

## Article

# Federated Quantum Machine Learning

Samuel Yen-Chi Chen \*  and Shinjae Yoo

Computational Science Initiative, Brookhaven National Laboratory, Upton, NY 11973, USA; sjyoo@bnl.gov

\* Correspondence: ychen@bnl.gov

**Abstract:** Distributed training across several quantum computers could significantly improve the training time and if we could share the learned model, not the data, it could potentially improve the data privacy as the training would happen where the data is located. One of the potential schemes to achieve this property is the federated learning (FL), which consists of several clients or local nodes learning on their own data and a central node to aggregate the models collected from those local nodes. However, to the best of our knowledge, no work has been done in quantum machine learning (QML) in federation setting yet. In this work, we present the federated training on hybrid quantum-classical machine learning models although our framework could be generalized to pure quantum machine learning model. Specifically, we consider the quantum neural network (QNN) coupled with classical pre-trained convolutional model. Our distributed federated learning scheme demonstrated almost the same level of trained model accuracies and yet significantly faster distributed training. It demonstrates a promising future research direction for scaling and privacy aspects.

**Keywords:** quantum machine learning; federated learning; quantum neural networks; variational quantum circuits; privacy-preserving AI



**Citation:** Chen, S.Y.-C.; Yoo, S. Federated Quantum Machine Learning. *Entropy* **2021**, *23*, 460. <https://doi.org/10.3390/e23040460>

Academic Editor: Rosario Lo Franco

Received: 23 March 2021

Accepted: 9 April 2021

Published: 13 April 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Recently, advances in machine learning (ML), in particular deep learning (DL), have found significant success in a wide variety of challenging tasks such as computer vision [1–3], natural language processing [4], and even playing the game of Go with a superhuman performance [5].

In the meantime, quantum computers are introduced to the general public by several technology companies such as IBM [6], Google [7], IonQ [8] and D-Wave [9]. Theoretically, quantum computing can provide exponential speedup to certain classes of hard problems that are intractable on classical computers [10,11]. The most famous example is the factorization of large numbers via Shor algorithm [12] which can provide exponential speedup. While the search in unstructured database via Grover algorithm [13] can provide quadratic speedup. However, currently available quantum computers are not equipped with quantum error correction [14,15] and would suffer from the device noise. Quantum computation tasks or quantum circuits with a large number of qubits and/or a long circuit depth cannot be faithfully implemented on these so-called noisy intermediate-scale quantum (NISQ) devices [16]. Therefore, it is a highly challenging task to design applications with moderate quantum resources requirements which can leverage the quantum advantages on these NISQ devices.

With the above-mentioned two rapid growing fields, it is then natural to consider the combination of them. Especially the machine learning applications which can be implemented on NISQ devices. Indeed, the area *quantum machine learning* (QML) draws a lot of attention recently and there are several promising breakthroughs. The most notable progress is the development of variational algorithms [17–19] which enable the quantum machine learning on NISQ devices [20]. Recent efforts have demonstrated the promising application of NISQ devices in several machine learning tasks [21–46].

One of the common features of these successful ML models is that they are data-driven. To build a successful deep learning model, it requires a huge amount of data. Although there are several public datasets for research purpose, most advanced and personalized models largely depend on the collected data from users' mobile devices and other personal data (e.g., medical record, browsing habits and etc.). For example, ML/DL approaches also succeed in the field of medical imaging [47,48], speech recognition [49–51], to name a few. These fields rely critically on the massive dataset collected from the population and these data should not be accessed by unauthorized third-party. The use of these sensitive and personally identifiable information raises several concerns. One of the concerns is that the channel used to exchange with the cloud service providers can be compromised, leading to the leakage of high-value personal or commercial data. Even if the communication channel can be secured, the cloud service provider is also risky as malicious adversaries can potentially invade the computing infrastructure. There are several solutions to deal with such issues. One of them is called *federated learning* (FL), which focuses on the decentralized computing architecture. For example, users can train a speech recognition model on his cell phone and upload the model to the cloud in exchange of the global model without upload the recordings directly. Such framework is made possible due to the fact of recent advances in hardware development, making even the small devices so powerful. This concept not only helps the privacy-preserving practice in classical machine learning but also in the rapidly emerging *quantum machine learning* as researchers are trying to expand the machine learning capabilities by leveraging the power of quantum computers. To harness the power of quantum computers in the NISQ era, the key challenge is how to distribute the computational tasks to different quantum machines with limited quantum capabilities. Another challenge is the rising privacy concern in the use of large scale machine learning infrastructure. We address these two challenges by providing the framework of training quantum machine learning models in a federated manner.

In this paper, we propose the federated training on hybrid quantum-classical classifiers. We show that with the federated training, the performance in terms of the testing accuracy does not decrease. In addition, the model still converges quickly compared to the non-federated training. Our efforts not only help building secure QML infrastructure but also help the distributed QML training which is to better utilize available NISQ devices.

This paper is organized as follows—in Section 2, we introduce the concept of federated machine learning. In Section 3, we describe the variational quantum circuit architecture in details. In Section 4, we describe the transfer learning in hybrid quantum-classical models. Section 5 shows the performance of the proposed federated quantum learning on the experimental data, followed by further discussions in Section 6. Finally we conclude in Section 7.

## 2. Federated Machine Learning

Federated learning (FL) [52] emerges recently along with the rising privacy concerns in the use of large-scale dataset and cloud-based deep learning [53]. The basic components in a federated learning process are a *central node* and several *client nodes*. The central node holds the *global model* and receives the trained parameters from client devices. The central node performs the *aggregation* process to generate the new global model and share this new model to all of its client nodes. The client nodes will train locally with the received model with their own part of data, which in general is only a small portion. In our proposed framework, the local clients are quantum computers or quantum simulators with the circuit parameters trained via hybrid quantum-classical manner. In each training round, a specified number of client nodes will be selected to perform the local training. Once the client training is finished, the circuit parameters from all the client nodes will be aggregated by the central node. There are various methods to aggregate the model. In this work, we choose the *mean* of the client models. The scheme of federated quantum machine learning is shown in Figure 1.

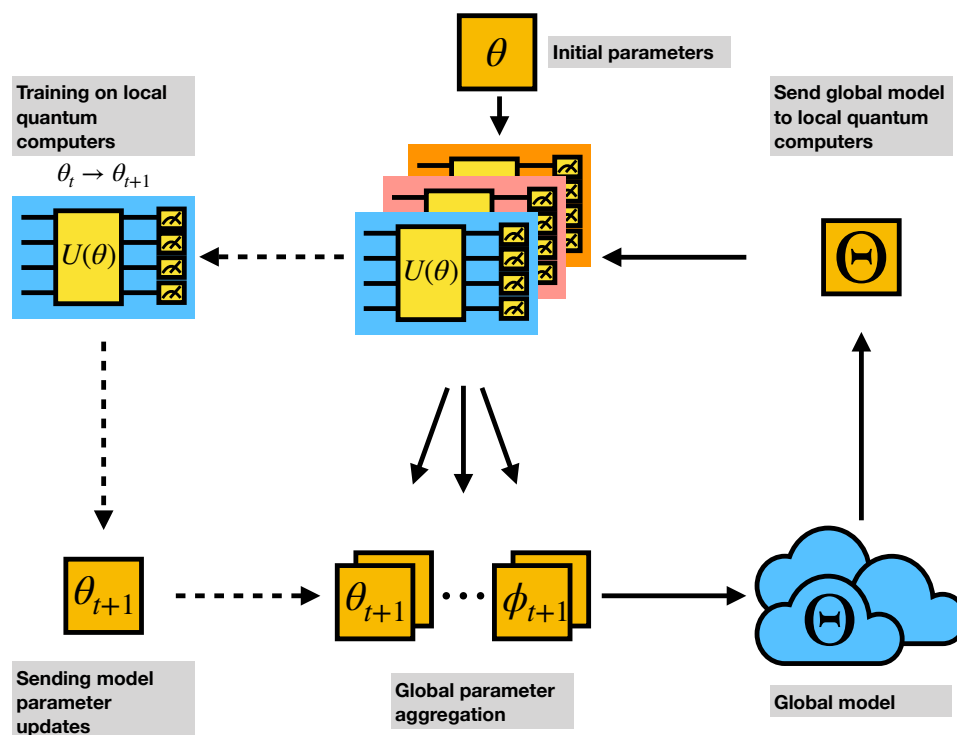


Figure 1. Federated Quantum Machine Learning.

For further discussion and advanced settings on federated learning, we refer to [54–61].

### 3. Variational Quantum Circuits

Variational quantum circuits (VQC) or quantum neural networks (QNN) are a special kind of quantum circuits with adjustable circuit parameters subject to optimization procedures developed by the classical machine learning community. In Figure 2 we introduce the general setting of a VQC in which the  $E(x)$  encodes the classical data into a quantum state which the quantum gates can actually operate on and the  $W(\phi)$  is the learnable block with parameters  $\phi$  which can be seen as the *weights* in classical neural network. There are *quantum measurements* in the final part of VQC which is to *readout* the information from a quantum circuit and these classical numbers can be further processed with other classical or quantum components.

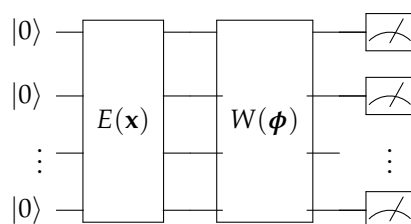


Figure 2. General structure for the variational quantum circuit (VQC). The  $E(x)$  is the quantum routine for encoding the classical data into the quantum state and  $W(\phi)$  is the variational quantum circuit block with the learnable parameters  $\phi$ . After the quantum operation, the quantum state is *measured* to retrieve classical numbers for further processing.

The general idea of VQC or QNN is that the circuit parameters are updated via iterative methods on a classical computer. Recent theoretical studies have also demonstrated that VQCs are more expressive than conventional neural networks [9,62–64] with respect to the number of parameters or the learning speed. In addition, in the work [30,65,66], it has been demonstrated via numerical simulation that certain hybrid quantum-classical

architectures reach higher accuracies than classical neural networks with a similar number of parameters.

Recent advances in VQC have demonstrated various applications in a wide variety of machine learning tasks. For example, VQC has been shown to be successful in the task of classification [20–29,65,67,68], function approximation [20,30,31], generative machine learning [32–37], metric learning [38,39], deep reinforcement learning [40–44,69], sequential learning [30,45,70] and speech recognition [46].

### 3.1. Quantum Encoder

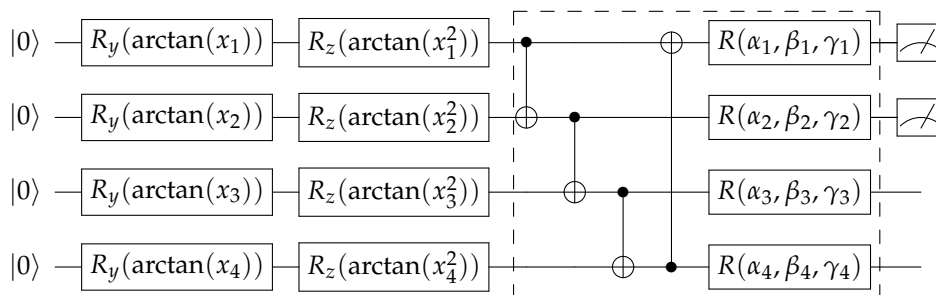
For a quantum circuit to operate on a classical dataset, the critical step is to define the *encoding* method which is to transform the classical vector into a quantum state. The encoding scheme is important as it is relevant to the efficiency of hardware implementation and potential quantum advantages. In NISQ era, the number of qubits as well as the circuit-depth are limited. Therefore, we need to encode the classical values with small number of qubits and without too many quantum operations. For more in-depth introduction of various kinds of encoding methods used in QML, refer to [71]. A general  $N$ -qubit quantum state can be represented as:

$$|\psi\rangle = \sum_{(q_1, q_2, \dots, q_N) \in \{0,1\}^N} c_{q_1, \dots, q_N} |q_1\rangle \otimes |q_2\rangle \otimes |q_3\rangle \otimes \dots \otimes |q_N\rangle, \quad (1)$$

where  $c_{q_1, \dots, q_N} \in \mathbb{C}$  is the *amplitude* of each quantum state and  $q_i \in \{0,1\}$ . The square of the amplitude  $c_{q_1, \dots, q_N}$  is the *probability* of measurement with the post-measurement state in  $|q_1\rangle \otimes |q_2\rangle \otimes |q_3\rangle \otimes \dots \otimes |q_N\rangle$ , and the total probability should sum to 1, that is,

$$\sum_{(q_1, q_2, \dots, q_N) \in \{0,1\}^N} ||c_{q_1, \dots, q_N}||^2 = 1. \quad (2)$$

In this work, we use the *variational encoding* scheme to encode the classical values into a quantum state. The basic idea behind this encoding scheme is to use the input values or their transformation as rotation angles for the quantum rotation gate. As shown in Figure 3, the encoding parts consist of single-qubit rotation gates  $R_y$  and  $R_z$  and use  $\arctan(x_i)$  and  $\arctan(x_i^2)$  as the corresponding transformations.



**Figure 3.** Variational quantum classifier. The variational quantum classifier includes three components: *encoder*, *variational layer* and *quantum measurement*. The encoder consists of several single-qubit gates  $R_y(\arctan(x_i))$  and  $R_z(\arctan(x_i^2))$  which represent rotations along  $y$ -axis and  $z$ -axis by the given angle  $\arctan(x_i)$  and  $\arctan(x_i^2)$ , respectively. These rotation angles are derived from the input values  $x_i$  and are not subject to iterative optimization. The variational layer consists of CNOT gates between each pair of neighbouring qubits which are used to entangle quantum states from each qubit and general single qubit unitary gates  $R(\alpha, \beta, \gamma)$  with three parameters  $\alpha, \beta, \gamma$ . Parameters labeled  $\alpha_i, \beta_i$  and  $\gamma_i$  are the ones for iterative optimization. The quantum measurement component will output the Pauli-Z expectation values of designated qubits. The number of qubits and the number of measurements can be adjusted to fit the problem of interest. In this work, we use the VQC as the final classifier layer, therefore the number of qubits equals to the latent vector size which is 4 and we only consider the measurement on the first two qubits for binary classification. The grouped box in the VQC may repeat several times to increase the number of parameters, subject to the capacity and capability of the available quantum computers or simulation software used for the experiments. In this work, the grouped box repeats for 2 times.

### 3.2. Quantum Gradients

The hybrid quantum-classical model can be trained in an end-to-end fashion, following the common backpropagation method used in training deep neural network. When it comes to the gradient calculation on quantum functions, *parameter-shift* method is employed. It can be used to derive the analytical gradient of the quantum circuits. The method is described in the reference [72,73]. The idea behind the parameter-shift rule is that given the knowledge of calculating the expectation of certain observable of quantum functions, the quantum gradients can be calculated without the use of finite difference method.

## 4. Hybrid Quantum-Classical Transfer Learning

In the NISQ era, the quantum computers are not error-corrected and thus cannot perform calculations in a fault-tolerant manner. The circuit depth and number of qubits are therefore limited and it is non-trivial to design the model architectures which can potentially harness the capabilities provided by near-term quantum computers. In this work, we employ the hybrid quantum-classical transfer learning scheme inspired by the work [24]. The idea is to use a pre-trained classical deep neural network, mostly convolutional neural networks (CNN) to extract the features from the images and compress the information into a latent vector  $x$  which is with much smaller dimension than the original image. Then the latent vector  $x$  is processed by the quantum circuit model to output the logits of each class. The scheme is presented in Figure 4. In this work, we employ the VGG16 [1] pre-trained model as the feature extractor.

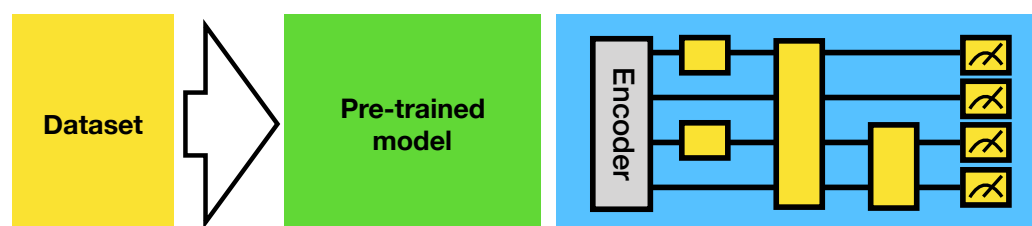


Figure 4. Hybrid Quantum-Classical Transfer Learning.

## 5. Experiments and Results

In this study we consider the following setting:

- Central node C: Receive the uploaded circuit parameters  $\theta_i$  from each local machine  $N_i$  and aggregate them into a global parameter  $\Theta$  and distributes to all local machines.
- Training points are equally distributed to the local machines and the testing points are on the central node to evaluate the aggregated global model.
- Individual local machines  $N_i$ : Each has a distinct part of the training data and will perform  $E$  epochs of the training locally with the batch size  $B$ .

The software we use for this work are PyTorch [74], PennyLane [73] and Qulacs [75].

### 5.1. Cats vs. Dogs

We perform the binary classification on the classic cats vs dogs dataset [76]. Each image in this dataset has slightly different dimensions, therefore we preprocessed to make all of the training and testing samples in the dimension of  $224 \times 224$ . In Figure 5 we show some of the examples from this dataset. Here, we have in total 23,000 training data and 2000 testing data. The testing data are on the central node which will be used to evaluate the aggregated model (global model) after each training round. The training data are equally distributed to the 100 local machines  $N_i$  where  $i \in \{1 \cdots 100\}$ . Therefore in each local machine, there are 230 training points. In each training round, 5 local machines will be randomly selected and each will perform 1, 2 or 4 epochs of training with its own training points. The batch size is  $S = 32$ . The trained model will then be sent to the central node for the aggregation. The aggregation method we use in this experiment is the collected model average. The aggregated model (global model) will then be shared to all the

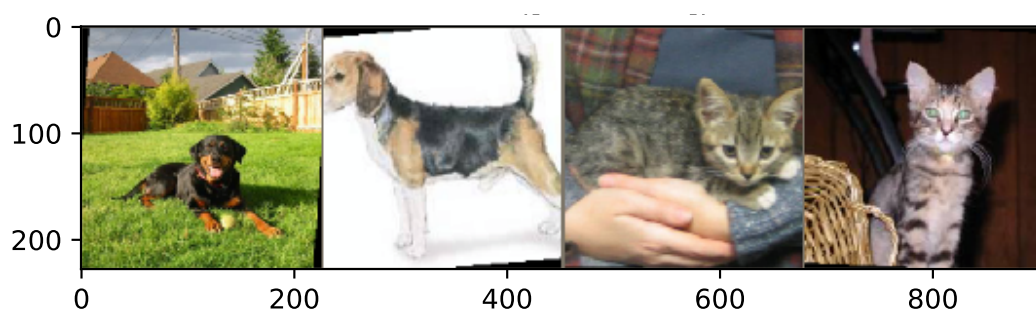


local machines. The hybrid model used in this experiment consists of pre-trained VGG16 model and a 4-qubit variational quantum circuit (VQC) as shown in Figure 3. The original classifier layer in the VGG16 model is replaced with the one shown in Table 1 in order to fit input dimension of the VQC layer. The dashed-box in the quantum circuit repeats twice, consisting of  $4 \times 3 \times 2 = 24$  quantum circuit parameters. The VQC receives 4-dimensional compressed vectors from the pre-trained VGG model to perform the classification task. The non-federated training for the comparison is with the same hybrid VGG-VQC architecture as the one used in the federated training. We perform 100 training rounds and the results are presented in Figure 6. We compare the performance of federated learning with non-federated learning with the same hybrid quantum-classical architecture and the same dataset.

**Table 1.** The trainable layer in our modified VGG model. This layer is designed to convert the output from the pretrained layer to a smaller dimensional one which is suitable for the VQC to process. The activation function used in this layer is ReLU. In addition, dropout layers with dropout rate = 0.5 are used.

	Linear	ReLU	Dropout ( $p = 0.5$ )	Linear	ReLU	Dropout ( $p = 0.5$ )	Linear
Input	25,088			4096			4096
Output	4096			4096			4

In the left three panels of Figure 6, we present the results of training the hybrid quantum model via federated setting with different number of local training epochs. Since the training data are distributed across different clients, we only consider the testing accuracies with the aggregated global model. In the considered Cats vs Dogs dataset, we observe that both the testing accuracies and testing loss reach the comparable level as the non-federated training. We also observe that the training loss, which is the average from clients, has fluctuations compared to non-federated training (shown in Table 2). The underlying reason might be that in each training round, different clients are selected, therefore the training data used to evaluate the training loss are different. Yet the training loss still converges after the 100 rounds of training. In addition, the testing loss and accuracies converge to comparable levels to the non-federal training, regardless of the local training epochs. Notably, we observe that a single epoch in local training is pretty enough to train a well-performed model. In each round of the federated training, the model updates are based on the samplings from 5 clients, with 1 local training epoch. The computing resources used are linear with  $230 \times 5 \times 1 = 1150$  in total. While for a full epoch of training with non-federated setting, the computing resources used are linear with 23,000. This results imply the potential of more efficient training on QML models with distributed schemes. This particularly benefits the training of quantum models when we are using high-performance simulation platform or an array of small NISQ devices, with the moderate communication overhead.



**Figure 5.** Cats vs Dogs Dataset [76].

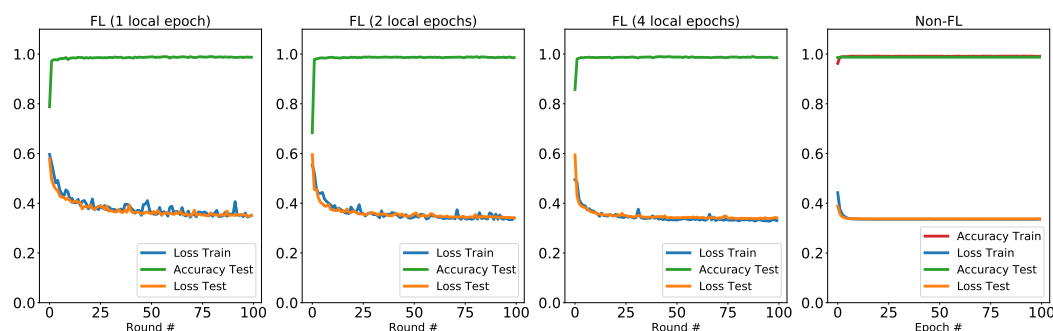


Figure 6. Results: Cats vs. Dogs.

Table 2. Comparison of performance in different training schemes with Cats vs Dogs dataset.

	Training Loss	Testing Loss	Testing Accuracy
Federated Training (1 local epoch)	0.3506	0.3519	98.7%
Federated Training (2 local epochs)	0.3405	0.3408	98.6%
Federated Training (4 local epochs)	0.3304	0.3413	98.6%
Non-Federated Training	0.3360	0.3369	98.75%

## 5.2. CIFAR (Planes vs. Cars)

In this experiment, we use the data from the CIFAR-10 dataset [77]. The dimension of the images in this dataset is  $32 \times 32$ . In Figure 7 we show some of the examples from this dataset. The hybrid quantum-classical models used in this experiment is the same as in the previous experiment. Here we have in total 10,000 training points and 2000 testing points. The testing points are on the central node (global model) which will be used to evaluate the aggregated model after each training round. The training points are equally distributed to the 100 local machines  $N_i$  where  $i \in \{1 \dots 100\}$ . In each training round, 5 local machines will be randomly selected and each will perform 1, 2 or 4 epochs of training with its own training points. The batch size  $S = 32$ . The trained model will then be sent to the central mode for aggregation. In the left three panels of Figure 8, we present the results of training the hybrid quantum model via federated setting with different number of local training epochs. Since the training data is distributed across different clients, we only consider the testing accuracies with the aggregated global model. In the considered Planes vs Cars dataset, we observe that both the testing accuracies and testing loss reach the comparable level as the non-federated training (shown in Table 3). Similar to the previous Cats vs Dogs dataset, we observe that the training loss, which is the average from clients, has fluctuations compared to non-federated training. In addition, the testing loss and accuracies converge to comparable levels to the non-federated training, regardless of the local training epochs. Notably, we observe that a single epoch in local training is pretty enough to train a well-performed model. In each round of the federated training, the model updates are based on the samplings from 5 clients, with 1 local training epoch. The computing resources used are linear with  $100 \times 5 \times 1 = 500$  in total. While for a full epoch of training with non-federated setting, the computing resources used are linear with 10,000. This results again imply the potential of more efficient training on QML models with distributed schemes. This particularly benefits the training of quantum models when we are using high-performance simulation platform or an array of small NISQ devices, with the moderate communication overhead.



Figure 7. Planes vs. Cars from CIFAR-10 Dataset [77].

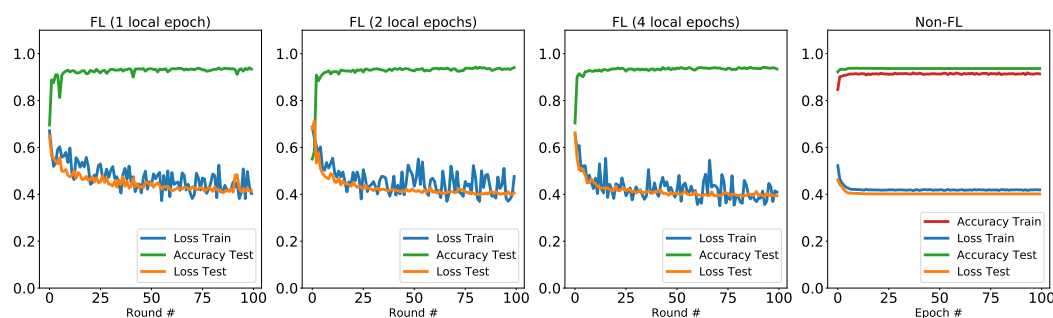


Figure 8. Results: Planes vs. Cars.

Table 3. Comparison of performance in different training schemes with CIFAR (Planes vs. Cars) dataset.

	Training Loss	Testing Loss	Testing Accuracy
Federated Training (1 local epoch)	0.4029	0.4133	93.40%
Federated Training (2 local epochs)	0.4760	0.4056	94.05%
Federated Training (4 local epochs)	0.4090	0.3934	93.45%
Non-Federated Training	0.4190	0.4016	93.65%

## 6. Discussion

### 6.1. Integration with Other Privacy-Preserving Protocols

In this study we consider the federated quantum learning framework. One of the limitation is that the process of exchanging model parameters can potentially be attacked. Moreover, we can not exclude the possibilities that malicious parties are joining the network, which will get the aggregated global model. The leaked model parameters can be used to deduce the training data of the model [78]. There are other protocols which can further boost the security. For example, it has been shown that trained models can be used to recover training entries [79]. In addition, it is also possible for adversaries to find out whether a specific entry is used in training process [80]. These possibilities raise serious concerns when the QML models are used to process private and sensitive data. One of the potential solution is to train the model with *differential privacy* (DP) [81]. With DP, it is possible to share the trained model and still keep the private information of the training data. Another direction is to incorporate the secure multi-party computation [82] which can further increase the security in decentralization computing. For example, a recent work



using universal blind quantum computation [83] provides an quantum protocol to achieve privacy-preserving multi-party quantum computation.

### 6.2. Different Aggregation Method

In this work, we use the simplest aggregation method which is simply calculating the average of parameters from each local machine (model). In a more realistic application scenario, clients may upload a corrupt trained model, or the communication channel may be interfered by noise, which can potentially compromise the global model if there is no other countermeasures. Several recent works present advanced aggregation schemes to address this issue [84,85]. The implementation of these advanced protocols with quantum machine learning is an interesting direction for future work.

### 6.3. Decentralization

This research presents a proof-of-concept federated training on quantum machine learning models. The scheme includes a *central node* to receive the trained models from clients, to aggregate them and to distribute the aggregated model to clients. This *central node* can be vulnerable to malicious attacks and the adversaries can compromise the whole network. Moreover, the communication bandwidth between clients and the central node may vary, leading some undesired effects in the synchronization process. To address these issues, recent studies propose various *decentralized* federated learning schemes [86–92]. For example, the *distributed ledger technologies* (DLT) [93–96] which power the development of blockchain have been applied in the decentralized FL [97–103]. The blockchain technologies are used to ensure the robustness and integrity of the shared information while remove the requirement of a central node. Blockchain-enabled FL can also be designed to encourage the data-owner participating in the model training process [104]. In addition, *peer-to-peer* protocols are also employed in FL to remove the need of a central node [105,106]. *Gossip learning* [107,108] is an alternative learning framework to FL [109–112]. Under gossip learning framework, no central node is required, nodes on the network exchange and aggregate models directly. The efficiencies and capabilities of these decentralized schemes such as blockchained FL and gossip learning in the quantum regime are left for future work.

In classical machine learning, distributed training frameworks are designed to scale up the model training to computing clusters [113], making the training on large-scale dataset and complex models possible. Potential direction is to apply the federated quantum learning to the high-performance quantum simulation.

### 6.4. Other Quantum Machine Learning Models

In this work, we consider the hybrid quantum-classical transfer learning architecture which includes a pre-trained classical model as the feature extractor. Currently the available quantum computers and simulation software are rather limited and do not possess large number of qubits. However, the proposed framework can be extended well beyond the transfer learning structure. Recently, a hybrid architecture combining tensor network and quantum circuit is proposed [114]. Such hybrid architecture is more generic than the pre-trained network used in this work. It is interesting to investigate the potential of decentralizing such kind of architectures. Moreover, it is possible to study the federated learning on quantum convolutional neural networks (QCNN) [65,66,115–119] when larger-scale quantum simulators or real quantum computers are available. The proposed model is not limited to learning classical data such as the ones presented in this work. It is possible to extend the scheme of this work to learn quantum data as well.

### 6.5. Potential Applications

This work can potentially be integrated with the work [46,120] for decentralizing the quantum-enhanced speech recognition. Another potential direction is in the use in healthcare in which a tremendous amount of sensitive personal data need to be processed

to train a reliable model. For example, the work [67] studied the application of VQC in dementia prediction which would benefit from the federated training to preserve the users' privacy. Recently, the application of quantum computing in financial industries have drawn a lot of attention [121]. It is expected that federated QML would play an important role in finance as well.

## 7. Conclusions

In this work, we provide the framework to train hybrid quantum-classical classifiers in a federated manner, which can help in preserving the privacy and distributing computational loads to an array of NISQ computers. We also show that the federated training in our setting does not sacrifice the performance in terms of the testing accuracy. This work should benefit the research in both the privacy-preserving AI and the quantum computing and pave a new direction for building secure, reliable and scalable distributed quantum machine learning architecture.

**Author Contributions:** Conceptualization, S.Y.-C.C.; methodology, S.Y.-C.C., S.Y.; software, S.Y.-C.C.; validation, S.Y.; writing—original draft preparation, S.Y.-C.C.; writing—review and editing, S.Y.; visualization, S.Y.-C.C.; supervision, S.Y.; project administration, S.Y.; funding acquisition, S.Y. Both discussed the results and wrote the paper. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work is supported by the U.S. Department of Energy, Office of Science, Advanced Scientific Computing Research under Award Number DE-SC-0012704 and the Brookhaven National Laboratory LDRD #20-024.

**Data Availability Statement:** Publicly available datasets were analyzed in this study. This two datasets used in this study can be found here: <https://www.cs.toronto.edu/~kriz/cifar.html> and <https://www.kaggle.com/karakaggle/kaggle-cat-vs-dog-dataset> (accessed on 13 April 2021).

**Conflicts of Interest:** The authors declare no conflict of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

VQC	Variational Quantum Circuit
QNN	Quantum Neural Network
QML	Quantum Machine Learning
FL	Federated Learning
NISQ	Noisy Intermediate Scale Quantum
DP	Differential Privacy
DLT	Distributed Ledger Technology

## References

1. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
2. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 12 June 2015; pp. 1–9.
3. Voulodimos, A.; Doulamis, N.; Doulamis, A.; Protopapadakis, E. Deep Learning for Computer Vision: A Brief Review. *Comput. Intell. Neurosci.* **2018**, *2018*, 1–13. [CrossRef] [PubMed]
4. Sutskever, I.; Vinyals, O.; Le, Q.V. Sequence to sequence learning with neural networks. *Adv. Neural Inf. Process. Syst.* **2014**, *2*, 3104–3112.
5. Silver, D.; Huang, A.; Maddison, C.J.; Guez, A.; Sifre, L.; van den Driessche, G.; Schrittwieser, J.; Antonoglou, I.; Panneershelvam, V.; Lanctot, M.; et al. Mastering the game of Go with deep neural networks and tree search. *Nature* **2016**, *529*, 484–489. [CrossRef] [PubMed]
6. Cross, A. The IBM Q experience and QISKit open-source quantum computing software. In Proceedings of the APS Meeting Abstracts, Los Angeles, CA, USA, 5–9 March 2018.
7. Arute, F.; Arya, K.; Babbush, R.; Bacon, D.; Bardin, J.C.; Barends, R.; Biswas, R.; Boixo, S.; Brandao, F.G.; Buell, D.A.; et al. Quantum supremacy using a programmable superconducting processor. *Nature* **2019**, *574*, 505–510. [CrossRef] [PubMed]

8. Grzesiak, N.; Blümel, R.; Wright, K.; Beck, K.M.; Pienti, N.C.; Li, M.; Chaplin, V.; Amini, J.M.; Debnath, S.; Chen, J.S.; et al. Efficient arbitrary simultaneously entangling gates on a trapped-ion quantum computer. *Nat. Commun.* **2020**, *11*, 1–6. [\[CrossRef\]](#)
9. Lanting, T.; Przybysz, A.J.; Smirnov, A.Y.; Spedalieri, F.M.; Amin, M.H.; Berkley, A.J.; Harris, R.; Altomare, F.; Boixo, S.; Bunyk, P.; et al. Entanglement in a quantum annealing processor. *Phys. Rev. X* **2014**, *4*, 021041. [\[CrossRef\]](#)
10. Harrow, A.W.; Montanaro, A. Quantum computational supremacy. *Nature* **2017**, *549*, 203–209. [\[CrossRef\]](#)
11. Nielsen, M.A.; Chuang, I. Quantum Computation and Quantum Information. *Am. J. Phys.* **2002**, *70*. [\[CrossRef\]](#)
12. Shor, P.W. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Rev.* **1999**, *41*, 303–332. [\[CrossRef\]](#)
13. Grover, L.K. Quantum mechanics helps in searching for a needle in a haystack. *Phys. Rev. Lett.* **1997**, *79*, 325. [\[CrossRef\]](#)
14. Gottesman, D. Stabilizer codes and quantum error correction. *arXiv* **1997**, arXiv:quant-ph/9705052.
15. Gottesman, D. Theory of fault-tolerant quantum computation. *Phys. Rev. A* **1998**, *57*, 127. [\[CrossRef\]](#)
16. Preskill, J. Quantum Computing in the NISQ era and beyond. *Quantum* **2018**, *2*, 79. [\[CrossRef\]](#)
17. Peruzzo, A.; McClean, J.; Shadbolt, P.; Yung, M.H.; Zhou, X.Q.; Love, P.J.; Aspuru-Guzik, A.; O'Brien, J.L. A variational eigenvalue solver on a photonic quantum processor. *Nat. Commun.* **2014**, *5*, 4213. [\[CrossRef\]](#)
18. Cerezo, M.; Arrasmith, A.; Babbush, R.; Benjamin, S.C.; Endo, S.; Fujii, K.; McClean, J.R.; Mitarai, K.; Yuan, X.; Cincio, L.; et al. Variational Quantum Algorithms. *arXiv* **2020**, arXiv:2012.09265.
19. Bharti, K.; Cervera-Lierta, A.; Kyaw, T.H.; Haug, T.; Alperin-Lea, S.; Anand, A.; Degroote, M.; Heimonen, H.; Kottmann, J.S.; Menke, T.; et al. Noisy intermediate-scale quantum (NISQ) algorithms. *arXiv* **2021**, arXiv:2101.08448.
20. Mitarai, K.; Negoro, M.; Kitagawa, M.; Fujii, K. Quantum circuit learning. *Phys. Rev. A* **2018**, *98*, 032309. [\[CrossRef\]](#)
21. Schuld, M.; Bocharov, A.; Svore, K.; Wiebe, N. Circuit-centric quantum classifiers. *arXiv* **2018**, arXiv:1804.00633.
22. Farhi, E.; Neven, H. Classification with quantum neural networks on near term processors. *arXiv* **2018**, arXiv:1802.06002.
23. Benedetti, M.; Lloyd, E.; Sack, S.; Fiorentini, M. Parameterized quantum circuits as machine learning models. *Quantum Sci. Technol.* **2019**, *4*, 043001. [\[CrossRef\]](#)
24. Mari, A.; Bromley, T.R.; Izaac, J.; Schuld, M.; Killoran, N. Transfer learning in hybrid classical-quantum neural networks. *arXiv* **2019**, arXiv:1912.08278.
25. Abohashima, Z.; Elhosen, M.; Houssein, E.H.; Mohamed, W.M. Classification with Quantum Machine Learning: A Survey. *arXiv* **2020**, arXiv:2006.12270.
26. Easom-McCaldin, P.; Bouridane, A.; Belatreche, A.; Jiang, R. Towards Building A Facial Identification System Using Quantum Machine Learning Techniques. *arXiv* **2020**, arXiv:2008.12616.
27. Sarma, A.; Chatterjee, R.; Gili, K.; Yu, T. Quantum Unsupervised and Supervised Learning on Superconducting Processors. *arXiv* **2019**, arXiv:1909.04226.
28. Chen, S.Y.C.; Huang, C.M.; Hsing, C.W.; Kao, Y.J. Hybrid quantum-classical classifier based on tensor network and variational quantum circuit. *arXiv* **2020**, arXiv:2011.14651.
29. Stein, S.A.; Baheri, B.; Tischio, R.M.; Chen, Y.; Mao, Y.; Guan, Q.; Li, A.; Fang, B. A Hybrid System for Learning Classical Data in Quantum States. *arXiv* **2020**, arXiv:2012.00256.
30. Chen, S.Y.C.; Yoo, S.; Fang, Y.L.L. Quantum Long Short-Term Memory. *arXiv* **2020**, arXiv:2009.01783.
31. Kyriienko, O.; Paine, A.E.; Elfving, V.E. Solving nonlinear differential equations with differentiable quantum circuits. *arXiv* **2020**, arXiv:2011.10395.
32. Dallaire-Demers, P.L.; Killoran, N. Quantum generative adversarial networks. *Phys. Rev. A* **2018**, *98*, 012324. [\[CrossRef\]](#)
33. Li, J.; Topaloglu, R.; Ghosh, S. Quantum Generative Models for Small Molecule Drug Discovery. *arXiv* **2021**, arXiv:2101.03438.
34. Stein, S.A.; Baheri, B.; Tischio, R.M.; Mao, Y.; Guan, Q.; Li, A.; Fang, B.; Xu, S. QuGAN: A Generative Adversarial Network Through Quantum States. *arXiv* **2020**, arXiv:2010.09036.
35. Zoufal, C.; Lucchi, A.; Woerner, S. Quantum generative adversarial networks for learning and loading random distributions. *NPJ Quantum Inf.* **2019**, *5*, 1–9. [\[CrossRef\]](#)
36. Situ, H.; He, Z.; Li, L.; Zheng, S. Quantum generative adversarial network for generating discrete data. *arXiv* **2018**, arXiv:1807.01235.
37. Nakaji, K.; Yamamoto, N. Quantum semi-supervised generative adversarial network for enhanced data classification. *arXiv* **2020**, arXiv:2010.13727.
38. Lloyd, S.; Schuld, M.; Ijaz, A.; Izaac, J.; Killoran, N. Quantum embeddings for machine learning. *arXiv* **2020**, arXiv:2001.03622.
39. Nghiem, N.A.; Chen, S.Y.C.; Wei, T.C. A Unified Classification Framework with Quantum Metric Learning. *arXiv* **2020**, arXiv:2010.13186.
40. Chen, S.Y.C.; Yang, C.H.H.; Qi, J.; Chen, P.Y.; Ma, X.; Goan, H.S. Variational quantum circuits for deep reinforcement learning. *IEEE Access* **2020**, *8*, 141007–141024. [\[CrossRef\]](#)
41. Lockwood, O.; Si, M. Reinforcement Learning with Quantum Variational Circuit. In Proceedings of the 16th AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, Worcester, MA, USA, 19–23 October 2020; pp. 245–251.
42. Wu, S.; Jin, S.; Wen, D.; Wang, X. Quantum reinforcement learning in continuous action space. *arXiv* **2020**, arXiv:2012.10711.
43. Jerbi, S.; Trenkwalder, L.M.; Nautrup, H.P.; Briegel, H.J.; Dunjko, V. Quantum enhancements for deep reinforcement learning in large spaces. *arXiv* **2019**, arXiv:1910.12760.

44. CHEN, C.C.; SHIBA, K.; SOGABE, M.; SAKAMOTO, K.; SOGABE, T. Hybrid quantum-classical Ulam-von Neumann linear solver-based quantum dynamic programming algorithm. *Proc. Annu. Conf. JSAI* **2020**, JSAI2020, 2K6ES203. [\[CrossRef\]](#)
45. Bausch, J. Recurrent quantum neural networks. *arXiv* **2020**, arXiv:2006.14619.
46. Yang, C.H.H.; Qi, J.; Chen, S.Y.C.; Chen, P.Y.; Siniscalchi, S.M.; Ma, X.; Lee, C.H. Decentralizing Feature Extraction with Quantum Convolutional Neural Network for Automatic Speech Recognition. *arXiv* **2020**, arXiv:2010.13309.
47. Suzuki, K. Overview of deep learning in medical imaging. *Radiol. Phys. Technol.* **2017**, *10*, 257–273. [\[CrossRef\]](#) [\[PubMed\]](#)
48. Lundervold, A.S.; Lundervold, A. An overview of deep learning in medical imaging focusing on MRI. *Z. Med. Phys.* **2019**, *29*, 102–127. [\[CrossRef\]](#) [\[PubMed\]](#)
49. Deng, L.; Hinton, G.; Kingsbury, B. New types of deep neural network learning for speech recognition and related applications: An overview. In Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, Vancouver, BC, Canada, 26–31 May 2013; pp. 8599–8603.
50. Amodei, D.; Ananthanarayanan, S.; Anubhai, R.; Bai, J.; Battenberg, E.; Case, C.; Casper, J.; Catanzaro, B.; Cheng, Q.; Chen, G.; et al. Deep speech 2: End-to-end speech recognition in english and mandarin. In Proceedings of the International conference on machine learning, New York, NY, USA, 19–24 June 2016; pp. 173–182.
51. Hannun, A.; Case, C.; Casper, J.; Catanzaro, B.; Diamos, G.; Elsen, E.; Prenger, R.; Satheesh, S.; Sengupta, S.; Coates, A.; et al. Deep speech: Scaling up end-to-end speech recognition. *arXiv* **2014**, arXiv:1412.5567.
52. McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; y Arcas, B.A. Communication-efficient learning of deep networks from decentralized data. In Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, Fort Lauderdale, FL, USA, 9–11 May 2017; pp. 1273–1282.
53. Shokri, R.; Shmatikov, V. Privacy-preserving deep learning. In Proceedings of the 22nd ACM SIGSAC conference on Computer and Communications Security, Denver, CO, USA, 12–16 October 2015; pp. 1310–1321.
54. Kulkarni, V.; Kulkarni, M.; Pant, A. Survey of Personalization Techniques for Federated Learning. *arXiv* **2020**, arXiv:2003.08673.
55. Kairouz, P.; McMahan, H.B.; Avent, B.; Bellet, A.; Bennis, M.; Bhagoji, A.N.; Bonawitz, K.; Charles, Z.; Cormode, G.; Cummings, R.; et al. Advances and open problems in federated learning. *arXiv* **2019**, arXiv:1912.04977.
56. Lim, W.Y.B.; Luong, N.C.; Hoang, D.T.; Jiao, Y.; Liang, Y.C.; Yang, Q.; Niyato, D.; Miao, C. Federated learning in mobile edge networks: A comprehensive survey. *IEEE Commun. Surv. Tutor.* **2020**, *22*, 2031–2063 [\[CrossRef\]](#)
57. Yang, Q.; Liu, Y.; Chen, T.; Tong, Y. Federated machine learning: Concept and applications. *ACM Trans. Intell. Syst. Technol. (TIST)* **2019**, *10*, 1–19. [\[CrossRef\]](#)
58. Li, T.; Sahu, A.K.; Talwalkar, A.; Smith, V. Federated learning: Challenges, methods, and future directions. *IEEE Signal Process. Mag.* **2020**, *37*, 50–60. [\[CrossRef\]](#)
59. Li, Q.; Wen, Z.; He, B. Federated learning systems: Vision, hype and reality for data privacy and protection. *arXiv* **2019**, arXiv:1907.09693.
60. Wang, X.; Han, Y.; Leung, V.C.; Niyato, D.; Yan, X.; Chen, X. Convergence of edge computing and deep learning: A comprehensive survey. *IEEE Commun. Surv. Tutor.* **2020**, *22*, 869–904. [\[CrossRef\]](#)
61. Semwal, T.; Mulay, A.; Agrawal, A.M. FedPerf: A Practitioners' Guide to Performance of Federated Learning Algorithms. In Proceedings of the 34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, BC, Canada, 7–11 December 2020.
62. Sim, S.; Johnson, P.D.; Aspuru-Guzik, A. Expressibility and Entangling Capability of Parameterized Quantum Circuits for Hybrid Quantum-Classical Algorithms. *Adv. Quantum Technol.* **2019**, *2*, 1900070. [\[CrossRef\]](#)
63. Du, Y.; Hsieh, M.H.; Liu, T.; Tao, D. The expressive power of parameterized quantum circuits. *arXiv* **2018**, arXiv:1810.11922.
64. Abbas, A.; Sutter, D.; Zoufal, C.; Lucchi, A.; Figalli, A.; Woerner, S. The power of quantum neural networks. *arXiv* **2020**, arXiv:2011.00027.
65. Chen, S.Y.C.; Wei, T.C.; Zhang, C.; Yu, H.; Yoo, S. Quantum Convolutional Neural Networks for High Energy Physics Data Analysis. *arXiv* **2020**, arXiv:2012.12177.
66. Chen, S.Y.C.; Wei, T.C.; Zhang, C.; Yu, H.; Yoo, S. Hybrid Quantum-Classical Graph Convolutional Network. *arXiv* **2021**, arXiv:2101.06189.
67. Sierra-Sosa, D.; Arcila-Moreno, J.; Garcia-Zapirain, B.; Castillo-Olea, C.; Elmaghraby, A. Dementia Prediction Applying Variational Quantum Classifier. *arXiv* **2020**, arXiv:2007.08653.
68. Wu, S.L.; Chan, J.; Guan, W.; Sun, S.; Wang, A.; Zhou, C.; Livny, M.; Carminati, F.; Di Meglio, A.; Li, A.C.; et al. Application of Quantum Machine Learning using the Quantum Variational Classifier Method to High Energy Physics Analysis at the LHC on IBM Quantum Computer Simulator and Hardware with 10 qubits. *arXiv* **2020**, arXiv:2012.11560.
69. Jerbi, S.; Gyurik, C.; Marshall, S.; Briegel, H.J.; Dunjko, V. Variational quantum policies for reinforcement learning. *arXiv* **2021**, arXiv:2103.05577.
70. Takaki, Y.; Mitarai, K.; Negoro, M.; Fujii, K.; Kitagawa, M. Learning temporal data with variational quantum recurrent neural network. *arXiv* **2020**, arXiv:2012.11242.
71. Schuld, M.; Petruccione, F. Information Encoding. In *Supervised Learning with Quantum Computers*; Springer International Publishing: Cham, Switzerland, 2018; pp. 139–171. [\[CrossRef\]](#)
72. Schuld, M.; Bergholm, V.; Gogolin, C.; Izaac, J.; Killoran, N. Evaluating analytic gradients on quantum hardware. *Phys. Rev. A* **2019**, *99*, 032331. [\[CrossRef\]](#)



73. Bergholm, V.; Izaac, J.; Schuld, M.; Gogolin, C.; Blank, C.; McKiernan, K.; Killoran, N. PennyLane: Automatic differentiation of hybrid quantum-classical computations. *arXiv* **2018**, arXiv:1811.04968.
74. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. Pytorch: An imperative style, high-performance deep learning library. *arXiv* **2019**, arXiv:1912.01703.
75. Suzuki, Y.; Kawase, Y.; Masumura, Y.; Hiraga, Y.; Nakadai, M.; Chen, J.; Nakanishi, K.M.; Mitarai, K.; Imai, R.; Tamiya, S.; et al. Qulacs: A fast and versatile quantum circuit simulator for research purpose. *arXiv* **2020**, arXiv:2011.13524.
76. Elson, J.; Douceur, J.J.; Howell, J.; Saul, J. Asirra: A CAPTCHA that Exploits Interest-Aligned Manual Image Categorization. In Proceedings of 14th ACM Conference on Computer and Communications Security (CCS), Alexandria, VA, USA, 29 October–2 November 2007.
77. Krizhevsky, A. *Learning Multiple Layers of Features from Tiny Images*; Technical Report; CiteSeerX Publishing: Princeton, NJ, USA, 2009.
78. Dwork, C.; Roth, A. The algorithmic foundations of differential privacy. *Found. Trends Theor. Comput. Sci.* **2014**, *9*, 211–407. [[CrossRef](#)]
79. Fredrikson, M.; Jha, S.; Ristenpart, T. Model inversion attacks that exploit confidence information and basic countermeasures. In Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, Denver, CO, USA, 12–16 October 2015; pp. 1322–1333.
80. Shokri, R.; Stronati, M.; Song, C.; Shmatikov, V. Membership inference attacks against machine learning models. In Proceedings of the 2017 IEEE Symposium on Security and Privacy (SP), San Jose, CA, USA, 25 May 2017; pp. 3–18.
81. Abadi, M.; Chu, A.; Goodfellow, I.; McMahan, H.B.; Mironov, I.; Talwar, K.; Zhang, L. Deep learning with differential privacy. In Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, 24–28 October 2016; pp. 308–318.
82. Goryczka, S.; Xiong, L.; Sunderam, V. Secure multiparty aggregation with differential privacy: A comparative study. In Proceedings of the Joint EDBT/ICDT 2013 Workshops, Genoa, Italy, 18–22 March 2013; pp. 155–163.
83. Li, W.; Lu, S.; Deng, D.L. Quantum Private Distributed Learning Through Blind Quantum Computing. *arXiv* **2021**, arXiv:2103.08403.
84. Pillutla, K.; Kakade, S.M.; Harchaoui, Z. Robust aggregation for federated learning. *arXiv* **2019**, arXiv:1912.13445.
85. Ang, F.; Chen, L.; Zhao, N.; Chen, Y.; Wang, W.; Yu, F.R. Robust federated learning with noisy communication. *IEEE Trans. Commun.* **2020**, *68*, 3452–3464. [[CrossRef](#)]
86. Savazzi, S.; Nicoli, M.; Rampa, V. Federated learning with cooperating devices: A consensus approach for massive IoT networks. *IEEE Internet Things J.* **2020**, *7*, 4641–4654. [[CrossRef](#)]
87. Wittkopp, T.; Acker, A. Decentralized Federated Learning Preserves Model and Data Privacy. *arXiv* **2021**, arXiv:2102.00880.
88. Pokhrel, S.R.; Choi, J. A decentralized federated learning approach for connected autonomous vehicles. In Proceedings of the 2020 IEEE Wireless Communications and Networking Conference Workshops (WCNCW), Seoul, Korea, 25–28 May 2020; pp. 1–6.
89. Bonawitz, K.; Eichner, H.; Grieskamp, W.; Huba, D.; Ingerman, A.; Ivanov, V.; Kiddon, C.; Konečný, J.; Mazzocchi, S.; McMahan, H.B.; et al. Towards federated learning at scale: System design. *arXiv* **2019**, arXiv:1902.01046.
90. Xiao, Y.; Ye, Y.; Huang, S.; Hao, L.; Ma, Z.; Xiao, M.; Mumtaz, S. Fully Decentralized Federated Learning Based Beamforming Design for UAV Communications. *arXiv* **2020**, arXiv:2007.13614.
91. Lalitha, A.; Shekhar, S.; Javidi, T.; Koushanfar, F. Fully decentralized federated learning. In Proceedings of the Third workshop on Bayesian Deep Learning (NeurIPS), Montreal, QC, Canada, 7 December 2018.
92. Lu, S.; Zhang, Y.; Wang, Y. Decentralized federated learning for electronic health records. In Proceedings of the 2020 54th Annual Conference on Information Sciences and Systems (CISS), Princeton, NJ, USA, 18–20 March 2020; pp. 1–5.
93. Nakamoto, S. Bitcoin: A peer-to-peer electronic cash system. *Bitcoin. Org* **2008**, *4*. [[CrossRef](#)]
94. Zyskind, G.; Nathan, O. Decentralizing privacy: Using blockchain to protect personal data. In Proceedings of the 2015 IEEE Security and Privacy Workshops, San Jose, CA, USA, 21–22 May 2015; pp. 180–184.
95. Cai, W.; Wang, Z.; Ernst, J.B.; Hong, Z.; Feng, C.; Leung, V.C. Decentralized applications: The blockchain-empowered software system. *IEEE Access* **2018**, *6*, 53019–53033. [[CrossRef](#)]
96. Pandl, K.D.; Thiebes, S.; Schmidt-Kraepelin, M.; Sunyaev, A. On the convergence of artificial intelligence and distributed ledger technology: A scoping review and future research agenda. *IEEE Access* **2020**, *8*, 57075–57095. [[CrossRef](#)]
97. Qu, Y.; Gao, L.; Luan, T.H.; Xiang, Y.; Yu, S.; Li, B.; Zheng, G. Decentralized privacy using blockchain-enabled federated learning in fog computing. *IEEE Internet Things J.* **2020**, *7*, 5171–5183. [[CrossRef](#)]
98. Ramanan, P.; Nakayama, K. Baffle: Blockchain based aggregator free federated learning. In Proceedings of the 2020 IEEE International Conference on Blockchain (Blockchain), Rhodes Island, Greece, 2–6 November 2020; pp. 72–81.
99. Awan, S.; Li, F.; Luo, B.; Liu, M. Poster: A reliable and accountable privacy-preserving federated learning framework using the blockchain. In Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, London, UK, 11–15 November 2019; pp. 2561–2563.
100. Qu, Y.; Pokhrel, S.R.; Garg, S.; Gao, L.; Xiang, Y. A blockchained federated learning framework for cognitive computing in industry 4.0 networks. *IEEE Trans. Ind. Inf.* **2020**, *17*, 2964–2973 [[CrossRef](#)]
101. Zhao, Y.; Zhao, J.; Jiang, L.; Tan, R.; Niyato, D.; Li, Z.; Lyu, L.; Liu, Y. Privacy-preserving blockchain-based federated learning for IoT devices. *IEEE Internet Things J.* **2020**, *8*, 1817–1829. [[CrossRef](#)]

102. Bao, X.; Su, C.; Xiong, Y.; Huang, W.; Hu, Y. Flchain: A blockchain for auditable federated learning with trust and incentive. In Proceedings of the 2019 5th International Conference on Big Data Computing and Communications (BIGCOM), QingDao, China, 9–11 August 2019; pp. 151–159.
103. Kim, H.; Park, J.; Bennis, M.; Kim, S.L. Blockchain-based on-device federated learning. *IEEE Commun. Lett.* **2019**, *24*, 1279–1283. [\[CrossRef\]](#)
104. Liu, Y.; Ai, Z.; Sun, S.; Zhang, S.; Liu, Z.; Yu, H. Fedcoin: A peer-to-peer payment system for federated learning. In *Federated Learning*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 125–138.
105. Roy, A.G.; Siddiqui, S.; Pölsterl, S.; Navab, N.; Wachinger, C. Braintorrent: A peer-to-peer environment for decentralized federated learning. *arXiv* **2019**, arXiv:1905.06731.
106. Lalitha, A.; Kilinc, O.C.; Javidi, T.; Koushanfar, F. Peer-to-peer federated learning on graphs. *arXiv* **2019**, arXiv:1901.11173.
107. Hegedűs, I.; Berta, Á.; Kocsis, L.; Benczúr, A.A.; Jelasity, M. Robust decentralized low-rank matrix decomposition. *ACM Trans. Intell. Syst. Technol. (TIST)* **2016**, *7*, 1–24. [\[CrossRef\]](#)
108. Ormándi, R.; Hegedűs, I.; Jelasity, M. Gossip learning with linear models on fully distributed data. *Concurr. Comput. Pract. Exp.* **2013**, *25*, 556–571. [\[CrossRef\]](#)
109. Hegedűs, I.; Danner, G.; Jelasity, M. Gossip learning as a decentralized alternative to federated learning. In *IFIP International Conference on Distributed Applications and Interoperable Systems*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 74–90.
110. Hegedűs, I.; Danner, G.; Jelasity, M. Decentralized recommendation based on matrix factorization: a comparison of gossip and federated learning. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 317–332.
111. Hegedűs, I.; Danner, G.; Jelasity, M. Decentralized learning works: An empirical comparison of gossip learning and federated learning. *J. Parallel Distrib. Comput.* **2021**, *148*, 109–124. [\[CrossRef\]](#)
112. Hu, C.; Jiang, J.; Wang, Z. Decentralized federated learning: a segmented gossip approach. *arXiv* **2019**, arXiv:1908.07782.
113. Sergeev, A.; Balso, M.D. Horovod: fast and easy distributed deep learning in TensorFlow. *arXiv* **2018**, arXiv:1802.05799.
114. Chen, S.Y.C.; Huang, C.M.; Hsing, C.W.; Kao, Y.J. An end-to-end trainable hybrid classical-quantum classifier. *arXiv* **2021**, arXiv:2102.02416.
115. Cong, I.; Choi, S.; Lukin, M.D. Quantum convolutional neural networks. *Nat. Phys.* **2019**, *15*, 1273–1278. [\[CrossRef\]](#)
116. Li, Y.; Zhou, R.G.; Xu, R.; Luo, J.; Hu, W. A quantum deep convolutional neural network for image recognition. *Quantum Sci. Technol.* **2020**, *5*, 044003. [\[CrossRef\]](#)
117. Oh, S.; Choi, J.; Kim, J. A Tutorial on Quantum Convolutional Neural Networks (QCNN). *arXiv* **2020**, arXiv:2009.09423.
118. Kerenidis, I.; Landman, J.; Prakash, A. Quantum algorithms for deep convolutional neural networks. *arXiv* **2019**, arXiv:1911.01117.
119. Liu, J.; Lim, K.H.; Wood, K.L.; Huang, W.; Guo, C.; Huang, H.L. Hybrid Quantum-Classical Convolutional Neural Networks. *arXiv* **2019**, arXiv:1911.02998.
120. Qi, J.; Yang, C.H.H.; Tejedor, J. Submodular rank aggregation on score-based permutations for distributed automatic speech recognition. In Proceedings of the ICASSP 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 3517–3521.
121. Egger, D.J.; Gambella, C.; Marecek, J.; McFaddin, S.; Mevissen, M.; Raymond, R.; Simonetto, A.; Woerner, S.; Yndurain, E. Quantum computing for Finance: state of the art and future prospects. *IEEE Trans. Quantum Eng.* **2020**, *1*, 3101724. [\[CrossRef\]](#)