVQC.

Elements	Value	Description
Libraries and Frameworks	Numpy, scipy, matplotlib, mpl_toolkits, sklearn, pandas, Qiskit, qiskit-machine-learning, qiskit-algorithms	Libraries and frameworks used for implementing and running VQC (Variational Quantum Classifier) and related benchmarks. Includes libraries for classical machine learning, data manipulation, visualization, and quantum computing (Qiskit).
Training dataset size	640	Number of samples in the training dataset.
Testing dataset size	160	Number of samples in the testing dataset.
Features	32	Total number of features in the dataset.
Iterations	300	Number of iterations for running the quantum experiment.
Qubits	5	Number of qubits used in the VQC and benchmark models.
Feature Map repetitions	3	Number of times the Feature Map encoding method is repeated.
Quantum simulator	Statevector simulator	Simulator used for executing local quantum experiments with Qiskit.

Authorized licensed use limited to: Zhejiang University. Downloaded on April 26,2025 at 01:24:25 UTC from IEEE Xplore. Restrictions apply.

TABLE II
BSETUP: THE SETUP FOR THE SECOND EXPERIMENTAL ENVIRONMENT OF PHISHVQC.

Elements	Value	Description
Libraries and Frameworks	Numpy, scipy, matplotlib, mpl_toolkits, sklearn, pandas, Qiskit, qiskit-machine-learning, qiskit-algorithms	Libraries and frameworks used for implementing and running VQC (Variational Quantum Classifier) and related benchmarks. Includes libraries for classical machine learning, data manipulation, visualization, and quantum computing (Qiskit).
Training dataset size	960	Number of samples in the training dataset.
Testing dataset size	240	Number of samples in the testing dataset.
Features	32	Total number of features in the dataset.
Iterations	300	Number of iterations for running the quantum experiment.
Qubits	5	Number of qubits used in the VQC and benchmark models.
Feature Map repetitions	4	Number of times the Feature Map encoding method is repeated.
Quantum simulator	Statevector simulator	Simulator used for executing local quantum experiments with Qiskit.

TABLE III
PHISHVQC PERFORMANCE WITH REALAMPLITUDE AND EFFICIENTSU2 IN ASETUP AND BSETUP.

Setup	Ansatz	Macro Phish Precision	Macro Phish Recall	Macro Avg F1-Score
ASetup	Realamplitude	0.97	0.81	0.89
	EfficientSU2	0.93	0.87	0.89
BSetup	Realamplitude	0.96	0.80	0.88
	EfficientSU2	0.90	0.84	0.87

TABLE IV
COMPARISON OF THE PHISHVQC MODEL WITH EXISTING MODELS

Model	Macro Avg F1-Score
VQC[[10]]	0.67
PhishVQC	0.89

In Table III In the ASetup, Realamplitude demonstrates a strong performance with a phishing precision of 0.97 and a recall of 0.81 leading to a macro average F1-score of 0.89. This indicates that while Realamplitude excels at identifying phishing URLs with high precision, its ability to capture all phishing instances is slightly less effective, as shown by the recall value.EfficientSU2, on the other hand, shows a good balance with a phishing precision of 0.93 and a higher recall of 0.87 resulting in an equivalent macro average F1-score of 0.89. This highlights EfficientSU2's effectiveness in both identifying phishing URLs and covering a wider range of instances in the

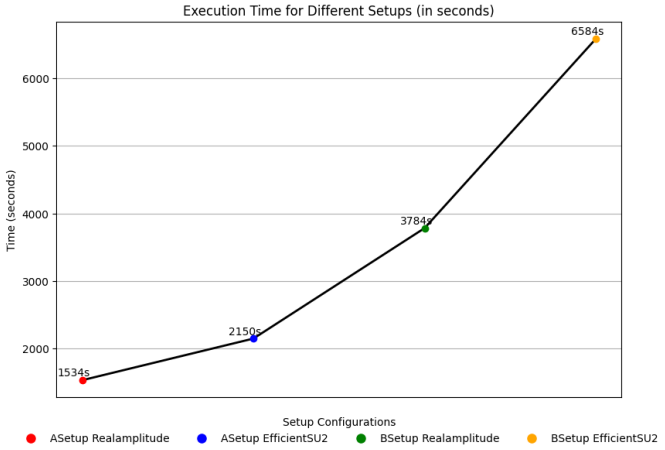

Fig. 5. Wall Time of Execution for PhishVQC with Different Setups.

ASetup.

In the BSetup, Realamplitude's phishing precision remains high at 0.96 but its recall drops slightly to 0.80 leading to a macro F1-score of 0.88. This reduction in recall suggests that while the model maintains high precision, it is less effective at identifying all phishing instances in this setup.EfficientSU2 in BSetup shows a decline in phishing precision (0.90), but its recall improves to 0.84 resulting in a slightly lower F1-score of 0.87. This demonstrates that EfficientSU2's performance is still reliable but sees a small drop in phishing precision compared to ASetup.

Overall while both models appears with strong phishing detection capabilities, Realamplitude's performance is more consistent in terms of precision while EfficientSU2 shows better recall in the BSetup. However, the analysis of legitimate URLs is less significant in this context as the primary focus is on phishing URL detection though the macro scores in this case also remains 0.83 to 0.89.But falsely identifying a phishing URL as legitimate poses a significant security risk potentially allowing harmful websites to be accessed , the accurate detection of phishing URLs far more critical in ensuring security. In table IV our PhishVQC shows an improvement of 0.22 in the macro average F1-score which indicates a 22% increase in overall performance from previous study.

The figure 5 compares execution wall times for setups with varying data scales. As BSetup's datapoints increase by 50% compared to ASetup, the wall time increases by almost three times. In ASetup, EfficientSU2 is about 40% slower than Realamplitude (1534s). Similarly, BSetup EfficientSU2 (6584s) takes nearly twice as long as Realamplitude. This highlights that EfficientSU2 consistently requires more time than Realamplitude across both setups.

V. DISCUSSIONS AND FUTURE WORK

In this study the focus is placed on addressing the cyber-security challenge of phishing URL detection in the context of quantum computing. The primary goal is to explore the

Authorized licensed use limited to: Zhejiang University. Downloaded on April 26,2025 at 01:24:25 UTC from IEEE Xplore. Restrictions apply.

potential of applying quantum machine learning techniques to improve phishing URL detection with a larger dataset and to identify an effective variational quantum classifier for URL classification.

PhishVQC shows that increasing the dataset size and feature map or ansatz repetitions in BSetup leads to minor improvements in performance but the key takeaway is that feature map repetitions have a more significant impact on the model's ability to capture phishing URLs. The larger dataset alone does not drastically change results. This highlights the importance of circuit complexity, especially through feature map repetitions over merely increasing dataset size for improving phishing detection accuracy.An increase in dataset size impacts the wall time execution of quantum circuits and the repetition of feature maps in the ansatz can also extend the execution time. Compared to the previous study PhishVQC shows a 22% improvement in macro F1-score.

In the transition from classical to quantum machine learning the current hardware complexity and challenges in processing large datasets make utilizing large sample sizes for such tasks still difficult. Nevertheless, this study highlights the essence of quantum machine learning (QML) for phishing URL detection which could inspire the exploration of other quantum algorithms such as Quantum Support Vector Machines (QSVM), Quantum Neural Networks (QNN) and Quantum Convolutional Neural Networks (QCNN) in the near future.

REFERENCES

[1] "Apwg trends report q1 2024, available:https://docs.apwg.org/reports/textbackslash apwg_ trends_report_q1_2024.pdf, Last access: 12 October 2024."

[2] F. Carroll, J. A. Adejobi, and R. Montasari, "How good are we at detecting a phishing attack? investigating the evolving phishing attack email and why it continues to successfully deceive society," vol. 3, no. 2, p. 170.

[3] K. Keplinger, "Is quantum computing becoming relevant to cybersecurity?," vol. 2018, no. 9, pp. 16–19.

[4] V. Chauhan, S. Negi, D. Jain, P. Singh, A. K. Sagar, and A. K. Sharma, "Quantum computers: A review on how quantum computing can boom AI," in *2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, pp. 559–563, IEEE.

[5] N. Innan, M. A.-Z. Khan, and M. Bennai, "Financial fraud detection: A comparative study of quantum machine learning models," vol. 22, no. 2, p. 2350044.

[6] A. Tudisco, D. Volpe, G. Ranieri, G. Curato, D. Ricossa, M. Graziano, and D. Corbelletto, "Evaluating the computational advantages of the variational quantum circuit model in financial fraud detection," vol. 12, pp. 102918–102940.

[7] N. Reyes-Dorta, P. Caballero-Gil, and C. Rosa-Remedios, "Detection of malicious URLs using machine learning," vol. 30, no. 9, pp. 7543–7560.

[8] M. A. Hall, "Correlation-based feature selection for machine learning,"

[9] M. Grossi, N. Ibrahim, V. Radescu, R. Loredo, K. Voigt, C. Von Altrock, and A. Rudnik, "Mixed quantum–classical method for fraud detection with quantum feature selection," vol. 3, pp. 1–12.

[10] A. Ray, S. Sakunthala, and A. Prabhakar, "Improving phishing detection in ethereum transaction network using quantum machine learning," in *2023 IEEE International Conference on Quantum Computing and Engineering (QCE)*, pp. 1107–1113, IEEE.

[11] D. Abreu, C. E. Rothenberg, and A. Abelém, "QML-IDS: Quantum machine learning intrusion detection system," in *2024 IEEE Symposium on Computers and Communications (ISCC)*, pp. 1–6, IEEE.

[12] A. Prasad and S. Chandra, "PhiUSIIL: A diverse security profile empowered phishing URL detection framework based on similarity index and incremental learning," vol. 136, p. 103545.

[13] Qiskit Community, "Qiskit machine learning: Rawfeaturevector." https://qiskit-community.github.io/qiskit-machine-learning/stubs/qiskit_machine_learning.circuit.library.RawFeatureVector.html.

[14] Qiskit, "Qiskit circuit library: Realamplitudes." https://docs.quantum.ibm.com/api/qiskit/qiskit.circuit.library.RealAmplitudes.

[15] Qiskit, "Qiskit primitives: Statevectorsampler." https://docs.quantum.ibm.com/api/qiskit/qiskit.primitives.StatevectorSampler.

Authorized licensed use limited to: Zhejiang University. Downloaded on April 26,2025 at 01:24:25 UTC from IEEE Xplore. Restrictions apply.