Measurements With A Quantum Vision Transformer: A Naive Approach

Dominic Pasquali[1,2], *Michele* Grossi[2], and *Sofia* Vallecorsa[2]

[1]Physics Department, University of California Santa Cruz, Santa Cruz, CA 95064
[2]CERN, 1211 Geneva 23, Switzerland

Abstract. In mainstream machine learning, transformers are gaining widespread usage. As Vision Transformers rise in popularity in computer vision, they now aim to tackle a wide variety of machine learning applications. In particular, transformers for High Energy Physics (HEP) experiments continue to be investigated for tasks including jet tagging, particle reconstruction, and pile-up mitigation.

An improved Quantum Vision Transformer (QViT) with a quantum-enhanced self-attention mechanism is introduced and discussed. A shallow circuit is proposed for each component of self-attention to leverage current Noisy Intermediate Scale Quantum (NISQ) devices. Variations of the hybrid architecture/model are explored and analyzed.

The results demonstrate a successful proof of concept for the QViT, and establish a competitive performance benchmark for the proposed design and implementation. The findings also provide strong motivation to experiment with different architectures, hyperparameters, and datasets, setting the stage for implementation in HEP environments where transformers are increasingly used in state of the art machine learning solutions.

1 Introduction

Much of the rise in performance and power behind modern day machine learning revolves around transformers and by extension self-attention. Made famous by the 2017 paper *Attention Is All You Need* [1], transformers and self-attention have revolutionized Natural Language Processing as such architectures have formed the bedrock for GPT-3 [2], LLaMa [3], and others [4–10]. The idea of Vision Transformers was introduced in the 2020 paper *An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale* [11] which applied transformers to images. With the onset of Vision Transformers, transformers can be used for computer vision tasks such as classification [11] and object detection [12].

At a very high level, self-attention takes inputs (in tensor form) and multiplies those inputs by three separate learned matrices to generate unique "q" (query), "k" (key), and "v" (value) matrices for each input[1]. Query and key matrix elements are multiplied together and then passed to a softmax operation to create the attention matrix, and the attention matrix is then multiplied by the value matrix elements. All of the resulting outputs are then concatenated

[1] The terms query, key, and value come from the days of retrieval systems when a search engine would map the Query (e.g. the text in a search bar) against the Keys (e.g. given descriptors like video title, description, etc...) of indexed items, and then the search engine would return the best matched items (Values) to the user.

and output together. The query matrix is then shared and used in the same operation with every input generated key and value matrix that the vision transformer takes in [1]; it is this sharing of the query matrix between the other separate key and query matrices from which self-attention gets its power and performance.

As Transformers expand to different variations and applications, such architecture has inspired and motivated the development of Quantum-enhanced Vision Transformers (QViT). The QViT replaces classical self-attention with a version of quantum self-attention, and such QViT design resolves issues with past implementations and whose performance is compared against a classical ViT.

2 Previous Quantum Self-Attention Architectures

Previous works attempted to implement quantum (self) attention with various levels of performance [5, 13]. However such implementations leave room for improvements which are outlined below and set the stage for a novel QViT implementation.

Attempts at applying attention to quantum vision transformer networks include [13], however while such work employs attention it lacks self-attention. Such self-attention is essential for comparing encoded inputs across multiple other inputs, however [13] makes no mention of the encoded query, key, or value parts of the input, and does not compute the query and the key parts of the input together.

Another drawback of previous works for quantum (self) attention in vision transformers is that they don't consider the sizes for the inputs of the attention mechanism to be an adjustable hyperparameter. This hyperparameter allows for the user to tune the projected space that is input to the (self) attention mechanism, and such tuning can be done on an case-by-case basis for purposes of biasing an input to have more of an effect than another input. For example it may be advantageous to give the query vector more influence than the key vector when they are combined, and likewise it might be more advantageous to give the attention vector more weight than the value vector. With past approaches such inputs were all treated equally and gave no room or operation to do so otherwise. Classical vision transformers don't typically consider this flexibility either as the projected sizes for their query, key, and value components are traditionally a function of the input, however given that the self-attention mechanism is now being dealt in a quantum mechanical manner, it's worth considering and fine-tuning the flexibility for the size of the inputs to the quantum mechanical self-attention mechanism.

Also the approach proposed in this work is hardware agnostic as one could tune the inputs to the quantum circuit to fit the available input size to the applied quantum hardware of interest. If using a simulated approach this ability to tune the size of the inputs as well as their proportions allow for effective use of the simulated quantum circuit, since too large of an input would slow down the simulation while too small of an input would not capture the full effects of the input in a reduced dimension. In contrast to the above motivation for a quantum vision transformer, the paper [13] passed the full width of the input to their quantum circuits without regard to current, future, or available quantum circuitry, real or simulated.

Other work addressing the quantum self-attention in a transformer architecture includes [5] which more closely resembles a true transformer structure with some interesting features. [5] divides the inputs into three parts and then entangles the query and key inputs via a CNOT gate before the self-attention is conducted; combining the query and key inputs constitutes creating some type of attention between the query and key. However the query, key, and value matrices should remain separate and independent until after they've been combined for self-attention.

The technique proposed in [5] also indicates that at the end of their quantum circuit the previous inputs are summed with the product of the proposed attention and value steps. This incorporation of the input to the output is more reminiscent of a UNet [14] than a transformer in which such an operation is absent.

[5] indicates that the query and key components of self-attention are coupled together, and then after measurements are taken for their respective values the query and keys are subtracted from one another and then put through a softmax (of sorts) of these differences. This approach creates noise in the output when evaluating the difference of queries being compared against keys from the same input, thus a difference wouldn't yield how much self-attention an input should give to itself since the items being compared are already entangled from previous operations.

Despite the aforementioned works, the proposed approach resolves the above points with its own novel implementation for quantum-enhanced self-attention vision transformers.

3 Methodology

The most simple approach to create a novel QViT simultaneously addressing the issues above while competing with a ViT [11] would be to replace the dot-product attention inside the ViT with a trained variational quantum circuit.

The following subsections will discuss how Quantum Self-Attention was implemented within the ViT. Variations and their motivations will be addressed in later sections. Unless otherwise noted, the QViT has identical default settings and parameters as the classical ViT. The performance of the QViT will then be benchmarked against its classical ViT counterpart.

3.1 Quantum Self-Attention

This work focused on developing Quantum Self-Attention within the ViT. However since self-attention within the ViT relies on dot-product attention, the attention is replaced with a classical-quantum hybrid architecture (Figure 1) which when inserted into the ViT architecture becomes Quantum Self-Attention within the ViT.

To begin, assume that the tensors for the query, key, and value for attention are given with their encoding masks already applied and the scaling factor is applied to the query and key.

3.2 Creating The Attention Mask

To create the quantum attention mask, the query and key pass through a classical linear layer to project their respective size down to five nodes. The resulting query and key vectors are concatenated together to form a single vector which is then passed to a hyperbolic tangent activation function, whose output is multiplied by a scalar, α. The vector is then passed to the quantum circuit where it undergoes an R_y rotational encoding step for 10 wires, after which the quantum circuit applies four iterations of PennyLane's StronglyEntanglingLayers (with a single iteration consisting of single generalized qubit rotations on each wire and CNOT gates connecting pairs of wires across the entire circuit) [15]. The quantum circuit measures the expectation value of the Pauli Z matrix for each of the wires of the circuit. The resulting measurements get sent through a softmax layer which generates the attention vector. These steps can be seen in the top half of Figure 1.

3.3 Generating The Output

Creating the output with the attention mask (vector) and value is nearly the same process as described in Section 3.2, with the exception that there is no softmax operation.

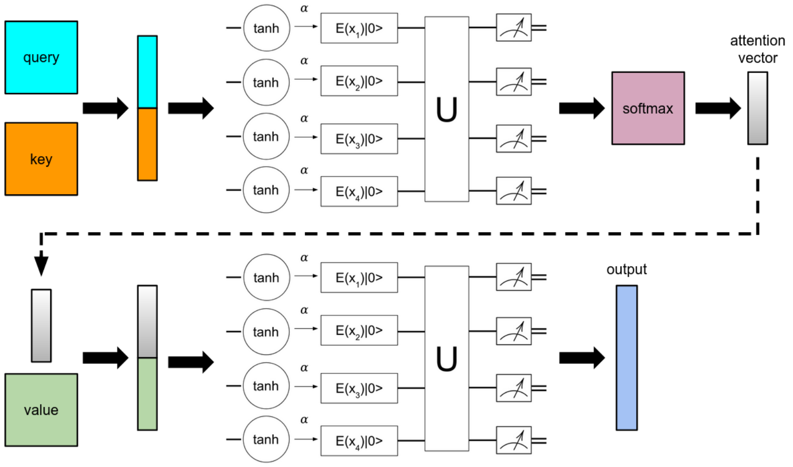

Figure 1. This figure depicts the flow of information through quantum attention, though the circuit parameters can be found in Section 3.3. The query and key matrices are flattened into vectors. The vectors are then concatenated together and are passed through a hyperbolic tangent activation function, whose outputs are then multiplied by a scalar, α. The inputs enter the quantum circuit and are encoded via an encoding method, $E(x_i)$ (where x_i is an input to the quantum circuit. Note that the illustration stops at x_4 for brevity). The above process is repeated for the value matrix whose resulting vector is concatenated with the attention vector, and the result after the imposed variational quantum circuit forms the output for Quantum Self-Attention.

To create the output, the attention vector and the value matrix pass through a classical linear layer to project their respective sizes down to five nodes. Then the attention and value are concatenated together to form a single vector which is then passed to a hyperbolic tangent activation function, whose output is multiplied by a scalar, α. The vector is then passed to the quantum circuit where it undergoes an R_y rotational encoding step for 10 wires, after which the quantum circuit then applies four layers of PennyLane's StronglyEntanglingLayers [15]. The quantum circuit then measures the expectation value of the Pauli Z matrix for each of the wires of the circuit. The resulting measurements form the output of the entire attention mechanism. The above creation of the aforementioned output can be seen in the bottom half of Figure 1.

After the output is generated, it passes through a classical linear layer to project back into the appropriate number of dimensions to continue being passed to the rest of the QViT.

4 Software, Data, Models & Experiment

4.1 Software and Data

The MNIST digits dataset [16] is used as a classification task to compare the performance of the ViT and proposed QViT.

PyTorch [17] was used to supply the MNIST dataset which was brought in via the EMNIST dataset [18] with the MNIST split. The data was then normalized with a mean of 0.1307 and a standard deviation of 0.3081 over the only channel. The training and test datasets were defined by their respective default files as called by PyTorch.

The ViT was taken from the PyTorch implementation of the Vision Transformer[19]. The defined architecture variables were taken to be the following: image size of 28 × 28 pixels; patch size of seven pixels; one layer; two heads; a latent (hidden) dimension of 128; multilayer perceptron dimension of 128; one channel; and 10 classes.

ADAM was the chosen optimizer and Cross Entropy was the loss function.

PennyLane [20] was used for simulating and training the quantum circuits. All functions for encoding data into the quantum circuits and constructing the quantum circuit learned parameters were done with the PennyLane API.

4.2 Models

Constructing the QViT begins with the ViT. Within the ViT, the dot-product attention is exchanged with the Quantum Self-Attention mechanism described in Section 3 and illustrated in Figure 1. After the output is generated, a linear layer is used to increase the output dimensions to the shape the output would have been if classical self-attention was used instead of Quantum Self-Attention.

α was chosen with regards to the rotational encoding step, R_y. Consider that since the classical information passes through a hyperbolic tangent activation function before reaching α, then all of the information will be projected to span from -1 to 1. Therefore one value for α was chosen to be $\alpha = \frac{\pi}{2}$; this is an arbitrary choice, with a careful eye for the projected space of the inputs to be not too big but not too small either, as the projected data will span from $-\frac{\pi}{2}$ to $\frac{\pi}{2}$ when passed to R_y. The other value for α was chosen to be $\alpha = \pi - 0.01$ to ensure that all encoded data remained unique after being passed to the R_y rotational encoding step and did not overlap at $\alpha = \pm\pi$.

There is a question of dimensionality when the output of the softmax operation should or should not match the dimensions of the value vector. In some models (as addressed below) the attention vector passes through a linear layer to scale up to the equivalent shape of the value vector, and then both the attention vector and the value vector are scaled down with the help of a linear layer to five nodes, after which the resulting vectors are concatenated together and passed to the quantum circuit. In other models the attention vector passes through a linear layer that scales the vector down to five nodes, and then a separate linear layer scales the value vector down to five nodes. After this the vectors are concatenated together and passed through the quantum circuit. The scaling tests to see if the initial projected dimensional space of the attention and value vectors change the performance when they're scaled down; this will be discussed further in Section 5.

From the above, four models are explored with the following titles and parameters:

- *QViT With* $\frac{\pi}{2}$ *Encoding*
  - the attention vector is rescaled to the same shape as the value vector with the help of a classical linear layer, and $\alpha = \frac{\pi}{2}$
- *QViT With* $\frac{\pi}{2}$ *Encoding And No Rescaling*
  - the attention vector is scaled down with the help of a classical linear layer, and $\alpha = \frac{\pi}{2}$
- *QViT With* $\pi - 0.01$ *Encoding*
  - the attention vector is rescaled to the same shape as the value vector with the help of a classical linear layer, and $\alpha = \pi - 0.01$
- *QViT With* $\pi - 0.01$ *Encoding And No Rescaling*
  - the attention vector is scaled down with the help of a classical linear layer, and $\alpha = \pi - 0.01$

5 Results And Discussion

As seen in Figure 2, when comparing the test and training statistics of the QViT and the (classical) ViT, it is found that the QViT and classical ViT are of comparable performance. It can be argued that the QViT gives a slight boost in performance when examining the test accuracy, however upon closer inspection of the y-axis' scale it can be immediately rendered that such differences do not amount to any statistical significance.

A similar result to the above analysis can also be found with the test loss. While the test loss for the QViT models on average outperform the classical ViT, such differences are minute when examining their scale via the y-axis.

The training loss tells the above story in perhaps the most stark terms; by inspection the training loss yields no statistical difference between the classical ViT and the QViT.

What is interesting is that varying α or the projected dimensions for the attention mask do not change the behavior of the QViT. It may very well be that the imposed quantum circuit allows for the resulting attention vector to be in a sufficiently separate dimensional space such that any extra alteration yields no advantage over the ViT or difference between the presented projected dimensions.

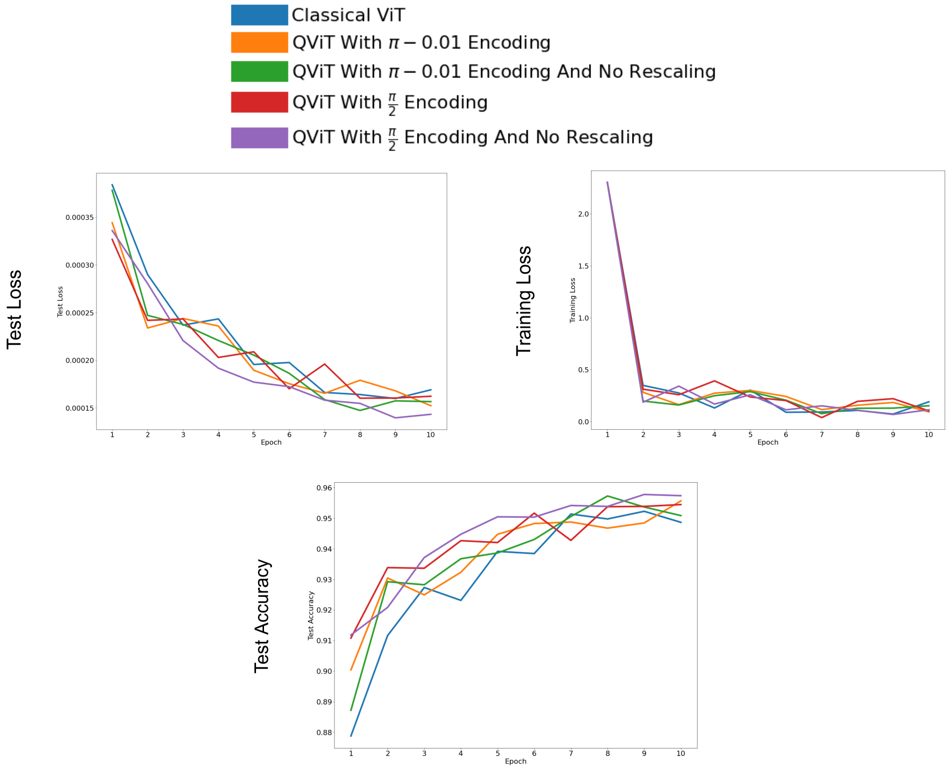

Figure 2. Plots depicting the training loss, test loss, and the test accuracy. Given the scale of the y-axis, the plots indicate that the proposed QViT is comparable in performance with the classical ViT.

6 Remarks

As inspired by modern day Vision Transformers, this work proposes a Quantum-enhanced Vision Transformer in which a novel approach to Quantum Self-Attention is introduced and implemented. The performance of the QViT is benchmarked with the MNIST digits dataset and whose performance is compared against a classical ViT. Careful steps are taken to ensure that the encoding of information from classical to quantum maximizes the possible encoding space while also preventing an overlap in data encoding, as discussed in Section 4.2. Upon examining the models and their resulting statistics after training and testing on the MNIST dataset, it is found that a naive and simple implementation of a QViT is of comparable and competitive performance with the classical ViT.

Future work includes changing the encoding method of classical to quantum information, as well as varying the quantum circuit architecture. Also changing the ratio of the information contributed by the query and key vectors, and the value and attention vectors in their respective operations may very well alter the resulting performance.

Acknowledgements

The authors would like to thank Eliana Stefani for helpful feedback. The authors are thankful to CERN Openlab for supplying dedicated compute nodes for this project.

References

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, I. Polosukhin, *Attention Is All You Need* (2023), arXiv:1706.03762

[2] T.B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell et al., *Language Models are Few-Shot Learners* (2020), arXiv:2005.14165

[3] *LLaMA: Open and Efficient Foundation Language Models, author=Hugo Touvron and Thibaut Lavril and Gautier Izacard and Xavier Martinet and Marie-Anne Lachaux and Timothée Lacroix and Baptiste Rozière and Naman Goyal and Eric Hambro and Faisal Azhar and Aurelien Rodriguez and Armand Joulin and Edouard Grave and Guillaume Lample* (2023), arXiv:2302.13971

[4] J. Devlin, M.W. Chang, K. Lee, K. Toutanova, *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding* (2019), arXiv:1810.04805

[5] G. Li, X. Zhao, X. Wang, *Quantum Self-Attention Neural Networks for Text Classification* (2023), arXiv:2205.05625

[6] D.Q. Nguyen, T. Vu, A. Tuan Nguyen, *BERTweet: A pre-trained language model for English Tweets*, in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, edited by Q. Liu, D. Schlangen (Association for Computational Linguistics, Online, 2020), pp. 9–14, `https://aclanthology.org/2020.emnlp-demos.2`

[7] N.L. Tran, D.M. Le, D.Q. Nguyen, *BARTpho: Pre-trained Sequence-to-Sequence Models for Vietnamese* (2022), arXiv:2109.09701

[8] M.K. Eddine, A.J.P. Tixier, M. Vazirgiannis, *BARThez: a Skilled Pretrained French Sequence-to-Sequence Model* (2021), arXiv:2010.12321

[9] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, L. Zettlemoyer, *BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension* (2019), arXiv:1910.13461

[10] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, R. Soricut, *ALBERT: A Lite BERT for Self-supervised Learning of Language Representations* (2020), arXiv:1909.11942

[11] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly et al., *An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale* (2021), arXiv:2010.11929

[12] Y. Li, H. Mao, R. Girshick, K. He, *Exploring Plain Vision Transformer Backbones for Object Detection* (2022), arXiv:2203.16527

[13] E.A. Cherrat, I. Kerenidis, N. Mathur, J. Landman, M. Strahm, Y.Y. Li, *Quantum Vision Transformers* (2022), arXiv:2209.08167

[14] O. Ronneberger, P. Fischer, T. Brox, *U-Net: Convolutional Networks for Biomedical Image Segmentation* (2015), arXiv:1505.04597

[15] Xanadu, *qml.StronglyEntanglingLayers* (2022), `https://docs.pennylane.ai/en/stable/code/api/pennylane.StronglyEntanglingLayers.html`

[16] LeCun, Yann and Cortes, Corinna and Burges, Christopher J.C. (2010), `http://yann.lecun.com/exdb/mnist/`

[17] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga et al., in *Advances in Neural Information Processing Systems 32*, edited by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché Buc, E. Fox, R. Garnett (Curran Associates, Inc., 2019), pp. 8024–8035, `http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf`

[18] G. Cohen, S. Afshar, J. Tapson, A. van Schaik, *EMNIST: an extension of MNIST to handwritten letters* (2017), arXiv:1702.05373

[19] PyTorch, *Vision Transformer* (2022), `https://pytorch.org/vision/master/models/vision_transformer.html`

[20] V. Bergholm, J. Izaac, M. Schuld, C. Gogolin, S. Ahmed, V. Ajith, M.S. Alam, G. Alonso-Linaje, B. AkashNarayanan, A. Asadi et al., *PennyLane: Automatic differentiation of hybrid quantum-classical computations* (2022), arXiv:1811.04968