Quantum Machine Learning: Unraveling a New Paradigm in Computational Intelligence

Sachin Khurana[1] and manisha nene[1]

[1]Defence Institute of Advanced Technology

March 14, 2024

Abstract

Quantum Machine Learning (QML) is an advanced discipline that emerges from the combined power of machine learning and quantum computing that has the ability to address intricate challenges in several domains. The domain of quantum machine learning investigates the development and execution of quantum software with the potential to facilitate machine learning at a much superior pace compared to traditional computers. This research delves into the fundamental principles of quantum mechanics and their crucial role in quantum computing, emphasizing the potential of various quantum algorithms to surpass classical algorithms in specific computational tasks, and then methodically navigates through the quantum machine learning algorithms, offering profound insights into their application potential in revolutionizing data analysis and complex problem-solving methodologies, including their importance in the Language Learning Models (LLM) and Language Analysis Models (LAM). The study also provides insights into the various quantum platforms, encompassing both hardware and software aspects for the implementation of QML algorithms, and also explores the challenges prevalent in QML, with a particular focus on the limitations imposed by existing quantum hardware and the intricate nuances of data processing within quantum frameworks. This study contributes by presenting the basis for future research work related to the development of algorithms in the field of quantum machine learning and anticipating the far-reaching impact of QML across diverse scientific and technological domains.

Received <day> <Month>, <year>; | Revised <day> <Month>, <year>; | Accepted <day> <Month>, <year>

DOI: xxx/xxxx

REVIEW PAPER

Quantum Machine Learning: Unraveling a New Paradigm in Computational Intelligence

Sachin Khurana*[1] | Manisha J Nene[2]

[1]SoCE&MS, Defence Institute of Advanced Technology, Pune, India
[2]SoCE&MS, Defence Institute of Advanced Technology, Pune, India

Correspondence
*Sachin Khurana, DIAT, Pune. Email: sachinkhurana005@gmail.com

Present Address
DIAT, Pune

Abstract

Quantum Machine Learning (QML) is an advanced discipline that emerges from the combined power of machine learning and quantum computing that has the ability to address intricate challenges in several domains. The domain of quantum machine learning investigates the development and execution of quantum software with the potential to facilitate machine learning at a much superior pace compared to traditional computers. This research delves into the fundamental principles of quantum mechanics and their crucial role in quantum computing, emphasizing the potential of various quantum algorithms to surpass classical algorithms in specific computational tasks, and then methodically navigates through the quantum machine learning algorithms, offering profound insights into their application potential in revolutionizing data analysis and complex problem-solving methodologies, including their importance in the Language Learning Models (LLM) and Language Analysis Models (LAM). The study also provides insights into the various quantum platforms, encompassing both hardware and software aspects for the implementation of QML algorithms, and also explores the challenges prevalent in QML, with a particular focus on the limitations imposed by existing quantum hardware and the intricate nuances of data processing within quantum frameworks. This study contributes by presenting the basis for future research work related to the development of algorithms in the field of quantum machine learning and anticipating the far-reaching impact of QML across diverse scientific and technological domains.

KEYWORDS:
machine learning, quantum gates, quantum circuits, quantum algorithms, quantum machine learning

1 | INTRODUCTION

The advent of quantum computing, alongside the advancements in classical machine learning (ML), brought forth a new age in computational capabilities, fundamentally redefining the landscape of data processing and analysis. Central to this paradigm shift is the concept of qubits, which contrast the binary bits of traditional computing. Qubits are capable of existing in multiple states concurrently utilising the principle of superposition. Coupled with quantum entanglement, where qubits in a quantum state can be correlated with each other regardless of distance, quantum computers possess the ability to process and analyze data at a scale and speed unattainable by classical computers. Concurrently, ML has seen significant progress, developing algorithms

[0]**Abbreviations:** QML, Quantum Machine Learning; QSVM, Quantum Support Vector Machine; QNN, Quantum Neural Network; QPCA, Quantum Principal Component Analysis

that facilitate computers to learn from data, thereby making precise predictions and decisions. These advancements in ML have led to its integration into various complex systems, from natural language processing to sophisticated image recognition technologies. However, as the complexity and volume of data grow, classical ML encounters limitations in processing speed and computational efficiency, creating challenges in handling large-scale, intricate datasets.

1.1 | Classical Machine Learning and it's Challenges

Classical Machine Learning (ML) has achieved remarkable success across various domains, however, it faces significant challenges, particularly in terms of scalability and efficiency, as the volume and complexity of data continue to grow exponentially. One of the most pressing challenges is the scalability of classical ML algorithms. As datasets become increasingly large, the computational resources required to process and analyze this data grow substantially. This escalation in data volume can lead to longer training times, especially for more complex models such as deep neural networks. In scenarios where real-time data analysis and decision-making are critical, such as in financial trading algorithms or emergency response systems, the inability of classical ML models to swiftly process large volumes of data can be a significant limitation. While classical ML techniques are powerful, they are often constrained by the capabilities of the hardware on which they run, the processing power of conventional CPUs, and even GPUs, can be a bottleneck, particularly for algorithms that require intensive computation and therefore, high-dimensional data, common in areas like genomics or complex sensor networks and LLMs, can significantly slow down the performance of ML models, making it difficult to extract meaningful insights in a timely manner.

Moreover, the inherent limitations of classical computing architectures are evident when dealing with complex, non-linear problems that are prevalent in many real-world applications. Classical ML algorithms sometimes struggle to capture the intricate relationships within such data, leading to suboptimal performance. Lastly, the interpretability of ML models, particularly those employing complex algorithms like deep learning, is a growing concern because of the 'black box' nature of these models, which can be problematic in domains where transparency and accountability are crucial. While classical ML has revolutionized many aspects of technology and industry, it faces significant challenges related to scalability, efficiency, data dependency, and interpretability. The potential for QML lies in its ability to solve complex problems and in handling vast amounts of data like learning language patterns more efficiently, and potentially solve NLP problems that are currently beyond the reach of classical ML models[1].

1.2 | Quantum Machine Learning

Quantum Machine Learning (QML) emerges as an interdisciplinary field that leverages the distinct characteristics of quantum mechanics to provide notable improvements in computational performance. Quantum algorithms possess the capability to conduct computations exponentially faster than classical algorithms, thereby significantly enhancing the efficacy of data processing and pattern recognition, pivotal in ML. QML excels at managing complex, high-dimensional data, which conventional algorithms often struggle with and this capability is particularly beneficial in analyzing quantum data, such as that derived from quantum simulations in physics and chemistry. QML's application across various industries illustrates its potential to transform traditional methodologies for problem-solving. Quantum computing has facilitated progress in drug development and molecular simulation, notably in the simulation of caffeine molecules, through collaborations between quantum computing enterprises and pharmaceutical corporations[2,3]. Financial institutions like JPMorgan Chase[5] use quantum computing for portfolio optimization and risk analysis, resulting in enhanced financial models and strategies[4,6]. In the energy sector, quantum computing is used to improve grid operations and energy demand forecasting, focusing on renewable energy sources[7]. Aerospace companies like Airbus explore quantum computing for aircraft design optimization and flight path efficiencies[8]. Furthermore, the field of cryptography and cyber security is undergoing a shift with the development of robust cryptographic algorithms to counter the potential challenges posed by future quantum capabilities[9,10,11,12]. QML can significantly accelerate language processing tasks, such as semantic analysis, language translation, and sentiment analysis[13,14]. The application of QML in Language Learning and Analysis Models can open up new possibilities in NLP as in LLMs, quantum-enhanced algorithms could enable faster and more accurate training of models on large linguistic datasets, leading to more sophisticated understanding and generation of human language. Similarly, LAMs can benefit from QML by achieving more efficient linguistic pattern recognition and analysis, which is crucial for applications like voice recognition software, automated translation services, and AI-driven content creation. The quantum approach could also enable deeper insights into linguistic subtleties by handling complex, high-dimensional language data more effectively. The wide range of applications of QML serves to showcase its adaptability and highlight its capability to effectively tackle intricate difficulties in diverse areas. The ongoing advancement of quantum technology is anticipated to result in an expansion in both the range and quantity of applications, hence emphasising the profound impact that QML can have.

This study presents a comprehensive overview of Quantum Machine Learning (QML), aiming to explore the integration of quantum principles into ML algorithms for tackling complex computational problems. The study has been organised as follows: Section II covers foundational concepts; Section III discusses significant quantum algorithms; Section IV provides an overview of QML algorithms; Section V examines various quantum platforms, including hardware and software; and Section VI delves into the challenges and future prospects in the advancement of QML.

2 | FOUNDATIONAL CONCEPTS

2.1 | Computing Paradigms

Classical computing, the cornerstone of modern digital technology, is founded on the principles of classical mechanics and operates on binary logic. At the heart of classical computing are bits, the fundamental units of information, which can have two distinct states: 0 or 1 and these bits can be manipulated through logical operations governed by Boolean algebra, forming the basis for all classical computational processes [15]. The manipulation and interaction of these bits follow deterministic rules, leading to predictable and well-defined outcomes for any given input. Classical computers are structured around the architecture proposed by John von Neumann, which includes a CPU, memory, input, and output units [16]. The CPU, executing instructions from software programs, performs arithmetic and logical operations, while memory units store data and instructions. This architecture enables classical computers to execute a wide range of tasks, from basic arithmetic to complex simulations, making them versatile and powerful tools for problem-solving across various domains. However, the efficiency of classical computing is limited by the physical constraints of the hardware, such as processing speed and memory capacity, and by the linear nature of its operations. This means that as problems increase in complexity or scale, the resources required for their computation grow linearly or even exponentially, presenting significant challenges in fields like cryptography, large-scale data analysis, and simulation of complex systems.

Quantum computing is a transformative change in the field of computational theory and practice, which is based on the principles of quantum mechanics. Quantum computing, in contrast to classical computing, operates using quantum bits, or qubits, instead of traditional bits. Superposition allows a qubit to carry out several computations simultaneously, while entanglement, a distinctively quantum phenomenon in which the state of one qubit is influenced by the state of another, enables complex correlations that are unattainable in classical computing. Quantum computing employs quantum gates to manipulate qubits, which are the quantum counterparts of classical logical gates, possess the potential to execute intricate transformations owing to the unique characteristics of qubits [16]. The quantum algorithm, exemplified as Shor's algorithm for factorization and Grover's algorithm for database search, utilises these features to tackle particular computational problems with significantly higher efficiency compared to traditional algorithms [17]. Quantum computing has significant potential for resolving specific issues that are now unsolvable for classical computers, particularly those related to extensive optimisation, complicated simulations, and cryptography. Nevertheless, the field is currently in its early phases, facing difficulties like as error correction, maintaining qubit coherence, and creating quantum systems that can be scaled up. Despite these challenges, the theoretical and emerging practical capabilities of quantum computing suggest a profound impact on various fields, including material science, pharmaceuticals, cryptography, and complex system modeling. Table 1 presents the comparative analysis between the classical and the quantum paradigm of computing.

2.2 | Quantum Mechanics

Quantum mechanics serves as the foundational theory for quantum computing, which has a significant impact on the field of Quantum Machine Learning. It provides unique concepts, including superposition and entanglement, which enhance computational efficiency and enable the processing of complex and large datasets [18]. These concepts enable quantum algorithms to possibly surpass classical analogues in particular tasks, including data processing and optimisation. Figure 1 represents two entangled qubits and their associated superposition values [19].

TABLE 1 Comparative Analysis

Aspect	Classical Computing	Quantum Computing
Basic Unit	Bits, which exist in one of two states: 0 or 1	Qubits, which can exist in multiple states simultaneously due to superposition
Data Processing	Deterministic and sequential processing	Parallel processing due to superposition and entanglement
Operations	Logical operations based on Boolean algebra	Operations using quantum gates that can create complex correlations
Problem Solving	Effective for a wide range of tasks but limited by linear or exponential scaling with data size	Can solve certain problems exponentially faster, particularly those that are intractable for classical computers
Limitations	Scalability issues with large-scale data or complex computations	Practical challenges include error rates, qubit coherence, and current hardware limitations
Examples	Sorting algorithms (QuickSort, MergeSort), Binary Search, algorithms in cryptography	Shor's algorithm (integer factorization), Grover's algorithm (database searching)

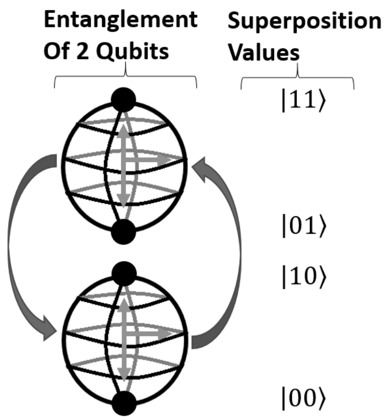

FIGURE 1 A Simplistic Representation of Two Entangled Qubits

2.2.1 | Superposition

The fundamental concept of quantum mechanics known as the principle of superposition [18] enables a quantum system, such as a qubit, to concurrently occupy multiple states. In classical computing, a bit is considered the fundamental unit of information [18], capable of assuming one of two states, namely 0 or 1. However, in the field of quantum computing, it is observed that a qubit has the ability to exist in a superposition state, simultaneously encompassing multiple states, and state of a qubit in superposition [18,19] is denoted as:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \tag{1}$$

In above equation, $|\psi\rangle$ symbolises the quantum state of the qubit, while $|0\rangle$ and $|1\rangle$ denote the basis states. The coefficients α and β are complex numbers referred to as probability amplitudes are responsible for determining the probabilities associated with the measurement outcomes of the qubit in each state. Specifically, the probability of measuring the qubit [16] in the state represented by $|0\rangle$ is given by $|\alpha|^2$, while the probability of measuring it in the state represented by $|1\rangle$ is given by $|\beta|^2$. The total of these probabilities must always be equal to 1, hence guaranteeing the normalisation of the quantum state. The utilisation of the superposition principle enables quantum computers to concurrently process a significant quantity of information. For example, although a conventional computer consisting of n bits has the capability to exist in any of the 2^n potential configurations at a specific moment, a quantum computer consisting of n qubits has the ability to exist in a superposition of all 2^n states concurrently. The aforementioned characteristic is the key factor that endows quantum computing with the capacity for speedup in particular tasks involving computation when compared to classical computers. When a measurement is performed on a qubit that is in a superposition state, the qubit undergoes a collapse, resulting in its state being determined as one of the basis states. Superposition holds significance beyond its theoretical implications since it implies practical implications in the domain of quantum computing, facilitating intricate calculations, especially in domains like cryptography [39], optimisation, and quantum system modelling.

2.2.2 | Entanglement

Entanglement is an important principle in the field of quantum mechanics where qubits establish a correlation such that the state of one particle cannot be characterised in isolation from the states of the other particles, irrespective of their spatial separation [17,18]. This interconnectedness means that a change in the state of one entangled particle is instantaneously reflected in the state of the other. Quantum entanglement is mathematically characterised by the utilisation of the quantum state of the system. Let us contemplate a simplistic system comprising of two qubits in which, when qubits become entangled, it is not feasible to depict their collective state as a multiplication of their individual states. Conversely, it is imperative to collectively describe the state of the complete system. For example, the Bell state [24] is widely recognised as one of the most famous entangled states and is represented as:

$$|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \tag{2}$$

If the first qubit is measured and determined to be in the state $|0\rangle$, the second qubit will also be measured and found to be in the state $|0\rangle$, and the same applies to the state $|1\rangle$. The coefficients $\frac{1}{\sqrt{2}}$ ensure that the total state is normalised, meaning the probabilities of the outcomes sum to one [17,23].

Entanglement is not just limited to pairs of qubits but it can also occur in systems of multiple qubits, leading to even more complex entangled states. These states are central to many quantum computing algorithms and applications, such as quantum teleportation, quantum cryptography, and superdense coding [6,12] . The principle of entanglement challenges our classical intuitions about the separability and independence of objects in the universe. It allows for the creation of correlations between qubits that can be exploited to execute calculations that are infeasible for classical computers. It also allows quantum computers to effectively explore numerous computing paths concurrently, hence offering a significant advantage in addressing particular types of complex problems. Quantum entanglement enables the creation of intricate quantum states in which information is not limited to each qubit but rather distributed throughout the entire system [25]. This characteristic empowers quantum computers to execute tasks with a level of efficiency and speed that surpasses the capabilities of conventional computers.

2.2.3 | Interference

Quantum interference constitutes a key principle within the field of quantum mechanics; it emerges due to the wave-like characteristics shown by quantum particles, such as electrons or photons, and has a key role in realising the computational capabilities of quantum algorithms. In classical physics, it is observed that waves possess the ability to engage in either constructive or destructive interference. When two waves are in phase, they combine, resulting in constructive interference and a wave with increased amplitude. In contrast, when waves are not in sync, they nullify each other, causing destructive interference, which leads to a wave with decreased amplitude or complete cancellation. In quantum computing, the states of qubits are described by probability amplitudes, which, like waves, can interfere with each other, and when a quantum system evolves, these amplitudes can add together or cancel each other out, a phenomenon that is crucial for the functioning of quantum algorithms. For example, consider a simple quantum system with a superposition of states [24] represented as:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \tag{3}$$

If this system undergoes an operation that alters the phases of these stages, the system may transition to a new state:

$$|\psi'\rangle = \alpha'|0\rangle + \beta'|1\rangle \tag{4}$$

in which α' and β' represent the new amplitudes[25]. The likelihood of measuring a specific state, such as $|0\rangle$, is based upon interference of all the paths leading to that state. If the paths leading to $|0\rangle$ interfere constructively, the probability of measuring $|0\rangle$ increases and if they interfere destructively, the probability decreases.

Quantum algorithms are designed to exploit the principle of interference by carefully orchestrating the phases of the probability amplitudes, thereby enhancing the amplitudes of 'correct' answers through constructive interference and diminishing the amplitudes of 'incorrect' answers via destructive interference. It helps quantum algorithms, such as Shor's algorithm for factoring large numbers or Grover's algorithm for exploring unsorted databases, achieve increased efficiency compared to classical algorithms[26,27]. By controlling the interference of probability amplitudes, quantum algorithms exhibit superior efficiency compared to traditional algorithms for specific problems[25]. This principle is a key reason why quantum computing holds such promise for revolutionising computing, offering new ways to process information and solve complex problems.

2.2.4 | Qubits

Qubits represents a substantial shift from the traditional bits used in classical computing[15], distinguished by its binary nature, encompassing two distinct states: 0 or 1. A qubit distinguishes itself from a classical bit by virtue of being capable of existing in a superposition state, which facilitates a qubit to execute several computations concurrently, a capability that is unattainable with classical bits[16]. The superposition principle is the fundamental basis for the much enhanced computational capabilities of quantum computing in comparison to classical computing, particularly in relation to specific problem domains[24,25]. One notable characteristic of qubits is their capacity to become entangled with additional qubits, which means the state of one qubit in an entangled pair cannot be characterised independently of the state of the other. Qubits can be implemented using diverse physical systems, including electron spins, photon polarisations, and atomic energy levels. The implementation of qubits involves sophisticated technology to isolate, control, and measure these quantum systems, often requiring conditions like ultra-low temperatures or vacuum environments to maintain their quantum properties. Thus, qubits serve as a core constituent of quantum computers, employing the fundamental concepts of quantum mechanics to store and manipulate information in manners that completely differ from those of classical computing. The distinctive characteristics exhibited by qubits present opportunities for more efficient solutions to complex problems compared to classical computers, hence signifying a noteworthy advancement in the area of computation.

2.2.5 | Quantum Gates

Quantum gates play a crucial role in the functioning of quantum computers, similar to how classical logic gates are essential in conventional computers. However, unlike classical gates, which manipulate binary bits, a quantum gate executes a quantum operation on a set of qubits, changing their state according to the rules of quantum mechanics[17]. These operations are reversible and can be represented mathematically by unitary matrices. Here are some common quantum gates and their corresponding matrices[24,25]:

1. Pauli-X Gate: Pauli-X flips the state of a qubit and is represented by the matrix:

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

2. Pauli-Y Gate: Pauli-Y gate introduces both bit flip as well as phase shift and is represented by the matrix:

$$Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$$

3. Pauli-Z Gate: Pauli-Z gate changes the phase of the qubit without changing its amplitude and is represented by the matrix:

$$Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

4. Hadamard Gate (H Gate): Hadamard Gate creates a superposition of $|0\rangle$ and $|1\rangle$ from a definite state.It is crucial for creating superpositions in quantum algorithms and is represented by:

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

5. Controlled NOT Gate (CNOT Gate): A two-qubit gate in which the second qubit (target) undergoes a bit flip if and only if the first qubit (control) is in the state $|1\rangle$. It is essential for creating entanglement between qubits and is represented by:

$$\text{CNOT} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

These gates can be combined in a quantum circuit to perform complex operations and algorithms. The choice and arrangement of gates determine the computation performed by the quantum computer. The ability to manipulate qubits with these gates, exploiting superposition and entanglement, is what gives quantum computers their unique computational capabilities.

2.2.6 | Quantum Circuits

Quantum circuits serve as the fundamental framework for performing quantum computations, bearing similarities to classical circuits in digital computing yet functioning in a unique way. The fundamental constituents of a quantum circuit consist of quantum gates, which are systematically employed to execute a precise operation that modifies the state of one or more qubits. These operations are reversible and hence guarantee the preservation of the overall probability within the quantum system. The arrangement and configuration of these gates dictate the characteristics and intricacy of the computation executed by a quantum circuit. Gates like the Hadamard gate have the capability to induce superposition in qubits, enabling them to concurrently embody several states[24]. The CNOT gate is applied for entangling of qubits, leading to the establishment of intricate interdependencies within their states. The phenomenon of entanglement plays a crucial role by facilitating the establishment of correlations among qubits that are unattainable in classical computing systems. Quantum parallelism represents a noteworthy characteristic of quantum circuits, which allows a quantum circuit to simultaneously process several possibilities. The presence of parallelism is the key factor that endows quantum computing with the capacity for significant computational prowess, especially when addressing certain problem domains. In addition, quantum circuits employ the principle of interference to amalgamate probabilities originating from distinct computational pathways. Through meticulous gate sequence design, quantum algorithms have the potential to increase the probability of obtaining accurate outcomes while simultaneously reducing the possibility of wrong results.

Quantum circuits serve as the fundamental framework for a range of quantum algorithms, including Shor's algorithm and Grover's algorithm[26,27]. The design and implementation of these circuits play a crucial role, as they have a direct influence on the efficiency and efficacy of the quantum algorithm. Nevertheless, quantum circuits encounter substantial obstacles, namely in the domains of error correction and noise mitigation. Furthermore, the issue of scalability poses a substantial barrier to the progress of quantum computing. The development of quantum circuits on a vast scale, capable of executing intricate computations while preserving their quantum properties, is an essential challenge that requires attention.

3 | QUANTUM ALGORITHMS

Quantum algorithms operate within the realm of quantum mechanics, utilizing the unique properties of quantum bits or qubits. Quantum algorithms leverage quantum gates to perform operations on qubits, and these operations can be vastly more complex than their classical counterparts due to the properties of superposition and entanglement[17]. This enables quantum algorithms to potentially solve problems exponentially faster than classical algorithms. Notable examples of quantum algorithms include Grover's algorithm for database searching, which offers a quadratic speedup over classical search algorithms[26]. The efficiency of quantum algorithms is not only based on the speed of computation but also on their ability to handle problems intractable for classical algorithms, particularly in the fields of cryptography, optimization, and simulation of quantum systems[2,11]. However, the practical implementation of quantum algorithms faces significant challenges, including error rates in quantum computing, qubit coherence, and the physical limitations of current quantum computing hardware[16]. Despite these challenges, quantum algorithms represent a groundbreaking shift in computational capabilities, promising to revolutionize areas where classical algorithms fall short.

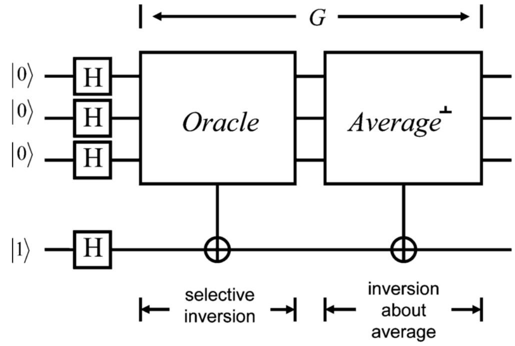

FIGURE 2 Block diagram of Grover's search algorithm

3.1 | Grover's Algorithm

Grover's algorithm, a seminal quantum algorithm, is renowned for its ability to search unstructured databases that exhibit a quadratic speedup compared to conventional methods[26]. This quantum algorithm is particularly notable for demonstrating a clear advantage over classical approaches in specific problem types, particularly in unstructured search problems. The essence of Grover's algorithm lies in its ability to locate a particular item inside an unordered database containing N items in $O(\sqrt{N})$ time[26] which is a significant enhancement compared to the $O(N)$ time complexity of the classical algorithms. The algorithm begins by initialising n qubits to the state $|0\rangle$, followed by the application of a Hadamard transform, resulting in a superposition of all conceivable states[25], represented as:

$$|\psi\rangle = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} |x\rangle \tag{5}$$

where $N = 2^n$. Figure 2 illustrates the block diagram of Grover's search method, which encompasses stepwise operations.

The key component of Grover's algorithm[26] is the oracle U_f, a quantum operation that changes the polarity of the amplitude of the desired state $|w\rangle$. For the marked item w, the function $f(w) = 1$, and for all other states, $f(x) = 0$. After applying the oracle, the Grover diffusion operator D is applied to amplify the amplitude of the marked state by inverting the amplitude of each state about the average amplitude. The operator is mathematically defined as:

$$D = 2|\psi\rangle\langle\psi| - I \tag{6}$$

The algorithm iterates the combination of the oracle and the diffusion operator approximately $\sqrt{N}$ times. This iterative process is key to maximising the likelihood of measuring the marked state, thereby efficiently collapsing the state of superposition to the desired item upon measurement[25,26]. Grover's algorithm has been extended and adapted through various variants, each tailored to optimize specific aspects of quantum search or address unique computational challenges. These variants demonstrate the adaptability and applicability of Grover's foundational principles to a wide range of quantum computing scenarios, enhancing the efficiency and scope of quantum search algorithms in solving complex problems across diverse fields[40,41]. In QML, Grover's algorithm has significant implications as it can be adapted to address optimisation challenges, which are frequently encountered in machine learning, by efficiently searching through potential solutions to find the optimal or near-optimal ones[31]. This capability is also beneficial in feature selection, where the algorithm can search through combinations of features in large datasets to identify the most relevant set for a given model. Additionally, Grover's algorithm can enhance clustering algorithms in machine learning, particularly in handling large datasets, by expediting the search through data points to identify clusters[33]. Moreover, within the context of data mining tasks related to machine learning, the algorithm's efficacy in efficiently identifying patterns or specific data points inside extensive databases holds significant value. The potential applications of Grover's algorithm in QML encompass optimisation, feature selection, and data mining, highlighting its significance in quantum-enhanced machine learning.

3.2 | Quantum Fourier Transform

The Quantum Fourier Transform (QFT) is an essential technique in the realm of quantum computing, it has the ability to perform Fourier transforms exponentially faster than classical algorithms, rendering it an essential element in many quantum algorithms and providing the computational advantages in various QML algorithms[34]. Mathematically, the QFT operates on a quantum state composed of n qubits, transforming a state $|x\rangle$ into $|y\rangle$ through a specific transformation. This transformation is expressed as:

$$|x\rangle \rightarrow \frac{1}{\sqrt{2^n}} \sum_{y=0}^{2^n-1} e^{2\pi i x y/2^n} |y\rangle \tag{7}$$

here, x and y denote the qubit's state in integer form, and $e^{2\pi i x y/2^n}$ introduces complex exponential phase relationships[18]. The QFT's implementation involves a series of Hadamard gates, which generate superpositions, and controlled phase rotation gates, which establish the necessary phase relationships between qubits[19].

The application of QFT in QML is very significant, especially in the process of extracting features and compressing data, especially in tasks that include pattern recognition and classification[35]. The utilisation of the Fourier basis in the process of transforming data enables the examination of frequency components inside intricate datasets, which is a crucial step in numerous machine learning applications[36]. QFT's role in quantum phase estimation, which is foundational for algorithms like the quantum linear systems algorithm, indirectly influences QML, especially in solving linear equations. Furthermore, the applications of QFT encompass the realm of quantum signal processing, when it fulfils a vital function in the processing of time-series data or images, which are frequently encountered in machine learning tasks. The Quantum Fourier Transform (QFT) possesses significant use in managing superpositions and entanglements, rendering it vital for the advancement of quantum algorithms in clustering and classification, potentially enhancing the capabilities of classical machine learning techniques.

3.3 | HHL algorithm

The HHL algorithm represents a significant development in quantum computing, particularly to solve linear systems of equations. It exhibits remarkable characteristics in terms of its ability to solve specific linear equations at an exponential rate in comparison to the most advanced conventional algorithms, a capability that holds significant implications for the domain of QML[32]. The primary objective of the HHL algorithm is to address linear equations of the type:

$$Ax = b \tag{8}$$

where A represents a Hermitian matrix and b denotes a vector with known values. The aim of this task is to determine the vector x. The initial phase requires the encoding of the vector b into a quantum state $|b\rangle$. This stage often entails the utilisation of quantum algorithms to transform classical data into quantum states. Subsequently, the Quantum Phase Estimation (QPE) technique[43] is utilised to approximate the eigenvalues of the matrix A, effectively encoding these eigenvalues into quantum states by means of the unitary operation e^{iAt}. One crucial stage of the HHL method entails the utilisation of controlled rotation gates, which leverage the eigenvalue data to execute rotations contingent upon these specific eigenvalues. Following that, the QPE[43] procedure is 'uncomputed' to disentangle the eigenvalue register from the rest of the system, an essential step in order to isolate the solution. The ultimate phase involves the process of measurement, which has a high likelihood of producing an estimation of the resultant vector x which is represented in a quantum state.

The utilisation of the HHL algorithm in quantum machine learning exhibits a wide range of applications that have significant implications. Within the domain of data fitting and prediction, specifically in the linear regression models, the HHL algorithm demonstrates the capability to efficiently calculate model weights, hence offering notable advantages for datasets of considerable magnitude[32]. Furthermore, the approach has the potential to be modified for the purpose of principal component analysis (PCA) and other methods of dimensionality reduction. This adaptation allows for faster processing of data with a high number of dimensions. Furthermore, inside machine learning techniques such as SVMs, the HHL algorithm demonstrates the capability to effectively calculate inner products within feature spaces that possess high dimensions[32]. The HHL algorithm has the potential to provide exponential improvements in the efficiency of solving linear problems when compared to traditional approaches. As quantum computing continues to progress, the HHL algorithm is anticipated to assume a pivotal position in facilitating the development of more efficient and potent machine learning models, opening new frontiers in data analysis and computational science.

3.4 | Quantum Approximate Optimisation Algorithm (QAOA)

QAOA is a significant algorithm particularly for solving combinatorial optimisation issues as it stands out for its capacity to address these issues more efficiently than conventional algorithms and is especially suited for near-term quantum computers,

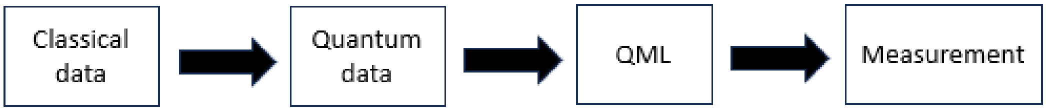

FIGURE 3 Basic structure of QML

which are characterised by limited qubits and higher error rates[19]. The algorithm operates through a series of quantum gates parameterized by classical variables, and these parameters are optimised using classical techniques. The optimisation problem is encoded in a problem Hamiltonian H_P, constructed so that its ground state represents the optimal solution. Alongside, a mixing Hamiltonian H_M, usually a sum of Pauli-X operators applied to each qubit, is defined to facilitate transitions between states. The QAOA circuit alternates between applying $e^{-i\beta_k H_M}$ and $e^{-i\gamma_k H_P}$, where β_k and γ_k are variational parameters[14]. These parameters are crucial as they are optimised classically to minimise the expectation value of H_P in the quantum state prepared by the circuit. The classical optimisation of β and γ seeks the values that minimise the expectation value of H_P, representing the most efficient solution to the initial problem.

QAOA's applications are diverse and impactful in QML, like training of neural networks or optimising feature sets, can be presented in the form of optimisation problems where QAOA can provide more efficient solutions. This is particularly relevant in scenarios involving complex optimization landscapes or large parameter spaces. QAOA also finds application in quantum-assisted feature selection, to determine the most relevant features from large datasets, a critical step in many machine learning models. Furthermore, the algorithm can be adapted for clustering and classification tasks, where the objective is to optimally group or classify data points. Additionally, QAOA presents a promising approach to tackle NP-hard problems in data analysis, such as graph partitioning or network optimisation, which are often encountered in advanced data analysis tasks. As quantum technology continues to evolve, QAOA is anticipated to have a crucial impact in solving complex problems that are currently challenging classical computing techniques, thereby opening new frontiers in optimisation and data analysis[44].

4 | QUANTUM MACHINE LEARNING

QML complements classical ML techniques with the concepts of quantum computing to enhance the efficiency of data-driven algorithms and establish novel computational frameworks[33]. By utilising concepts of quantum parallelism and entanglement[37,38], QML endeavours to enhance the efficiency of computational processes such as optimisation, classification and clustering, which are fundamental to classical ML. There are four distinct methodologies[16] depending upon the type of data and type of computing used, as shown in Table 2. The term "CC" denotes the utilisation of classical data being processed in a classical manner, while the term "QC" pertains to the utilisation of classical data being processed on a quantum computer. Our primary emphasis will be on the CQ and QQ methodologies, which involve the utilisation of quantum computing for classical and quantum data, respectively. The CQ methodology involves utilising quantum processor for processing of classical data, while the QQ approach involves the utilisation of quantum computing for quantum data. Figure 3 depicts how input classical data is converted to quantum data before application of QML algorithm[22].

TABLE 2 Methodologies based on Computing and Data Types

Methodology	Computing Type	Data Type	Problem Domains
CC (Classical-Classical)	Classical Computing	Classical Data	Data Analysis, Web Browsing
QC (Quantum-Classical)	Quantum Computing	Classical Data	Traffic Flow Optimization
CQ (Classical-Quantum)	Classical Computing	Quantum Data	Quantum State Characterization
QQ (Quantum-Quantum)	Quantum Computing	Quantum Data	Secure Communications, Quantum Simulations

The quantum speedup in quantum machine learning (QML) offers a substantial potential advantage compared to classical machine learning, deriving from the underlying principles of quantum computing[21]. Table 3 represents the comparative analysis for machine learning algorithms in both classical and quantum computing environments by examining their respective computational complexities. However, the amount of speedup achieved depends on the type of specific quantum algorithm used in the machine learning tasks.

TABLE 3 Comparative Analysis

Algorithm	**Classical Algorithm Complexity**	**Quantum Algorithm Complexity**	**Remarks**
SVM	O(poly(NM))	O(log(NM))	QSVM typically utilises quantum kernels
NN	O(N^k)	O(log(N))	QNNs can exploit quantum parallelism and may use variational circuits.
K-means	O(NMI)	$O\sqrt{(NMI)}$	Quantum K-means may use Grover's search for distance calculations and amplitude amplification.
PCA	O(N^3)	O(log(N))	QPCA leverages quantum phase estimation and quantum singular value decomposition.

where N represents the number of training data points, M represents the number of features, k is the number of layers in neural network, and I represents the number of iterations in K-means.This table offers a comprehensive comparison at a macro level[32,33] as for each machine learning algorithm, the level of complexity may vary depending on the specific implementation and the arrangement of the data, and the exact quantum computing architecture used.

4.1 | Supervised Algorithms

Supervised learning, a fundamental paradigm in machine learning in which a model is trained on a labeled dataset. In the quantum realm, supervised learning algorithms are being developed to exploit the unique features of quantum computing in order to potentially enhance the efficacy and performance of learning tasks like classification and regression[32].

4.1.1 | Quantum Support Vector Machines (QSVM)

QSVM represent a quantum-enhanced adaptation of classical SVM, a widely used machine learning model for mainly classification tasks. QSVM leverages the principles of quantum computing, potentially offering significant speedups, particularly in processing large and complex datasets. The main objective of conventional SVM is to identify a hyperplane in an N-dimensional space that effectively separates the data points into various classes. This hyperplane as shown in figure 4 is given by the equation:

$$w \cdot x - b = 0 \tag{9}$$

where w represents the vector normal to the hyperplane and b represents the bias[28]. The optimisation problem in SVMs involves maximising the margin between this hyperplane and the closest data points from each class, which are referred to as support vectors. SVMs utilise the kernel method to handle data that is not linearly separable by expressing it into a higher-dimensional space where these data points become linearly separable[48,49,50].

QSVM adapts this framework to quantum computing, aiming to exploit quantum parallelism and entanglement for more efficient computation. The process begins with quantum state preparation using encoding schemes, which maps classical vectors into the amplitudes of a quantum state. The most resource-intensive aspect of SVM is the the calculation of the kernel matrix, where QSVM brings a significant advantage. This step is performed using a quantum circuit that estimates the inner products

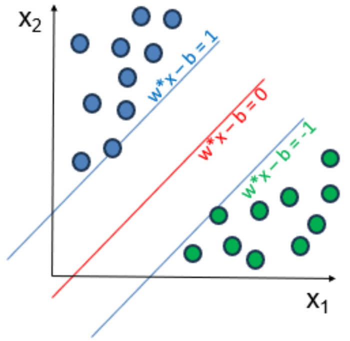

FIGURE 4 SVM

between data points in the transformed feature space, thus efficiently computing its kernel.In QSVM[28,32], concept of a quantum kernel K, a measure of similarity between $|\psi\rangle$ and $|\phi\rangle$ is given by:

$$K(|\psi\rangle, |\phi\rangle) = |\langle\psi|\phi\rangle|^2 \tag{10}$$

Furthermore, the optimisation problem central to SVMs, which involves maximizing the margin or minimizing the norm of w, can potentially be addressed using quantum optimization algorithms like QAOA[44]. This quantum approach to optimisation could further enhance the efficiency of QSVM. The applications of QSVM in machine learning are promising, particularly in handling large datasets. Due to the quantum advantage in kernel computation, QSVM can potentially manage larger datasets more efficiently than classical SVMs. This efficiency is also beneficial in recognizing complex patterns in data, especially where classical SVMs struggle due to computationally intensive kernel calculations.

Pseudocode for QSVM Algorithm:

1. **Data Preprocessing.** Input: Classical data set
   - Normalize and preprocess the data.
   - Split the dataset into training and testing sets.
2. **Quantum State Preparation.**
   - Map the preprocessed classical data vectors into quantum states.
   - This mapping encodes classical data into the amplitudes of a quantum state.
3. **Quantum Kernel Estimation.**
   - For each pair of data points in the transformed feature space:
     a. Prepare the quantum states corresponding to these points.
     b. Apply a quantum circuit that performs an inner product estimation between these states.
     c. Measure the output to estimate the inner product.
   - Construct the kernel matrix K using the estimated inner products.
4. **Training the QSVM.**
   - Utilize the quantum kernel matrix in the SVM training process.
   - Solve the SVM optimization problem (maximizing margin/minimizing norm of w).
5. **Making Predictions.** For a new input data point:
   - Map the input data point to a quantum state.

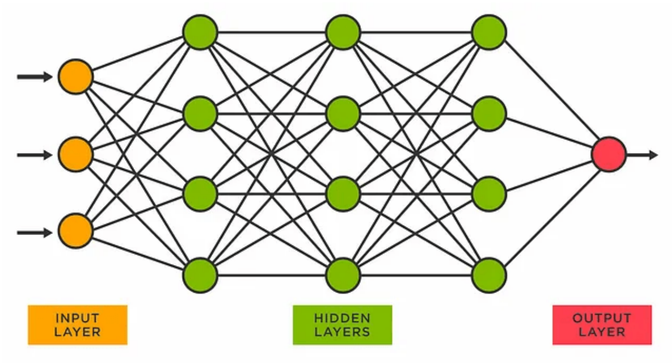

FIGURE 5 Classical Neural Network

- Use the trained QSVM model to predict the class of the new data point.

6. **Post-processing.**
   - Convert the quantum measurement outcomes to classical format.
   - Interpret the results to obtain the final classification or prediction.

Additionally, QSVM may offer advantages in scenarios with a high number of features, as quantum states can compactly represent high-dimensional data[28]. As quantum computing technology continues to evolve, QSVMs hold the promise of significantly enhancing the capabilities of machine learning models in handling large-scale and complex data analysis tasks.

4.1.2 | Quantum Neural Networks (QNN)

QNNs are designed to use the principles of quantum mechanics to improve the efficiency and capabilities of conventional neural network models. In classical neural networks[53], information processing occurs through layers of neurons, where each neuron typically applies a weighted summation on the inputs, followed by the application of an activation function as shown in figure 5. While QNNs are still in the developmental phase and encompass various approaches, they commonly utilise quantum circuits to carry out computations similar to those performed by traditional neural networks.

In QNNs, this paradigm is shifted to the quantum domain, where quantum circuits and qubits replace classical neurons and bits. The mathematical framework of QNNs begins with the encoding of classical input data into quantum states. Quantum gates then function analogously to neurons in classical networks, applying transformations to the qubits. Classical neural networks employ a series of linear transformations for approximating an arbitrary function $f(x)$. In QNNs, the conventional linear transformations and activation functions are substituted by unitary operations denoted as U and quantum measurements, respectively[30]. Consequently, the function $f(x)$ is transformed into a quantum operation in which $|\psi_{\text{in}}\rangle$ and $|\psi_{\text{out}}\rangle$ represent the input and output quantum states respectively[51,52] as shown in figure 6:

$$|\psi_{\text{out}}\rangle = U_f |\psi_{\text{in}}\rangle \tag{11}$$

QNNs utilises quantum phenomena like entanglement and superposition that enable complex correlations and parallel processing capabilities, potentially offering significant advantages over traditional neural networks. The outcome of a QNN is obtained by performing a measurement on the quantum state. This measurement causes the quantum state to collapse into classical information, which can subsequently be interpreted as the output of the network[37].

Pseudocode for Quantum Neural Network (QNN) Algorithm:

1. **Initialize:**
   - Define the structure of the QNN (number of qubits, quantum gates, etc.)
2. **Data Encoding:**
   - Input: Classical data set.

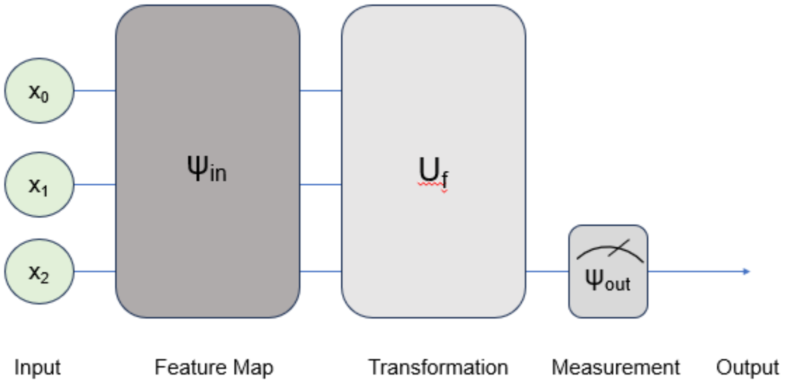

FIGURE 6 Quantum Neural Network

- Encode the classical input data into quantum states.
- For each data point x:
  a. Map x to a quantum state using an appropriate encoding scheme.

3. **Quantum Processing:**
   - Apply a series of quantum gates (unitary operations U) on the qubits.
   - These gates function similarly to neurons in classical neural networks.
   - Each gate manipulates the quantum state, analogous to processing in neural network layers.
4. **Entanglement and Superposition:**
   - Utilize quantum phenomena like entanglement and superposition to establish complex correlations between qubits.
   - This step enhances the parallel processing capabilities of the QNN.
5. **Measurement:**
   - After applying quantum gates, perform a measurement on the final quantum state.
   - This measurement collapses the quantum state to classical information.
6. **Interpret Output:**
   - Convert the measurement results from quantum to classical form.
   - Interpret these results as the output of the QNN.
   - In case of a classification task, determine the class label based on the output.
7. **Training the QNN (if applicable):**
   - Define a cost function to evaluate the performance of the QNN.
   - Adjust the parameters of the quantum gates (analogous to weights in a classical NN) to minimize the cost function.
   - Use quantum optimization algorithms if applicable for efficient training.
8. **Repeat Steps 2-7** for each iteration or data point.
9. **End Process:**
   - Conclude after processing all data points or reaching convergence in training.

QNNs possess significant potential for a wide range of applications as quantum parallelism can be efficiently utilised in cases where there is a need for efficient solutions in pattern recognition and classification tasks[42]. QNNs offer potential benefits in the realm of feature extraction for datasets with high dimensions, since they leverage quantum superposition to enhance the processing of intricate data structures with improved efficiency. Moreover, in the case of data that possesses inherent quantum properties, such as information derived from quantum simulations or quantum sensors, quantum neural networks have the potential to offer a seamless and effective computational framework. Also, QNNs can be specifically designed for various applications, for example, NLP tasks, offering potentially faster and more efficient language processing capabilities for language learning and analysis models which require better performance and faster processing[31].

4.2 | Unsupervised Algorithms

Unsupervised machine learning focuses on identifying patterns, structures, or attributes in data that does not include explicit labels. Quantum computing presents innovative methodologies for unsupervised learning problems, including clustering and dimensionality reduction[20], by utilising the fundamental properties of quantum mechanics[56]. The objective of these quantum techniques is to augment or expedite the process of learning in comparison to conventional techniques.

4.2.1 | Quantum K-means Clustering

Quantum K-means clustering is a quantum counterpart of the conventional K-means clustering method that incorporates principles of superposition and quantum parallelism to boost the efficiency of data clustering, particularly in the context of large datasets[32]. By employing conventional K-means, one can divide a set of n data points into k clusters, every data point is assigned to the cluster whose mean is closest to it. The process entails minimising the within-cluster sum of squares (WCSS), which is determined by summing the squared distances between each data point x_i from the centroid of the cluster to which it is assigned μ_j. Mathematically, WCSS can be expressed as:

$$WCSS = \sum_{i=1}^{n} \min_{j}(\|x_i - \mu_j\|^2) \tag{12}$$

The process starts by selecting k initial centroids, followed by assigning each data point to its nearest centroid, typically utilising the Euclidean distance[55]:

$$d(x_i, \mu_j) = \sqrt{\sum_{d}(x_{id} - \mu_{jd})^2} \tag{13}$$

After assignment, centroids are updated by determining the mean of every data point in each cluster, and these steps are repeated until convergence[46,47]. The key quantum enhancement is the ability to perform distance calculations in parallel due to quantum superposition. The quantum algorithm then calculates distances using quantum subroutines, which can compute the Euclidean distance in a superposed state, allowing for parallel computation of distances[54,56]:

$$U_{\text{distance}}|x_i\rangle|\mu_j\rangle|0\rangle = |x_i\rangle|\mu_j\rangle|d(x_i, \mu_j)\rangle \tag{14}$$

However, updating centroids in quantum K-means is challenging due to quantum measurement constraints. After quantum computations, the results are measured and fed back into the classical part of the algorithm for further iterations.
Pseudocode for Quantum K-means Clustering Algorithm:

1. **Initialize:**
   - Choose the number of clusters, k.
   - Select k initial centroids randomly from the dataset.
2. **Quantum State Preparation:**
   - Encode the dataset into quantum states.
   - Prepare quantum states for centroids.
3. **Quantum Distance Calculation:**

- For each data point and each centroid:
  a. Apply a quantum subroutine $U\_distance$ to compute the Euclidean distance in superposition.
  b. Calculate $U\_distance$.
- Perform this step in parallel for all data points and centroids due to quantum superposition.

4. **Cluster Assignment:**
   - Measure the quantum states to obtain the distances.
   - Assign each data point to the cluster of the nearest centroid based on the measured distances.
5. **Update Centroids:**
   - Classically compute the new centroids as the mean of the data points in each cluster.
   - Encode the new centroids into quantum states for the next iteration.
6. **Check for Convergence:**
   - If the centroids do not change significantly, or a maximum number of iterations is reached, end the algorithm.
   - Otherwise, return to step 3.
7. **Final Output:**
   - Output the clustering result with data points assigned to their respective clusters.

Quantum K-means clustering aims to harness quantum computing for more efficient clustering, especially beneficial for large-scale and high-dimensional datasets. While promising, its practical implementation is an active research area, focusing on challenges like quantum state preparation, centroid updating, and integrating quantum-classical computing steps. Quantum K-Means can be utilized for various applications, including grouping similar linguistic patterns or documents, which can be beneficial for Language Learning and Analysis Models.

4.2.2 | Quantum Principal Component Analysis (QPCA)

The Quantum Principal Component Analysis (QPCA) algorithm is a quantum computing methodology that improves the efficiency of the classical Principal Component Analysis (PCA) method, commonly employed in data analytics. The utilisation of quantum mechanics concepts in the extraction of principal components has the potential to provide exponential speedup, making it particularly attractive for large datasets. Principal components are the crucial dimensions that capture the highest amount of variance in the data. Nevertheless, the primary objective of conventional PCA is to reduce the dimensionality of a given dataset while maintaining a significant amount of variation[57]. The procedure entails the identification of the primary components, which refer to the directions within the dataset that optimise variance. Classical Principal Component Analysis (PCA) performs diagonalisation[29] of the covariance matrix C for a specific data collection x:

$$C = \frac{1}{n} \sum_{i=1}^{n} \vec{x}_i \vec{x}_i^T \tag{15}$$

here, n represents the total number of data points. The principal components are subsequently identified as the eigenvectors of matrix C, wherein their associated eigenvalues signify the extent to which each component captures variance. However, QPCA operates directly on the quantum states that represent the data. QPCA aims to identify the principal components of the density matrix ρ, given by:

$$\rho = |\Psi\rangle\langle\Psi| \tag{16}$$

where $|\Psi\rangle$ denotes the encoding of the data set[12].

The initial step for the implementation of QPCA involves the preparation of the quantum state which involves encoding of data matrix X into a quantum state[32]. The density matrix ρ, which is the quantum counterpart of C represents the quantum state of the covariance matrix C. Next stage involves the use of QPE technique to estimate the eigenvalues of the density matrix ρ. Nevertheless, the process of recovering primary components from quantum measurements, which involves reconstructing

classical information, can be intricate and challenging. The mathematical complexity of QPCA is attributed to its utilisation principles of superposition and entanglement to execute computations that are highly demanding in terms of resources on conventional computers, particularly when dealing with high-dimensional datasets.
Pseudocode for Quantum Principal Component Analysis (QPCA) Algorithm:

1. **Data Encoding:**
    - Input: Classical data set X.
    - Encode the classical data matrix X into a quantum state.
    - Prepare the density matrix.
2. **Quantum State Preparation:**
    - Initialize quantum registers to represent the quantum state corresponding to the density matrix.
3. **Quantum Phase Estimation (QPE):**
    - Apply QPE to the quantum state representing the density matrix.
    - QPE is used to estimate the eigenvalues of the density matrix, which are crucial for identifying the principal components.
4. **Measurement:**
    - Perform quantum measurements to obtain the eigenvalues and eigenvectors of the density matrix.
    - These measurements yield information about the principal components.
5. **Post-Processing:**
    - Convert the measured quantum information into classical form.
    - Identify the principal components based on the measured eigenvalues and eigenvectors.
    - In classical PCA, these components correspond to the directions that maximize variance in the data.
6. **Dimensionality Reduction:**
    - Use the identified principal components to perform dimensionality reduction on the dataset.
    - This step involves projecting the original data onto the space spanned by the principal components.
7. **Output:**
    - Output the reduced-dimensionality data and the principal components.

The prospective applications of QPCA in the domains of machine learning and data analysis hold considerable importance, especially in scenarios characterised by large datasets or computationally intensive principal component extraction requirements.

5 | QUANTUM PLATFORMS

Currently, quantum computers exhibit a significant level of specialisation and are predominantly limited to use within research institutions, therefore making them ill-suited for applications beyond the confines of this particular domain. As a result, the usage of quantum machines for public utilisation is made easier through the use of cloud-based platforms, which allow for online access over the pre-existing internet infrastructure. Quantum computing encompasses a range of approaches to quantum hardware, each with distinct strengths and challenges that arise from technical advancements. Superconducting qubits, which are being developed by firms such as Google, IBM, and Rigetti [58], are known for their scalability and rapid gate operation times. These qubits are manufactured utilising procedures similar to those employed in standard semiconductor fabrication processes. Nevertheless, these systems face challenges in terms of coherence times, which are limited, and greater error rates and their operation requires cryogenic temperatures. Trapped ion qubits, which have been implemented by IonQ and Honeywell's Quantinuum [58,59], exhibit superior fidelity due to their extended coherence periods and reduced error rates. Moreover, their fully interconnected architecture enables each ion qubit to communicate with any other qubit. However, it is worth noting that these systems encounter challenges in terms of slower gate operations and difficulty in scaling. Quantum Dots, pursued by Intel and Toshiba, offer scalability and potential for higher operational temperatures but struggle with maintaining coherence. Microsoft's Topological Qubits, offer greater error resistance, potentially reducing extensive error correction needs. Photonic Qubits, explored by Xanadu and PsiQuantum[58], operate at room temperature and could integrate with existing optical networks, yet face challenges in error correction and photon loss. Neutral Atom Qubits, investigated by Pasqal and ColdQuanta[58,59], promise scalability and long coherence times but require precise control over neutral atoms and complex engineering for interactions. This rich quantum computing landscape, with each approach carving its path with distinct advantages and facing unique challenges, is a tapestry of technological exploration. The field may see certain technologies emerge as leaders or a hybrid approach combining multiple platforms' strengths as research progresses.

Alongwith the quantum hardware approaches their associated Quantum Software Development Kits (SDKs) and libraries are of significant importance in facilitating the accessibility, programming, and experimentation with quantum computers for academics and developers. Qiskit[60], an open-source SDK developed by IBM, is specifically designed for Python users. It offers a seamless interface with IBM's quantum computers and simulators. Qiskit has gained significant recognition in both academic and industry communities due to its intuitive tools which assist in the creation, analysis, and execution of quantum circuits. Cirq[60,61], a notable open-source Python library developed by Google, focuses on the creation, manipulation, and optimisation of quantum circuits. This library is specifically designed to complement Google's quantum computing capabilities, while also offering adaptability for utilisation with other processors. The Quantum Development Kit (QDK) developed by Microsoft encompasses the Q# programming language[61], which is proficient at articulating quantum algorithms and facilitating simulation. Furthermore, it seamlessly connects with Microsoft's array of tools and services, so offering a full solution for the development of quantum applications. The Ocean SDK, developed by D-Wave Systems[59], is specifically designed for quantum annealing and serves as a platform for solving intricate optimisation issues. It leverages D-Wave's quantum annealers, highlighting the unique capabilities of quantum computing in specialised domains. The libraries of Xanadu, including Strawberry Fields and PennyLane[61,62], have a primary focus on the fields of QML, quantum computing, and chemistry. Strawberry Fields is specifically dedicated to the advancement of photonic quantum computing, while PennyLane aims to facilitate quantum circuit optimisation across different hardware platforms. The ProjectQ framework, developed by researchers at ETH Zurich, provides a software framework to implement quantum programmes using the Python programming language. This framework is designed to be compatible with various backends, such as local simulators and actual quantum hardware[61,62].

Each software development kit (SDK) and library inside the quantum computing ecosystem fulfils a crucial role by providing distinct features and abstractions. Not only do they enhance the accessibility of quantum computing, but they also foster innovation and facilitate the implementation of actual solutions in this advanced domain. The progression of quantum technology is anticipated to result in the advancement of these tools, hence providing enhanced capabilities and facilitating the utilisation of a broader spectrum of quantum computing platforms.

6 | CHALLENGES AND FUTURE PROSPECTS

Although, the field of QML is encompassing significant potential but also encountering a range of technical challenges. The challenges cover a range of factors including restrictions in hardware capabilities, complexities in algorithm design, intricacy in data processing, requirements for software development, and significant resource demands[63]. One of the foremost challenges encountered in QML pertains to the constraints imposed by hardware. Current quantum computers face limitations in terms of coherence times, which impose restrictions on the duration for which qubits may preserve their quantum state[64]. As a result, these limitations impose constraints on the complexity and length of calculations that can be effectively performed. Furthermore, the heightened susceptibility of quantum systems to external perturbations results in increased error rates and the issue of scalability also presents an additional obstacle in the development of quantum computers[63,64]. As the number of qubits increases, the challenge of maintaining stable quantum states and effectively handling error rates becomes increasingly intricate.

The task of developing quantum algorithms that surpass their classical counterparts in performance present a formidable challenge. Numerous algorithms of this nature are currently in the nascent phases of research or continue to exist solely in theoretical form. Adding hybrid algorithms, which combine the most advantageous aspects of both computing paradigms, makes things even more complicated, especially when it comes to managing how these two types of computation interact with each other[65].

TABLE 4 Quantum Computing Platforms

Quantum Platform	Developing Company	Characteristics	Challenges	Associated SDK/Library
Superconducting Qubits	Google, IBM, Rigetti	Scalable, rapid gate operation times	Limited coherence times, cryogenic operation	Qiskit (IBM), Cirq (Google), Forest (Rigetti)
Trapped Ion Qubits	IonQ, Honeywell's Quantinuum	Superior fidelity, extended coherence, fully interconnected architecture	Slower gate operations	SDKs like Qiskit, Cirq can be used
Quantum Dots	Intel, Toshiba	Scalability, potential for higher operational temperatures	Maintaining coherence	Not commonly available
Topological Qubits	Microsoft	Stable and greater error resistance	Material and fabrication difficult, scalability issues	Q# (Microsoft QDK)
Photonic Qubits	Xanadu, PsiQuantum	Operates at room temperature, integration with optical networks	Challenges in error correction and photon loss	Strawberry Fields, PennyLane (Xanadu)
Neutral Atom Qubits	Pasqal, ColdQuanta	Scalability, long coherence times	Precise control over neutral atoms, complex engineering	Quantum Toolbox in Python (QuTiP)
Quantum Annealing	D-Wave Systems	Specialized for solving optimization problems	Only applicable to optimization problems	Ocean SDK (D-Wave)

The field of QML is faced with unique hurdles when it comes to data encoding and processing. The technique of converting classical data into quantum representation with high efficiency is not a simple task and often requires a significant allocation of resources. Moreover, the quantum paradigm of data processing exhibits fundamental distinctions from classical methodologies, hence requiring novel methodologies for obtaining significant insights from quantum computations. The domain is also confronted with a scarcity of sophisticated software tools and programming languages, hence rendering the development and testing of quantum algorithms more arduous compared to traditional computing [61,62]. Bringing together quantum computing solutions into existing classical systems introduces an additional layer of intricacy, necessitating the establishment of compatibility in terms of both hardware and software components. The resource demands associated with quantum computing are enormous, frequently requiring significant computational resources, which encompass classical computer capacity for both data preparation and postprocessing. The maintenance of quantum computers necessitates a substantial allocation of resources and involves intricate processes, including the provision of cryogenic temperatures to sustain superconducting qubits. In the domain of QML, key problems encompass the intricate nature of quantum models and the constraints imposed by the availability of training data. Quantum models have a higher level of complexity compared to conventional models, hence rendering them more challenging to comprehend and analyse. Additionally, the efficacy of QML algorithms is frequently contingent upon the accessibility and calibre of the training data, so introducing an additional facet to the obstacles encountered in this pioneering domain.

Notwithstanding these obstacles, the domain of QML is swiftly progressing, propelled by a sequence of developing patterns and prospective advancements that have the capacity to fundamentally transform the field of computation and data analysis. The primary focus of these improvements lies in the endeavour to build robust and fault-tolerant quantum computers, which are crucial for effectively addressing the defects that naturally arise while scaling quantum systems. Simultaneously, there is a notable emphasis on the development and optimisation of algorithms for quantum systems to achieve efficient solutions to intricate problems [45]. The emergence of hybrid quantum-classical systems is driven by the need to address existing hardware limits with the aim of maximising the utilisation of quantum resources while capitalising on the reliability of classical computing [65].

In conjunction with these technological breakthroughs, the integration of QML with artificial intelligence (AI) represents a significant development with the potential to revolutionise the field[66]. This integration holds the promise of augmenting AI algorithms and facilitating the creation of more robust and advanced systems. This integration explores how quantum computing can address complex AI challenges, such as neural network optimization and large-scale data analysis. Quantum simulation, namely in the domains of material science and chemistry, is a very auspicious field that holds the potential for significant advancements in drug discovery and material design[67]. Moreover, the advancement of quantum communication and networking is poised to have a substantial influence on quantum machine learning (QML), facilitating novel approaches to distributed quantum computing and machine learning frameworks. With the continuous expansion of data volumes, quantum computing presents a promising prospect for effectively handling and examining large-scale datasets. The investigation into the cross-disciplinary utilisation of Quantum Machine Learning (QML) in various domains such as climate modelling, financial modelling, and logistics is currently gaining momentum. With the advancement of quantum computing, it is expected that these areas of research will undergo development and broadening, leading to the emergence of novel computing and data analysis paradigms and ushering in a new era of technological progress[64,66].

CONCLUSION

The exploration into Quantum Machine Learning (QML) has revealed its extensive capabilities and the complex obstacles it encounters. QML, an emerging discipline situated at the confluence of quantum computing and machine learning, offers an unique methodology for data processing and analysis. It holds the potential to outperform classical computing in a multitude of complex tasks. The exploration of quantum computing concepts, classical machine learning methods, and their integration has yielded a detailed examination of the present status and future potential of QML. The fundamental principles of quantum computing, such as superposition, entanglement, and quantum circuits, provide a basis for understanding how quantum computing can improve machine learning tasks. The investigation into several quantum algorithms employed in Quantum Machine Learning (QML) has unveiled the extensive range of possible applications that quantum computing can offer within the domain of machine learning. Nevertheless, there are certain technical obstacles that currently hinder the complete realisation of Quantum Machine Learning's (QML) potential. The obstacles encompass various aspects, including constraints in quantum hardware, such as coherence time and error rates, complexities in algorithms, difficulties in data encoding and processing, and the requirement for more advanced software tools and programming languages. Notwithstanding these challenges, the prospects for QML are promising, as ongoing research consistently endeavours to expand the frontiers of its potential. Looking into the future, the progression of QML is positioned to have a profound impact on the fields of technology and research. It is anticipated that the progressive improvements in quantum hardware, error correction methodologies, and algorithmic developments would gradually surmount the existing constraints. The fusion of quantum computing and machine learning is expected to result in substantial progress in computational capabilities, enabling the formulation of innovative solutions for highly intricate and data-intensive issues. Although the current level of development is preliminary, the advancements achieved thus far and the future prospects it presents are noteworthy and encouraging. From the latest developments and ongoing research,QML is anticipated to assume a prominent role in the forthcoming era of technological progress, and it will revolutionise our problem-solving strategies and data analysis methodologies.

CONFLICT OF INTEREST

Authors have no conflict of interest relevant to this article.

References

1. M. Abbaszade, V. Salari, S. S. Mousavi, M. Zomorodi and X. Zhou, "Application of Quantum Natural Language Processing for Language Translation," in IEEE Access, vol. 9, pp. 130434-130448, 2021, doi: 10.1109/ACCESS.2021.3108768.

2. von Lilienfeld, O. A. "Quantum machine learning in chemical compound space," *Angew. Chem. Int. Ed.*, vol. 57, no. 16, pp. 4164-4169, Apr. 2018.

3. Cirac, J. I. "Quantum computing and simulation," *Nanophotonics*, vol. 10, no. 1, pp. 453–456, 2021.

4. Jones, C. "AI, Big data, Quantum Computing and Financial Exclusion: tempering enthusiasm and offering a human centric approach to policy," 2021.

5. Jpmorgan chase and qc ware evolve hedging for a quantum future. (n.d.). Retrieved 19 January 2024, from https://www.jpmorgan.com/technology/news/jpmorganchase-qcware-evolve-hedging-for-a-quantum-future

6. Mathur, P. *Machine Learning Applications Using Python: Cases Studies from Healthcare, Retail and Finance*, Apress, 2019.

7. Ajagekar, A., & You, F. (2019). Quantum computing for energy systems optimization: Challenges and opportunities. Energy, 179, 76–89. https://doi.org/10.1016/j.energy.2019.04.186

8. (N.d.). Retrieved 19 January 2024, from https://www.airbus.com/en/innovation/disruptive-concepts/quantum-technologies

9. Sharma, M. Kr, and Nene, M. J. "Two-factor Authentication Using Biometric Based Quantum Operations," *SECURITY AND PRIVACY*, vol. 3, no. 3, May 2020, p. e102.

10. Sharma, A. and Nene, M. J. "Qubit Transmission Techniques," *2021 International Conference on Intelligent Technologies (CONIT)*, Hubli, India, 2021, pp. 1-6.

11. Sharma, M. Kumar, and Nene, M. J. "Dual Factor Third-party Biometric-based Authentication Scheme Using Quantum One Time Passwords," *SECURITY AND PRIVACY*, vol. 3, no. 6, Nov. 2020, p. e129.

12. Nema, P. and Nene, M. J. "Pauli Matrix based Quantum Communication Protocol," *2020 IEEE International Conference on Advent Trends in Multidisciplinary Research and Innovation (ICATMRI)*, Buldhana, India, 2020, pp. 1-6.

13. Coecke, B., de Felice, G., Meichanetzidis, K., and Toumi, A. "Foundations for near-term quantum natural language processing," *arXiv:2012.03755*, 2020.

14. Guarasci, R., De Pietro, G., & Esposito, M. (2022). Quantum natural language processing: Challenges and opportunities. Applied Sciences, 12(11), 5651. https://doi.org/10.3390/app12115651

15. Galindo, A., & Martín-Delgado, M. A. (2002). Information and computation: Classical and quantum aspects. Reviews of Modern Physics, 74(2), 347–423. https://doi.org/10.1103/RevModPhys.74.347

16. Montiel Ross, O. H. (2020). A review of quantum-inspired metaheuristics: Going from classical computers to real quantum computers. IEEE Access, 8, 814–838. https://doi.org/10.1109/ACCESS.2019.2962155

17. Gill, S. S., Kumar, A., Singh, H., Singh, M., Kaur, K., Usman, M., & Buyya, R. (2022). Quantum computing: A taxonomy, systematic review and future directions. Software: Practice and Experience, 52(1), 66–114. https://doi.org/10.1002/spe.3039

18. Feynman, R. P. (1982). Simulating physics with computers. *International Journal of Theoretical Physics, 21*(6), 467–488. https://doi.org/10.1007/BF02650179

19. Preskill, John. “Quantum Computing in the NISQ Era and Beyond.” *Quantum*, vol. 2, Aug. 2018, p. 79. https://quantum-journal.org

20. Lloyd, S., Mohseni, M., and Rebentrost, P. “Quantum algorithms for supervised and unsupervised machine learning,” 2013, pp. 1–11.

21. Rønnow, T. F., et al. “Defining and detecting quantum speedup,” *Science*, vol. 345, no. 6195, 2014, pp. 420–424.

22. Schuld, M., Sweke, R., & Meyer, J. J. (2021). Effect of data encoding on the expressive power of variational quantum-machine-learning models. Physical Review A, 103(3), 032430. https://doi.org/10.1103/PhysRevA.103.032430

23. Dirac, P. A. M. "A new notation for quantum mechanics," *Math. Proc. Cambridge Philos. Soc.*, vol. 35, no. 3, pp. 416-418, 1939.

24. Vedral, V., and Plenio, M. B. "Basics of quantum computation," *Prog. Quantum Electron.*, vol. 22, no. 1, pp. 1-39, 1998.

25. Steane, A. "Quantum computing," *Reports Prog. Phys.*, vol. 61, no. 2, pp. 117-173, Feb. 1998.

26. Grover, L. K. "Quantum mechanics helps in searching for a needle in a haystack," *Phys. Rev. Lett.*, vol. 79, no. 2, pp. 325-328, 1997.

27. Shor, P. W. "Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer," *SIAM Journal on Computing*, vol. 26, no. 5, Oct. 1997, pp. 1484-509. https://arxiv.org/abs/quant-ph/9508027

28. Rebentrost, P., et al. "Quantum Support Vector Machine for Big Data Classification," *Physical Review Letters*, vol. 113, no. 13, Sept. 2014, p. 130503.

29. Lloyd, S., et al. "Quantum Principal Component Analysis," *Nature Physics*, vol. 10, no. 9, Sept. 2014, pp. 631–33.

30. Farhi, E., and Neven, H. "Classification with Quantum Neural Networks on Near Term Processors," *arXiv*, 30 Aug. 2018.

31. Shi, J., Li, Z., Lai, W., Li, F., Shi, R., Feng, Y., & Zhang, S. (2023). Two end-to-end quantum-inspired deep neural networks for text classification. IEEE Transactions on Knowledge and Data Engineering, 35(4), 4335–4345. https://doi.org/10.1109/TKDE.2021.3130598

32. Biamonte, J., et al. "Quantum Machine Learning," *Nature*, vol. 549, no. 7671, Sept. 2017, pp. 195–202.

33. Jhanwar, A. and Nene, M. J. "Enhanced Machine Learning using Quantum Computing," in *2021 Second International Conference on Electronics and Sustainable Communication Systems (ICESC)*, Coimbatore, India, 2021, pp. 1407-1413.

34. Weinstein, Y. S., et al. "Implementation of the Quantum Fourier Transform," *Physical Review Letters*, vol. 86, no. 9, Feb. 2001, pp. 1889–91. https://doi.org/10.1103/PhysRevLett.86.1889

35. Ruiz-Perez, L., and Garcia-Escartin, J. C. "Quantum Arithmetic with the Quantum Fourier Transform," *Quantum Information Processing*, vol. 16, no. 6, Apr. 2017, p. 152. https://doi.org/10.1007/s11128-017-1603-1

36. Camps, D., et al. "Quantum Fourier Transform Revisited," *Numerical Linear Algebra with Applications*, vol. 28, no. 1, Jan. 2021, p. e2331. https://doi.org/10.1002/nla.2331

37. Gupta, M., and Nene, M. J. "Quantum Computing: A Measurement and Analysis Review," *Concurrency and Computation: Practice and Experience*, vol. 33, no. 20, Oct. 2021, p. e6344. https://doi.org/10.1002/cpe.6344

38. Gupta, M., and Nene, M. J. "Quantum Computing: An Entanglement Measurement," in *2020 IEEE International Conference on Advent Trends in Multidisciplinary Research and Innovation (ICATMRI)*, Buldhana, India, 2020, pp. 1-6.

39. Upadhyay, G., and Nene, M. J. "One time pad generation using quantum superposition states," in *2016 IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)*, Bangalore, India, 2016, pp. 1882-1886, doi: 10.1109/RTEICT.2016.7808162.

40. Grover, L. K. "A Fast Quantum Mechanical Algorithm for Database Search," in *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing - STOC '96*, ACM Press, 1996, pp. 212–19. https://doi.org/10.1145/237814.237866

41. Gilliam, A., et al. "Optimizing Quantum Search Using a Generalized Version of Grover's Algorithm," *arXiv*, 26 May 2020. https://doi.org/10.48550/arXiv.2005.06468

42. Pronin, C. B., Maksimychev, O. I., Ostroukh, A. V., Volosova, A. V., and Matukhina, E. N. "Creating Quantum Circuits for Training Perceptron Neural Networks on the Principles of Grover's Algorithm," *2022 Systems of Signals Generating and Processing in the Field of on Board Communications*, Moscow, Russia.

43. Duan, B., et al. "A Survey on HHL Algorithm: From Theory to Application in Quantum Machine Learning," *Physics Letters A*, vol. 384, no. 24, Aug. 2020, p. 126595. https://doi.org/10.1016/j.physleta.2020.126595

44. Farhi, E., and Harrow, A. W. "Quantum Supremacy through the Quantum Approximate Optimization Algorithm," *arXiv:1602.07674*, 20 Oct. 2019. https://doi.org/10.48550/arXiv.1602.07674

45. Adcock, J., Allen, E., Day, M., et al. "Advances in quantum machine learning," *arXiv preprint arXiv:1512.02900*, 2015.

46. Deville, Y., and Deville, A. "Single-Preparation Unsupervised Quantum Machine Learning: Concepts and Applications," *IEEE Transactions on Quantum Engineering*, vol. 2, 2021, pp. 1–24. https://doi.org/10.1109/TQE.2021.3121797

47. Cong, I., and Duan, L. "Quantum discriminant analysis for dimensionality reduction and classification," *New J Phys.*, 2016;18(7):073011

48. Burges, C. J. C. "A tutorial on support vector machines for pattern recognition," *Data Min Knowl Disc*, 1998;2(2):121-167.

49. Suykens, J. A. K., and Vandewalle, J. "Least squares support vector machine classifiers," *Neural Process Lett*, 1999;9(3):293-300.

50. Anguita, D., Ridella, S., Rivieccio, F., and Zunino, R. "Quantum optimization for training support vector machines," *Neural Netw*, 2003;16(5-6):763-770.

51. Cao, Y., Guerreschi, G. G., and Aspuru-Guzik, A. "Quantum neuron: an elementary building block for machine learning on quantum computers," 2017.

52. Zhao, J., Zhang, Y. H., Shao, C. P., Wu, Y. C., Guo, G. C., and Guo, G. P. "Building quantum neural networks based on swap test," 2019.

53. Allcock, J., Hsieh, C.-Y., Kerenidis, I., and Zhang, S. "Quantum algorithms for feedforward neural networks," *ACM Trans. Quantum Comput.*, vol. 1, no. 1, 2020.

54. Likas, A., Vlassis, N., and Verbeek, J. J. "The global k-means clustering algorithm," *Pattern Recogn.*, 2003;36(2):451-461.

55. Xiao, J., et al. "A Quantum-Inspired Genetic Algorithm for k-Means Clustering," *Expert Systems with Applications*, vol. 37, no. 7, July 2010, pp. 4966–73. https://doi.org/10.1016/j.eswa.2009.12.017

56. Aïmeur, E., et al. "Quantum Speed-up for Unsupervised Learning," *Machine Learning*, vol. 90, no. 2, Feb. 2013, pp. 261–87. https://doi.org/10.1007/s10994-012-5316-5

57. Jolliffe, I. *Principal Component Analysis*. New York, NY: Springer, 2011, pp. 1094-1096.

58. De Leon, N. P., et al. "Materials Challenges and Opportunities for Quantum Computing Hardware," *Science*, vol. 372, no. 6539, Apr. 2021, p. eabb2823. https://doi.org/10.1126/science.abb2823

59. De Leon, N. P., Itoh, K. M., Kim, D., Mehta, K. K., Northup, T. E., Paik, H., Palmer, B. S., Samarth, N., Sangtawesin, S., & Steuerman, D. W. (2021). Materials challenges and opportunities for quantum computing hardware. Science, 372(6539), eabb2823. https://doi.org/10.1126/science.abb2823

60. Cross, A. W., Bishop, L. S., Smolin, J. A., & Gambetta, J. M. (2017). Open quantum assembly language (arXiv:1707.03429). arXiv. https://doi.org/10.48550/arXiv.1707.03429

61. Fingerhuth, M., Babej, T., & Wittek, P. (2018). Open source software in quantum computing. PLOS ONE, 13(12), e0208561. https://doi.org/10.1371/journal.pone.0208561

62. Khan, A. A., Ahmad, A., Waseem, M., Liang, P., Fahmideh, M., Mikkonen, T., & Abrahamsson, P. (2023). Software architecture for quantum computing systems—A systematic review. Journal of Systems and Software, 201, 111682. https://doi.org/10.1016/j.jss.2023.111682

63. Houssein, E. H., Abohashima, Z., Elhoseny, M., & Mohamed, W. M. (2022). Machine learning in the quantum realm: The state-of-the-art, challenges, and future vision. Expert Systems with Applications, 194, 116512. https://doi.org/10.1016/j.eswa.2022.116512

64. Cerezo, M., Verdon, G., Huang, H.-Y., Cincio, L., & Coles, P. J. (2022). Challenges and opportunities in quantum machine learning. Nature Computational Science, 2(9), 567–576. https://doi.org/10.1038/s43588-022-00311-3

65. Perdomo-Ortiz, A., Benedetti, M., Realpe-Gómez, J., & Biswas, R. (2018). Opportunities and challenges for quantum-assisted machine learning in near-term quantum computers. Quantum Science and Technology, 3(3), 030502. https://doi.org/10.1088/2058-9565/aab859

66. E. P. DeBenedictis, "A Future with Quantum Machine Learning," in Computer, vol. 51, no. 2, pp. 68-71, February 2018, doi: 10.1109/MC.2018.1451646.

67. Khan, A., & Ahmed, F. (2023). Future prospects of quantum machine learning in accelerating drug discovery processes. Emerging Trends in Machine Intelligence and Big Data, 15(10), 50–66. http://orientreview.com/index.php/etmibd-journal/article/view/32

How to cite this article: Sachin Khurana and Manisha J Nene (2024), Quantum Machine Learning: Unraveling a New Paradigm in Computational Intelligence, *Engineering Reports*, *2024*.