International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

An analysis of the State Preparation Techniques for Quantum Machine Learning

Subodh Nath Pushpak[1*], Sarika Jain[1], Siddharth Kalra[2]

Submitted: 05/05/2024 **Revised**: 18/06/2024 **Accepted**: 25/06/2024

Abstract: The study discusses various methods for preparing quantum-states from classical data, which is crucial for quantum machine learning (QML). It analyzes the complexity of these state preparation techniques, highlighting their efficiency and potential challenges. Effective state preparation plays a key role in connecting classical data with quantum systems, allowing quantum algorithms to be utilized in solving machine learning challenges. This paper reviews the related work of state preparation, introduces a variety of state preparation schemes currently proposed, describes the implementation process of these schemes, and summarizes and analyzes the complexity of these schemes. The paper covers different encoding methods, such as basis coding, amplitude coding, and quantum sampling coding. Finally, prospects for conducting research in the area of state preparation have been identified. Furthermore, the document examines potential future research avenues in field of quantum-state preparation and its impact on QML algorithms.

Keywords: *Quantum-state Preparation, quantum machine learning (QML), quantum data encoding*

1 Introduction

QML is a rapidly growing research field that uses principles of quantum information science in the field of artificial intelligence. The incredible potential of this technology lies in its ability to revolutionize the approach to computational tasks by leveraging the remarkable qualities of quantum systems, including superposition and entanglement. Superposition is the intriguing ability of a quantum system to be in combination with two or more states simultaneously until it is measured [1]. Entanglement refers to the quantum feature wherein the quantum properties of two or more objects are interrelated so that the quantum property of each object is related to the state of other objects, even when separated by a substantial distance. This interdependence enables quantum systems to execute specific computational tasks, such as quantum parallel processing, with greater efficiency in contrast to their classical counterparts. QML algorithms may potentially encode classical data and perform computations more efficiently than classical algorithms by representing classical dataset properties into quantum-states [2]. Nonetheless, preparing quantum-states from classical data is a complex process that demands careful consideration of various methods.

Quantum computing, as a new computing model, has the ability to exponentially accelerate some specific algorithms compared to classical computing and is expected to provide sufficient computing power for machine learning [3]. When using quantum computing to handle machine learning tasks, the representation of interesting properties of the dataset plays a vital role. The initial step is to identify the classical data properties of the dataset and ways to represent them so that they can be used in quantum algorithms. One of the fundamental challenges in this domain is the efficient preparation of quantum-states that can be utilized for ML tasks.

ML is a science in Artificial Intelligence (AI) that trains known data through computer learning and uses the trained data model to predict information about unknown data. With the increase in computer performance, machine learning algorithms have significantly enhanced their ability to process voluminous data. Several ways exist to process and train classical data, such as neural networks, clustering, etc. The selection of training methods needs to refer to the corresponding data types to extract the features of unknown data. When processing large-scale data, deep learning methods are often adopted to obtain data features, such as neural networks containing billions of weights, which fully demonstrates the effect of deep learning in processing big data [4].

For QML algorithms to work, the classical properties of data need to be converted in quantum information to harness the computational power of quantum information. The technique of transforming classical data for use in a quantum algorithm is known as the state-preparation [5]. In the preparation of quantum-states, various methods are employed to convert classical properties of data into the corresponding quantum information and states. Additionally, classical data can be mapped to Hamiltonians

[1]*Amity University, Sector 125 Noida, India*
[2]*Capgemini, Australia*
Email Addresses*:*
Subodh Nath Pushpak: subodh.pushpak@s.amity.edu
Sarika Jain: sjain@amity.edu, ORCID: 0000-0002-4752-339X
Siddharth Kalra: siddharthkalra@gmail.com
****Corresponding author:*** *Subodh Nath Pushpak, Amity University, Sector 125, Noida, India. Email: subodh.pushpak@s.amity.edu, ORCID: https://orcid.org/0000-0002-2152-9873*

using specific techniques. The techniques used to build the quantum-state directly influence the selection of the machine learning algorithm. This means that various state preparation methods result in differences in extracting classical data information. These differences may then impact the operations and the steps involved in the QML algorithms and the algorithmic complexity. The accuracy and success rate of state preparation significantly impact the effectiveness of the quantum algorithm. This paper explores the following aspects:

- Approaches for preparing quantum-states
- Simulation of the states of quantum systems
- Applications: The research also explores potential applications of these prepared and simulated quantum-states, potentially in areas such as quantum computation or information processing.

2 State Preparation

In the realm of QML algorithms, quantum computers are revolutionizing the processing of classical data by leveraging the representation of classical data in quantum systems. This process of converting classical data so that it may be consumed by quantum algorithms is known as state preparation [6]..

Traditional methods for quantum-state preparation include initializing qubits in computational basis states (e.g., |0> or |1>) and generating superposition states using Hadamard gates. However, preparing more complex states, especially as the number of qubits increases, poses significant challenges [7].

There are different methods for preparing quantum-states, most of which involve converting that classically represented data into corresponding quantum-states [8]. The method by which the quantum-state is prepared directly affects the choice of executing machine learning algorithms, which means that different state preparation methods determine the differences in extracting classical data information and affect subsequent operations in quantum systems. For QML algorithms, the precision and attainment rate of state preparation are pivotal in determining the comprehensive efficacy and performance of the machine learning algorithm.

The need for state preparation extends beyond the scope of machine learning applications. It is also the basis of some algorithms, such as the HHL [9] or VQE [10], which are used to solve linear equations. The quantum Principal Component Analysis (QPCA) is used for clustering and feature recognition. There are also support vector machine algorithms, which classify large-scale data. The common denominator of this kind of quantum algorithm is to solve practical classical problems, and it needs to use classical data as input and output.

State preparation is a stepwise process and it entails:

1. Transform the classical dataset data to a quantum-state.
2. Apply quantum gates for the unitary transformation of the quantum-state.
3. Finally, evaluate the results using probabilistic quantum measurement multiple times.

The quantum algorithm complexity may be expressed by the count of quantum operations or gates utilized in the circuit of the quantum algorithm. The quantum algorithm efficiency may also be evaluated based on the number of execution queries required. The number of execution steps or time needed for the algorithm is called query complexity. Query complexity is particularly important in quantum computing, which helps quantify the advantage quantum algorithms might have over classical ones. Grover's quantum algorithm for unstructured search requires a query complexity of $O(\sqrt{N})$, and offers quadratic speedup over to classical algorithms [11]. Query complexity is closely related to other complexity measures like time complexity and space complexity. However, Query complexity only indicates the number of queries, and does not account for other computational costs that may be involved.

3 Literature Review

Several studies have investigated various techniques for state preparation in the context of QML algorithms and quantum computing [12]. In their 2022 paper, Cerezo et al. [10] discuss the challenges and opportunities in QML algorithm, highlighting state preparation. Meanwhile, Abrams and Williams (1999) [11] explores the use of quantum algorithms for numerical linear algebra, emphasizing the need for efficient state preparation methods. The paper proposes a state preparation technique based on the Qiskit runtime, a cloud-based quantum computing service.

The related work on amplitude coding for state preparation is extensive. In addition to ordinary coding methods, amplitude coding was explored in the work of Grover and Rudolph in 2002 [13], where they prepared a data distribution that satisfies conditional integrability into a quantum-state. In 2005, Soklakov and Schack [14] used other forms of black boxes to propose an effective probabilistic algorithm under certain restrictions. Another approach is the quantum random access memory method, which directly obtains a new quantum-state from classical data starting from a known quantum-state.

Overall, the literature highlights the importance of state preparation in quantum computing and the need for efficient techniques to enable practical quantum machine learning applications.

4 Encoding

Encoding is the procedure of expressing the classical dataset information into quantum-states [15]. The process involves a unitary transformation that transitions the ground state to the target quantum-state within the quantum system. Encoding is a technique used to compose the state of quantum information using classical information. This stage is indispensable for translating conventional data into quantum format, a pivotal phase in constructing quantum computational algorithms. The goal of encoding is to represent the data in a quantum system.

4.1 Basis coding

In the Basis coding process, binary data-vectors are transformed into the of the quantum-state basis format by using Pauli-X gates to encode the data in the binary representation of the basis state on the corresponding qubits. To translate a binary sequence of length n into a quantum-state composed of n qubits states given by |x⟩=|ix⟩, the process involves transforming |ix⟩, a computational basis state, into the quantum-state |x⟩. This transformation effectively represents the binary string x in the form of a quantum-state. The qubit count required for mapping *N* features of the dataset is *N*. The runtime of preparing the quantum-state for *M* data points with N features is *f(MN).*

4.2 Amplitude coding

The predominant technique for state preparation in quantum computing involves the encoding of data using the amplitude of the quantum-state. This method lays the groundwork for the manipulation and representation of data within a quantum system, serving as a fundamental basis for quantum algorithms and computations. The data vector can be a continuous variable, and the data feature information is expressed as the amplitude of the qubit.

$$|\mathbf{x}\rangle = \sum_{i}^{N} x_i |i\rangle$$

where $\{|i\rangle\}$ This computational basis is essential for the Hilbert space, and it's crucial that the input meets the normalization condition.: $|\mathbf{x}|^2 = 1$; since the amplitudes of a quantum-state are evaluated by the classical information associated with the system. The count of qubits needed for mapping *N* features of the dataset is log_2N. The runtime of preparing the quantum-state for *M* data points with N features is *f(log(MN)).*

4.3 Angle encoding

Angle encoding is a technique that utilizes quantum rotation gates (Rx, Ry, Rz) to transform classical information x. In this method, the classical-data provided sets the parameters for rotating the gate, which are then used to encode the information. Mathematically it is expressed as

$$|\mathbf{x}\rangle = \bigotimes_{i}^{n} R(\mathbf{x}_i)|0^n\rangle$$

where R can be one of Rx, Ry, Rz.

This encoding leverages the relationship between the phase of quantum-states, represented by complex numbers, and the probabilities of observing specific outcomes. In quantum algorithms, phase expressed as angles is crucial. Parameterized quantum circuits tune circuit parameters for desired computations. Angle encoding integrates classical information into quantum-states, enabling quantum computers to process classical data effectively. The count of qubits required for mapping *N* features of the dataset is *N*. The runtime of preparing the quantum-state for *M* data points with N features is *f(MN).*

4.4 Hamiltonian Evolution Ansatz Encoding

The Hamiltonian encoding approach, also known as dynamic encoding, involves embedding conventional information to the dynamics of a quantum-state by manipulating the hamiltonian energy of the system. This approach leverages the quantum-state's evolution over time to store and process information, offering potential advantages in quantum information processing and quantum communication. Unlike directly readying a quantum-state containing the desired feature distribution, this approach implicitly encodes feature information by allowing it to define the progression of the quantum information. Specifically, the data is used to construct a Hamiltonian operator, and then the ground state is developed under the influence of this hamiltonian for a specified duration. This method constructs a Hamiltonian state whose ground state represents the desired quantum-state.

It uses a Trotter formula to approximate an evolution and is useful in obtaining the ground state of a Hubbard model

$$|\mathbf{x}\rangle = \left(\prod_{i=1}^{n} R_{Z_iZ_{i+1}}\left(\frac{t}{T}x_i\right) R_{Y_iY_{i+1}}\left(\frac{t}{T}x_i\right) R_{X_iX_{i+1}}\left(\frac{t}{T}x_i\right)\right)^T \bigotimes_{i=1}^{n+1} |\psi_i\rangle$$

where $R_{XX}, R_{YY}, R_{ZZ}$are the rotation gates, $|\psi_i\rangle$ is a Haar-random single-qubit, and the total count of Trotter steps is given as T.

The Hamiltonian-evolution approach is particularly advantageous when aiming to construct the ground state of a physically realizable hamiltonian. This method facilitates the direct and precise preparation of the desired quantum-state.

5 Encoding for Quantum Image Processing (QIP)

QIP techniques use quantum techniques and algorithms to represent, manipulate, and manage images. It is noteworthy that not all classical image operations can be implemented in quantum; however, there are scenarios where encoding and processing images as quantum data does provide benefits over classical algorithms. The most important aspect of QIP is the representation of images in a way that is conducive to quantum processing.

The pixel values are encoded as probability amplitudes and the position is encoded as the basis states of Hilbert space.

5.1 The Qubit Lattice

This scheme represents a pixel of an image as the amplitude of the qubit. Therefore a pixel at the ith row and jth column is given as

$$|pixel_{i,j}\rangle = \cos\frac{\theta_{i,j}}{2}|0\rangle + \sin\frac{\theta_{i,j}}{2}|1\rangle$$

5.2 Flexible representation of quantum images (FRQI)

In this representation, the amplitude of the qubit denotes the grayscale value of every pixel in the image.. Additionally, an ancilla qubit is employed to denote the relative position of each individual pixel. given by the below equation

$$\langle image\rangle = \frac{1}{2^n}\sum_{i=0}^{2^{2n}-1}(\cos\theta_i|0\rangle + \sin\theta_i|1\rangle)|i\rangle$$

where $\theta_i \in \left[0, \frac{\pi}{2}\right]$

As a result of superposition, the available representation space decreases exponentially when compared to classical images. This phenomenon has significant implications for data storage and processing, particularly in the context of quantum computing and information theory. In FRQI, only 2n + 1 qubits are required since a qubit represents the qubit's gray value, and the entire image is represented by a quantum superposition of the row and column coordinates.

5.3 Novel enhanced quantum representation (NEQR)

NEQR maps grayscale pixel values of an image to qubit basis-states. This approach contrasts with FRQI in that it utilizes the basis-state of an array of qubits to preserve each pixel 's grayscale value, rather than encoding probability amplitude within a qubit. The NEQR representation for a 2n × 2n image is expressed as

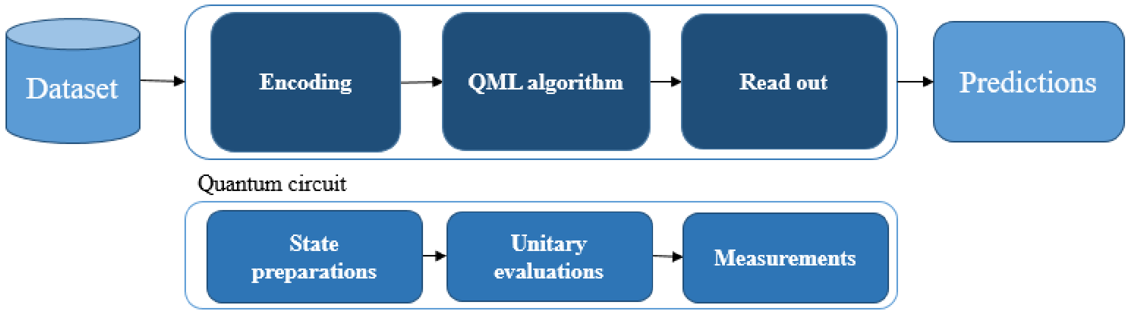
$$\langle image\rangle = \frac{1}{2^n}\sum_{y=0}^{2^{2n}-1}\sum_{x=0}^{2^{2n}-1}|f(y,x)\rangle|yx\rangle$$

where |f(y, x) represent the color information and f (y, x) is the pixel intensity

NEQR technology, through its advanced algorithms and computational methods, demonstrates remarkable precision in the extraction of digital images from quantum images. This innovative approach significantly expedites the image creation process, offering a substantial quadratic speedup compared to conventional methods..

6 Methodology

In this paper, we have implemented various state preparation methods and evaluated their performance on real datasets. Our results demonstrate the trade-offs between the different state preparation methods in terms of accuracy, complexity, and practical feasibility. The dataset attributes encompass various data types, including categorical variables, strings, and integers.



The steps involved are:

1. Encode the input conventional data into the quantum-state using a state preparation technique.
2. Apply a series of parametrized gates to the encoded form to perform feature extraction.
3. Measure the quantum circuit output and measure outcomes to update the parameters of the quantum gates.
4. Repeat steps 2-3 until the model converges.
5. Use the trained quantum circuit and implement the hybrid quantum-classical ML methodology.
6. Optimize the hybrid model using classical backpropagation.
7. Evaluate the performance of the hybrid model on test data.
8. Repeat the process for each state preparation technique and compare the results.

9. Analyze the trade-offs between the different state preparation methods in terms of accuracy, complexity, and practical feasibility.
10. Conclude with the recommended state preparation method based on the application requirements.

6.1 Quantum feature map-based data encoding

A quantum feature-map serves as a method for converting classical dataset properties to the quantum-state space. The selection of an appropriate feature map is of paramount importance and is contingent upon the specific characteristics of the dataset being analyzed for classification purposes.. The feature map contains layers of Hadamard gates and entangling blocks.. The quantum SVM kernel is based on the Pauli, Zfeature, and ZZFeature maps. To prepare quantum kernel matrices for training and testing, a feature map is applied to each pair of training data points first, followed by testing data points. In a later stage, classification support vector machines are trained and tested using quantum kernel matrices [16].

ZZ Feature Map

The Quantum ZZMap is a distinct feature map used in QSVM. The circuit is a second-order Pauli-Z evolution. The ZZ Feature Map requires the input of two parameters: the total count of features and the count of repeated circuits. The ZZ Feature Map is difficult to reproduce using conventional methods and is executed using circuits with low depth on quantum devices that are nearly fully developed. The objective is to build the kernel of the Support Vector Machine (SVM) by using quantum-states. The ZZMap focuses explicitly on the interaction between qubits in the quantum circuit, which introduces a phase shift (the "ZZ" part).

By exploiting quantum entanglement and superposition the ZZMap enhances the expressive power of the feature map.

$$U(\alpha,\beta) = \left(\exp\left(i\sum_j \phi_j(\mu_i)\big(Z_j \otimes Z_k\big)\right)\exp\left(i\sum_j \phi_j(\mu_i)\big(Z_j\big)\right)H^{\otimes n}\right)^d$$

7 Results

The characteristics of the specific dataset strongly influence the performance of quantum encoding, the selected encoding technique, and the quantum algorithm utilized. For the Iris dataset, the Hamiltonian evolution encoding achieved the highest classification accuracy of 97.3% on the test set.

In contrast, the ZZ-feature encoding and IQP encoding achieved slightly lower accuracies of 96.7% and 96.5%, respectively.

8 Conclusion

Machine learning algorithms rely on the quality of the dataset used for learning, and for quantum algorithms, the conventional data must be converted using appropriate encoding techniques to be effective and efficient. Designing efficient and effective encoding strategies is crucial for realizing the full potential of quantum machine learning [17].

The performance of QML techniques is highly dependent on the specific dataset properties, the chosen encoding technique, and the quantum algorithm employed [18].

New encoding techniques, such as Sparse Encoding, which explores efficient encoding methods for sparse datasets, common in many real-world applications, and Hierarchical Encoding [19], which represents hierarchical structures with multiple levels of abstraction, provide ample opportunities for further research.

Data-driven encoding, which aims to Design encoding schemes that adapt to the specific characteristics of the data, is also an active area of research [20].

Some potential research directions include developing techniques for encoding image datasets into quantum-states, investigating quantum encoding techniques for textual data, and exploring approaches to handling time-series data within the quantum computational framework.

References

[1] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe and S. Lloyd, "Quantum machine learning," Nature, vol. 549, p. 195–202, September 2017.

[2] K. Bharti, T. Haug, V. Vedral and L. C. Kwek, Machine learning meets quantum foundations: A brief survey, 2020.

[3] V. Dunjko and H. J. Briegel, "Machine learning & artificial intelligence in the quantum domain: A review of recent progress," Reports on Progress in Physics, vol. 81, no. 7, 2018.

[4] Z. Abohashima, M. Elhoseny, E. H. Houssein and W. M. Mohamed, Classification with Quantum Machine Learning: A Survey, 2020.

[5] H.-Y. Huang, M. Broughton, M. Mohseni, R. Babbush, S. Boixo, H. Neven and J. R. McClean, Power of data in quantum machine learning, vol. 12, 2021.

[6] J. Allcock and S. Zhang, Quantum machine learning, vol. 6, 2019.

[7] R. Li, J. Xu, J. Yuan and D. Li, "An Introduction to Quantum Machine Learning Algorithms," in Advances in Intelligent Systems and Computing, 2021.

[8] D. V. Fastovets, Y. I. Bogdanov, B. I. Bantysh and V. F. Lukichev, Machine learning methods in quantum computing theory, 2019.

[9] H.-Y. Huang, M. Broughton, M. Mohseni, R. Babbush, S. Boixo, H. Neven and J. R. McClean,

"Power of data in quantum machine learning," Nature Communications, vol. 12, 2020.

[10] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio and P. J. Coles, "Variational quantum algorithms," Nature Reviews Physics, vol. 3, p. 625–644, 2021.

[11] D. S. Abrams and C. P. Williams, Fast quantum algorithms for numerical integrals and stochastic processes, arXiv, 1999.

[12] Z. Zhao, J. K. Fitzsimons, P. Rebentrost, V. Dunjko and J. Fitzsimons, "Smooth input preparation for quantum and quantum-inspired machine learning," Quantum Machine Intelligence, vol. 3, 2018.

[13] L. Grover and T. Rudolph, Creating superpositions that correspond to efficiently integrable probability distributions, Quantum Physics (quant-ph), FOS: Physical sciences, FOS: Physical sciences, 2002.

[14] N. Soklakov and R. Schack, "Efficient state preparation for a register of quantum bits," Physical Review A, vol. 73, p. 012307, January 2006.

[15] S. C. Lu, Y. Zheng, X. T. Wang and R. B. Wu, Quantum machine learning, vol. 34, 2017.

[16] R. Mengoni and A. D. Pierro, "Kernel methods in Quantum Machine Learning," Quantum Machine Intelligence, vol. 1, no. 3-4, 2019.

[17] J. Gao, L. F. Qiao, Z. Q. Jiao, Y. C. Ma, C. Q. Hu, R. J. Ren, A. L. Yang, H. Tang, M. H. Yung and X. M. Jin, "Experimental Machine Learning of Quantum-states," Physical Review Letters, vol. 120, no. 24, 2018.

[18] M. Grossi, N. Ibrahim, V. Radescu, R. Loredo, K. Voigt, C. von Altrock and A. Rudnik, "Mixed Quantum–Classical Method for Fraud Detection With Quantum Feature Selection," IEEE Transactions on Quantum Engineering, vol. 3, p. 1–12, 2022.

[19] S. Y. C. Chen, C. M. Huang, C. W. Hsing and Y. J. Kao, Hybrid quantum-classical classifier based on tensor network and variational quantum circuit, 2020.

[20] M. Schuld and N. Killoran, "Quantum Machine Learning in Feature Hilbert Spaces," Physical Review Letters, vol. 122, February 2019.