

Optimizing Quantum Neural Networks: A Comparative Study of QNN Architectures and Training Techniques

Desi Amirullah¹, Lipantri Mashur Gultom²
Teknik Informatika, Politeknik Negeri Bengkalis^{1,2},
andes@polbeng.ac.id¹, lipantri@polbeng.ac.id²

Abstract

This study explores various architectures of Quantum Neural Networks (QNN) that utilize quantum computing to enhance machine learning model performance. The architectures examined include Hybrid Quantum Neural Network (H-QNN), Deep Quantum Neural Network (DQNN), Quantum Convolutional Neural Network (QCNN), and Coherent Feed Forward Quantum Neural Network. Additionally, the EfficientSU2 architecture demonstrated superior performance in accuracy and cost reduction. Optimization techniques such as Adaptive Moment Estimation (ADAM), Analytic Quantum Gradient Descent (AQGD), and Nakanishi-Fujii-Todo (NFT) were integrated to improve training efficiency. The study results show that EfficientSU2 with 10,000 iterations yielded the best results in terms of cost and accuracy across all tested datasets, with accuracy improvements reaching up to 97%. In conclusion, the EfficientSU2 architecture and the applied optimization techniques significantly enhanced the processing of complex data, paving the way for substantial advancements in quantum-based artificial intelligence applications.

Keywords : Quantum Computing, Quantum Neural Networks, Quantum Gradient Descent,

1. PENDAHULUAN

Perkembangan teknologi komputasi kuantum telah memberikan kontribusi signifikan dalam bidang *machine learning*, khususnya *deep learning* dalam pengembangan *Quantum Neural Networks* (QNN). Berbagai arsitektur QNN menawarkan cara-cara baru untuk memproses informasi dengan memanfaatkan prinsip-prinsip kuantum. Dalam penelitian ini, penting untuk mengeksplorasi arsitektur-arsitektur yang berbeda, yang masing-masing memiliki karakteristik dan keunggulan unik.

Salah satu pendekatan yang awal ditemukan adalah *Hybrid Quantum Neural Network* (H-QNN) (Chen, 2023), yang mengintegrasikan elemen kuantum dan klasik untuk meningkatkan kinerja model. Dengan menggabungkan kekuatan komputasi kuantum dan algoritma pembelajaran klasik, jaringan ini menawarkan cara baru untuk memproses informasi secara efisien. Salah satu contoh dari pendekatan ini adalah *Hybrid Quantum-Classical Neural Network* (HQ-CNN) (Arthur, 2022), yang memanfaatkan arsitektur klasik untuk pengolahan data sambil memanfaatkan keunggulan kuantum dalam pengoptimalan dan representasi informasi. Pendekatan ini menunjukkan potensi dalam aplikasi yang memerlukan komputasi yang cepat dan akurat, seperti dalam *quantum deep learning*, di mana kompleksitas dan kedalaman model dapat dioptimalkan lebih lanjut.

Kemudian dikembangkan satu arsitektur *Deep Quantum Neural Network* (DQNN) (Mingard, 2024). Model ini memungkinkan aliran informasi dari lapisan input ke output tanpa adanya umpan balik, mirip dengan jaringan saraf klasik. Namun, keunggulan DQNN terletak pada kemampuannya untuk memanfaatkan superposisi kuantum, yang meningkatkan kapasitas dan kecepatan pemrosesan data, memungkinkan model untuk menangani informasi yang lebih kompleks. Arsitektur lain yang patut diperhatikan adalah *Quantum Convolutional Neural Network* (QCNN) (Mahmud, 2024). QCNN dirancang untuk pengolahan data spasial, seperti gambar, dengan menggunakan operasi konvolusi kuantum. Pendekatan ini tidak hanya mempercepat proses pengolahan, tetapi juga mampu mempertahankan fitur-fitur penting dalam data, sehingga meningkatkan akurasi dalam tugas-tugas seperti klasifikasi dan deteksi objek.

Selain itu, arsitektur *Deep Quantum Neural Network* memanfaatkan lapisan-lapisan yang lebih dalam untuk meningkatkan kapasitas model dalam menangkap pola kompleks. Implementasi teknik seperti *EfficientSU2* (Liu, 2024) pada arsitektur ini dapat meningkatkan efisiensi dalam pengoptimalan parameter, sehingga meningkatkan performa model secara keseluruhan. *Coherent Feed Forward Quantum Neural Network* (Singh, 2024) juga merupakan inovasi menarik dalam bidang ini, yang memfokuskan pada pengolahan informasi dengan cara yang lebih terstruktur dan sistematis. Dengan memanfaatkan prinsip-prinsip kuantum, arsitektur ini menjanjikan peningkatan akurasi dan kecepatan dalam berbagai aplikasi pembelajaran mesin. Secara keseluruhan, beragam arsitektur dalam QNN menunjukkan potensi besar untuk merevolusi cara pemrosesan informasi, membuka jalan bagi kemajuan teknologi di berbagai bidang.

Integrasi teknik optimasi seperti *Adaptive Moment Estimation* (ADAM) (Schuld, 2019), *Analytic Quantum Gradient Descent* (AQGD) (Mitarai, 2018), dan *Nakanishi-Fujii-Todo* (NFT) (Nakanishi, 2020) semakin memperkuat efektivitas pelatihan QNN. Ketiga metode ini menawarkan cara baru untuk mengatasi tantangan dalam konvergensi dan efisiensi model serta mendukung kemampuan jaringan dalam menghadapi data kompleks. Kombinasi inovasi dalam arsitektur dan teknik optimasi ini membuka peluang besar untuk kemajuan dalam kecerdasan buatan berbasis kuantum yang dapat diterapkan dalam berbagai bidang.

2. TINJAUAN PUSTAKA

Quantum Machine Learning (QML) (Schuld, 2021) adalah cabang ilmu yang menggabungkan prinsip-prinsip komputasi kuantum dengan teknik-teknik pembelajaran mesin. Tujuan utamanya adalah untuk memanfaatkan keunggulan komputasi kuantum untuk meningkatkan efisiensi dan efektivitas algoritma pembelajaran mesin tradisional. Di dalam QML, algoritma dan model pembelajaran mesin dirancang untuk dijalankan pada komputer kuantum yang dapat memproses informasi dengan cara yang berbeda dibandingkan dengan komputer klasik. Beberapa aspek kunci dari QML meliputi superposisi dan *entanglement* dimana komputer kuantum dapat memproses banyak kemungkinan sekaligus berkat superposisi, serta memanfaatkan *entanglement* untuk hubungan yang lebih kompleks antar data.

Quantum Deep Learning (QDL) (Pointing, 2024) adalah gabungan antara prinsip-prinsip *deep learning* dan komputasi kuantum. Tujuannya adalah untuk memanfaatkan kekuatan komputasi kuantum guna meningkatkan kemampuan model-model *deep learning* tradisional dalam memproses dan menganalisis data yang sangat besar dan kompleks. Beberapa aspek kunci dari *Quantum Deep Learning* meliputi arsitektur jaringan kuantum dimana arsitektur jaringan saraf yang dirancang untuk dijalankan di komputer kuantum, seperti *Quantum Neural Networks* (QNN) dan *Quantum Convolutional Neural Networks* (QCNN) (MacCormack, 2022). Arsitektur ini dapat memanfaatkan superposisi dan *entanglement* untuk meningkatkan efisiensi pemrosesan.

Quantum Neural Networks (QNN) (Abbas, 2021) adalah model pembelajaran mesin yang menggabungkan prinsip-prinsip dari jaringan saraf klasik dengan komputasi kuantum. QNN dirancang untuk memanfaatkan karakteristik unik komputasi kuantum, seperti superposisi dan *entanglement*, untuk meningkatkan kemampuan dalam memproses dan menganalisis data. QNN sering kali memiliki struktur yang mirip dengan jaringan saraf tradisional dengan lapisan input, lapisan tersembunyi, dan lapisan output. Namun, di dalam QNN unit-unit dalam lapisan tersebut dioperasikan menggunakan mekanisme kuantum. Salah satu keunggulan utama dari QNN adalah kemampuannya untuk memproses banyak keadaan data sekaligus berkat superposisi. Selain itu, *entanglement* memungkinkan hubungan yang kompleks antara data, yang dapat meningkatkan kapasitas model untuk menangkap pola-pola tersembunyi.

EfficientSU2 (Liu, 2024) adalah metode yang digunakan dalam konteks *Quantum Machine Learning* dan *Quantum Neural Networks* untuk mengoptimalkan parameter dalam

sirkuit kuantum. Ini adalah salah satu algoritma *Parameterized Quantum Circuit* (PQC) yang dirancang untuk meningkatkan efisiensi pelatihan model kuantum. *EfficientSU2* membangun sirkuit kuantum yang terdiri dari gerbang-unitari $SU(2)$, yang merupakan representasi umum dari gerbang kuantum. Struktur ini memungkinkan fleksibilitas dalam merancang sirkuit yang dapat memodelkan berbagai fungsi kuantum. *EfficientSU2* berusaha mengurangi kompleksitas sirkuit kuantum, yang membantu dalam mengurangi kebutuhan sumber daya komputasi dan meningkatkan kecepatan pelatihan.

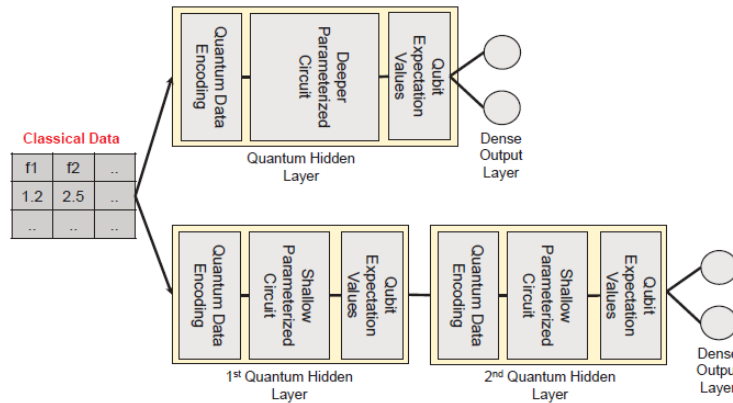
Adaptive Moment Estimation (ADAM) (Schuld, 2019) adalah algoritma optimasi yang populer digunakan dalam pelatihan model pembelajaran mesin, terutama dalam konteks *deep learning*. ADAM menggabungkan keuntungan dari dua algoritma sebelumnya, yaitu AdaGrad dan RMSProp, untuk memberikan pengoptimalan yang lebih efisien dan stabil. ADAM dapat menyesuaikan laju pembelajaran untuk setiap parameter secara individual, berdasarkan estimasi momen pertama (rata-rata gradien) dan momen kedua (rata-rata kuadrat gradien). Ini memungkinkan algoritma untuk lebih responsif terhadap perilaku gradien selama pelatihan. ADAM melakukan koreksi bias untuk estimasi momen pertama dan kedua, terutama pada awal pelatihan ketika nilai-nilai ini masih kecil. Ini membantu mencegah bias dalam pembaruan parameter. ADAM memiliki beberapa parameter yang dapat disesuaikan, seperti laju pembelajaran (*learning rate*), faktor *decay* untuk momen pertama dan kedua, serta *epsilon* untuk menghindari pembagian dengan nol.

Analytic Quantum Gradient Descent (AQGD) (Mitarai, 2018) adalah metode optimasi yang dirancang khusus untuk digunakan dalam konteks komputasi kuantum dan pembelajaran mesin kuantum. Metode ini berfokus pada perhitungan gradien secara analitik, yang memungkinkan pembaruan parameter yang lebih efisien dalam pelatihan model kuantum. Dengan menghitung gradien secara analitik, AQGD dapat mengurangi jumlah pengukuran yang diperlukan pada komputer kuantum. Ini penting karena pengukuran dalam sistem kuantum sering kali memerlukan banyak sumber daya dan dapat memperkenalkan *noise*.

Nakanishi-Fujii-Todo (NFT) (Nakanishi, 2020) adalah algoritma optimasi yang digunakan dalam konteks pembelajaran mesin kuantum, khususnya dalam pelatihan model-model kuantum seperti *Quantum Neural Networks*. Algoritma ini dirancang untuk meningkatkan efisiensi dan akurasi dalam proses pembaruan parameter model kuantum. NFT merupakan metode yang mengadopsi pendekatan *gradient descent*, di mana algoritma ini melakukan pembaruan parameter berdasarkan informasi gradien dari fungsi objektif. Dengan cara ini, NFT berusaha meminimalkan kesalahan model dengan efisien. NFT dirancang khusus untuk diintegrasikan dengan *Quantum Neural Networks*, memberikan pendekatan yang lebih baik dalam pelatihan model-model kuantum yang kompleks.

3. METODE PENELITIAN

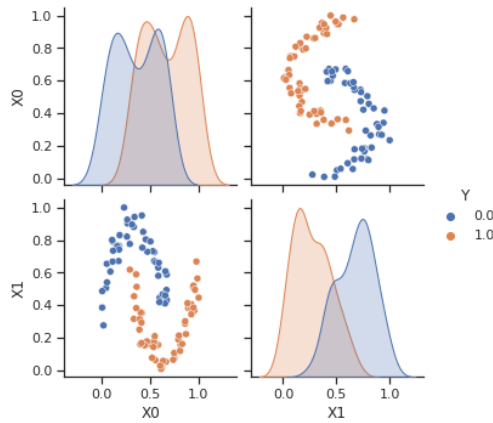
Seluruh proses pada penelitian ini menggunakan Qiskit pada IBM Quantum Lab dengan *Qiskit backend statevector simulator* serta dibantu dengan perangkat keras Dell inspiron 3881 dengan RAM memory 16 GB (DDR4 SDRAM), Processor Intel(R) Core(TM) i7-10700F CPU @ 2.90 GHz (8 Core) dan Sistem Operasi Windows 10 Home Single Language 64 Bit. yang terdiri dari beberapa tahapan berikut:



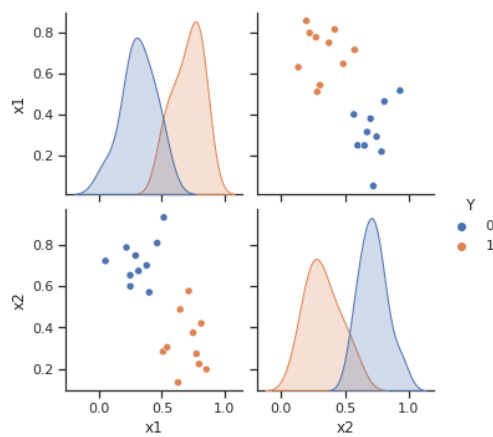
Gambar 3.1. Tahapan penelitian (Alam, 2022)

3.1 Dataset

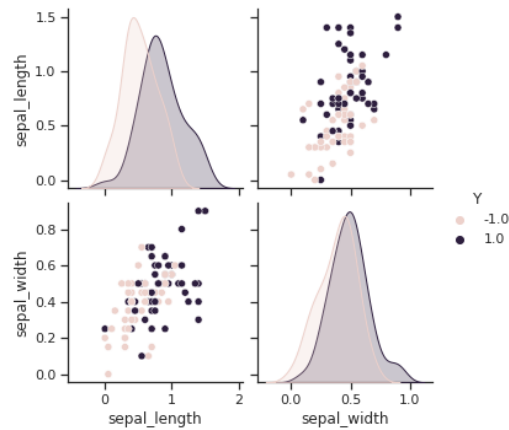
Dataset yang digunakan bersumber dari UCI *Machine learning repository* yang terdiri dari 3 dataset yaitu, moon, gaussian dan iris. Dimana setiap dataset akan dibagi menjadi dua bagian yaitu untuk proses pelatihan (70%) dan proses pengujian (30%)



Gambar 3.2. Distribusi dataset moon kedalam dua kelas 0 dan 1



Gambar 3.3. Distribusi dataset gaussian kedalam dua kelas 0 dan 1



Gambar 3.4. Distribusi dataset gaussian kedalam dua kelas -1 dan 1

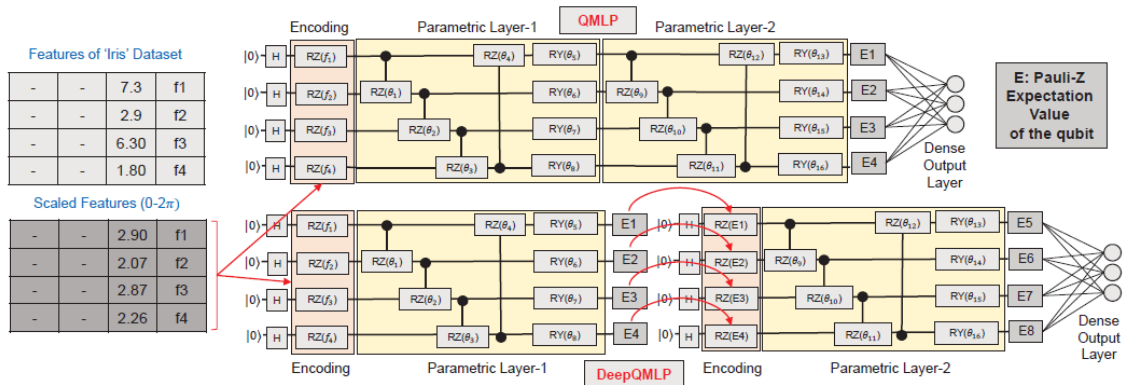
kemudian setiap dataset tersebut ditransformasikan kedalam *qubit* dengan metode *encoding ZZFeatureMap* (Pointing, 2024).

3.2 Arsitektur

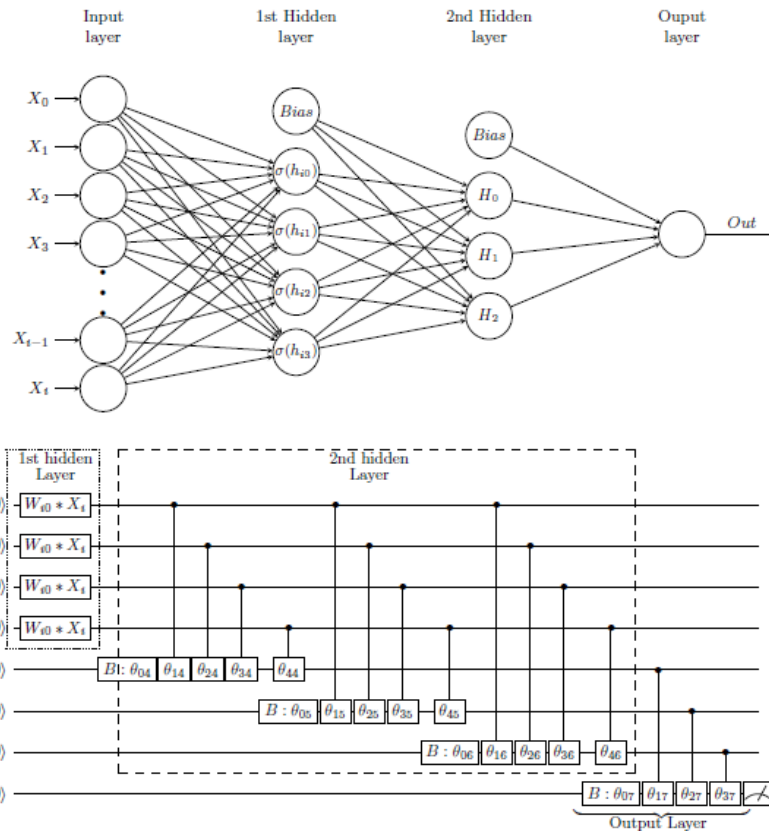
Ada beberapa jenis arsitektur jaringan saraf dengan *quantum circuit* dari penelitian terdahulu sebagai eksperimen dalam penelitian ini.



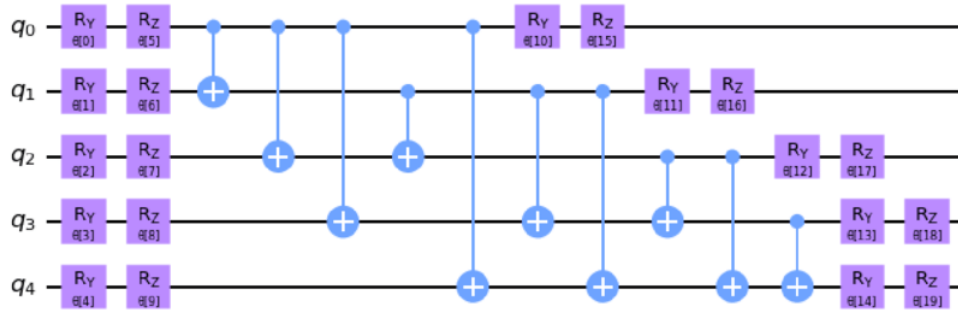
Gambar 3.5. Hybrid Quantum Neural Network (H-QNN) (Chen, 2023)



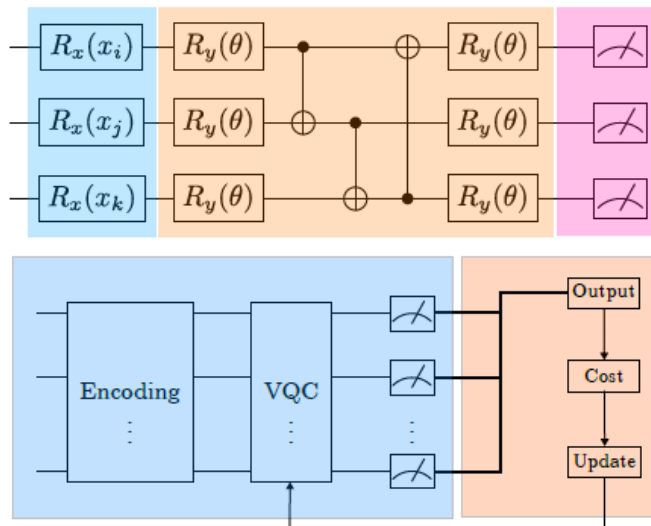
Gambar 3.6. Deep QMLP (Alam, 2022)



Gambar 3.7. Coherent Feed Forward – QNN (Singh, 2024)



Gambar 3.8. EfficientSU2 (Liu, 2024)



Gambar 3.9. Variational quantum circuits (VQC) (Smaldone, 2024)

3.3 Optimasi

Tabel 3.1. Konfigurasi hyperparameter dari setiap metode optimasi

Metode	Hyperparameter
ADAM	Iterasi : 100, 1.000 dan 10.000
	Learning rate : 0.01, 0.001 dan 0.0001
AQGD	<i>default</i>
NFT	<i>default</i>

4. HASIL PENELITIAN DAN PEMBAHASAN

Hasil dari penelitian ini terdiri dari nilai *cost*, akurasi pelatihan dan akurasi validasi ditampilkan pada beberapa tabel berikut:

Tabel 4.1. Hasil dari dataset moon dengan optimasi ADAM

arsitektur		iterasi								
		100			1.000			10.000		
		learning rate			learning rate			learning rate		
		0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
H-QNN	Cost	0.84	0.76	0.73	0.67	0.56	0.53	0.57	0.38	0.23
	Acc train	0.53	0.66	0.69	0.67	0.72	0.85	0.87	0.92	0.95
	Acc validation	0.55	0.69	0.73	0.62	0.73	0.88	0.89	0.97	0.97
Deep QML	Cost	0.63	0.55	0.47	0.58	0.45	0.27	0.21	0.13	0.08
	Acc train	0.65	0.73	0.77	0.77	0.79	0.85	0.88	0.92	0.97
	Acc validation	0.67	0.71	0.78	0.78	0.8	0.86	0.83	0.94	0.97
Coherent Feed Forward – QNN	Cost	0.63	0.62	0.53	0.35	0.31	0.12	0.05	0.03	0.03
	Acc train	0.64	0.62	0.71	0.76	0.78	0.86	0.87	0.89	0.93
	Acc validation	0.62	0.62	0.71	0.75	0.79	0.88	0.88	0.93	0.94
EfficientSU2	Cost	0.55	0.53	0.4	0.33	0.3	0.24	0.16	0.13	0.11
	Acc train	0.74	0.82	0.83	0.93	0.95	0.95	0.96	0.97	0.97
	Acc validation	0.76	0.88	0.89	0.94	0.95	0.95	0.97	0.97	0.97
VQC	Cost	0.57	0.55	0.52	0.55	0.52	0.5	0.33	0.32	0.32
	Acc train	0.83	0.93	0.95	0.93	0.93	0.94	0.91	0.94	0.94
	Acc validation	0.89	0.94	0.95	0.93	0.94	0.95	0.95	0.95	0.95

Untuk hasil pada tahap ini diperoleh dengan nilai *cost* dan akurasi terbaik pada EfficientSU2 dengan iterasi 10.000.

Tabel 4.2. Hasil dari dataset gaussian dengan optimasi ADAM

arsitektur		iterasi								
		100			1000			10000		
		learning rate			learning rate			learning rate		
		0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
H-QNN	Cost	0.73	0.65	0.62	0.56	0.45	0.42	0.46	0.27	0.12
	Acc train	0.54	0.67	0.7	0.68	0.73	0.86	0.88	0.93	0.96
	Acc validation	0.56	0.7	0.74	0.63	0.74	0.89	0.9	0.98	0.98
Deep QML	Cost	0.52	0.44	0.36	0.47	0.34	0.16	0.1	0.02	0.03
	Acc train	0.66	0.74	0.78	0.78	0.8	0.86	0.89	0.93	0.98
	Acc validation	0.68	0.72	0.79	0.79	0.81	0.87	0.84	0.95	0.98
Coherent Feed Forward – QNN	Cost	0.52	0.51	0.42	0.24	0.2	0.01	0.06	0.08	0.08
	Acc train	0.75	0.73	0.82	0.87	0.89	0.97	0.98	0.98	0.98
	Acc validation	0.73	0.73	0.82	0.86	0.9	0.99	0.99	0.99	0.99
EfficientSU2	Cost	0.44	0.42	0.29	0.22	0.19	0.13	0.05	0.02	0.02
	Acc train	0.85	0.93	0.94	0.96	0.96	0.96	0.96	0.96	0.96
	Acc validation	0.87	0.93	0.93	0.96	0.96	0.96	0.96	0.96	0.96
VQC	Cost	0.46	0.44	0.41	0.44	0.41	0.39	0.22	0.21	0.21
	Acc train	0.72	0.82	0.84	0.82	0.82	0.83	0.8	0.83	0.83
	Acc validation	0.78	0.83	0.84	0.82	0.83	0.84	0.84	0.84	0.84

Untuk hasil pada tahap ini diperoleh dengan nilai *cost* dan akurasi terbaik pada EfficientSU2 dengan iterasi 10.000.

Tabel 4.3. Hasil dari dataset iris dengan optimasi ADAM

arsitektur		iterasi								
		100			1000			10000		
		learning rate			learning rate			learning rate		
		0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
H-QNN	Cost	0.69	0.61	0.58	0.52	0.41	0.38	0.42	0.23	0.08
	Acc train	0.68	0.81	0.84	0.82	0.87	0.88	0.88	0.88	0.88
	Acc validation	0.7	0.84	0.88	0.77	0.88	0.88	0.88	0.88	0.88
Deep QML	Cost	0.48	0.4	0.32	0.43	0.3	0.12	0.06	0.02	0.07
	Acc train	0.8	0.88	0.92	0.92	0.94	0.95	0.95	0.95	0.95
	Acc validation	0.82	0.86	0.93	0.93	0.95	0.95	0.95	0.95	0.95
Coherent Feed Forward – QNN	Cost	0.48	0.47	0.38	0.2	0.16	0.03	0.1	0.12	0.12
	Acc train	0.79	0.77	0.86	0.91	0.93	0.93	0.93	0.93	0.93
	Acc validation	0.77	0.77	0.86	0.9	0.94	0.94	0.94	0.94	0.94
EfficientSU2	Cost	0.4	0.38	0.25	0.18	0.15	0.09	0.01	0.02	0.04
	Acc train	0.89	0.97	0.98	0.98	0.98	0.98	0.98	0.98	0.98
	Acc validation	0.91	0.93	0.93	0.93	0.93	0.93	0.93	0.93	0.93
VQC	Cost	0.42	0.4	0.37	0.4	0.37	0.35	0.18	0.17	0.17
	Acc train	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98
	Acc validation	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97

Untuk hasil pada tahap ini diperoleh dengan nilai cost dan akurasi terbaik pada EfficientSU2 dengan iterasi 10.000.

Tabel 4.4. Hasil dari dataset moon dengan optimasi AQGD

arsitektur			
H-QNN	Cost	0.29	
	Acc train	0.87	
	Acc validation	0.89	
Deep QML	Cost	0.34	
	Acc train	0.88	
	Acc validation	0.83	
Coherent Feed Forward – QNN	Cost	0.43	
	Acc train	0.87	
	Acc validation	0.88	
EfficientSU2	Cost	0.07	
	Acc train	0.92	
	Acc validation	0.91	
VQC	Cost	0.05	
	Acc train	0.91	
	Acc validation	0.89	

Untuk hasil pada tahap ini diperoleh dengan nilai cost dan akurasi terbaik pada EfficientSU2 VQC.

Tabel 4.5. Hasil dari dataset gaussian dengan optimasi AQGD

arsitektur			
H-QNN	Cost	0.28	
	Acc train	0.87	
	Acc validation	0.89	
Deep QML	Cost	0.33	
	Acc train	0.88	
	Acc validation	0.83	
Coherent Feed Forward – QNN	Cost	0.42	
	Acc train	0.87	
	Acc validation	0.88	
EfficientSU2	Cost	0.07	
	Acc train	0.89	

VQC	Acc validation	0.9
	Cost	0.05
	Acc train	0.9
	Acc validation	0.89

Untuk hasil pada tahap ini diperoleh dengan nilai cost dan akurasi terbaik pada EfficientSU2 VQC.

Tabel 4.6. Hasil dari dataset iris dengan optimasi AQGD

arsitektur

H-QNN	Cost	0.3
	Acc train	0.87
	Acc validation	0.89
Deep QML	Cost	0.32
	Acc train	0.88
	Acc validation	0.83
Coherent Feed Forward – QNN	Cost	0.44
	Acc train	0.87
	Acc validation	0.88
EfficientSU2	Cost	0.05
	Acc train	0.88
	Acc validation	0.88
VQC	Cost	0.05
	Acc train	0.89
	Acc validation	0.89

Untuk hasil pada tahap ini diperoleh dengan nilai cost dan akurasi terbaik pada EfficientSU2 VQC.

Tabel 4.7. Hasil dari dataset moon dengan optimasi NFT

arsitektur

H-QNN	Cost	0.4
	Acc train	0.89
	Acc validation	0.92
Deep QML	Cost	0.34
	Acc train	0.87
	Acc validation	0.83
Coherent Feed Forward – QNN	Cost	0.43
	Acc train	0.87
	Acc validation	0.88
EfficientSU2	Cost	0.05
	Acc train	0.89
	Acc validation	0.87
VQC	Cost	0.05
	Acc train	0.91
	Acc validation	0.89

Untuk hasil pada tahap ini diperoleh dengan nilai cost dan akurasi terbaik pada EfficientSU2 VQC.

Tabel 4.8. Hasil dari dataset gaussian dengan optimasi NFT

H-QNN	Cost	0.4
	Acc train	0.88
	Acc validation	0.9
Deep QML	Cost	0.32
	Acc train	0.88
	Acc validation	0.88
Coherent Feed Forward – QNN	Cost	0.38
	Acc train	0.9
	Acc validation	0.91
EfficientSU2	Cost	0.06
	Acc train	0.91
	Acc validation	0.9
VQC	Cost	0.08
	Acc train	0.9
	Acc validation	0.9

Untuk hasil pada tahap ini diperoleh dengan nilai cost dan akurasi terbaik pada EfficientSU2 VQC.

Tabel 4.9. Hasil dari dataset iris dengan optimasi NFT

H-QNN	Cost	0.37
	Acc train	0.88
	Acc validation	0.89
Deep QML	Cost	0.35
	Acc train	0.87
	Acc validation	0.85
Coherent Feed Forward – QNN	Cost	0.41
	Acc train	0.86
	Acc validation	0.87
EfficientSU2	Cost	0.08
	Acc train	0.87
	Acc validation	0.88
VQC	Cost	0.05
	Acc train	0.89
	Acc validation	0.89

Untuk hasil pada tahap ini diperoleh dengan nilai cost dan akurasi terbaik pada EfficientSU2 VQC.

Dari data yang disajikan, terlihat bahwa berbagai arsitektur jaringan memiliki kinerja yang berbeda tergantung pada pengaturan *learning rate* dan jumlah iterasi. Pada H-QNN, meskipun menunjukkan peningkatan akurasi dalam data pelatihan dan validasi, masih memiliki nilai biaya (*cost*) yang relatif tinggi pada *learning rate* yang lebih besar. Sementara itu, arsitektur Deep QML menunjukkan penurunan yang signifikan pada biaya dan peningkatan akurasi saat *learning rate* berkurang, menandakan bahwa model ini lebih sensitif terhadap pengaturan *learning rate*. Secara umum, biaya berkurang seiring dengan meningkatnya jumlah iterasi, yang mengindikasikan bahwa model dapat belajar lebih efektif dari data yang diberikan.

Di sisi lain, arsitektur EfficientSU2 dan Coherent Feed Forward – QNN menunjukkan kinerja yang konsisten dengan biaya yang lebih rendah dan akurasi yang meningkat di semua

pengaturan. EfficientSU2, khususnya, menonjol dengan biaya yang paling rendah dan akurasi validasi tertinggi pada semua level *learning rate* yang diuji, menunjukkan kemampuannya untuk generalisasi dengan baik terhadap data yang tidak terlihat. VQC juga menunjukkan hasil yang baik, tetapi dengan fluktuasi yang lebih besar dalam nilai biaya dan akurasi, yang mungkin menunjukkan bahwa model ini lebih mudah terpengaruh oleh variasi dalam data atau pengaturan *hyperparameter*. Secara keseluruhan, pemilihan arsitektur dan pengaturan *hyperparameter* yang tepat sangat penting untuk mencapai kinerja optimal dalam pengembangan model berbasis jaringan saraf.

5. KESIMPULAN DAN SARAN

Dari hasil eksperimen menunjukkan bahwa beragam arsitektur Quantum Neural Networks (QNN), seperti Hybrid Quantum Neural Network (H-QNN) dan EfficientSU2, menawarkan keunggulan signifikan dalam efisiensi dan akurasi dalam pengolahan data. Dari Data eksperimen menunjukkan bahwa pengaturan *learning rate* dan jumlah iterasi sangat mempengaruhi kinerja model seperti DQNN dan Coherent Feed Forward QNN yang menunjukkan hasil konsisten dalam pengurangan biaya dan peningkatan akurasi. Selain itu, integrasi teknik optimasi seperti ADAM, AQGD dan NFT meningkatkan efektivitas pelatihan, memungkinkan model untuk menghadapi kompleksitas data dengan lebih baik. Dengan demikian, inovasi ini membuka peluang besar bagi kemajuan teknologi di berbagai bidang, menggarisbawahi potensi besar QNN dalam merevolusi metode pemrosesan informasi.

6. DAFTAR PUSTAKA

- Abbas, A., Sutter, D., Zoufal, C., Lucchi, A., Figalli, A., & Woerner, S. (2021). The power of quantum neural networks. *Nature Computational Science*, 1(6), 403-409.
- Alam, M., & Ghosh, S. (2022). Deepqmlp: A scalable quantum-classical hybrid deep neural network architecture for classification. In 2022 35th International Conference on VLSI Design and 2022 21st International Conference on Embedded Systems (VLSID) (pp. 275-280). IEEE.
- Arthur, D. (2022). A hybrid quantum-classical neural network architecture for binary classification. arXiv preprint arXiv:2201.01820.
- Chen, H. Y., Chang, Y. J., Liao, S. W., & Chang, C. R. (2023). Hybrid Quantum Neural Network in High-dimensional Data Classification.
- Liu, C. Y., Kuo, E. J., Lin, C. H. A., Chen, S., Young, J. G., Chang, Y. J., & Hsieh, M. H. (2024). Training classical neural networks by quantum machine learning. arXiv preprint arXiv:2402.16465.
- MacCormack, I., Delaney, C., Galda, A., Aggarwal, N., & Narang, P. (2022). Branching quantum convolutional neural networks. *Physical Review Research*, 4(1), 013117.
- Mahmud, J., Mashtura, R., Fattah, S. A., & Saquib, M. (2024). Quantum convolutional neural networks with interaction layers for classification of classical data. *Quantum Machine Intelligence*, 6(1), 11.
- Mingard, C., Pointing, J., London, C., Nam, Y., & Louis, A. A. (2024). Exploiting the equivalence between quantum neural networks and perceptrons. arXiv preprint arXiv:2407.04371.
- Mitarai, K., Negoro, M., Kitagawa, M., & Fujii, K. (2018). Quantum circuit learning. *Physical Review A*, 98(3), 032309.
- Nakanishi, K. M., Fujii, K., & Todo, S. (2020). Sequential minimal optimization for quantum-classical hybrid algorithms. *Physical Review Research*, 2(4), 043158.
- Pointing, J. (2024). Do Quantum Neural Networks have Simplicity Bias?. arXiv preprint arXiv:2407.03266.
- Schuld, M. (2021). Supervised quantum machine learning models are kernel methods. arXiv preprint arXiv:2101.11020.

- Schuld, M., Bergholm, V., Gogolin, C., Izaac, J., & Killoran, N. (2019). Evaluating analytic gradients on quantum hardware. *Physical Review A*, 99(3), 032331.
- Singh, U., Goldberg, A. Z., & Heshami, K. (2024). Coherent Feed Forward Quantum Neural Network. arXiv preprint arXiv:2402.00653.
- Smaldone, A. M., Shee, Y., Kyro, G. W., Xu, C., Vu, N. P., Dutta, R., ... & Batista, V. S. (2024). Quantum Machine Learning in Drug Discovery: Applications in Academia and Pharmaceutical Industries. arXiv preprint arXiv:2409.15645.