Received: January 31, 2022. Revised: March 3, 2022.

INASS

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Hybrid Quantum Deep Learning and Variational Quantum Classifier-Based Model for Botnet DGA Attack Detection

Hatma Suryotrisongko[1]* **Yasuo Musashi[2]**

[1]*Graduate School of Science and Technology, Kumamoto University, Japan*
[1]*Faculty of Intelligent Electrical and Informatics Technology, Institut Teknologi Sepuluh Nopember, Indonesia*
[2]*Center for Management of Information Technologies, Kumamoto University, Japan*
* Corresponding author's Email: hatma@is.its.ac.id

Abstract: We explored the potentials of the quantum machine learning (ML) approach for cybersecurity domain generation algorithm (DGA) detection applications employing two quantum ML approaches: hybrid quantum-classical deep learning (DL) and variational quantum classifier (VQC) models. We implemented quantum noise models and benchmarked the hybrid quantum-classical DL with the VQC-based one. Our datasets include statistical analysis (entropy, relative entropy, information radius, and reputation score) of 1,000,000 Alexa and 803,333 DGA domain names. Qiskit and Pennylane were utilized for the quantum simulations in our experiments. We found that our proposed hybrid quantum DL outperforms the VQC-based model (94.9% maximum accuracy). Investigating the combinations of 12 types of optimizer algorithms, four feature maps, and four variational form circuits reveal that feature maps and variational forms are critical for the VQC algorithm. We show the usefulness of a quantum circuit architecture consisting of Angle Embedding and Random Layers for a hybrid quantum DL model.

Keywords: Botnet, Cybersecurity, DGA, Quantum computing, Quantum machine learning, VQC.

1. Introduction

Botnet is notorious for using domain name system (DNS) traffic to hide their communication messages with a command and control (C&C) server and causing severe cyberattacks [1]. Domain generation algorithm (DGA)-based botnets' traffic is challenging to detect because it uses C&C domain names that are algorithmically generated [2].

Quantum machine learning (ML) research has been gaining more attention [3]. A quantum computer can improve ML tasks because quantum information processing could deliver exponential speedups on specific problems [4]. Extensive reviews/surveys are available in [5–7]. This paper explores the hybrid quantum-classical deep learning (DL) approach and variational quantum classifier (VQC) algorithm [8] for botnet DGA detection.

The VQC algorithm consists of both quantum (computed on a quantum computer device) and classical parts (calculated on a conventional computer) [8]. The first component within the quantum part is a quantum feature-mapping circuit to transform the dataset input from a classical data type (e.g., integer or floating number) into qubits [8]. The second component within the quantum part is the variational circuit, containing parameterized gates to be optimized during the training process. In the classical part of this algorithm, an optimization algorithm runs an objective function and updates the parameter values within the variational circuit to achieve an optimal classification model.

The availability of various DL tools (e.g., Keras, TensorFlow, PyTorch [9]) make it convenient to construct a DL model with multiple layers (e.g., convolution layers, recurrent layers) for computations in a conventional computer. Therefore, we investigate the inclusion of a quantum circuit (to be computed on a quantum computer device) as a layer in a DL model, which we refer to as the hybrid quantum DL model [10]. As achieving a quantum advantage for ML applications on current noisy

intermediate-scale quantum devices (NISQs) remains an open challenge [11,12], this paper also implements noise models for our hybrid DL model.

The strong points and contributions of our work in this paper are: (1) proposing models to detect DGA-based botnet traffic on two quantum ML approaches, which are hybrid quantum-classical DL and VQC algorithm; (2) experiment with quantum noise models; and (3) comparing the performance of hybrid quantum-classical DL model with the VQC algorithm-based model.

We organized the rest of this paper as follows: section 2 describes related works to point the position of our work. Section 3 elaborates our research's methodology. The discussions of our results are presented in section 4. Finally, we draw a conclusion in section 5.

2. Related works

Quantum-enhanced ML algorithms can surpass their classical ML counterpart algorithm in diverse applications [8, 13]. Taylor [14] argued that quantum computing is similar to a double-edged sword; it significantly improves information technologies and could cause catastrophic or irreversible events [15].

The hybrid quantum-classical DL approach, where a quantum device computes only part of the model, has potential for current NISQ technologies. The authors [16] proposed a hybrid quantum-classical convolutional neural network (CNN) model for COVID19 prediction. A hybrid quantum CNN model was introduced in [17] to secure models and avoid privacy leakage attacks.

The position of this paper is to continue our previous research [18, 19] (exploring quantum ML approaches for DGA detection) by experimenting with the VQC algorithm [8]. In contrast with [20] which only evaluates one variational circuit and two optimizers, our study investigates 12 types of optimizers, four types of feature maps, and four types of variational circuits. To the best of our knowledge, this paper is the first research on the VQC algorithm for the cybersecurity botnet DGA classification, which also includes a comparison with a hybrid quantum-classical DL approach.

3. Methods

3.1 Dataset and features

In this study, we used the dataset that we used in our previous research [21], which adopted the Patsakis' approach [22] in using the Alexa Top 1M domains and ten botnet DGA families (total 1,803,333 domain names) as the ground-truth dataset for botnet detection (Table 1).

Table 1. Datasets of DGA and legitimate domain names

Dataset	Number	Sample Content
Alexa	1,000,000	wikipedia.org
Conficker	100,000	hkcoaxcnjf.net
Cryptolocker	100,000	tgrmkncpkhaj.biz
Goz	1,667	atgxucqshdurghqdjyxti.ru
Matsnu	100,000	scoreadmireluckapplyfitcouple.com
New_Goz	1,666	1jwwz47ue0sakssvy4e1jx8z03.org
Pushdo	100,000	nafwupwi.ru
Ramdo	100,000	wsqwecgygoaakesq.org
Rovnix	100,000	theirtheandaloneinto.com
Tinba	100,000	ghlwtuutpkwm.com
Zeus	100,000	dd12071h5p3isqezvxr16k41j1.net

$$H(X) = -\sum_{i \in X} p_i \log_2 p_i \tag{1}$$

$$D_{KL}(P \,||\, Q) = \sum_i p_i \log\left(\frac{p_i}{q_i}\right) \tag{2}$$

$$JSD_{\pi_1, \ldots, \pi_n}(P_1, \ldots, P_n) = H\left(\sum_{i=1}^{n} \pi_i P_i\right) - \sum_{i=1}^{n} \pi_i H(P_i) \tag{3}$$

$$W_{N-gram}(i) = \log_{10}\left(\sum_{i=1}^{n} C_{N-gram}(i)\right) \tag{4}$$

The feature engineering processed the domain names into seven features: domain's character length (CharLength), entropy value, relative entropy (RE) from the Alexa domain names (REAlexa), the minimum of RE with botnet DGAs domain names (MinREBotnets), information radius (IRad) value, a new value generated by decision tree algorithm (TreeNewFeature), and domain's reputation score (ReputationAlexa). We released our codes [23] and datasets [21] to facilitate reproducible research.

The first feature is the domain name character length (CharLength). Then, the entropy values were calculated using Shannon's function Eq. (1), where *P* is the domain name's characters probability distribution. Our previous publication reported the importance of entropy value for statistical analysis for securing DNS traffic [24].

Table 2. Combinations of VQC algorithm's components

Feature Maps	Variational Forms	Optimizers
PauliFeature Map	EfficientSU2	AQGD, CG, COBYLA.
RawFeature Vector	Excitation Preserving	GSLS, L_BFGS_B, NELDER_MEAD.
ZFeatureMap	Real Amplitudes	NFT, P_BFGS, POWELL.
ZZFeatureMap	TwoLocal	SLSQP, SPSA, TNC.

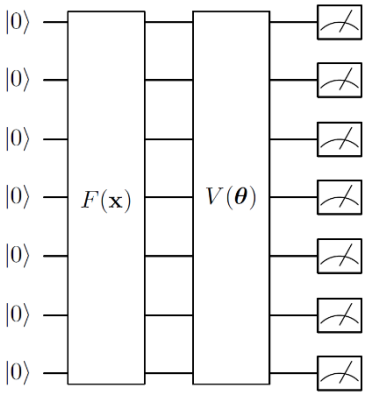

Figure. 1 Illustration of the quantum parts of VQC with seven qubits, where F(x) is encoding classical data into a quantum state, and $V(\theta)$ is the parameterized circuits with adjustable parameters θ

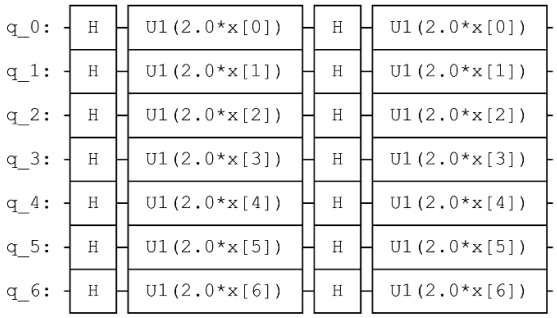

Figure. 2 A ZFeatureMap circuit with seven qubits

The RE values were computed using Kullback–Leibler divergence function Eq. (2) to measure the distance or similarity of a domain name with the character probability distributions of Alexa or DGA domain names, where *P* is the target domain name, *and Q* is the probability distributions to be compared. The IRad values were calculated using the Jensen–Shannon divergence function Eq. (3), which measures the similarity/distance of a domain name with two or more probability distributions, in our case, the ten botnet DGA families domains.

The next feature was generated using a decision tree algorithm (TreeNewFeature). We combined entropy, REAlexa, MinREBotnets, and CharLength features and used them to train a decision tree predictive model. The last feature was the Alexa reputation score (ReputationAlexa), which adopted Zhao's work [25] as shown in Eq. (4), where *W* is the weight matrix used to calculate the reputation score and *CN-gram* is the character *n*-gram frequencies.

3.2 VQC algorithm-based model

The first quantum ML-based classification algorithm used in this research is the VQC algorithm [8] (Fig. 1). Our research constraint uses the VQC algorithm implemented in the original paper [8], using Qiskit's VQC library [26] without modifications, to focus only on combining VQC components and investigating its performance.

As shown in Table 2, the focus of this study is to examine the combinations of 12 types of optimizer algorithms, four types of feature maps, and four types of variational form circuits to identify a combination that produces high accuracy for botnet DGA detection.

3.2.1. Quantum feature-mapping circuits

The quantum feature-mapping circuit in the VQC algorithm is used for quantum input state preparation (converting/transforming classical data input into the quantum state). This research uses four feature-mapping circuits (PauliFeatureMap, RawFeatureVector, ZFeatureMap, and ZZFeatureMap) (Table 2). We implemented all feature-mapping circuits using Qiskit's library [26].

We used the first feature map in the data-encoding circuits, the Pauli expansion circuit (PauliFeatureMap), which takes inputs from an initial set of data in classical form and builds derived values (known as features). The ZFeatureMap data encoding circuit is the first-order Pauli Z-evolution circuit (Fig. 2). It comprises several Hadamard and unitary gate sets. Hadamard gates function on a single qubit to change the qubit state into a superposition state, with an equal probability of becoming 0 or 1. Unitary gates conduct unitary operations for all involved qubits. This study's last feature map circuit is the second-order Pauli Z-evolution circuit (ZZFeatureMap data encoding).

3.2.2. Variational circuits

This research uses four variational forms, namely, EfficientSU2, ExcitationPreserving, RealAmplitudes, and TwoLocal (Table 2). The parameter of this second circuit is the training parameters of the classification model. This research observes the accuracy under different combinations of variational forms, feature maps, and optimization algorithms.

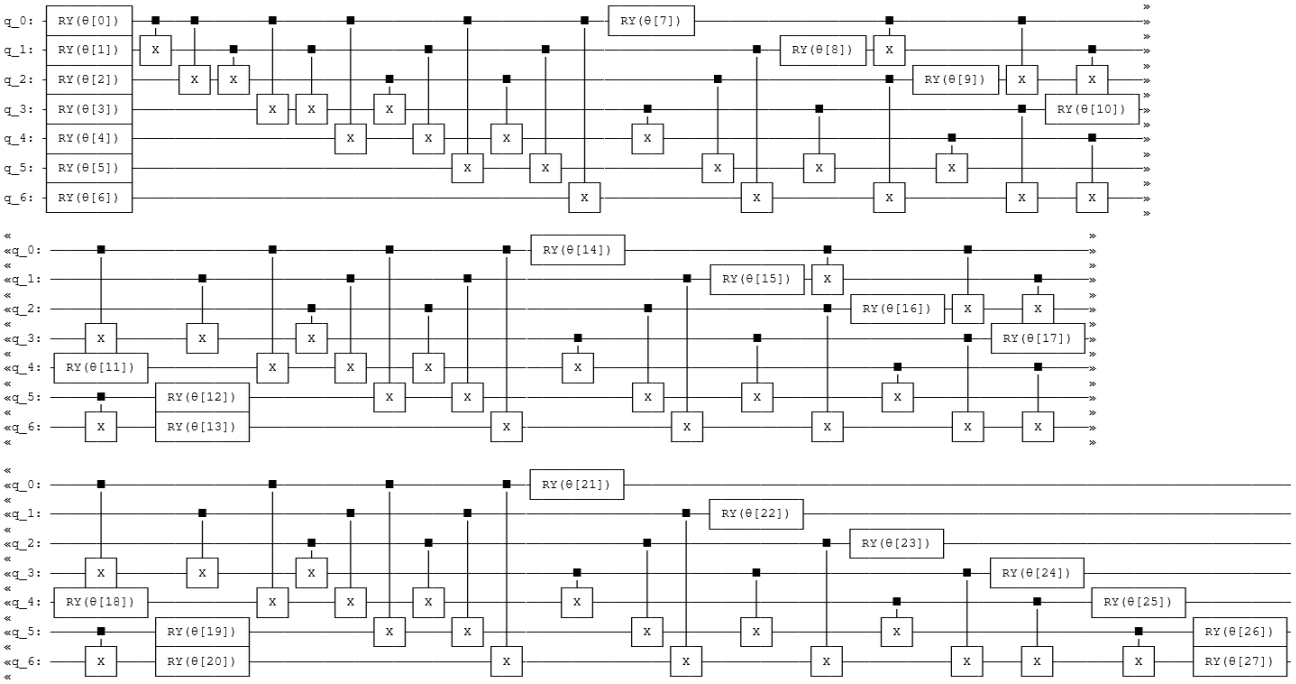

Figure. 3 Illustration of seven qubits RealAmplitudes circuit

Therefore, for this research's replicability, we purposely applied Qiskit's default settings [26].

Fig. 3 shows variational circuits using RealAmplitudes, comprising alternating Y rotations and CX entanglements. The rotation y (Ry) gate is a single-qubit rotation through angle θ (radians) around the y-axis. The CX gate refers to a controlled-X gate. RealAmplitudes prepares quantum states comprising only real amplitudes, whereas the complex part is always 0. The 2-local circuit comprises alternating entanglement and rotation layers. Single-qubit rotation gates are applied to all qubits. The EfficientSU2 consists of layers of single-qubit operations spanned by SU(2), referring to a particular unitary group of degree 2 as 2×2 unitary matrices with determinant 1 (e.g., Pauli rotation gates) and CX entanglements.

3.2.3. Optimizers

Optimization methods are used in ML algorithms to identify the best parameters for a given model. It comprises an objective function that will be maximized or minimized and some tunable parameters. We used 12 optimizer algorithms in Qiskit's library [26]; namely, analytic quantum gradient descent (AQGD), the conjugate gradient (CG), constrained optimization by linear approximation optimizer (COBYLA), Gaussian-smoothed line search (GSLS), limited-memory Broyden–Fletcher–Goldfarb–Shanno bound (L_BFGS_B), NELDER_MEAD, Nakanishi–Fujii–Todo (NFT), P_BFGS, POWELL, sequential least-squares programming (SLSQP), simultaneous perturbation stochastic approximation (SPSA), and truncated Newton (TNC) (Table 2).

3.2.4. VQC-based model experiments

Using the dataset explained in section 3.1 (selected randomly n = 1,000) and combinations of four feature maps, four variational forms, and twelve optimizer algorithms, we conducted 192 experiments that run on Qiskit's state vector simulator. We limit our study to using a simulator as conducting experiments on a real quantum computer will be costly. We aim to discover the effects of combinations of local optimizer algorithms, feature maps, and variational forms on accuracy results.

3.3 Hybrid quantum-classical DL model

Fig. 4 shows the architecture of our proposed hybrid quantum DL model. The classical parts of the model are Keras sequential models, comprising Keras dense layers with ReLU activation, a dropout layer, and a dense layer with sigmoid activation. Using the parameterized quantum circuits (PQCs) approach [27,28], we added a quantum layer between the dropout and dense layer. We conducted the implementation using Pennylane [29]. The quantum layer combines Pennylane's embedding to encode classical input data into a quantum state and layers' circuit (trainable parameterized circuit).

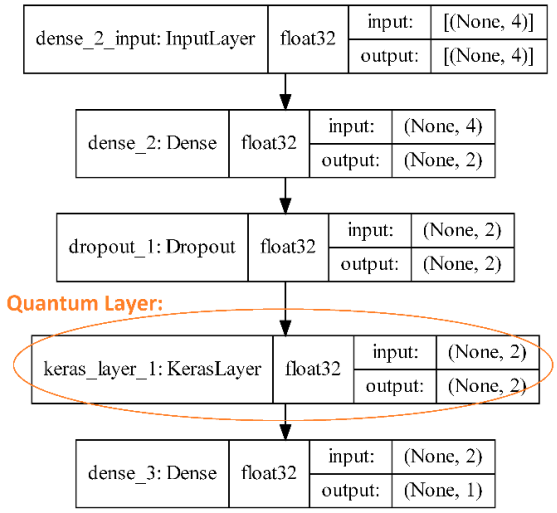

Figure. 4 Hybrid quantum-classical deep learning model

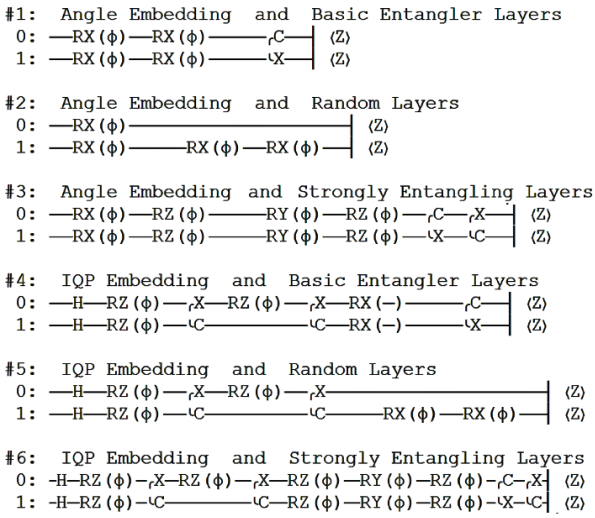

Figure. 5 Circuits of the quantum layers

Optimizer	Average	Max	Min
SLSQP	65,8%	84,4%	43,6%
TNC	64,9%	84,4%	43,6%
CG	62,7%	84,4%	43,6%
POWELL	62,3%	84,4%	43,6%
SPSA	62,1%	84,4%	43,6%
COBYLA	62,0%	84,4%	43,6%
P_BFGS	61,8%	84,4%	43,6%
L_BFGS_B	61,5%	84,4%	43,6%
NELDER_MEAD	61,5%	84,4%	43,6%
NFT	60,3%	78,0%	43,6%
GSLS	52,0%	84,0%	40,0%
AQGD	50,9%	56,0%	48,4%
Variational Form	**Average**	**Max**	**Min**
TwoLocal	67,3%	84,4%	47,2%
RealAmplitudes	65,5%	84,4%	41,6%
EfficientSU2	65,4%	84,4%	40,0%
ExcitationPreserving	49,3%	56,0%	43,6%
Feature Map	**Average**	**Max**	**Min**
RawFeatureVector	72,3%	84,4%	43,6%
ZFeatureMap	68,9%	78,4%	40,0%
PauliFeatureMap	51,3%	58,8%	41,6%
ZZFeatureMap	51,2%	57,6%	46,4%

Figure. 6 The average, maximum, and minimum accuracy of feature maps, variational forms, and optimizers

We ran experiments using six combinations of ansatz (Fig. 5). Angle embedding encodes features into rotation angles of qubits, and IQP embedding encodes features into qubits using the diagonal gates of an IQP circuit [8]. The basic entangler layer comprises single-qubit rotations on each qubit, followed by CNOT gates. The random layer randomly selects single-qubit rotations and 2-qubit entangling gates. The strongly entangling layer consists of single-qubit rotations and entanglers, inspired by the circuit-centric classifier design [30].

The experiments using this hybrid model used the same settings as those for the VQC-based model to produce comparable results. Furthermore, our proposed hybrid model was simulated using a qiskit.aer device of Pennylane-Qiskit plugin [31], which can simulate noise. We then applied noise models from eight IBM quantum computing devices [26]. Therefore, the simulations mimic various noises of actual computations on a real quantum computer.

4. Results and discussions

For now, using quantum computing technology as an accelerator for artificial intelligence (AI) or machine learning (ML) applications is still lacking in terms of practicality and cost-efficiency, compared with existing approaches such as the graphics processing unit (GPU)-based accelerators. Quantum computers are still developing and not yet widely adopted for applications in industries, including cybersecurity. However, research and experiments to explore the full potentials of the quantum ML approach for solving a real-life computing problem are critical.

4.1 VQC-based model

The 192 experiment results in Table 3 show that the maximum accuracy is 84.4%, whereas the average is 60.7%, and the minimum is 40.0%. The evaluation metric is given in Eq. (5), where *TP*, *TN*, *FP*, *FN* stands for true positive, true negative, false positive, and false negative, respectively.

Fig. 6 shows a gradient-colored representation to investigate details about each VQC algorithm's component (feature maps, variational forms, and optimizer). Most optimizer algorithms could reach the maximum accuracy (84.4%), whereas the AQGD algorithm has the worst accuracy (only 56.0% maximum accuracy).

$$\boldsymbol{Accuracy} = \frac{\boldsymbol{TP+TN}}{\boldsymbol{TP+TN+FP+FN}} \quad (5)$$

Table 3. VQC-based model's accuracy

Optimizer	Feature Map	TwoLocal	RealAmplitudes	EfficientSU2	ExcitationPreserving
SPSA	RawFeatureVector	84.4%	84.4%	84.4%	43.6%
	PauliFeatureMap	54.8%	50.0%	49.6%	48.4%
	ZFeatureMap	74.8%	77.2%	77.6%	56.0%
	ZZFeatureMap	56.4%	52.4%	46.4%	52.4%
AQGD	RawFeatureVector	59.2%	66.8%	61.6%	43.6%
	PauliFeatureMap	59.0%	51.6%	59.0%	48.4%
	ZFeatureMap	72.8%	74.0%	76.4%	56.0%
	ZZFeatureMap	59.0%	49.6%	59.0%	48.4%
CG	RawFeatureVector	84.4%	84.4%	84.4%	43.6%
	PauliFeatureMap	52.4%	58.8%	53.6%	48.4%
	ZFeatureMap	74.4%	76.8%	77.2%	56.0%
	ZZFeatureMap	52.0%	56.4%	51.2%	48.4%
COBYLA	RawFeatureVector	84.4%	84.4%	84.4%	43.6%
	PauliFeatureMap	47.2%	57.2%	53.6%	48.4%
	ZFeatureMap	77.2%	76.8%	77.6%	56.0%
	ZZFeatureMap	48.4%	53.2%	50.8%	48.4%
L_BFGS_B	RawFeatureVector	84.4%	84.4%	84.4%	43.6%
	PauliFeatureMap	52.0%	51.6%	50.7%	48.4%
	ZFeatureMap	75.2%	75.6%	78.4%	56.0%
	ZZFeatureMap	54.0%	49.2%	47.2%	48.4%
GSLS	RawFeatureVector	52.0%	84.0%	61.2%	43.6%
	PauliFeatureMap	52.0%	41.6%	48.0%	48.4%
	ZFeatureMap	52.0%	48.0%	40.0%	56.0%
	ZZFeatureMap	52.0%	53.2%	52.0%	48.4%
NELDER_MEAD	RawFeatureVector	84.4%	84.4%	83.6%	43.6%
	PauliFeatureMap	50.0%	56.4%	56.4%	48.4%
	ZFeatureMap	71.2%	72.8%	70.8%	56.0%
	ZZFeatureMap	52.4%	55.2%	49.2%	48.4%
NFT	RawFeatureVector	71.6%	60.0%	76.4%	43.6%
	PauliFeatureMap	57.6%	55.2%	50.0%	48.4%
	ZFeatureMap	78.0%	77.2%	77.2%	56.0%
	ZZFeatureMap	51.2%	50.8%	51.6%	48.4%
P_BFGS	RawFeatureVector	84.4%	84.4%	84.4%	43.6%
	PauliFeatureMap	52.4%	50.4%	50.8%	48.4%
	ZFeatureMap	76.4%	76.0%	76.8%	56.0%
	ZZFeatureMap	54.4%	50.8%	50.8%	48.4%
POWELL	RawFeatureVector	84.4%	84.4%	84.4%	43.6%
	PauliFeatureMap	54.0%	57.2%	54.4%	48.4%
	ZFeatureMap	74.4%	76.8%	73.6%	56.0%
	ZZFeatureMap	52.4%	53.2%	51.2%	48.4%
SLSQP	RawFeatureVector	84.4%	84.4%	84.4%	43.6%
	PauliFeatureMap	52.0%	51.6%	53.2%	48.4%
	ZFeatureMap	77.6%	76.0%	76.8%	56.0%
	ZZFeatureMap	51.2%	57.6%	51.6%	48.4%
TNC	RawFeatureVector	83.6%	84.0%	84.4%	43.6%
	PauliFeatureMap	52.0%	52.8%	53.6%	48.4%
	ZFeatureMap	71.6%	72.0%	77.2%	56.0%
	ZZFeatureMap	52.0%	57.2%	52.4%	48.4%

Table 4. Hybrid quantum DL models' accuracy (%)

Noise Model	Circuits of the quantum layers for the hybrid quantum DL model *					
	#1	#2	#3	#4	#5	#6
ibmq_16_ melbourne	86.4	93.9	84.9	51.7	93.4	89.4
ibmq_5_ yorktown	86.4	93.9	85.4	47.2	93.4	90.4
ibmq_athens	86.9	92.9	84.9	44.2	93.4	88.9
ibmq_belem	86.9	92.4	85.4	56.2	93.9	88.4
ibmq_lima	88.9	93.9	84.9	49.7	93.4	81.9
ibmq_quito	86.4	94.9	84.4	54.2	94.4	88.9
ibmq_ santiago	87.9	93.9	86.4	45.2	92.9	88.4
ibmqx2	86.4	92.4	83.9	46.2	79.3	79.8

* #1: Angle embedding and basic entangler layers, #2: Angle embedding and random layers, #3: Angle embedding and strongly entangling layers, #4: IQP embedding and basic entangler layers, #5: IQP embedding and random layers, and #6: IQP embedding and strongly entangling layers

For the variational forms, the best performance is the TwoLocal, whereas ExcitationPreserving has the lowest accuracy. The quantum circuit uses the variational form as part of a parameterized gate trained to function as the classifier. For the feature map, RawFeatureVector has the highest accuracy (84.4%), and ZZFeatureMap has the lowest accuracy (57.6%), but with only slight differences from the performance of PauliFeatureMap. Feature map is used to transform input data from classical data into a quantum bit. These experimental results show that the feature map and variational circuit selections should be meticulously considered.

The authors [32] analyzed the VQC approach using the wine, breast cancer, and MNIST datasets. Their maximum testing accuracy was 91%, higher than our results (84.4%). In another paper, [20], the authors implement VQC models for restaurant reviews dataset, with EfficientSU2 variational circuit and COBYLA and AQGD optimizers. Their AQGD optimizer-based model produced 77% accuracy, which is lower than our results (84.4%). The difference in datasets used in the experiment contributes to the accuracy result differences.

4.1.1. ANOVA analysis

This section presents our investigation on each VQC component's effects on accuracy. We conducted six ANOVA calculations using R [33], namely, ANOVA one-way analysis for optimizers, variational forms, feature maps, and ANOVA two-way analysis for feature maps with variational forms, feature maps with optimizers, and variational forms with optimizers.

From the ANOVA one way analysis results, optimizers have no statistically significant effect on accuracy ($p > 0.05$), with $F(11,166) = 0.654$, $p = 0.78$, and $\eta 2[g] = 0.42$. However, variational forms and feature maps have a statistically significant effect on accuracy, with variational forms' $p < 0.05$, with $F(3,174) = 18.999$, $p = 1.05 \times 10^{-10}$, and $\eta 2[g] = 0.247$, and feature maps' $p < 0.05$, with $F(3,174) = 47.523$, $p = 1.74 \times 10^{-22}$, and $\eta 2[g] = 0.45$.

From the two-way ANOVA results, a statistically significant interaction ($p < 0.05$) between feature maps and variational forms for accuracy occurs, with $F(9,162) = 34.250$, $p = 3.43 \times 10^{-33}$, and $\eta 2[g] = 0.622$. However, no significant interaction ($p > 0.05$) occurred between two-way ANOVA analysis of feature maps and optimizers ($F(33,136) = 0.627$, $p = 9.4 \times 10^{-1}$, and $\eta 2[g] = 0.32$), and similarly, between two-way ANOVA analysis of variational forms with optimizers ($F(30,134) = 0.121$, $p = 1.00$, and $\eta 2[g] = 0.026$).

This finding implies that thorough considerations on the feature maps and variational forms selection for building quantum circuits could affect the model's performance. Like in a neural network or deep learning, where the architecture has a significant effect on its performance, this paper contributes to the literature as a direction for future research on investigating various quantum circuits architecture for quantum ML applications.

4.2 Hybrid quantum-classical DL model

Table 4 shows the results of our hybrid quantum DL model experiments. Even after considering noise from the quantum computing devices, the angle embedding and random layers combination deliver the highest accuracy (as high as 94.9%).

Table 5 shows the detailed performance results of the experiment with the ibmq_quito noise model, where the metrics measurements during the training processes can be seen in Fig. 7. The combination of AngleEmbedding and RandomLayers circuits reached 94.7% accuracy, with a false-positive ratio or false alarm ratio of 5.2%.

This evidence shows the superior performance of our proposed hybrid quantum DL model compared to the VQC algorithm-based model (Table 3). The average accuracy of the angle embedding and random layers combination is 93.5%, with minimum accuracy at 92.4%, which is still higher than the maximum accuracy of the VQC-based model (84.4%, see Table 3).

Table 5. Detail metrics of the ibmq_quito experiment

Noise Model	Circuits of the quantum layers for the hybrid quantum DL model					
	#1	#2	#3	#4	#5	#6
TP	78	98	74	32	98	83
FP	2	5	2	20	6	2
TN	94	91	94	76	90	94
FN	25	5	29	71	5	20
Accuracy	86.4	94.9	84.4	54.2	94.4	88.9
Precision	97.5	95.1	97.3	61.5	94.2	97.6
Recall	75.7	95.1	71.8	31.0	95.1	80.5
AUC	97.8	97.7	97.1	49.2	95.4	97.6

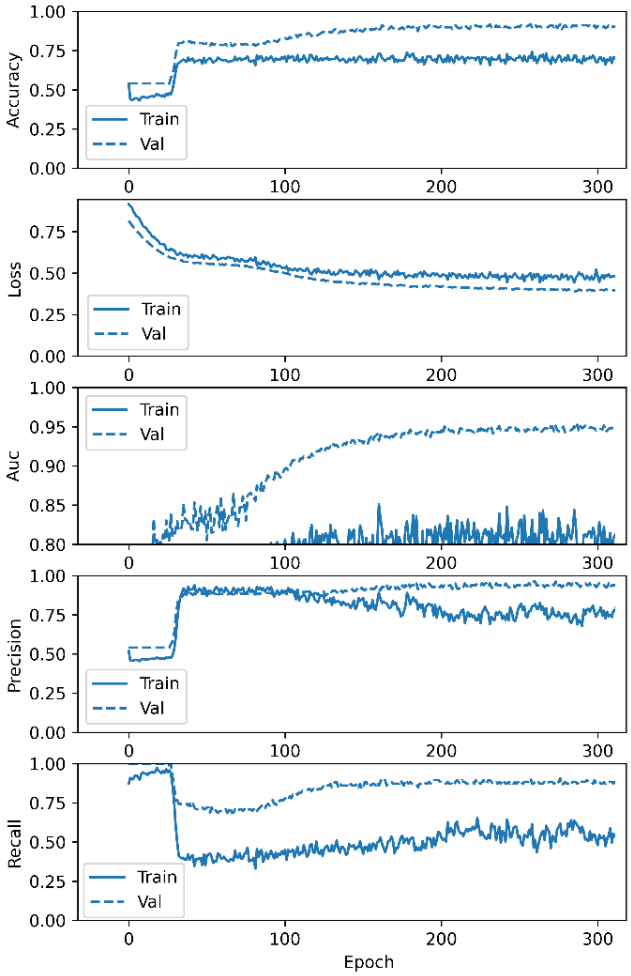

Figure. 7 The plots of accuracy, loss, AuC, precision, and recall from the ibmq_quito experiment

From these results, combining IQP embedding and basic entangler layers had the poorest performance. The accuracy was 56.2%, 49.3%, and 44.2% for the maximum, average, and minimum accuracy. This result indicates that the IQP embedding and basic entangler layers combination has inferior performance compared to the other combinations in our case.

These results show that the architecture selection is crucial in the deep learning and neural network fields as it affects performance. The quantum circuit architecture consisting of angle embedding and Random Layers delivers a high-performance result, with an average of 93.5% accuracy.

4.2.1. Statistical t-test analysis

We statistically analyzed the performance (accuracy) significance of the VQC-based model (Table 3) and our proposed hybrid quantum DL model (Table 4). First, we investigated whether the mean of the hybrid quantum model's accuracies (*mA*) equals the mean of the VQC-based model's accuracy (*mB*). The results show a low significance level of the *t-test* (p = 8.376 x 10^{-13}), with a *t-test* statistic value (t) = −8.804 and degrees of freedom (df) = 67.213. Therefore, we reject the *H0: mA = mB* hypothesis and accept the alternative hypothesis *Ha: mA ≠ mB*.

Next, we performed *t-tests* for *H0: mA ≤ mB* and *H0: mA ≥ mB*. The first results show strong evidence against the null hypothesis (*H0: mA ≤ mB*) because p = 4.188 x 10^{-13}, t = −8.804, and df = 67.213. The subsequent results (p = 1, t = −8.804, and df = 67.213) indicate strong evidence to accept the null hypothesis (*H0: mA ≥ mB*).

This finding implies that our hybrid DL model performs better than the VQC-based model. Including traditional layers in the hybrid quantum DL model's architecture may give a potential to harness all the goodness of current well-developed traditional DL techniques for our hybrid quantum DL model.

5. Conclusion

Our VQC-based experiments discovered that variational forms and feature maps significantly affect accuracy. The ANOVA analysis results in section 4.1.1. confirm that feature maps and variational forms are essential parts of the VQC algorithm based on our case study results.

We spotlight the novelty of our hybrid quantum-classical DL model, which delivers the benefit of higher performance (94.9% maximum accuracy) than the VQC algorithm-based model (84.4% maximum accuracy). The *t*-test analysis results in section 4.2.1. become statistical evidence supporting our claim that our proposed hybrid quantum DL model's performances are superior to the VQC-based model for accuracy.

Also, we found the effectiveness of Angle Embedding and Random Layers quantum circuits for a hybrid quantum DL's architecture.

As the existing conventional DL approach already delivers practicality with satisfactory

performance, future research should focus on finding the right tasks and problems where the quantum ML could improve the performances of its traditional algorithm counterpart. In addition, as encryption currently plays a vital role in delivering privacy and confidentiality in confidential ML applications, research of quantum-based confidential ML approach is needed. Future works should also investigate the randomness contribution to the accuracy performance by using entropy analysis.

Conflicts of interest

The authors declare no conflict of interest.

Author contributions

Conceptualization, H. Suryotrisongko; methodology, H. Suryotrisongko; software, H. Suryotrisongko; validation, H. Suryotrisongko and Y. Musashi; investigation, H. Suryotrisongko and Y. Musashi; data curation, H. Suryotrisongko; writing—original draft preparation, H. Suryotrisongko; writing—review and editing, Y. Musashi; visualization, H. Suryotrisongko; supervision, Y. Musashi; funding, Y. Musashi.

Acknowledgments

This work was financially and technically supported by Kumamoto University's Center for Management of Information Technology (CMIT) researchers and staff members, and professors in the Faculty of Advanced Science and Technology. Also, the authors thank all the professors and colleagues from Institut Teknologi Sepuluh Nopember.

References

[1] M. Singh, M. Singh, and S. Kaur, "Issues and challenges in DNS based botnet detection: A survey", *Computers & Security*, Vol. 86, pp. 28–52, 2019, doi: 10.1016/j.cose.2019.05.019.

[2] T. S. Wang, H. T. Lin, W. T. Cheng, and C. Y. Chen, "DBod: Clustering and detecting DGA-based botnets using DNS traffic analysis", *Computers & Security*, Vol. 64, pp. 1–15, 2017, doi: 10.1016/j.cose.2016.10.001.

[3] L. Buffoni and F. Caruso, "New trends in quantum machine learning", *Europhysics Letter*, Vol. 132, No. 6, p. 60004, 2020, doi: 10.1209/0295-5075/132/60004.

[4] K. Mitarai, M. Negoro, M. Kitagawa, and K. Fujii, "Quantum circuit learning", *Phys. Rev. A*, Vol. 98, No. 3, p. 032309, 2018, doi: 10.1103/PhysRevA.98.032309.

[5] 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, doi: 10.1038/nature23474.

[6] V. Dunjko and H. J. Briegel, "Machine learning & artificial intelligence in the quantum domain: a review of recent progress", *Rep. Prog. Phys.*, Vol. 81, No. 7, p. 074001, 2018, doi: 10.1088/1361-6633/aab406.

[7] M. Schuld, I. Sinayskiy, and F. Petruccione, "An introduction to quantum machine learning", *Contemporary Physics*, Vol. 56, No. 2, pp. 172–185, 2015, doi: 10.1080/00107514.2014.964942.

[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, doi: 10.1038/s41586-019-0980-2.

[9] S. Pouyanfar, S. Sadiq, Y. Yan, H. Tian, Y. Tao, M. P. Reyes, M. L. Shyu, S. C. Chen, and S. S. Iyengar, "A Survey on Deep Learning: Algorithms, Techniques, and Applications", *ACM Comput. Surv.*, Vol. 51, No. 5, Art. No. 92, pp. 1-36, 2019, doi: 10.1145/3234150.

[10] S. Garg and G. Ramakrishnan, "Advances in Quantum Deep Learning: An Overview", *arXiv:2005.04316 [quant-ph]*, May 2020, Accessed: Mar. 29, 2021. [Online]. Available: http://arxiv.org/abs/2005.04316

[11] D. Riste, M. P. D. Silva, C. A. Ryan, A. W. Cross, A. D. Córcoles, J. A. Smolin, J. M. Gambetta, J. M. Chow, and B. R. Johnson, "Demonstration of quantum advantage in machine learning", *Npj Quantum Information*, Vol. 3, No. 1, pp. 1–5, 2017, doi: 10.1038/s41534-017-0017-3.

[12] A. Chalumuri, R. Kune, and B. S. Manoj, "A hybrid classical-quantum approach for multi-class classification", *Quantum Inf Process*, Vol. 20, No. 3, p. 119, 2021, doi: 10.1007/s11128-021-03029-9.

[13] C. Zoufal, A. Lucchi, and S. Woerner, "Quantum Generative Adversarial Networks for learning and loading random distributions", *Npj Quantum Inf*, Vol. 5, No. 1, p. 103, 2019, doi: 10.1038/s41534-019-0223-2.

[14] R. D. Taylor, "Quantum Artificial Intelligence: A 'precautionary' U.S. approach?", *Telecommunications Policy*, p. 101909, 2020, doi: 10.1016/j.telpol.2020.101909.

[15] P. Choy, D. Cates, F. Chehwan, C. Rodriguez, A. Leider, and C. C. Tappert, "Cryptography in Quantum Computing", *Advances in Intelligent Systems and Computing*, Vol. 1069, pp. 377–393, 2020, doi: 10.1007/978-3-030-32520-6_30.

[16] E. H. Houssein, Z. Abohashima, M. Elhoseny, and W. M. Mohamed, "Hybrid quantum convolutional neural networks model for COVID-19 prediction using chest X-Ray images", *arXiv:2102.06535 [cs, eess]*, 2021, Accessed: Mar. 25, 2021. [Online]. Available: http://arxiv.org/abs/2102.06535

[17] C. H. H. Yang, J. Qi, S. Y. C. Chen, P. Y. Chen, S. M. Siniscalchi, X. Ma, and C. H. Lee, "Decentralizing feature extraction with quantum convolutional neural network for automatic speech recognition", In: *Proc. of ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6523–6527, 2021. doi: 10.1109/ICASSP39728.2021.9413453.

[18] H. Suryotrisongko and Y. Musashi, "Evaluating hybrid quantum-classical deep learning for cybersecurity botnet DGA detection", *Procedia Computer Science*, Vol. 197, pp. 223–229, 2022, doi: 10.1016/j.procs.2021.12.135.

[19] H. Suryotrisongko and Y. Musashi, "Hybrid Quantum Deep Learning with Differential Privacy for Botnet DGA Detection", In: *Proc. of 2021 13th International Conference on Information Communication Technology and System (ICTS)*, pp. 68–72, 2021. doi: 10.1109/ICTS52701.2021.9608217.

[20] N. Joshi, P. Katyayan, and S. A. Ahmed, "Evaluating the Performance of Some Local Optimizers for Variational Quantum Classifiers", *J. Phys.: Conf. Ser.*, Vol. 1817, No. 1, p. 012015, 2021, doi: 10.1088/1742-6596/1817/1/012015.

[21] H. Suryotrisongko and Y. Musashi, "Botnet DGA Dataset", *IEEE Dataport*, 2020, doi: https://dx.doi.org/10.21227/rg6z-z622.

[22] C. Patsakis, F. Casino, and V. Katos, "Encrypted and covert DNS queries for botnets: Challenges and countermeasures", *Computers & Security*, Vol. 88, p. 101614, 2020, doi: 10.1016/j.cose.2019.101614.

[23] H. Suryotrisongko, "Botnet DGA Detection", *IEEE Code Ocean*, doi: https://doi.org/10.24433/CO.4005597.v2.

[24] Y. Matsubara, Y. Musashi, K. Sugitani, and T. Moriyama, "Open DNS Resolver Activity in Campus Network System", In: *Proc. of 2015 8th International Conference on Intelligent Networks and Intelligent Systems (ICINIS)*, pp. 145–148, 2015. doi: 10.1109/ICINIS.2015.30.

[25] H. Zhao, Z. Chang, G. Bao, and X. Zeng, "Malicious Domain Names Detection Algorithm Based on N-Gram", *Journal of Computer Networks and Communications*, Vol. 2019, p. 4612474, 2019, doi: 10.1155/2019/4612474.

[26] G. Aleksandrowicz, T. Alexander, P. Barkoutsos, L. Bello, Y. B. Haim, D. Bucher, F. J. C. Hernández, J. C. Franquis, A. Chen, C. F. Chen, and J. M. Chow, "Qiskit: An Open-Source Framework for Quantum Computing", *International Business Machine Corporation (IBM)*, 2019, doi: 10.5281/zenodo.2562111.

[27] M. Benedetti, E. Lloyd, S. Sack, and M. Fiorentini, "Parameterized quantum circuits as machine learning models", *Quantum Sci. Technol.*, Vol. 4, No. 4, p. 043001, 2019, doi: 10.1088/2058-9565/ab4eb5.

[28] Y. Du, M. H. Hsieh, T. Liu, and D. Tao, "Expressive power of parametrized quantum circuits", *Phys. Rev. Research*, Vol. 2, No. 3, p. 033125, 2020, doi: 10.1103/PhysRevResearch.2.033125.

[29] V. Bergholm, J. Izaac, M. Schuld, C. Gogolin, M. S. Alam, S. Ahmed, J. M. Arrazola, C. Blank, A. Delgado, S. Jahangiri, and K. McKiernan, "PennyLane: Automatic differentiation of hybrid quantum-classical computations", *arXiv:1811.04968 [physics, physics:quant-ph]*, 2020, Accessed: Mar. 25, 2021. [Online]. Available: http://arxiv.org/abs/1811.04968

[30] M. Schuld, A. Bocharov, K. M. Svore, and N. Wiebe, "Circuit-centric quantum classifiers", *Phys. Rev. A*, Vol. 101, No. 3, p. 032308, 2020, doi: 10.1103/PhysRevA.101.032308.

[31] R. LaRose, "Overview and Comparison of Gate Level Quantum Software Platforms", *Quantum*, Vol. 3, p. 130, 2019, doi: 10.22331/q-2019-03-25-130.

[32] S. Mardirosian, "Quantum-enhanced Supervised Learning with Variational Quantum Circuits, Thesis Master Computer Science", *Leiden Institute of Advanced Computer Science (LIACS)*, Leiden University, Leiden, Netherlands, 2019.

[33] R. B. Dessau and C. B. Pipper, ""R"--project for statistical computing", *Ugeskrift for laeger*, Vol. 170, No. 5, pp. 328–330, 2008