Your Original and Relevant Course Project Title

Quantum Algorithms — Course paper

Chenyu Shi
s3500063@umail.leidenuniv.nl
LIACS, Leiden University
Leiden, Netherlands

Shupei Li
s3430863@umail.leidenuniv.nl
LIACS, Leiden University
Leiden, Netherlands

ABSTRACT

KEYWORDS

keyword1, keyword2, keyword3, social network analysis, network science

ACM Reference Format:
Chenyu Shi and Shupei Li. 2022. Your Original and Relevant Course Project Title: Quantum Algorithms — Course paper. In *Proceedings of Quantum Algorithms Course 2022 (QA '22).* ACM, New York, NY, USA, 3 pages.

1 INTRODUCTION

In this paper, we consider the classification problem from the perspective of quantum algorithms. Classification is an important data analysis task in machine learning field, where a lot of algorithms designed for classical computers have been developed. However, the volume and the variety of data have been growing rapidly in recent several decades. To handle the high-dimensional data with complicated features, we need more advanced techiques and more powerful computers. Developments in quantum computing provide a possible solution for the big data challenge — introducing quantum algorithms into machine learning to obtain the quantum speedups. In theory, quantum machine learning algorithms outperform classical algorithms by leveraging quantum phenomena (e.g. entanglement and coherence) to reduce the time complexity of information processing and analysis [1]. Researchers have successfully designed the quantum version of several machine learning algorithms and implemented them on quantum computers, including both supervised learning and unsupervised learning. For example, Lu and Braunstein proposed the quantum decision tree algorithm based on the quantum fidelity metric [5]. Lloyd et al. performed k-means algorithm with the quantum version of Lloyd's algorithm [4]. There are also attempts to generalize the deep neural networks to quantum computers [2]. In our project, we focus on using the quantum version of support vector machines (SVMs) to solve the supervised classification problem.

Classical SVMs are widely used in image classification, text mining, etc. The SVM constructs nonlinear decision boundaries by first transforming the feature space and then learning linear boundaries on the transformed space from data. To develop quantum SVMs, we need to map the data into a quantum space and create the corresponding quantum circuits. Havlíček et al. [3] proposed two strategies to build the quantum SVM model. One strategy is optimizing a variational quantum circuit that is also known as the parametrized quantum circuit. The other is estimating the quantum kernel. Considering the limitations of quantem kernel estimator, we investigate the variational quantum circuit in experiments.

There is a problem in the construction of variational quantum circuits — the limited number of qubits. Although quantum computers have enormous potential of fast computing, the Noisy Intermediate-Scale Quantum (NISQ) devices suffer from non-negligible noise and limited qubit counts [7]. Therefore, it is meaningful to decompose a large quantum circuit into smaller ones that can be deployed on NISQ devices while maintaining the comparable performance to the original circuit. In the project, we try to simulate the large variational quantum classifier with multiple small quantum circuits. We also explore the generalization ability of our circuit decomposition scheme.

The remainder of the paper is structured as follows. Section 2 reviews previous work related to variation quantum algorithms and quantum circuit cutting. Section 3 explains the variational quantum classifier and quantum circuit cutting strategy in detail. Section 4 introduces datasets we use in the project. We present our experimental set-up and results in Section 5. The paper ends with a conclusion section.

2 RELATED WORK

Our project is inspired by the variation quantum algorithms as well as the quantum circuit cutting. We provide an overview on related work in both research directions.

2.1 Variation Quantum Algorithms

2.2 Quantum Circuit Cutting

3 APPROACH

3.1 Variational Quantum Classifier

Variational Quantum Classifier(VQC) is a kind of parameterized quantum circuit used in classification tasks [3]. Conventional machine learning methods, such as SVM and neural networks, often take advantage of a parameterized model as classifiers. During training process, these methods use a target function or loss function to measure the difference between prediction results and ground true labels and update the parameters in the model to improve its performance. After several training epochs on adequate data, these machine learning methods can achieve great performance on prediction on unseen data as well. VQC works in a similar way, whereas applying a quantum circuit as its parameterized model. The abstract model of VQC is shown in Figure 1, including data encoding (or feature mapping) $U_{enc}(x)$, variational model $U_{train}(\theta)$, measurement, and parameter updating(usually via gradient decent). In the following subsection 3.1.1 to 3.1.4, we will introduce these processing steps in detail. And then, in subsection 3.1.5, the overall working principle of our VQC circuit will be discussed.

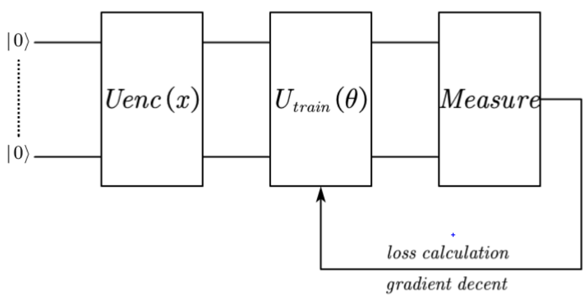

Figure 1: Abstract model of VQC: A Variational Quantum Classifier contains these processing steps: data encoding, varational model, measurement and parameter updating.

3.1.1 *Data Encoding.*

As what we have discussed above, VQC attempts to apply parameterized quantum circuit as its learning model. However, nowadays most of data is stored in classical computers which quantum circuit cannot directly access to. Hence, it's necessary to encode classical data into quantum state that can be preocessed by quantum circuit at first.

There are many basic methods to achieve this encoding preocess, such as basis encoding, amplitude encoding and angle encoding. Besides, several higher order encoding methods which combines different basic encoding methods are studied as well. In this paper, our proposal data encoding method is xxxx, one of the most popular higher order encoding methods. The specific quantum circuit is shown in Figure 2.

3.1.2 *Variational Model.*

Variational model is a quantum circuit with trainalbe parameters. During the training process, these parameters will be updated to improve the model's performance. There are various kinds of variational model can be applied in VQC. In this paper, the circuit of our variational model is shown in Figure 3.

3.1.3 *Measurement.*

When training most of machine learning models, we need to compare the prediction results and the labels to update parameters, which works the same way in VQC. Nevertheless, after data encoding and variational model, the results are still in quantum states. Hence, we need to extract prediction results from these quantum states. This process is accomplished by measurement. We simply measure all qubits in the circuit and run the circuit for many times. Then we can obtain a distribution of these possible results.

In binary classification tasks, a common way to extract prediction results is to converge the result states distribution into a binary distribution of odd parity states and even parity states. In other words, we consider the parity distribution as the prediction probability of two classes. For example, we run a 2-qubit circuit for 100

Figure 2: Data encoding circuit: Description

Figure 3: Variational model circuit: Description

times and obtain the measurement result which is [00:21, 01:33, 10:28, 11:18], so the prediction probability for each class shall be [0.39, 0.61]. Therefore, the second class will be predicted as the final result.

3.1.4 *Parameter Updating.*

Parameter updating process is the most significant part of machine learning, which includes loss calculation and gradient decent. Loss calculation is the same as the conventional machine learning. When we obtain the predcition probability distribution of each class, we compare the difference between it and ground true label through a loss function. Despite popularity of cross entropy loss in classification tasks, we still choose MAE loss in our model.

In the conventional machine learning, such as neural network, gradient decent based method is famous and efficient to update parameters in the training process. However, it's hard to calculate gradient as in the conventional machine learning due to the fact that our model is based on the quantum circuit. Hence, [6] proposed a method using difference to subsitute real gradients. In detail, for each parameter θ, we run the circuit and calculate the loss value based on the prediction probability distribution results for parameter $\theta + s$ and $\theta - s$, namely $L(\theta + s)$ and $L(\theta - s)$. Then using the following formula to calculate the gradient:

$$gradient(\theta) = \frac{L(\theta + s) - L(\theta - s)}{2s}$$

where s is usually chosen as a small real positive number. Then this proximate gradient is used to update model's parameters just as the same way in the conventional machine learning.

3.1.5 Overall Analysis.

In this subsection, we make a overal analysis on our VQC and the circuit theoretically to illustrate why this process makes a good classifier.

According to the analysis in the previous subsections, our ansatz can be written in the mathematical form:

$$U(x, \theta) = U_{train}(\theta) U_{enc}(x)$$

Before the measurement, the following state is prepared:

$$|\psi(x, \theta)\rangle = U(x, \theta) \left|0^n\right\rangle$$

And then we measure the observable $Z^{\otimes n}$ on the above state to implement the function:

$$f_\theta(x) = \langle \psi(x, \theta) | Z^{\otimes n} | \psi(x, \theta) \rangle$$

This function is actually the expectation of the observing result distribution on the computational basis of even parity states and odd parity states, when giving even parity states value 1 (represent class 0) and odd parity states value -1 (represent class 1). To be more specific, let us use a 2-qubit example to illustrate it.

Suppose we have a general 2-qubit state $|\phi\rangle = a|00\rangle + b|01\rangle + c|10\rangle + d|11\rangle$ as $|\psi(x, \theta)\rangle$. Feed this state into the above function, and the result is actually computed as:

$$\begin{aligned} f_\theta(x) &= \langle \phi | Z^{\otimes n} | \phi \rangle \\ &= \begin{pmatrix} a & b & c & d \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} \\ &= a^2 - b^2 - c^2 + d^2 \end{aligned}$$

When we observe $|\phi\rangle$ on the computational basis, the final probability to obtain odd and even parity state shall be:

$$p(even) = p(|00\rangle) + p(|11\rangle) = a^2 + d^2$$

$$p(odd) = p(|01\rangle) + p(|10\rangle) = b^2 + c^2$$

Remember we have given even parity states value 1 and odd parity states value -1 to make it a classifier, so the expectation value of this probability distribution will be:

$$E = 1 * p(even) + (-1) * p(odd) = a^2 - b^2 - c^2 + d^2$$

It's actually the same result compared to $f_\theta(x)$. However, in real cases, we couldn't

4 DATA

5 EXPERIMENTS

6 CONCLUSION

ACKNOWLEDGMENTS

REFERENCES

[1] Jacob Biamonte, Peter Wittek, Nicola Pancotti, Patrick Rebentrost, Nathan Wiebe, and Seth Lloyd. 2017. Quantum machine learning. *Nature* 549, 7671 (Sep 2017), 195–202.

[2] Edward Farhi and Hartmut Neven. 2018. Classification with Quantum Neural Networks on Near Term Processors. (2018). https://arxiv.org/abs/1802.06002

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

[4] Seth Lloyd, Masoud Mohseni, and Patrick Rebentrost. 2013. Quantum algorithms for supervised and unsupervised machine learning. (2013). https://arxiv.org/abs/1307.0411

[5] Songfeng Lu and Samuel L. Braunstein. 2013. Quantum decision tree classifier. *Quantum Information Processing* 13 (2013), 757 – 770.

[6] Maria Schuld, Alex Bocharov, Krysta M Svore, and Nathan Wiebe. 2020. Circuit-centric quantum classifiers. *Physical Review A* 101, 3 (2020), 032308.

[7] Wei Tang, Teague Tomesh, Martin Suchara, Jeffrey Larson, and Margaret Martonosi. 2021. CutQC: using small Quantum computers for large Quantum circuit evaluations. In *Proceedings of the 26th ACM International Conference on Architectural Support for Programming Languages and Operating Systems.* ACM.