Quantum Machine Learning Adapters

[1,2]Snehal Raj, [1]Brian Coyle

[1]*QC Ware, Palo Alto, USA and Paris, France*

[2]*Laboratoire d'Informatique de Paris 6, CNRS, Sorbonne Université, Paris, France*

Contact e-mail: snehal.raj@qcware.com

Abstract

Fine-tuning techniques are crucial in adapting large pre-trained machine learning models to specific tasks and avoiding complete model retraining. As model size continues to grow, traditional fine-tuning (which requires updating all model parameters) becomes computationally expensive and less feasible, so parameter-efficient methods are essential. In this work, we propose classical and quantum *Adapters* inspired from Hamming-weight preserving quantum circuits from quantum machine learning literature. We evaluate the effectiveness of these quantum-inspired techniques on the MNIST dataset, and compare against the widely used classical adapter method, Low Rank Adaptation (LoRA). Our results suggest that these techniques could serve as an effective substitute for other parameter-efficient fine-tuning methods currently used in the task-specific adaptation of large language models. Finally, we discuss the linear combination of unitaries (LCU) framework as a quantum parameterization for Adapters in both applications using quantum data and classical data in quantum states. We discuss opportunities and challenges of such models relating to barren plateaus and classical simulability.

Introduction Pre-trained Language Models (PLMs) have shown exceptional performance in numerous natural language processing (NLP) tasks [1]. To optimize these models for specific tasks, fine-tuning adapts PLMs to task-specific data. Yet, traditional fine-tuning, which updates *all* model parameters, becomes computationally prohibitive as PLMs grow larger. To address this challenge, Parameter Efficient Fine Tuning (PEFT) methods are widely used by industry practitioners [2]. So-called *Adapters* have the role of *adapting* the full model to the fine-tuning scenario. Low-Rank Adaptation (LoRA) [3] is perhaps the most widely used technique for PEFT that efficiently reduces the number of trainable parameters by leveraging low-rank matrix decompositions. A basic adapter works as follows. Assume a pre-trained weight matrix $W_0 \in \mathbb{R}^{d\times k}$, producing output $\boldsymbol{h}$ given input $\boldsymbol{x}$. For example, LoRA modifies the model by adding an update $\Delta W := BA$, via summation, where $B \in \mathbb{R}^{d\times r}$, $A \in \mathbb{R}^{r\times k}$, with $r \leqslant \min(d,k)$ enforcing the low-rank condition. The output of a LoRA adapted layer is then $\boldsymbol{h}'$ as shown below. During adaptation, only A and B are trainable, while W_0 remains frozen, significantly reducing the computational overhead.

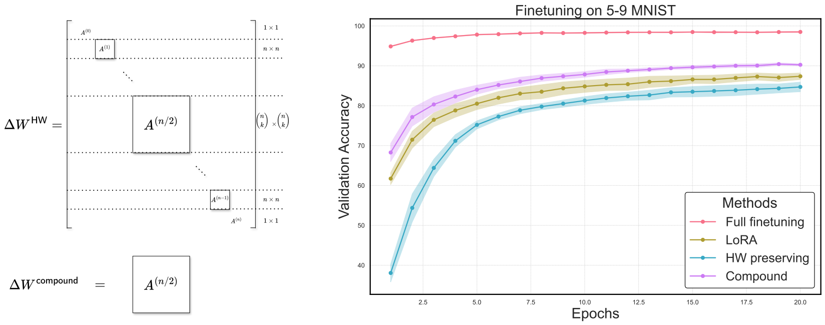
$$\mathsf{PLM} \rightarrow \boldsymbol{h} = W_0\boldsymbol{x} \implies \mathsf{PLM} + \mathsf{LoRA} \rightarrow \boldsymbol{h}' = W_0\boldsymbol{x} + \Delta W\boldsymbol{x} = (W_0 + BA)\boldsymbol{x}$$

Quantum-inspired classical Adapters Using quantum, or quantum-inspired methods, we aim to discover novel Adapters, either for quantum or classical models, which either outperform the current state-of-the-art in terms of metrics such as accuracy, and/or give more efficient parameterizations of Adapters. Our first proposal in this direction is a novel *quantum-inspired* (classical) parameterization for adapter layers inspired from Hamming weight (HW) preserving quantum circuits proposed by Refs. [4, 5, 6]. A specific instance of these circuits is known as the quantum *compound* layer, which can be shown to be a quantum efficient parameterization for *compound* matrices [4], where a compound matrix is defined as follows. Given a 'base' matrix, $A \in \mathbb{R}^{n\times n}$, the compound matrix $A^{(k)}$ for $k \in [n]$ is the $\binom{n}{k} \times \binom{n}{k}$ dimensional matrix with entries $A^{(k)}_{IJ} := \det(A_{IJ})$. The 'compound Adapters' which will serve the basis of our proposal are quantum-inspired because, for a constant $k = \mathcal{O}(1)$, these matrices can be efficiently classically simulated, although may be implemented via quantum circuits to enable a polynomial speedups. We discuss two possibilities in the following.

Direct compound parameterization of Adapters The first proposal is to *directly* use the k^{th} order compound matrix as the adapter matrix. By matching $\binom{n}{k}$ to the size of the original pre-trained matrix W_0 we get the following adapter update rule:

$$\Delta W^{\mathsf{compound}} := A^{(k)}$$

Since the compound matrices are determined by the parameters in their base, A, the compound adapter has a comparable quadratic-to-linear parameter reduction, similar to LoRA.



Method	(FT) {5-9}	(OG) {0-4}	N_p
Full	98.50%	44.97%	8645
LoRA	87.37%	54.30%	256
Comp.	**90.26%**	53.61%	256
HW	84.71%	**60.24%**	144

(a) (b) (c)

Figure 1: **Overview of finetuning outcomes using different models and techniques.** a) Hamming-weight & compound adapter parameterizations. HW is block-diagonal matrix composed of compound matrices, and compound is a specific order compound matrix b) Validation accuracy while finetuning using different techniques. c) Comparison of model performance on the original dataset (MNIST) after finetuning for 20 epochs (FT, MNIST {5-9}). Quantum inspired parameterizations are able to achieve similar performance to LoRA while demonstrating lesser forgetting (OG, MNIST {0-4}).

Hamming weight preserving parameterization of Adapters Secondly, we propose to use *all* compound matrices in a block-diagonal (direct sum) form as the adapter as seen in Fig. 1a. In the context of quantum circuits, such matrices can correspond to *Hamming-weight preserving* unitaries, where each compound matrix $A^{(k)}$ acts exclusively on a fixed (k) Hamming-weight subspace of the full 2^n dimensional Hilbert space. With this parameterization, it is sufficient to take an exponentially compression base $A \in \mathbb{R}^{\log(N)\times\log(N)}$ as an adapter for the original $W_0 \in \mathbb{R}^{N\times N}$.

$$\Delta W^{\mathsf{HW}} := \bigoplus_{k=0}^{n} A^{(k)}$$

Finally, we can increase the expressivity of these Adapters by taking a *linear combination* with parameters, $\boldsymbol{\alpha} := \{\alpha_i\}_i$ as follows (where $(*) \in \{\mathsf{compound}, \mathsf{HW}\}$):

$$\Delta W^*_{\mathsf{LC}} := \sum\nolimits_{i=1}^{R} \alpha_i \Delta W^*_i \implies \boldsymbol{h}' = W_0\boldsymbol{x} + \Delta W^*_{\mathsf{LC}}\boldsymbol{x}$$

Here, the number of sub-Adapters (R) added is a hyperparameter similar to the *rank* hyperparameter in LoRA.

Results In our experiments, we evaluate the efficacy of the proposed quantum-inspired fine-tuning Adapters using the MNIST dataset [7]. We partition MNIST into two subsets, $\mathcal{A}, \mathcal{B}$ based on class labels: $\mathcal{A} := \{0, 1, 2, 3, 4\}, \mathcal{B} := \{5, 6, 7, 8, 9\}$. The architecture of our neural network to adapt is straightforward, consisting of three linear layers of sizes $N_H \times N_H, N_H \times N_H$ and $N_H \times 5$[1] respectively. Initially, the network is trained to predict digits in subset $\mathcal{A}$, then we adapt the trained model to predict subset $\mathcal{B}$. The methods we compare are 1) Full fine-tuning, 2) Low-Rank Adaptation (LoRA), 3) Hamming-weight preserving adaptation, 4) direct compound adaptation. For the latter two (3, 4) we choose $R = 2$ in the linear-combination. In Fig. 1b we show the validation accuracy on MNIST 5-9 (subset $\mathcal{B}$) when fine-tuning using different strategies. We make two observations. First, even though the Hamming-weight preserving adapter $(\Delta W^{\mathsf{HW}}_{\mathsf{LC}})$ has fewer parameters, it is able to perform comparably to the SOTA LoRA. However, with comparable parameters, the compound adapter, $\Delta W^{\mathsf{compound}}_{\mathsf{LC}}$, can outperform LoRA by 3% in accuracy. Also when tested on the *original* dataset as shown in Table 1c, the Hamming-weight adapter, $\Delta W^{\mathsf{HW}}_{\mathsf{LC}}$ is capable of being less forgetful (losing less accuracy on original problem). Minimizing forgetfulness is especially important to induce generalization abilities when models are fine-tuned across various tasks. Therefore, both of these new approaches may be able to replace SOTA fine-tuning methods in the literature, for different purposes.

Quantum Adapters The above linear combination of adaptors ΔW^*_{LC} is essential to the performance of the adaptation. Taking inspiration from this, we propose to use the above models in a purely *quantum* framework,

[1] Since we have 5 possible output classes.

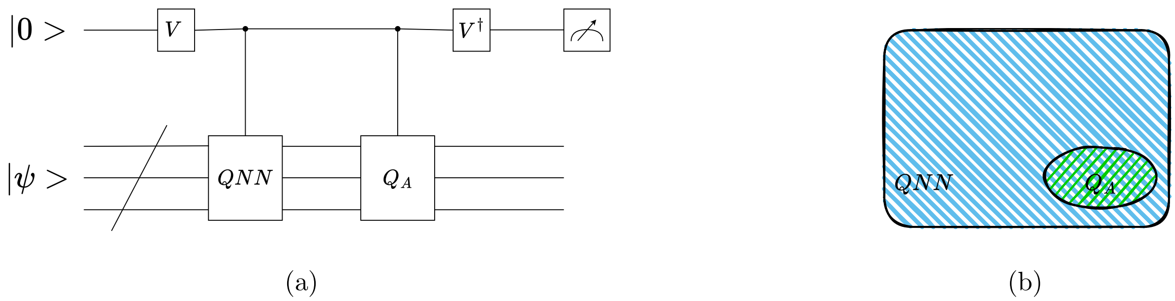


Figure 2: **Proposed quantum adapter for fine-tuning a quantum neural network (QNN).** a) A linear combination of unitaries (LCU) circuit for implementing an adapter circuit Q_A in conjunction with a QNN layer. b) Cartoon of dynamical Lie algebra (DLA) difference between a generic QNN and Q_A.

using the quantum circuits themselves as adaptors. In doing so, one could either extend the adaptors to the fully quantum realm - using restricted circuits to fine-tune to quantum datasets which originally require large or complex circuits for performance, or potentially perform the adaptation of classical models directly on quantum hardware using circuits. Specifically, compound circuits have been used in QML literature for their subspace preserving structure and low-dimensional dynamical Lie algebra (DLA) enabling the avoidance of barren plateaus (BPs) [8, 9, 10]. To bring adaptors into the quantum domain, we propose to use the well known a linear combination of unitaries[2] (LCU) framework which has been well used in quantum simulation [11] and recently adapted to quantum machine learning [12]. We depict such an approach using the fully quantum compound circuits as Adapters as shown in Fig. 2a. One potential direction is the following. First, one could train a low-dimensional DLA circuit (BP avoiding, classically simulatable), e.g. the fixed Hamming-weight compound circuit, classically for a given downstream task, and then combined with a more expressive (high DLA) quantum neural network (QNN as show in Fig. 2b), acting as a warm start, $(\alpha_1 U_{\mathsf{QNN}} + \alpha_2 U_{Q_A})\,|\psi\rangle$: Furthermore, this framework could be used to avoid or mitigate barren plateaus [12] or to increase the difficulty of classically simulating low-DLA models.

References

[1] Brown et al. Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33:1877–1901, 2020.

[2] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *International conference on machine learning*, pages 2790–2799. PMLR, 2019.

[3] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

[4] Iordanis Kerenidis and Anupam Prakash. Quantum machine learning with subspace states. *arXiv preprint arXiv:2202.00054*, 2022.

[5] Jonas Landman, Natansh Mathur, Yun Yvonna Li, Martin Strahm, Skander Kazdaghli, Anupam Prakash, and Iordanis Kerenidis. Quantum Methods for Neural Networks and Application to Medical Image Classification. *Quantum*, 6:881, December 2022.

[6] El Amine Cherrat, Snehal Raj, Iordanis Kerenidis, Abhishek Shekhar, Ben Wood, Jon Dee, Shouvanik Chakrabarti, Richard Chen, Dylan Herman, Shaohan Hu, Pierre Minssen, Ruslan Shaydulin, Yue Sun, Romina Yalovetzky, and Marco Pistoia. Quantum Deep Hedging. *Quantum*, 7:1191, November 2023.

[7] Li Deng. The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.

[8] Jarrod R McClean, Sergio Boixo, Vadim N Smelyanskiy, Ryan Babbush, and Hartmut Neven. Barren plateaus in quantum neural network training landscapes. *Nature communications*, 9(1):4812, 2018.

[9] Enrico Fontana, Dylan Herman, Shouvanik Chakrabarti, Niraj Kumar, Romina Yalovetzky, Jamie Heredge, Shree Hari Sureshbabu, and Marco Pistoia. The adjoint is all you need: Characterizing barren plateaus in quantum ans\" atze. *arXiv preprint arXiv:2309.07902*, 2023.

[10] Michael Ragone, Bojko N Bakalov, Frédéric Sauvage, Alexander F Kemper, Carlos Ortiz Marrero, Martin Larocca, and M Cerezo. A unified theory of barren plateaus for deep parametrized quantum circuits. *arXiv preprint arXiv:2309.09342*, 2023.

[11] Andrew M Childs and Nathan Wiebe. Hamiltonian simulation using linear combinations of unitary operations. *arXiv preprint arXiv:1202.5822*, 2012.

[12] Jamie Heredge, Maxwell West, Lloyd Hollenberg, and Martin Sevior. Non-unitary quantum machine learning. *arXiv preprint arXiv:2405.17388*, 2024.

[2] Given a sequence of unitaries, $\{U_k\}_k$ and coefficients $\{\alpha_k\}_k$, we can apply the non-unitary operation $\sum_k \alpha_k U_k$ in a probabilistic fashion using postselection on ancilliary qubits.