CLASSICAL-TO-QUANTUM TRANSFER LEARNING FOR SPOKEN COMMAND RECOGNITION BASED ON QUANTUM NEURAL NETWORKS

Jun Qi[1], Javier Tejedor[2]

[1]Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, USA
[2] Escuela Politecnica Superior, Universidad San Pablo-CEU, CEU Universities, Madrid, Spain

ABSTRACT

This work investigates an extension of transfer learning applied in machine learning algorithms to the emerging hybrid end-to-end quantum neural network (QNN) for spoken command recognition (SCR). Our QNN-based SCR system is composed of classical and quantum components: (1) the classical part mainly relies on a 1D convolutional neural network (CNN) to extract speech features; (2) the quantum part is built upon the variational quantum circuit with a few learnable parameters. Since it is inefficient to train the hybrid end-to-end QNN from scratch on a noisy intermediate-scale quantum (NISQ) device, we put forth a hybrid transfer learning algorithm that allows a pre-trained classical network to be transferred to the classical part of the hybrid QNN model. The pre-trained classical network is further modified and augmented through jointly fine-tuning with a variational quantum circuit (VQC). The hybrid transfer learning methodology is particularly attractive for the task of QNN-based SCR because low-dimensional classical features are expected to be encoded into quantum states. We assess the hybrid transfer learning algorithm applied to the hybrid classical-quantum QNN for SCR on the Google speech command dataset, and our classical simulation results suggest that the hybrid transfer learning can boost our baseline performance on the SCR task.

Index Terms— Quantum neural network, spoken command recognition, variational quantum circuit, transfer learning

1. INTRODUCTION

The state-of-the-art baseline system of spoken command recognition (SCR) [1, 2, 3] is built upon the advancement of deep learning (DL) technology [4]. The DL technologies highly rely on two important aspects: (1) powerful computational resources arising from the graphical processing unit (GPU); (2) the availability of access to a large amount of labeled or unlabelled training data [5, 6]. Despite the rapid empirical progress of the SCR systems, deep learning models are becoming computational expensive and, now that Moore's law is faltering [8], it is necessary to contemplate a future technology to further deal with a huge amount of speech data. However, new exciting possibilities are opening up due to the imminent advent of quantum computing devices that directly exploit the laws of quantum mechanics to evade the technological limits of classical computation [9].

The exploitation of quantum computing machines to carry out quantum machine learning (QML) [10] is still in its initial exploratory stage. In particular, a quantum neural network (QNN) [11], which is capable of carrying out a universal quantum computation, attracts much attention in the domain of QML because it can be seen as a quantum analog of the classical deep neural network (DNN). Hence, the QNN model allows the algorithm of back-propagation to

train the model parameters of the QNN. However, in the age of noisy intermediate-scale quantum (NISQ) devices [12], it is necessary to simulate quantum experiments with noisy quantum circuits that may degrade the baseline performance. A compromised QNN, namely variational quantum circuit (VQC), has been proposed to overcome the influence of quantum noise in the QNN. The advantages of the VQC in the QNN arise from many aspects: (1) a VQC is a quantum circuit with adjustable parameters that are optimized according to a predefined metric; (2) VQC is flexibly placed on a NISQ machine and can be resilient to the quantum noisy effects on quantum circuits.

Prominent examples in the hybrid QNN for machine learning tasks include quantum reinforcement learning [13], quantum image processing [14], and quantum circuit learning (QCL) [15], where the VQC model plays an important role as a quantum component. Particularly as for the application of QNNs for SCR, our pioneering work [16] investigates the use of quantum convolutional neural networks to extract quantum speech features as the input to state-of-the-art classical deep neural networks. However, the work [16] does not employ the QNN as an end-to-end SCR model, and it is still unknown about the performance of QNNs for SCR. Hence, in this work, we attempt to build a hybrid classical-quantum QNN model for SCR. In particular, we make use of a hybrid transfer learning approach to speed up the end-to-end training framework based on the QNN model.

Classical transfer learning [17, 18] is a typical example of machine learning that has been originally inspired by biological intelligence, and it originates from the knowledge acquired in a specific context which can be transferred to a different area. For example, when we learn a foreign language, we always make use of our previous linguistic knowledge to speed up the learning rate [19]. This general idea has been successfully applied to artificial intelligence and machine learning domains. It has been shown that in many situations, instead of training a full network from scratch, it should be more efficient to start from a pre-trained network so that only some model components need further fine-tuning for a particular task of interest. The transfer learning methodology is very related to the training of a hybrid classical-quantum QNN, where the classical component can be pre-trained in another generic model and directly transferred to the hybrid QNN.

This work aims to investigate a hybrid classical-to-quantum transfer learning paradigm in the context of a QNN-based SCR system. Our baseline system, namely CNN-DNN, consists of two core components: (1) convolutional neural network (CNN) for speech feature extraction; (2) DNN for recognizing commands. In this work, we first build a hybrid classical-quantum model, namely CNN-QNN, where the CNN is still kept for feature extraction and the DNN is replaced by the QNN model. In doing so, a classical CNN-DNN becomes a hybrid classical-quantum CNN-QNN. Al-

though the hybrid model follows an end-to-end learning pipeline, the frequent data conversion between classical and quantum states greatly slows down the computing efficiency in the training stage. Moreover, the current NISQ device admits only a small number of qubits for the QNN so the representation power of the CNN-QNN may become quite limited. Thus, a new training strategy needs to be proposed. Here, we put forth a novel hybrid transfer learning strategy. In more detail, given a well-trained classical CNN-DNN, the CNN model is taken to initialize the CNN component of the CNN-QNN and several additional training steps are conducted to fine-tune the parameters of the QNN.

The rest of the paper is organized as follows: Section 2 introduces some notations. Section 3 introduces the architecture of the VQC-based QNN. Section 4 presents the framework of our SCR system. Section 5 shows the hybrid transfer learning algorithm. Our simulation results will be reported in Section 6, and the paper is concluded in Section 7.

2. NOTATIONS FOR QUANTUM COMPUTING

We denote $\mathbb{R}^D$ as the D-dimensional real coordinate space. Given a vector $\mathbf{v} = [v_1, v_2, ..., v_D]^T \in \mathbb{R}^D$, a D-qubit quantum state $|\mathbf{v}\rangle = \otimes_{i=1}^{D}|v_i\rangle = |v_1\rangle \otimes |v_2\rangle \otimes \cdots \otimes |v_D\rangle$ is a quantum state associated with a 2^D-dimensional vector in a Hilbert space, where for a scalar v_i, the quantum state $|v_i\rangle$ can be written as Eq. (1).

$$|v_i\rangle = \cos v_i |0\rangle + \sin v_i |1\rangle = \begin{bmatrix} \cos v_i \\ \sin v_i \end{bmatrix}. \quad (1)$$

3. VARIATIONAL QUANTUM CIRCUIT-BASED QUANTUM NEURAL NETWORK

Figure 1 exhibits the architecture of the QNN including three components: (1) quantum encoding; (2) VQC; (3) measurement. The introduction of each component is shown as follows:

(a) Quantum encoding   (b) Variational Quantum Circuit   (c) Measurement
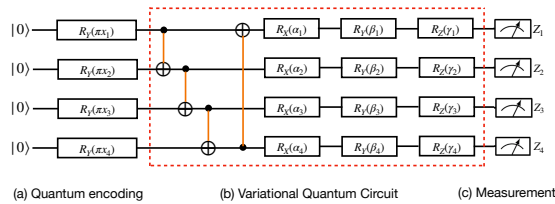
Fig. 1. *A framework of quantum neural network. $R_X(\cdot)$, $R_Y(\cdot)$, $R_Z(\cdot)$ separately denote Pauli rotation X, Y, Z gates. The circuits in the dash square correspond to the learnable model with the repeated copies.*

1. The framework of quantum encoding bridges the relationship between the classical data input $\mathbf{x}$ and its quantum state $|\mathbf{x}\rangle$. In other words, quantum encoding is associated with the generation of quantum embedding from the classical input vector $\mathbf{x} = [x_1, x_2, x_3, x_4]^T$. The quantum state $|\mathbf{x}\rangle$ can be written as Eq. (2).

$$|\mathbf{x}\rangle = |x_1\rangle \otimes |x_2\rangle \otimes |x_3\rangle \otimes |x_4\rangle$$
$$= \begin{bmatrix} \cos(x_1) \\ \sin(x_1) \end{bmatrix} \otimes \begin{bmatrix} \cos(x_2) \\ \sin(x_2) \end{bmatrix} \otimes \begin{bmatrix} \cos(x_3) \\ \sin(x_3) \end{bmatrix} \otimes \begin{bmatrix} \cos(x_4) \\ \sin(x_4) \end{bmatrix}$$
$$= \left(\otimes_{i=1}^{4} R_Y(2x_i) \right) |0\rangle^{\otimes 4}. \quad (2)$$

Then, the circuits of quantum encoding in Figure 1 produce the following quantum state as Eq. (3).

$$\left(\otimes_{i=1}^{4} R_Y(\pi x_i) \right) |0\rangle^{\otimes 4}$$
$$= \begin{bmatrix} \cos(\pi x_1) \\ \sin(\pi x_1) \end{bmatrix} \otimes \begin{bmatrix} \cos(\pi x_2) \\ \sin(\pi x_2) \end{bmatrix} \otimes \begin{bmatrix} \cos(\pi x_3) \\ \sin(\pi x_3) \end{bmatrix} \otimes \begin{bmatrix} \cos(\pi x_4) \\ \sin(\pi x_4) \end{bmatrix}. \quad (3)$$

2. The model of the VQC in the dashed square consists of CNOT gates and adjustable rotation gates R_X, R_Y, R_Z. The CNOT gates mutually impose quantum entanglement between any two quantum wires, so that the qubits from all the wires can be entangled. The rotation angles α_i, β_i, and γ_i are adjustable and can be taken as the trainable parameters for R_X, R_Y, and R_Z, respectively.

3. The output of the quantum state vector should be projected to classical observable $\hat{\mathbf{Z}} = [Z_1, Z_2, Z_3, Z_4]$, where each Z_i is an expected value along the z-axis.

Moreover, the rotation gates R_X, R_Y, and R_Z, which are associated with the unitary matrices in Figure 2, stand for a linear mapping between quantum inputs and quantum outputs. One key advantage of the QNN is that fewer model parameters are involved in the QNN model. For example, Figure 1 shows 4 quantum wires and 6 VQC layers, which result in 72 trainable model parameters in the QNN.

Fig. 2. *Unitary matrices of the quantum gates.*

Besides, the QNN model can be trained in an end-to-end pipeline based on the back-propagation algorithm with different stochastic gradient descent (SGD) optimizers such as Adam [20], RMSprop [21], and Adadelta [22]. This is because the update of the QNN parameters follows a first-order optimization technique to minimize a loss function over the dataset, which is represented as Eq. (4).

$$\Theta \leftarrow \Theta - \eta \frac{\partial \mathcal{L}(\Theta)}{\partial \Theta}, \quad (4)$$

where $\Theta = [\Theta_1, \Theta_2, ..., \Theta_D]^T$ are the parameters to be learnt, $\mathcal{L}$ is the loss over the data, and η is the learning rate. Given a small ϵ, the partial derivative term can be approximated by using the finite difference method as Eq. (5).

$$\frac{\partial C(\Theta_i)}{\partial \Theta_i} = \frac{\mathcal{L}(\Theta_i + \epsilon) - \mathcal{L}(\Theta_i - \epsilon)}{2\epsilon} + \mathcal{O}(\epsilon^2). \quad (5)$$
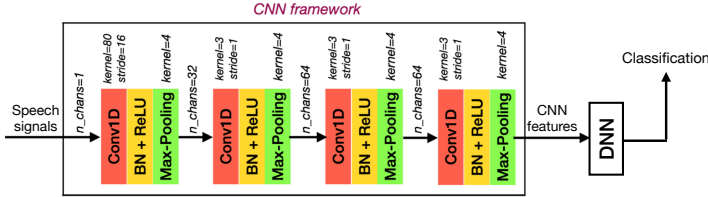
4. HYBRID CLASSICAL-QUANTUM QNN FOR SCR

Next, we illustrate the architecture of the hybrid classical-quantum QNN model for SCR in Figure 3. The CNN framework consists of four 1D convolutional layers (Conv1D) followed by batch normalization (BN) and the ReLU activation. A max-pooling layer with a kernel of 4 is also used after each Conv1D. The output of the CNN framework is a set of high-dimensional CNN features, which should be compressed to low-dimensional features (Dense) connected to the quantum encoding framework in a QNN. The encoded quantum states go through the QNN model and the measured outputs correspond to the classification labels for a certain task. The output of the QNN is connected to a classification layer by a non-trainable matrix. This hybrid classical-quantum QNN is denoted as CNN-QNN.
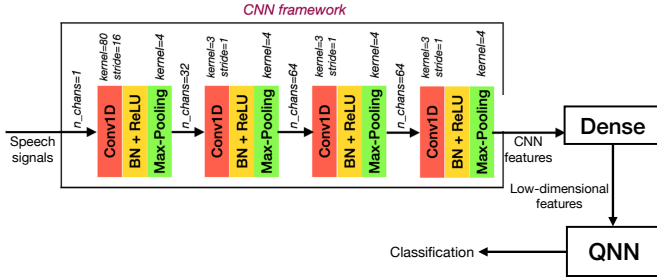
Fig. 3. *A classical-quantum hybrid QNN for SCR. CNN is utilized as feature extraction and QNN is applied for classification.*

Accordingly, a CNN-DNN model is shown in Figure 4, where the QNN is replaced by a classical DNN. The feature reduction component is removed, but more model parameters are included in the DNN model.

Fig. 4. *A classical CNN-DNN for SCR. CNN is utilized for feature extraction and DNN is used for classification.*

5. CLASSICAL-TO-QUANTUM TRANSFER LEARNING

As discussed in the introduction, hybrid classical-to-quantum transfer learning is significantly appealing in the current technological era of NISQ devices. Although there are so many technical limitations in the usage of quantum computing nowadays, e.g., a small number of available qubits and noisy quantum circuits, the NISQ computers are approaching the quantum supremacy milestone [23, 24]. At the same time, we could make use of the very successful and well-tested tools of classical deep learning, especially for speech and language processing tasks in which some specific datasets are not large enough to attain well-trained neural networks. However, prior knowledge of some networks that are pre-trained on generic datasets can be shared with a new model for a specific task.
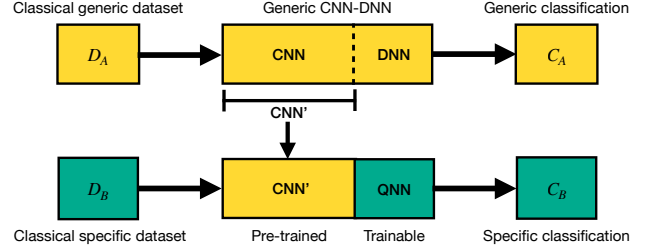
Fig. 5. *An illustration of hybrid transfer learning. A generic CNN-DNN is pre-trained, and the CNN model is transferred to the CNN-QNN. The parameters of the QNN need further fine-tuning based on a classical specific dataset.*

In classical machine learning, transfer learning has been widely used in certain applications. For example, the generative pre-trained transformer (GPT) [25] and bidirectional encoder representations from Transformers (BERT) [26] are always adapted to particular language processing tasks by simply appending a few neural networks on top of the pre-trained language models. Similarly, a hybrid classical-to-quantum transfer learning algorithm comprises of applying exactly those classical pre-trained models as feature extractors and then post-processing these features on a quantum computer. As for our transfer learning algorithm applied to SCR, a pre-trained generic CNN-DNN model is obtained beforehand based on a classical generic dataset as shown in Figure 5. The CNN model of the pre-trained CNN-DNN, namely CNN', is transferred to the CNN-QNN, which generates CNN'-QNN. During the training stage of the CNN'-QNN, the CNN' model is fixed, and only the parameters of the QNN need further fine-tuning on the classical specific dataset. Since the transferred CNN is not involved in the training process, a small number of specific training data is enough to optimize the model parameters of the QNN. Besides, the hybrid classical-to-quantum transfer learning significantly speeds up the training efficiency because the overhead of the frequent communication between classical and quantum devices can be avoided.

6. EXPERIMENTS

In this section, we assess our hybrid transfer learning algorithm for CNN-QNN in the following aspects:

1. Performance gain can be obtained by applying the hybrid transfer learning technique.

2. The hybrid transfer learning can work with both noiseless and noisy quantum circuits.

6.1. Data profile

Our SCR task was conducted on the Google speech command dataset [27]. The dataset is composed of $11,165$ development and $6,500$ test utterances, which come from 35 spoken commands, e.g., ['left', 'go', 'yes', 'down', 'up', 'on', 'right', ...]. The development data are randomly split into two parts: 90% is used for model training and 10% is used for model validation. All the audio files are about 1 second long, downsampled from 16KHz to 8KHz. The batch size was set to 256 in the training process, and the speech signals in a batch were configured as the same length by zero padding.

6.2. Experimental setup

Our baseline CNN-DNN system is based on the classical neural network architecture as shown in Figure 4, and our proposed CNN-QNN model is shown in Figure 3. The two models share the same CNN framework, which consists of 4 CNN blocks. Each CNN block is composed of a Conv1D followed by BN and ReLU, and a max-pooling is used to reduce the dimension of the speech features. The kernel size and stride are set as 80 and 16, for the first CNN block, and 3 and 1, respectively for the other CNN blocks. Moreover, the number of channels for each block follows the $1 - 32 - 64 - 64$ configuration. Besides, the kernel size was configured as 4 for the max-pooling layer of each CNN block. The time-series signal is directly fed into the first CNN block, and the output of the CNN framework is related to 64-dimensional abstract features.

As for the classical CNN-DNN, the setup of the DNN is as follows: the hidden layers of the DNN were configured as $64 - 128 - 256 - 512$, and the ReLU activation function was imposed upon each hidden layer except the top hidden layer. The output of the CNN-DNN was connected to 35 classes, which correspond to the spoken commands.

For the setup of the CNN-QNN, the number of qubits was set as 8 and the classical abstract features need to be further compressed to 8 dimensions. The 8 classical features were encoded into quantum embeddings that go through the VQC model. Since 8 channel wires were applied and 4 repetitive VQC models were used, this leads to 96 adjustable parameters in total.

6.3. Experimental results with noiseless quantum circuits

We first examine the performance of the CNN-QNN end-to-end model, where the noiseless quantum circuits are considered. In addition to the comparison with the CNN-DNN model, our method is also compared with prominent neural network models available in the literature, namely: DenseNet-121 benchmark [2], Attention-RNN [1, 28], and QCNN [16]. QCNN denotes the use of quantum convolutional features for the task. We extend the 10-class training setup in [16] to 35 classes. Moreover, all deployed models are trained with the same SCR dataset from scratch, without any data augmentation or pre-training techniques to make a fair architecture-wise study.

Models	Params. (Mb)	CE	Acc. (%)
DenseNet-121 [2]	7.978	0.473	82.11
Attention-RNN [1]	0.170	0.291	93.90
QCNN [16]	0.186	0.280	94.23
CNN-DNN	0.216	0.251	94.42
CNN-QNN	**0.071**	0.437	83.25

Table 1. The experimental results on the test dataset. Params. represents the number of model parameters; CE means the cross-entropy; and Acc. refers to the classification accuracy.

Table 1 shows the empirical results of various models. Although the classical CNN-DNN outperforms the DenseNet-121, Attention-RNN, and QCNN systems in terms of lower CE value and higher accuracy, our CNN-QNN model with much fewer parameters cannot reach improvement. This motivates us to investigate the hybrid classical-to-quantum transfer learning.

CNN-QNN_2 denotes a CNN-QNN model in which the CNN model is transferred and fixed. In other words, only the VQC parameters are trainable in the CNN-QNN_2 model. By comparison,

CNN-QNN_3 denotes that the CNN comes from a pre-trained CNN-DNN, and both CNN and QNN parameters need to be updated during the training stage.

Models	Params. (Mb)	CE	Acc. (%)
CNN-DNN	0.216	0.251	94.42
CNN-QNN_2	**0.00096**	**0.248**	94.58
CNN-QNN_3	0.071	0.267	**94.87**

Table 2. The experimental results on the test dataset. The experiments were conducted with noiseless quantum circuits.

The results of the hybrid transfer learning are shown in Table 2. Compared with CNN-DNN, CNN-QNN_2 attains better accuracy (94.58% vs. 94.42%) with the lowest CE value (0.248 vs. 0.251) and much fewer parameters (0.00096 vs. 0.216). On the other hand, CNN-QNN_3 achieves the best accuracy performance. The simulation results on noiseless circuits suggest the effectiveness of hybrid transfer learning.

6.4. Experimental results with noisy quantum circuits

Next, we discuss the hybrid transfer learning algorithm on the NISQ device, where noisy quantum circuits are considered. More specifically, we follow an established noisy circuit experiment with the NISQ device suggested by [13]. One major advantage of the setups is to observe the robustness and preserve the quantum advantages of a deployed QNN with a physical setting being close to quantum processing unit (QPU) experiments.

As for the detailed setup, we first use an IBM Q 20-qubit machine to collect channel noise in the real scenario for a deployed QNN and then upload the machine noise into our Pennylane-Qiskit [29] simulator. As shown in Table 3, the hybrid transfer learning algorithm still achieves good results with much fewer parameters. Although with the noisy quantum circuits, both CNN-QNN_2 and CNN-QNN_3 cannot outperform the classical CNN-DNN, their results are very close. Furthermore, the empirical performance of CNN-QNN_2 and CNN-QNN_3 would become much better when fault-tolerant quantum computers could become available.

Models	Params. (Mb)	CE	Acc. (%)
CNN-DNN	0.216	0.251	94.42
CNN-QNN_2	**0.00096**	0.274	93.38
CNN-QNN_3	0.071	0.269	93.84

Table 3. The experimental results on the test dataset. The simulations were conducted with noisy quantum circuits.

7. CONCLUSIONS

This work focuses on a hybrid classical-to-quantum transfer learning algorithm for QNNs applied to SCR. We first set up the VQC-based QNN, and then design a CNN-QNN-based SCR system. We employ hybrid transfer learning to transfer a pre-trained CNN framework to our CNN-QNN system so that better performance could be obtained. Our experiments on the Google speech command dataset show that the hybrid classical-to-quantum transfer learning is of significance in enhancing classification accuracy and lowering cross-entropy loss value for the CNN-QNN model.

8. REFERENCES

[1] Douglas Coimbra de Andrade, Sabato Leo, Martin Loesener Da Silva Viana, and Christoph Bernkopf, "A Neural Attention Model for Speech Command Recognition," *arXiv preprint arXiv:1808.08929*, 2018.

[2] Brian McMahan and Delip Rao, "Listening to the World Improves Speech Command Recognition," in *Proc. AAAI Conference on Artificial Intelligence*, 2018, vol. 32.

[3] Jaesung Bae and Dae-Shik Kim, "End-to-End Speech Command Recognition with Capsule Network," in *Proc. Interspeech*, 2018, pp. 776–780.

[4] Ian Goodfellow, Yoshua Bengio, and Aaron Courville, *Deep Learning*, MIT press, 2016.

[5] Jun Qi, Jun Du, Sabato Marco Siniscalchi, and Chin-Hui Lee, "A Theory on Deep Neural Network Based Vector-to-Vector Regression with An Illustration of Its Expressive Power in Speech Enhancement," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 12, pp. 1932–1943, 2019.

[6] Jun Qi, Jun Du, Sabato Marco Siniscalchi, Xiaoli Ma, and Chin-Hui Lee, "Analyzing Upper Bounds on Mean Absolute Errors for Deep Neural Network Based Vector-to-Vector Regression," *IEEE Transactions on Signal Processing*, vol. 68, pp. 3411–3422, 2020.

[7] Jun Qi, *Theoretical Error Performance Analysis for Deep Neural Network Based Regression Functional Approximation*, Ph.D. thesis, Georgia Institute of Technology, 2022.

[8] Thomas N Theis and H-S Philip Wong, "The End of Moore's Law: A New Beginning for Information Technology," *Computing in Science & Engineering*, vol. 19, no. 2, pp. 41–50, 2017.

[9] Jun Qi, Chao-Han Huck Yang, and Pin-Yu Chen, "QTN-VQC: An End-to-End Learning Framework for Quantum Neural Networks," *arXiv preprint arXiv:2110.03861*, 2021.

[10] Jacob Biamonte, Peter Wittek, Nicola Pancotti, Patrick Rebentrost, Nathan Wiebe, and Seth Lloyd, "Quantum Machine Learning," *Nature*, vol. 549, no. 7671, pp. 195–202, 2017.

[11] Jarrod R McClean, Sergio Boixo, Vadim N Smelyanskiy, Ryan Babbush, and Hartmut Neven, "Barren Plateaus in Quantum Neural Network Training Landscapes," *Nature Communications*, vol. 9, no. 1, pp. 1–6, 2018.

[12] John Preskill, "Quantum Computing In the NISQ Era and Beyond," *Quantum*, vol. 2, pp. 79, 2018.

[13] Samuel Yen-Chi Chen, Chao-Han Huck Yang, Jun Qi, Pin-Yu Chen, Xiaoli Ma, and Hsi-Sheng Goan, "Variational Quantum Circuits for Deep Reinforcement Learning," *IEEE Access*, vol. 8, pp. 141007–141024, 2020.

[14] Iris Cong, Soonwon Choi, and Mikhail D Lukin, "Quantum Convolutional Neural Networks," *Nature Physics*, vol. 15, no. 12, pp. 1273–1278, 2019.

[15] Kosuke Mitarai, Makoto Negoro, Masahiro Kitagawa, and Keisuke Fujii, "Quantum Circuit Learning," *Physical Review A*, vol. 98, no. 3, pp. 032309, 2018.

[16] Chao-Han Huck Yang, Jun Qi, Samuel Yen-Chi Chen, Pin-Yu Chen, Sabato Marco Siniscalchi, Xiaoli Ma, and Chin-Hui Lee, "Decentralizing Feature Extraction with Quantum Convolutional Neural Network for Automatic Speech Recognition," in *Proc. International Conference on Acoustics, Speech and Signal Processing*, 2021, pp. 6523–6527.

[17] Sinno Jialin Pan and Qiang Yang, "A Survey on Transfer Learning," *IEEE Transactions on knowledge and data engineering*, vol. 22, no. 10, pp. 1345–1359, 2009.

[18] Chuanqi Tan, Fuchun Sun, Tao Kong, Wenchang Zhang, Chao Yang, and Chunfang Liu, "A Survey on Deep Transfer Learning," in *Proc. International Conference on Artificial Neural Networks*. Springer, 2018, pp. 270–279.

[19] Eric Wehrli, Luka Nerima, and Yves Scherrer, "Deep Linguistic Multilingual Translation and Bilingual Dictionaries," in *Proc. The fourth workshop on statistical machine translation*. The Association for Computational Linguistics, 2009, pp. 90–94.

[20] Diederik P Kingma and Jimmy Ba, "Adam: A Method for Stochastic Optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[21] Fangyu Zou, Li Shen, Zequn Jie, Weizhong Zhang, and Wei Liu, "A Sufficient Condition for Convergences of Adam and RMSprop," in *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 11127–11135.

[22] Matthew D Zeiler, "Adadelta: An Adaptive Learning Rate Method," *arXiv preprint arXiv:1212.5701*, 2012.

[23] Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando GSL Brandao, David A Buell, et al., "Quantum Supremacy Using A Programmable Superconducting Processor," *Nature*, vol. 574, no. 7779, pp. 505–510, 2019.

[24] Aram W Harrow and Ashley Montanaro, "Quantum Computational Supremacy," *Nature*, vol. 549, no. 7671, pp. 203–209, 2017.

[25] Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al., "Language Models Are Few-Shot Learners," *arXiv preprint arXiv:2005.14165*, 2020.

[26] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova, "BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[27] Pete Warden, "Speech Commands: A Dataset for Limited-Vocabulary Speech Recognition," *arXiv preprint arXiv:1804.03209*, 2018.

[28] Chao-Han Yang, Jun Qi, Pin-Yu Chen, Xiaoli Ma, and Chin-Hui Lee, "Characterizing Speech Adversarial Examples Using Self-Attention U-Net Enhancement," in *Proc. International Conference on Acoustics, Speech and Signal Processing*, 2020, pp. 3107–3111.

[29] Ville Bergholm, Josh Izaac, Maria Schuld, Christian Gogolin, M Sohaib Alam, Shahnawaz Ahmed, Juan Miguel Arrazola, Carsten Blank, Alain Delgado, Soran Jahangiri, et al., "Pennylane: Automatic Differentiation of Hybrid Quantum-Classical Computations," *arXiv preprint arXiv:1811.04968*, 2018.