ESANN 2025 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 23-25 April 2025, i6doc.com publ., ISBN 9782875870933. Available from http://www.i6doc.com/en/.

Enhancing Machine Learning with Quantum Methods

Lautaro Hickmann and Markus Lange and Hans-Martin Rieser

DLR Institute for AI Safety and Security
Sankt Augustin and Ulm, Germany

Abstract. Quantum physics offers a new paradigm that promises to make certain computations faster and more efficient. The recent progress of quantum computers allows for more complex applications which lead to a rising interest in transferring machine learning methods to quantum hardware for practical applications. However, the development of quantum computers is still in its beginnings and currently these approaches require synergy with classical computers. We present some methods where this quantum-classical interplay is used to enhance machine learning approaches.

1 Introduction

For more than 80 years, development of digital computers has made exponential progress in their computing power and has enabled unprecedented economic and societal changes. Our whole modern way of life depends on these machines today. However, this growth has decreased in the last decades and is opposed to the huge demand of novel applications like artificial intelligence. Therefore, science seeks for novel ways of performing computations efficiently. One of these ways employs the linear algebra behind quantum physical phenomena to perform calculations – quantum computing.

In quantum computing, classical bits are replaced by qubits which are the quantum version of two-level systems. Unlike their classical counterparts, states of qubits can be in a superposition of their basis states which leads to a quantum probabilistic behaviour of the calculation. Furthermore, several qubits may be entangled which means their states directly depend on each other. This leads to the possibility of using an exponentially growing Hilbert space and a natural way of performing rotations in this space by unitary gates. However, realizing algorithms that really make full use of the possibilities of this huge state space is non-trivial [1, 2] – in particular for the currently available quantum computers from the noisy intermediate scale quantum (NISQ) era which severely limit both size and length of quantum programs.

The term quantum machine learning (QML) includes a variety of different ansaetze that include quantum parts within their pipeline. The idea behind QML is that the large available state space can lead to a higher expressivity while using less qubits. Moreover superposition and entanglement could lead to faster training or the need for less parameters. However, detailed theoretical analyses are still rare and heuristic results are somewhat limited due to the limitations of available quantum hardware and the exponential cost scaling for simulations of quantum computers.

Our article starts with an introduction into learning on quantum computers including model types and why data reuploading is necessary in many contemporary QML methods. We show a more detailed view on two specific architectures of QML. Quantum Kernel methods which offer better access to theoretical analysis and tensor networks (TN) which are used for their application centric ansatz with a direct correspondence to classical data and models. This enables an integration into hybrid ML/QML pipelines. However, without suitably encoded data no pipeline will work. Therefore we look into how to encode classical data in a quantum state, a main bottleneck for QML. Moreover, we briefly touch on optimizing ML models, both quantum and classical. In the end, we give an outlook to the most relevant open questions in QML from our point of view.

2 The Model of Quantum Machine Learning

A standard approach in quantum machine learning is based on embedding a classical data point into a subspace of the Hilbert space that is spanned by the qubits of the quantum system under consideration. The resulting quantum state is then used as an initial state of the learning algorithm. However, the actual learning, i.e., adjustment of weights, is performed on the classical outputs obtained by measuring the quantum system in its final state. To be more precise, let $\mathcal{X}$ be the space of input data and $\mathcal{F}$ denote the Hilbert space of $2^n \times 2^n$ Hermitian operators. Let $U^{\{\phi\}} : \mathcal{X} \to \mathcal{F}$ be a feature encoding unitary where ϕ indicates here that there are different choices of feature encoding maps. Examples of feature encodings are angle encoding and amplitude encoding [3]. The density matrix describing the n-qubit quantum state encoding the input data $x \in \mathcal{X}$ via the feature encoding $U_{(\phi)}$ is given by

$$\rho^{\{\phi\}}(x) = U^{\{\phi\}}(x)|0\rangle\langle 0|[U^{\{\phi\}}(x)]^*$$

where $|0\rangle \in \mathcal{H}$ denotes the n-qubit *zero* state in a chosen basis. Note that $\mathcal{H}$ denotes the Hilbert space of all possible n-qubit quantum states. Moreover, note that we will omit the dependence on the feature encoding map from here on out. To measure this state with respect to (w.r.t.) some observable $\mathcal{O} \in \mathcal{F}$ we take the expectation value of the state w.r.t. the observable

$$f(x) = \mathrm{Tr}[\rho(x)\mathcal{O}] = \langle \rho(x), \mathcal{O} \rangle_{\mathcal{F}} \,. \tag{1}$$

This is a linear function in the quantum feature space $\mathcal{F} \subset \mathcal{L}(\mathcal{H})$, which is a subspace of the space of linear operators on $\mathcal{H}$. Depending on the family of observables we have a so-called implicit or explicit quantum model [4].

2.1 Explicit quantum model

We define $\mathcal{O}_\theta := V_\theta^* \mathcal{O} V_\theta$ for a fixed observable $\mathcal{O} \in \mathcal{F}$ and a variational unitary operator $V_\theta : \mathcal{H} \to \mathcal{H}$ with parameter $\theta \in \mathbb{R}^m$, $m \in \mathbb{N}$. Due to the unitary evolution of quantum states, V_θ is normal, i.e., $V_\theta V_\theta^* = V_\theta^* V_\theta = 1$ for each fixed θ. Moreover, choosing an observable $\mathcal{O}_\theta$ for fixed θ amounts to selecting the vectors $\omega_\theta \in \mathcal{F}$ accessible to the linear model (1). These models are also called quantum neural networks [5, 6] since as in classical machine learning the parameter θ can be used to optimize the output of the model w.r.t. some learning objective given via some loss function.

2.2 Implicit quantum model

Another way to specify the model (1) is to use the kernel trick and construct a kernel from the state encoding density matrix. Similar to the classical case, the idea is to utilize the encoding into a higher dimensional space to decompose the input data space into separable subspaces [7, 8]. To do this, we consider as an observable $\mathcal{O} := \mathcal{O}_{\alpha,\mathcal{D}} = \sum_{k=1}^{K} \alpha_k \rho(x^{(k)})$ where $\mathcal{D} = \{x^{(1)}, \dots, x^{(K)}\} \subset \mathcal{X}$ is the training dataset. Note that usually the model is written in kernel form, i.e.,

$$f_{\alpha,\mathcal{D}}(x) = \sum_{k=1}^{K} \alpha_k K(x, x^{(k)})$$

with kernel $K(x, x^{(k)}) = \mathrm{Tr}[\rho(x)\rho(x^{(k)})]$. More details on quantum kernel models are given in Section 3.1.

2.3 Data reuploading quantum models

By intertwining feature encoding unitaries with the variational unitary operators defining the trainable parts of the linear model (1) so-called data reuploading models [9]

$$f_\theta(x) = \mathrm{Tr}[\rho_\theta(x)\mathcal{O}_\theta]\,, \quad x \in \mathcal{X}, \theta \in \mathbb{R}^m, m \in \mathbb{N} \tag{2}$$

can be constructed. Here $\rho_\theta(x) := U(x,\theta)|0\rangle\langle 0|U^*(x,\theta) \in \mathcal{F}$ denotes a hermitian operator intertwining feature encoding and variational steps, i.e.,

$$U(x,\theta) = U_1(x) \prod_{\ell=2}^{L} V_{\theta,\ell} U_\ell(x)$$

with variational unitaries $V_{\theta,\ell}$ and feature encoding unitaries $U_\ell(x)$ for some $x \in \mathcal{X}$. The observable in (2) is given as $\mathcal{O}_\theta = V_{\theta,1}^* \mathcal{O} V_{\theta,1}$ for some observable $\mathcal{O} \in \mathcal{F}$. Note that in general the feature encoding unitaries and the variational unitaries do not commute, i.e.,

$$[U_\ell(x), V_{\theta,\ell'}] = U_\ell(x)V_{\theta,\ell'} - V_{\theta,\ell'}U_\ell(x) \neq 0\,, \quad \forall \ell, \ell' \in \{1, \dots, L\}\,.$$

Hence, there is no straightforward representation as an explicit model with the same amount of qubits. However it is possible to find a mapping from data reuploading models to explicit models if additional ancilla qubits are added [4].

The encoding most often used for data reuploading models is parametrized Pauli rotational encoding. It is a form of time evolution encoding where the Hamiltonian is made up of 2×2 hermitian matrices acting on each single qubit separately. Any of these matrices H can be decomposed into

$$H = \alpha_0 \sigma^0 + \sum_{k \in \{x,y,z\}} \alpha_k \sigma^k$$

where $\alpha \in \mathbb{R}^4$ is a H-dependent coefficient and σ^i are the usual Pauli matrices with $\sigma^0 \equiv 1_{\mathbb{R}^{2\times 2}}$, the 2×2 unit matrix. Hence a scalar data point $x_j \in [0, 2\pi)$ can be associated with the angle of a parametrized Pauli rotation via the single qubit encoding

$$U(x_j) = e^{-iHx_j} = \exp(-i \left[\alpha_0 \sigma^0 + \alpha_x \sigma^x + \alpha_y \sigma^y + \alpha_z \sigma^z\right] x_j) \, .$$

For arbitrary multidimensional data $x \in \mathcal{X}$ a suitable scaling and a parallel data encoding strategy needs to be applied.

The model function of a data re-uploading model, can be expressed as a partial Fourier series. The nomenclature *partial* indicates that only a subset of the Fourier coefficients is nonzero. The integer frequency spectrum for which the Fourier coefficients are nonzero $\Omega \subset \mathbb{Z}^N$ is determined by the eigenvalues of the data encoding gates used in the encoding blocks. The remaining architecture, i.e. the variational blocks and the observable that defines the readout operation, characterize the coefficients c_ω that a quantum model can realize. The quantum model can thus be represented as [10]:

$$f_\theta(x) = \sum_{\omega \in \Omega} c_\omega(\theta) e^{i\omega x} \, .$$

Note that the expressivity of the model is directly visible from the partial Fourier representation.

In general, the expressivity of a given quantum learning model can be determined by estimating the randomness of states generated from the model in terms of the Kullback-Leibler (KL) divergence [11, 12]. In particular, there is a correspondence between the gradient of the cost landscape (trainability) and the expressibility of the quantum model [13].

3 Quantum Machine Learning Methods

Even more important than the expressivity of a quantum model is its ability to generalize to unseen data. Generalization has for example been investigated from a statistical learning theory perspective with the help of bounds based on the (quantum) Fisher information matrix [14, 15]. However, since the implicit (kernel) perspective is more amenable to theoretical analysis a lot of work on generalization capabilities takes this path.

3.1 Quantum Kernel Methods

From the view point of kernels, it was shown that generalization is impossible if the largest eigenvalue of the kernel integral operator is small and that generalization is unlikely if the rank of the kernel matrix is large [16, 17]. This points to the case, that learning is impossible for models with a large number of qubits unless the amount of training data provided grows exponentially with qubit count. Nevertheless, using a hyperparameter called the *kernel bandwidth* it has been shown that learning is possible even for high numbers of qubits if the kernel and the dataset are well aligned [18, 19].

Based on this an analytical and numerical investigation of the influence of expressivity on the generalization ability of quantum kernel models is performed in the contribution Gross et. al. [20]. In particular, a parallel data encoding strategy is used and it is shown, that from this a simple universal form of quantum kernels emerges. Moreover, using a qubit-dependent data re-scaling schemes, it is possible to exponentially vary the spectral content of the kernel and thereby control its simplicity bias.

Hence by analysing the respective kernel a lot of information on the generalization capabilities of the underlying quantum model can be gained.

3.2 Data Encoding of Images

For machine learning classical images are often used as inputs. Images contain large amounts of information that have to be encoded. The encoding of classical data is a critical step for QML run time and expressivity. Especially since most quantum machine learning algorithms only return probabilistic results, the whole routine, including state preparation and measurement, may have to be repeated several times [21]. A Flexible Representation of Quantum Images (FRQI) [22] was proposed as a method to encode Grey-value images into quantum computers. The intensity information and their corresponding position are encoded into a normalized quantum state representation of the image. Later the Multi-Channel representation for Quantum Images (MCQI) [23] was proposed as a generalization for RGB images.

Given a $2^n \times 2^n$ grey-valued image, using FRQI it can be encoded as:

$$|I(\theta)\rangle_{\text{FRQI}} = \frac{1}{2^n} \sum_{i=0}^{2^{2n-1}} \left(\cos\theta_i|0\rangle + \sin\theta_i|1\rangle\right)|i\rangle$$

with $\theta_i \in [0, \frac{\pi}{2}], i = 0, 1, \ldots 2^{2n-1}$ the re-scaled pixel intensities and by selecting an ordering of pixels from the 2D structure of the image, e.g. going through the image following a snake-like pattern. The representation consist of two parts, the computational basis quantum states $|i\rangle$ represent the position of each pixel and $\cos\theta_i|0\rangle + \sin\theta_i|1\rangle$ encodes their respective intensity information. The resulting state is then normalized.

ESANN 2025 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 23-25 April 2025, i6doc.com publ., ISBN 9782875870933. Available from http://www.i6doc.com/en/.

MCQI expands the encoding of intensities to RGB images with:

$$\begin{aligned}|C^i_{RGB}\rangle = &\cos\theta^i_R|000\rangle + \cos\theta^i_G|001\rangle + \cos\theta^i_B|010\rangle + \sin\theta^i_R|100\rangle \\ &+ \sin\theta^i_G|101\rangle + \sin\theta^i_B|110\rangle + \cos 0|110\rangle + \sin 0|111\rangle\end{aligned}$$

where $\{\theta^i_R, \theta^i_B, \theta^i_G\} \in [0, \frac{\pi}{2}]$ are three angles encoding the color value of the R, G, and B channels of the i-th pixel. For a chosen ordering of pixels the image is then encoded as

$$|I(\theta)\rangle_{\text{MCQI}} = \frac{1}{2^n + 1} \sum_{i=0}^{2^{2n-1}} |C^i_{RGB}\rangle|i\rangle \,.$$

A special kind of images generalising RGB channels to several emission spectra are so called Hyperspectral images. These images have a high dimensional data structure, with vectors of multiple intensities per pixel. Usually up to hundreds of spectral bands are used. Let $x \in \mathbb{R}^{2^n}$ be a normalized version of such a vector, possibly padded with zeros. A qubit-efficient way to encode this vector into a quantum state is amplitude encoding (AE). It encodes classical information into quantum state amplitudes

$$\mathbf{x} = \begin{pmatrix} x_1 & \cdots & x_{2^n} \end{pmatrix}^T \rightarrow |\psi_{\mathbf{x}}\rangle = \sum_{j=1}^{2^n} x_j|j\rangle$$

where $|j\rangle$ denotes the j-th basis state of the n-qubit quantum system.

Amplitude encoding uses a low number of qubits to store the classical information at the expense of increasing the complexity of the encoding process [21].

Fischbach et. al. [24] proposed to combine AE with MCQI by replacing $|C^i_{RGB}\rangle$ with $|\psi_{\mathbf{x}}\rangle$, thus enabling the encoding of hyperspectral images into a quantum state.

All three encoding methods require multi-controlled gates and deep circuit to realize the encoding. A novel way to alleviate this constraints is by using tensor networks to compress and represent the quantum states prior to encoding them into a circuit.

3.3 Quantum Tensor Networks

Tensor networks (TNs) originate as a method from quantum many body physics to reduce the numerical effort needed to describe quantum systems. The idea is to decompose large tensors – a generalization of matrices – into a network of smaller tensors. This approach significantly reduces storage size and computation demands when the chosen network layout fits to the underlying structure of the quantum system or machine learning problem and therefore removes redundancies. A major field of use for tensor networks in quantum computing is the classical simulation of circuits – the most efficient methods today make use of TNs [25].

TNs can also be used for machine learning [26]. In particular, MPS have been employed as convolutional layers together with neural pooling. The main

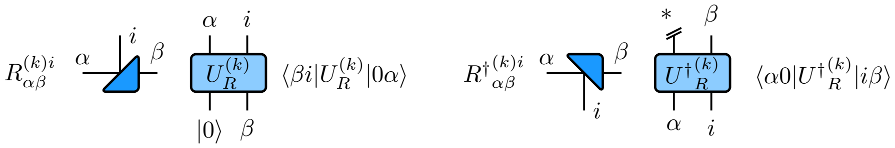

Fig. 1: Mapping from isometric tensors to unitary quantum gates. Whether a bond α, β is mapped to an incoming or outgoing qubit bundle, depends on whether the tensor is given in right or left isometric form. The free bonds i are represented by outgoing qubits. Adjoining a tensor flips its directions. Qubit preservation is taken into account by adding additional dummy qubits or discarding left over ones. Adapted from [28].

advantage of tensor network based approaches is their intrinsic locality and structure. This allows for very efficient optimization with dedicated approaches like density matrix renormalization group (DMRG) approaches operating on single tensor nodes. By limiting the sizes of the bonds, i.e. the number of connections or the bond dimension, between different nodes, a straight forward way of approximating the TN structure is available. However, choosing the right tensorial layout suiting the problem at hand is critical for achieving good accuracies. Furthermore, their origin as a numeric method for quantum many body systems leads to an intrinsic linear behaviour of TNs – integrating nonlinearities that are essential to modern ML approaches must be taken care of outside the tensor network itself.

Instead of using tensor networks as a trainable ansatz for a machine learning models, one can compress already trained networks since they are representing – very complex – tensorial structures. When limiting the bond dimensions, lossy approximations are possible. Jahromi et. al. [27] provide an investigation on how to use tensor networks to reduce the size of LLMs by a factor of 70% with only a minimal reduction in accuracy.

When using a quantum computer, the strict linearity of TNs is a major advantage: Mapping classical TNs to a quantum computer is straight forward as shown in Figure 1. The main difference is that classical tensor networks may make use of any kind of tensor while quantum nodes are restricted to unitary nodes. Standard TN manipulation methods like singular value decomposition produce already isometric tensors which can be padded with the tensors' kernels to get a unitary object [28]. However, DMRG does not preserve unitarity of tensors. This means, one needs to project to a unitary or enforce this behaviour as boundary condition as shown by Jäger et. al. [29].

A direct application of this classical–quantum mapping for TNs is related to the data encoding bottleneck mentioned in the Introduction since TNs can also be used to compressed the data necessary for quantum machine learning. The resulting object can be interpreted as a quantum state and even mapped to a two-qubit gate representation for a NISQ era quantum computer with reasonable accuracy [30]. This may be seen as a classical feature selection stage.

3.4 Quantum Computing and Optimizing ML

The feature selection process is a necessary step in any supervised machine learning pipelines. One can interpret these features as random variables in the model and use mutual information to quantify how much information can be gained from one of the features about any of the other ones. Thus mutual information can be used to optimise the feature selection and obtain a stable solution with reasonable usage of resources for the considered machine learning problem [31]. Assuming conditional independence the optimal feature combination can be phrased as a QUBO problem. Such problems are at the heart of many applications for example in finance, traffic management and machine learning. So-called annealing algorithms are a way to obtain a solution [32]. In contrast to simulated annealing which uses thermal fluctuations, quantum annealing employs quantum fluctuations (tunneling) to escape local minima and find the true ground state of the considered system by employing a special time evolution. For more details on quantum annealing and its application to optimization problems we refer to [33].

Pranjić et. al. [34] demonstrate that a best-fitting feature combinations can be obtained by using a hybrid approach that uses classical methods together with quantum annealing for solving the respective QUBOs. Their Mutual Information Quadratic Unconstrained Binary Optimization (MIQUBO) approach improves the prediction of machine learning models for datasets which have a small mutual information concentration w.r.t. the chosen feature that should be predicted.

4 Outlook

We have outlined, that there are promising new approaches in the field of QML. Quantum, Quantum Inspired and Hybrid Machine Learning techniques are gaining more and more track and might overcome some of the shortcomings of classical approaches, mainly on the (energy) efficiency and training effort side in the future. However, there are still major challenges to overcome. The most prominent is the unavailability of capable quantum computers. Noise on current devices severely limits the size of algorithms which also sets a hard limit on the complexity of data that can be evaluated. In addition, as most existing approaches are hybrid quantum-classical algorithms with many handovers, a tight interconnection between quantum and classical CPU-/GPU-based hardware is essential for making QML perform well.

On the software side, there are, in our opinion, three major open issues. Encoding data into a quantum system is notoriously hard and needs more suitable preprocessing methods. Moreover, there is not much experience with building well performing algorithms similar to the heuristic approach in classical machine learning. Finally, theoretical insights on the inner functioning of quantum algorithms is even more scarce than in classical machine learning.

Still, there are several near-term applications. First, quantum-inspired techniques already improve performance of machine learning, and also quantum simulations can benefit from the quantum–classical interplay. Second, deploying

ESANN 2025 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 23-25 April 2025, i6doc.com publ., ISBN 9782875870933. Available from http://www.i6doc.com/en/.

quantum machine learning directly to quantum data will circumvent the encoding bottleneck and therefore offers a more straightforward approach to quantum data evaluation.

Overall, quantum machine learning presents itself as a promising approach, that could have major impact on developing a solution to the current efficiency problem of large machine learning models.

References

[1] Peter W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26(5):1484–1509, 1997.

[2] Lov K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*, STOC '96, page 212–219, New York, NY, USA, 1996. Association for Computing Machinery.

[3] Manuela Weigold, Johanna Barzen, Frank Leymann, and Marie Salm. Encoding patterns for quantum algorithms. *IET Quantum Communication*, 2(4):141–152, December 2021.

[4] Sofiene Jerbi, Lukas J. Fiderer, Hendrik Poulsen Nautrup, Jonas M. Kübler, Hans J. Briegel, and Vedran Dunjko. Quantum machine learning beyond kernel methods. *Nature Communications*, 14(1), January 2023.

[5] K. Mitarai, M. Negoro, M. Kitagawa, and K. Fujii. Quantum circuit learning. *Physical Review A*, 98(3), September 2018.

[6] Edward Farhi and Hartmut Neven. Classification with quantum neural networks on near term processors, 2018.

[7] Vojtěch Havlíček, Antonio D. Córcoles, Kristan Temme, Aram W. Harrow, Abhinav Kandala, Jerry M. Chow, and Jay M. Gambetta. Supervised learning with quantum-enhanced feature spaces. *Nature*, 567(7747):209–212, March 2019.

[8] Maria Schuld and Nathan Killoran. Quantum machine learning in feature hilbert spaces. *Physical Review Letters*, 122(4), February 2019.

[9] Adri'an P'erez-Salinas, Alba Cervera-Lierta, Elies Gil-Fuster, and José Ignacio Latorre. Data re-uploading for a universal quantum classifier. *Quantum*, 4:226, 2020.

[10] Maria Schuld, Ryan Sweke, and Johannes Jakob Meyer. Effect of data encoding on the expressive power of variational quantum-machine-learning models. *Phys. Rev. A*, 103:032430, 03 2021.

[11] Sukin Sim, Peter D. Johnson, and Alán Aspuru-Guzik. Expressibility and entangling capability of parameterized quantum circuits for hybrid quantum-classical algorithms. *Advanced Quantum Technologies*, 2(12), October 2019.

[12] Kouhei Nakaji and Naoki Yamamoto. Expressibility of the alternating layered ansatz for quantum computation. *Quantum*, 5:434, April 2021.

[13] Zoë Holmes, Kunal Sharma, M. Cerezo, and Patrick J. Coles. Connecting ansatz expressibility to gradient magnitudes and barren plateaus. *PRX Quantum*, 3(1), January 2022.

[14] Amira Abbas, David Sutter, Christa Zoufal, Aurelien Lucchi, Alessio Figalli, et al. The power of quantum neural networks. *Nature*, 2021.

[15] Tobias Haug and M.S. Kim. Generalization of quantum machine learning models using quantum fisher information metric. *Physical Review Letters*, 133(5), July 2024.

[16] J. M. Kübler*, S. Buchholz*, and B. Schölkopf. The inductive bias of quantum kernels. In *Advances in Neural Information Processing Systems 34 (NeurIPS 2021)*, pages 12661–12673. Curran Associates, Inc., December 2021. *equal contribution.

[17] Hsin-Yuan Huang, Michael Broughton, Masoud Mohseni, Ryan Babbush, Sergio Boixo, Hartmut Neven, and Jarrod R. McClean. Power of data in quantum machine learning. *Nature Communications*, 12(1), May 2021.

[18] Ruslan Shaydulin and Stefan M. Wild. Importance of kernel bandwidth in quantum machine learning. *Phys. Rev. A*, 106:042407, Oct 2022.

[19] Akash Canatar, Eric Peters, Cengiz Pehlevan, and Stefan Wild. Bandwidth enables generalization in quantum kernel models. *Phys. Rev. X*, 2022.

[20] Markus Gross, Markus Lange, Bogusz Bujnowski, and Hans-Martin Rieser. Expressivity vs. generalization in quantum kernel methods. In *33st European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, ESANN 2025*, 2025.

[21] Maria Schuld and Francesco Petruccione. *Machine Learning with Quantum Computers.* Springer International Publishing, 2021.

[22] Phuc Q. Le, Fangyan Dong, and Kaoru Hirota. A flexible representation of quantum images for polynomial preparation, image compression, and processing operations. *Quantum Information Processing*, 10(1):63–84, April 2010.

[23] Bo Sun, Abdullah M. Iliyasu, Fei Yan, Fangyan Dong, and Kaoru Hirota. An rgb multichannel representation for images on quantum computers. *Journal of Advanced Computational Intelligence and Intelligent Informatics*, 17(3):404–417, 2013.

[24] Fabian Fischbach, Hans-Martin Rieser, and Oliver Sefrin. Encoding hyperspectral data with low-bond dimension quantum tensor networks. In *33st European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, ESANN 2025*, 2025.

[25] Siddhartha Patra, Saeed S. Jahromi, Sukhbinder Singh, and Román Orús. Efficient tensor network simulation of ibm's largest quantum processors. *Phys. Rev. Res.*, 6:013326, Mar 2024.

[26] William Huggins, Piyush Patil, Bradley Mitchell, K Birgitta Whaley, and E Miles Stoudenmire. Towards quantum machine learning with tensor networks. *Quantum Science and Technology*, 4(2):024001, January 2019.

[27] Saeed Jahromi, Uygar Kurt, Sukhi Singh, David Montero, Borja Aizpurua Altuna, and Román Orús Lacort. Compactifai: Extreme compression of large language models using quantum-inspired tensor networks. In *33st European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, ESANN 2025*, 2025.

[28] Hans-Martin Rieser, Frank Köster, and Arne Peter Raulf. Tensor networks for quantum machine learning. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 479(2275), July 2023.

[29] Gustav Jäger, Martin B. Plenio, and Hans-Martin Rieser. Quantum tensor network learning with dmrg. In *33st European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, ESANN 2025*, 2025.

[30] Shi-Ju Ran. Encoding of matrix product states into quantum circuits of one- and two-qubit gates. *Physical Review A*, 101(3), March 2020.

[31] Gerhard Hellstern, Vanessa Dehn, and Martin Zaefferer. Quantum computer based feature selection in machine learning. *IET Quantum Communication*, 5(3):232–252, February 2024.

[32] Gary Kochenberger, Jin-Kao Hao, Fred Glover, Mark Lewis, Zhipeng Lü, Haibo Wang, and Yang Wang. The unconstrained binary quadratic programming problem: a survey. *Journal of Combinatorial Optimization*, 28(1):58–81, April 2014.

[33] Arnab Das and Bikas Chakrabarti, editors. *Quantum Annealing and Other Optimization Methods.* Springer Berlin Heidelberg, 2005.

[34] Daniel Pranjić, Bharadwaj Chowdary Mummaneni, and Christian Tutschku. Quantum annealing based feature selection. In *33st European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, ESANN 2025*, 2025.