

Quantum-Enhanced Deep Learning-Based Lithology Interpretation From Well Logs

Naihao Liu^{1b}, Member, IEEE, Teng Huang, Jinghuai Gao^{1b}, Member, IEEE, Zongben Xu, Member, IEEE, Daxing Wang, and Fangyu Li^{1b}, Member, IEEE

Abstract—Lithology interpretation is important for understanding subsurface properties. Yet, the common manual well log interpretation is usually with low efficiency and bad consistency. Therefore, the automatic well log interpretation tools based on machine learning and deep learning have been developed. Although the state-of-the-art sophisticated models can show fine interpretation performance with acceptable accuracies, “blind” tests do not always exhibit satisfactory results because of the complexity of lithology interpretation with respect to subsurface rock properties and the data-labeling quality. To solve this generalization challenge, we propose to leverage the parameterized quantum circuits in the deep-learning model. The quantum computing takes advantages of the superposition and entanglement quantum systems, which could potentially endow the generalization power or capability to the deep-learning model. Using the proposed quantum-enhanced deep-learning (QEDL) model, we have tested the model performance on field well log data from different wells. Compared with the classic fine convolutional neural network (CNN) model and the long short-term memory (LSTM) model, the proposed QEDL model achieves comparable model performance with a clearly improved generalization power for interpreting both thin and thick lithology layers. In addition, because of the quantum circuit structure, the QEDL model needs much fewer model parameters than LSTM and CNN models, i.e., the QEDL parameter number in our study can be approximately 75% less than that of LSTM and 89% less than that of CNN.

Index Terms—Deep learning, lithology interpretation, quantum computing.

Manuscript received November 24, 2020; revised February 7, 2021 and May 23, 2021; accepted May 27, 2021. Date of publication June 15, 2021; date of current version January 12, 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 41904102, in part by the National Postdoctoral Program for Innovative Talents under Grant BX20190279, in part by the China Postdoctoral Science Foundation under Grant 2019M653584, and in part by the National Key Research and Development Program of China under Grant 2020YFA0713400 and Grant 2020YFA0713404. (Corresponding author: Fangyu Li.)

Naihao Liu, Teng Huang, and Jinghuai Gao are with the School of Information and Communications Engineering, Xi'an Jiaotong University, Xi'an 710049, China (e-mail: naihao_liu@mail.xjtu.edu.cn; 919369017@qq.com; jhgao@mail.xjtu.edu.cn).

Zongben Xu is with the School of Mathematics, Xi'an Jiaotong University, Xi'an 710049, China (e-mail: zbxu@mail.xjtu.edu.cn).

Daxing Wang is with the PetroChina Changqing Oilfield Company, Xi'an 710018, China, and also with the National Engineering Laboratory for Exploration and Development of Low Permeability Oil and Gas Fields, Xi'an 710018, China (e-mail: wdx1_cq@petrochina.com.cn).

Fangyu Li is with the Department of Electrical and Computer Engineering, Kennesaw State University, Marietta, GA 30060 USA (e-mail: fli6@kennesaw.edu).

Digital Object Identifier 10.1109/TGRS.2021.3085340

I. INTRODUCTION

QUANTUM-ENHANCED machine learning solves the machine-learning tasks by using the quantum algorithms, thereby improving classical machine-learning model performances. Typically, the quantum computing depends on the use of the engineered quantum system, which is based on the quantum behavior phenomena, such as the entanglement and superposition, resulting in a system “difficult to simulate” by a classical computer [1]. In terms of the quantum computing embedded machine learning, although multiple research works have proposed the quantum neural networks (QNNs) by using a quantum perceptron [2]–[5], the real implementation was not broadly available until recently. In early 2020, Google introduced the TensorFlow Quantum (TFQ) library [6] for rapid prototyping of the hybrid quantum-classical models based on the quantum circuit simulation framework Cirq [7]. Thanks to the Cirq, it is possible to avoid encoding the input classical data into a real quantum computer to operate the quantum algorithms. Because of the TFQ, the quantum machine-learning algorithms, such as the QNN, are not purely theoretical anymore. We can implement the quantum computing enhanced algorithms on classical computers to experience the advantages brought by the quantum computing.

The lithology interpretation by using well logs extracts reservoir parameters and provides the foundation for geological research studies [8]. By utilizing well log data, many mathematical algorithms have been adopted to interpret lithology [9], [10]. The cross-plotting and statistical analysis-based approaches are two main traditional approaches for the lithology prediction by using well logs. We can adopt two or more well logs for the lithology interpretation through the cross-plot technique [11]. The multivariate statistical analysis-based approaches have also been widely adopted for the lithology interpretation, such as discriminant function analysis (DFA) [12] and principal component analysis (PCA) [13]. Nevertheless, these methods depend on plenty of samples, which is not only a time-consuming work but also difficult to be obtained in exploration geophysics. Moreover, well logs are affected by the rock properties, such as the effective pressure, saturation levels, and pore fluids [14]. In consequence, the lithology interpretation is a typical nonlinear issue and difficult to be effectively solved by using these traditional methods. Hence, we can develop an appropriate and effective nonlinear method for solving the lithology interpretation issue [15].

Recently, there are machine-learning-based approaches employed to solve the lithology interpretation issues, such as the random forest (RF)-based classifiers [16], support vector machine (SVM) [17], and neural networks (NNs) [18]. Among these machine-learning-based techniques, the fully connected (FC) NNs are adopted to predict lithology by using well logs [19], such as the artificial NN (ANN)- and back-propagation NN (BPNN)-based models. For the prediction of reservoir parameters, these NNs only adopt well log curves of the corresponding depth. Hence, the influence of well logs before and after the specific depth are neglected, and the predicted lithology results cannot guarantee the dependability and authenticity. Moreover, when processing huge well log data, these NNs are easy to fall into local minima, resulting in a low accuracy. In addition, a disadvantage of the FC NNs is that the extracted information will not be transmitted between the same layer because there is no connection between neurons in the same layer. Hence, it is difficult to accurately and effectively predict lithology by using the FC NNs based on well logs. There are also many improved works about the FC NNs for solving the lithology prediction issue, but the implementation process is very complex [15], [20].

With the development of the convolutional NNs (CNN), there are many CNN-based models for seismic interpretation [21]–[23]. However, it seems only a few researches for the lithology interpretation are based on CNN [24]. A deep residual CNN is introduced to automatically identify lithological facies by using wellbore image logs [25]. A hybrid model is designed for the prediction of lithology/fluid properties [26], which combines the time-frequency analysis and 1-D CNN (1D-CNN). As state-of-the-art NN-based approaches, the CNN models are data-driven and nonlinear, therefore suitable for solving the geological issues [27], [28]. However, the traditional CNN models lack the generalization property when dealing with a limited dataset (i.e., manually interpreted lithology labels) [29], [30], which is also because of the complexity of the lithology interpretation with respect to the subsurface rock properties and the data-labeling quality [15].

Besides the networks mentioned above, the recurrent NN (RNN) is a specialized one, designed for processing sequential signals [31]. The RNN has drawn wide attention because of its good performances in the modeling of sequential data, such as handwriting recognition [32], streaming sensor data analysis [33], time series prediction [34], speech recognition [35], and text classification [36], which has also been applied in geoscience fields [37], [38]. Tian *et al.* [39] combined the hidden Markov models and RNNs to classify lithologies by taking the spatial context into account. Imamverdiyev and Sukhostat compared the effectiveness of the CNNs and RNNs for lithology facies classification [24]. They concluded that the proposed CNN model provides more accurate results than the RNN model. Moreover, the RNNs are difficult to train mainly because of the exploding and vanishing gradient issues, especially when handling long sequences [36], [40]. There are also several variations of the RNNs developed to solve these issues, such as gated recurrent unit (GRU) [41], [42] and long short-term memory (LSTM) [15], [43].

Meanwhile, in real applications, we aim to apply a well-trained model to other well logs, which are not used in the training dataset. This is because the model generalization is important for the lithology interpretation in practice. To improve the generalization in the CNN-based models, often large training datasets are used [44]. Nevertheless, obtaining sufficient training data and corresponding labels is extremely time-consuming and expensive, if not impossible.

Thanks to the quantum computing nature, the QNN has an advanced generalization capacity [6], which means the initialized and well-trained networks can find approximate solutions, which generalize well to unseen/untrained samples. The classical machine-learning and deep-learning methods could be possibly enhanced by incorporating the parameterized quantum circuits (PQCs), especially in terms of the generalization power. The generalization power is extremely important for complex problems, such as the lithology interpretation where training data and labels are limited for a specific survey.

Here, we propose a quantum-enhanced deep-learning (QEDL) model for the lithology interpretation. The quantum computing takes advantages of the superposition and entanglement quantum systems. In this article, we cover the quantum computing basics, quantum operations, and QNN principles. By introducing the leverage of the PQC in the deep-learning model, we could potentially endow the generalization power or capability to the deep-learning model. The extensive quantitative analysis of field well logs from different wells demonstrates the promising performances of the trained QEDL model. Compared with the conventional CNN model and the LSTM model, our QEDL model shows superior behaviors in terms of the efficiency, convergence rate, model complexity, accuracy, and generalization power. The contributions of our work are as follows:

- 1) To the best of our knowledge, this is the first time to incorporate the powerful quantum computing in the lithology interpretation.
- 2) The proposed QEDL model significantly improves the deep-learning generalization capacity, which is highly desired for complicated geoscience applications.
- 3) Our QEDL model will benefit the following quantum computing related geophysical data-processing works in terms of the motivation, solution, model architecture, result evaluation, and applications.

The rest of this article is organized as follows: We first introduce several related works about the machine-learning-based lithology interpretation in Section II. Then, we present the QEDL-based model with the design of the architecture, loss function, and related theoretical principles in Section III. (It should be noticed that related technical backgrounds are summarized in Appendices VI to VI-C.) In Section IV, we use a field application to analyze the performances of our proposed model explicitly and quantitatively, and make detailed comparisons with the CNN and LSTM models. In Section V, the benefits and limitations of the proposed model are discussed in detail. In Section VI, conclusion is finally drawn.

II. MACHINE-LEARNING-BASED LITHOLOGY INTERPRETATION

A. SVM and ANN

SVM was proposed for solving the pattern recognition issues [45], which distinguishes different classes by constructing a hyperplane or setting of hyperplanes [46]. To solve a specific pattern recognition issue, we first assume the training pairs $\{(x_1, y_1), (x_2, y_2), \dots, (x_l, y_l)\}$, $x_l \in \mathcal{R}^N$, and $y_l \in (0, 1)$, where x_l and y_l denote the l th sample and its label, l is the number of the input samples. We then write the SVM as follows:

$$\max W(\alpha) = \sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{i=1}^l \sum_{j=1}^l \alpha_i y_i \alpha_j y_j K(x_i, x_j) \quad (1)$$

$$\text{s.t. } \sum_{i=1}^l \alpha_i y_i = 0, \quad \alpha_i \in [0, C], \quad i = 1, \dots, l \quad (2)$$

where $K(x_i, x_j)$ denotes a support vector (SV) kernel, $C > 0$ is a penalty parameter, α_i and α_j , $i, j = 1, \dots, l$, are the Lagrange multipliers. Then, the decision function is defined as

$$f(\mathbf{x}) = \text{sgn} \left(\sum_{i=1}^l \alpha_i y_i K(x_i, x) + b \right) \quad (3)$$

where sgn denotes the sign function and b is a unique constant.

Using (1)–(3), the training samples in the input space is mapped into a feature map. It should be noted that the mapped samples in the feature map are linearly separable [47]. Moreover, to determine the final results, the SVM achieves good robustness by only using a small number of the SVs. This not only saves lots of calculation time by reducing a large number of the redundant samples but also helps capture the key features [48]. Hence, the SVM is often adopted to solve the binary classification issues. However, the SVM-based models suffer from several limitations. The most obvious drawback is that it apparently only handles the binary classification issues [49]. Definitely, we have to train multiseperated SVMs when recognizing a multiclass classification, which is time consuming when handling a large dataset. Moreover, the SVM does not always perform well when the target classes are seriously overlapping.

The ANN is introduced to solve the complex classification problems by mimicking the behavior of the human brain and nervous system to learn from the given training samples [50], [51]. We briefly introduce the ANN as follows [52]:

$$y_j = \phi \left(c_j + \sum_{k=1}^K v_{kj} \left(\phi \left(\sum_{l=1}^L w_{lk} x_l + b_k \right) \right) \right). \quad (4)$$

In (4), y_j denotes the j th component of the output \mathbf{y} . x_l , $l = 1, 2, \dots, L$ is the l th input component where M denotes the input number. K is the number of hidden layer neurons. c_j and b_k denote the biases of the j th output node and k th hidden node. $\phi(\cdot)$ denote the activation function between the hidden and output layers, whereas $\varphi(\cdot)$ is the activation function between the input and hidden layers. v_{kj} is the weight between the k th hidden node and j th output node, whereas w_{lk} is the weight between the l th input and k th hidden node.

Using (4), the ANN learns from the given training samples and use these training samples to adjust its weights w_{lk} and v_{kj} . Then, it captures the relationship between the input samples and corresponding outputs [53]. Therefore, the ANN-based models do not need any prior knowledge about the nature of the relationship between the input and output samples. Nevertheless, the ANN-based models also suffer from several unsolved issues, such as the vanishing gradient and overfitting [54], [55], which often result in a low accuracy. Moreover, it should be noted that the ANN is one of the typical feedforward networks. Hence, the captured features by using the ANN cannot be transferred between the same layer because there is no connection between neurons in the same layer, which makes it inefficiently predict lithology by using well log data.

B. CNN

The CNN is a widely used machine-learning-based tool for the classification tasks, such as seismic fault detection [28], [56] and image recognition [57], [58]. Generally, the CNN is composed of a sequence of interleaved layers. An intermediate 2-D array of pixels in each layer, termed as a feature map, is produced from the previous one, presented in Fig. 1. Although typically there are multiple feature maps in the CNN layers, we only consider one feature map for one layer for simplicity.

The convolution layer calculates the pixel values $x_{ij}^{(\ell)}$ from a linear combination of the nearby pixel values, defined as

$$x_{i,j}^{(\ell)} = \sum_{a,b=1}^w w_{a,b} x_{i+a,j+b}^{(\ell-1)} \quad (5)$$

where $w_{a,b}$ denote the weights.

The Max-pooling layer takes the maximum value from a few contiguous pixels, which helps to reduce the feature map size. A nonlinear (activation) function often follows a pooling layer. Here, we use the softmax function in our model. Variables are fixed for a specific CNN [58], such as the number of convolution and pooling layers, and the size of the weight matrices $w_{a,b}$ (i.e., hyperparameters). The CNN's key properties are thus translationally invariant convolution and pooling layers, each of them characterized by a constant number of parameters.

In this study, we adopt a simplified CNN-based architecture to interpret lithology by using well logs. Fig. 1 shows the simplified architecture of the CNN-based lithology interpretation adopted in this study. It can be seen that each convolutional layer is followed by a Softmax operator. Additionally, we introduce a dropout (0.5) to avoid the overfitting. By using the features extracted by the previous convolutional layers, a followed FC layer outputs the classification labels which the model is trying to predict. It should be noted that the hyperparameters of the used CNN model in our study are elaborated in Section IV-B.

III. QUANTUM-ENHANCED NNs

Using the quantum search and piecewise weight learning, a QNN model based on quantum circuit gates is proposed in [59]. A quantum auto-encoder for the quantum

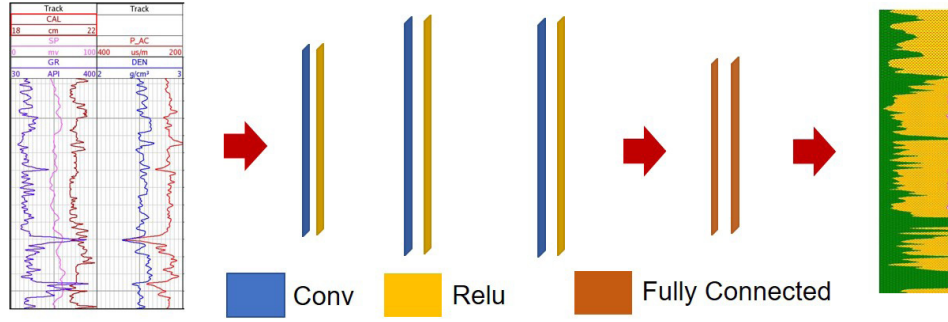


Fig. 1. Simplified illustration of CNN-based lithology analysis. A sequence of processing layers (convolution and softmax) transforms input well logs into a probability distribution of the rock types, which is used to predict the lithology.

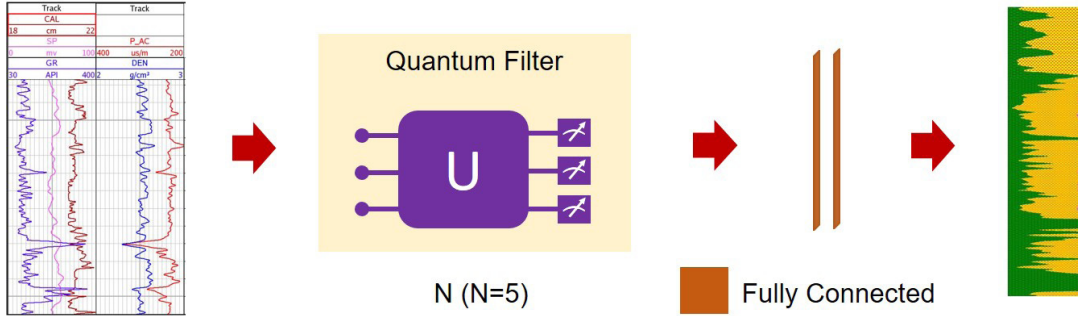


Fig. 2. Simplified illustration of the proposed QEDL model. Compared with Fig. 1, a quantum filter (circuit structure) is inserted to replace the conventional convolution filters.

state compression is introduced in [60]. The hybrid models with the quantum and classical approaches usually have the computations carried on the quantum-engineered devices and classical computers [61]–[64]. Reversible PQC have been widely adopted to construct the QNN with a back-propagation-kind training strategy. Considering the presentation brevity, we put the basics of the quantum computing in Appendices VI to VI-C.

A. General QNNs

Classical NNs build a relationship between the input (x_1, x_2, \dots, x_d) and output y as $(x_1, x_2, \dots, x_d) \rightarrow y$. Whereas QNNs can be generally expressed as a quantum computation with an additional operation based on a permutation matrix¹ to convert the mapping relation to $(x_1, x_2, \dots, x_d, 0) \rightarrow (x'_1, x'_2, \dots, x'_d, y)$ [65]. We can represent the input component of the vector (x_1, \dots, x_d) through a quantum state $|\psi\rangle_{1,\dots,d}$ to ensure the input representation unitary. Thus, in terms of the quantum computations, the transformation is expressed as

$$|\psi\rangle_{1,\dots,d}|0\rangle \rightarrow |\psi'\rangle_{1,\dots,d}|y\rangle \quad (6)$$

where $|\psi\rangle_{1,\dots,d}$ denote input in qubits (more details can be found in Appendix VI), for which a $d + 1$ bit unitary matrix is applied to transform it to the output $|\psi'\rangle_{1,\dots,d}|y\rangle$.

¹Mathematically, a permutation matrix P needs to be a square matrix with either 0 or 1 as elements and $PP^T = I$.

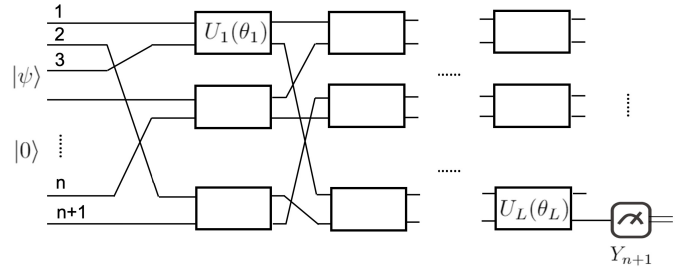


Fig. 3. Illustration of a L -layered QNN where the input is $|\psi\rangle|0\rangle$ and the output is measured through a Pauli- y operator, which is defined in (VI.5) on the ancillary bit.

To capture detailed input information, a L layer QNN can be implemented following the classical deep NNs with a set of variational circuits:

$$U(\Theta) = U_L(\theta_L)U_{L-1}(\theta_{L-1}) \dots U_1(\theta_1) \quad (7)$$

where the i th layer unitary matrix $U_i(\theta_i)$ has a set of the model parameters θ_i , as shown in Fig. 3. $\Theta = \{\theta_1, \dots, \theta_L\}$ represents all parameters.

B. Loss Function

Because the lithology interpretation can be formulated as a classification problem, here, we define the loss function of the QNN. Given the training data $\{(|\psi_m\rangle, y_{m,n}) : m = 1, 2, \dots, M, n = 1, 2, \dots, N\}$, where input states $|\psi_m\rangle$ are

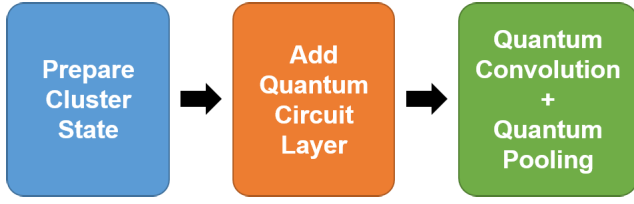


Fig. 4. Implementation of the QEDL using the TFQ typically follows this architecture. First, we prepare a cluster state, which is shown in Fig. 5. The rest of the model's quantum components are constructed with adding the quantum circuit layers, as shown in Fig. 6. Then parameterized quantum circuits (PQCs) or multilayer QNNs (MQNNs) are used to implement the quantum convolution and quantum pooling, as shown in Fig. 7.

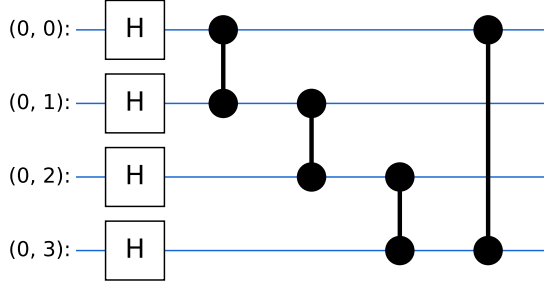


Fig. 5. Cluster state is a type of highly entangled state of the multiple qubits [66]. The cluster states are generated in lattices of the qubits with the Ising-type interactions. 1-D cluster states ($d = 1$) example with $n = 4$, where n is the number of qubits. $|\phi\rangle = (1/2)(|+0+0\rangle + |+0-1\rangle + |-1-0\rangle + |-1+1\rangle)$.

associated with the binary classification labels $y_{m,n} = 0$ or 1 , the categorical cross entropy (CCE) loss can be defined as

$$\text{CCE} = - \sum_{m=1}^M \sum_{n=1}^N (\hat{y}_{m,n} * \log y_{m,n}) \quad (8)$$

where $m = 1, 2, \dots, M$ and $n = 1, 2, \dots, N$ denote the number of samples and categories, respectively. $\hat{y}_{m,n}$ represents the expected output label corresponding to the input $|\psi_m\rangle$.

C. TFQ-Based QEDL

Motivated by the CNN architecture shown in Fig. 1, we propose a quantum circuit-embedded hybrid deep-learning model, shown in Fig. 2. The proposed QEDL model implementation using the TFQ typically follows the architecture shown in Fig. 4. In terms of the model hyper parameters, the numbers of the quantum convolution and pooling layers are determined by trials, whereas the unitary operator properties are learned during the training process.

The TFQ library is based on the Cirq, which is an open-source framework [7]. There are basic quantum circuit structures, including the quantum gates, qubits, measurement operators, and so on. The TFQ-designed quantum computations, which are operated in simulations, can also be executed on real quantum systems. The TFQ provides the cluster states (e.g., Fig. 5), layer classes (e.g., Fig. 6), and quantum convolution/pooling layers (e.g., Fig. 7) designed for in-graph circuit construction. The hyperparameters of the proposed QEDL model are elaborated in Section IV-B.

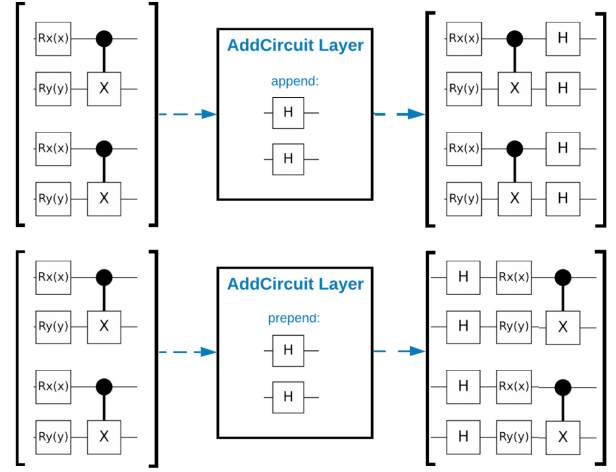


Fig. 6. `tfq.layers.AddCircuit` function in the TFQ can either prepend or append to the input batch of the quantum circuits.

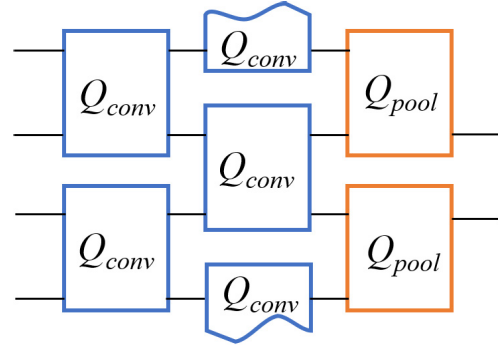


Fig. 7. Combination of the quantum convolution Q_{conv} (blue) on 4 qubits and quantum pooling Q_{pool} (orange) that reduces the system size from 4 qubits to 2 qubits.

D. Comparisons Between CNN and QEDL

The proposed QEDL model in Fig. 2 is also based on the CNN model in Fig. 1, but with a quantum filter. Hence, the QEDL model can naturally solve the multiclass classification issues. In addition, different approaches for training deep networks have been studied and applied to try to address the vanishing gradient and overfitting issues, such as the pretraining, better random initialization, better optimization methods, specific architectures, alternative activation functions, more sophisticated training, and testing strategy, and so on. These are the advantages brought from the deep-learning models and CNN models. And in Section IV-D, we carry out a comprehensive model evaluation. Moreover, by adopting the superposition and entanglement quantum systems, the quantum computing could potentially endow the generalization power to the deep-learning model. Therefore, the proposed algorithm could provide a robust model with a powerful generalization property, which benefits for the lithology interpretation based on well logs. It should be noted that there have been attempts to improve the model generalization, such as the transfer learning [67]. We will have a detailed discussion addressing the difference between the transfer learning and QEDL in Section V.

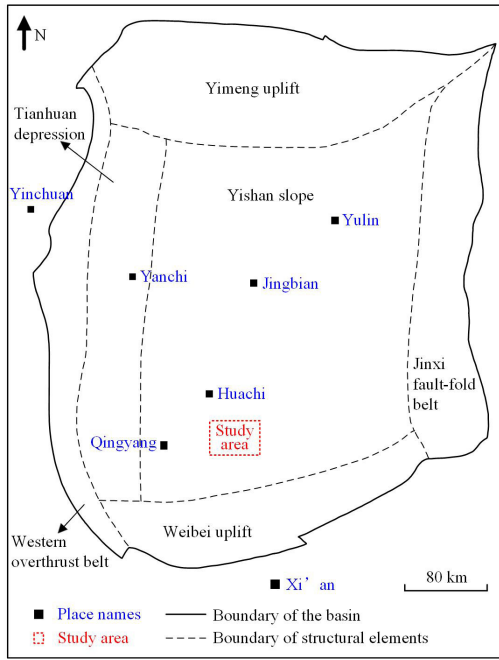


Fig. 8. Study area locates at the Ordos basin, Northwest of China, indicated by the red dotted rectangle.

IV. FIELD DATA APPLICATIONS

A. Study Area and Well Logs

The study area locates at the Ordos basin, Northwest of China, with an area of 4910 km², indicated in Fig. 8. The study area is near the western margin of the North China Craton, which is a low-porosity and low-permeability gas reservoir [68]. It should be noted that the sedimentary strata of the study area consist of the marine sediment in the bottom strata and continental deposit in the upper, in which the Mesoproterozoic mainly consists of continental clastic deposits [15]. Moreover, there mainly exists three types of lithologies in the studied reservoir area, i.e., mudstone, sand-mudstone, and sandstone.

Because of the lack of manual interpretation results and the confidentiality agreement, we cannot obtain sufficient well log data in this study area. Hence, in this study, we only introduce four well borehole data to testify the performance and generalization property of the QEDL model and make detailed comparisons with the state-of-the-art CNN and LSTM models. Fig. 9 shows a seismic time slice at 950 ms with four well boreholes denoted by the red dots. W1, W2, W3, and W4 denote four well boreholes located at this seismic survey, whereas the red line presents a seismic section cross four well boreholes. We then extract the 2-D seismic section cross four wells, shown in Fig. 10. The red lines in Fig. 10 denote the well boreholes and the blue curves present the synthetic seismograms [69].

There are often dozens of well log curves. In this study, we select five of them as the training data to interpret the lithology, i.e., acoustic (AC), borehole diameter (CAL), density (DEN), natural gamma ray (GR), and spontaneous potential (SP). The reason to select these logs is that they

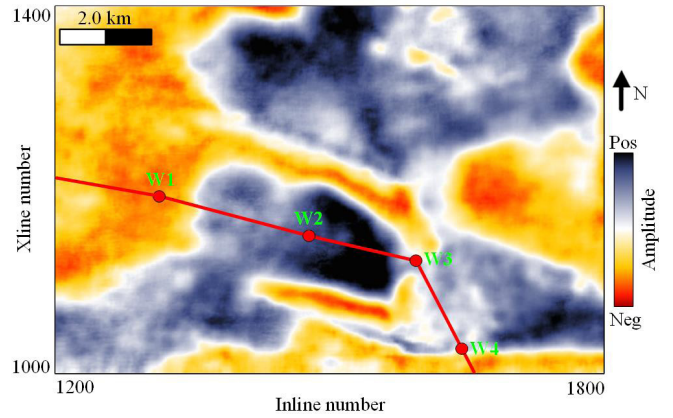


Fig. 9. Seismic time slice at 950 ms with four well boreholes denoted by the red dots. W1, W2, W3, and W4 denote four well boreholes. The red line presents a seismic section cross four well boreholes.

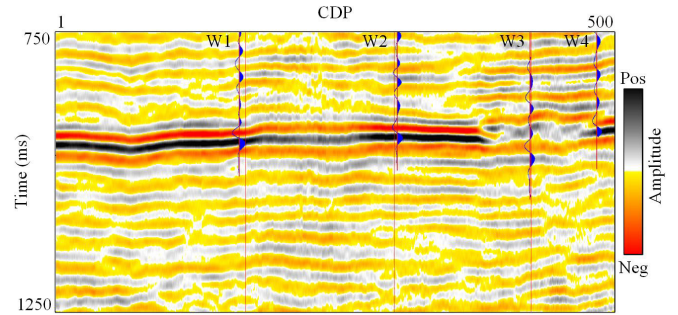


Fig. 10. 2-D seismic section cross four well boreholes, presented by the red line in Fig. 9. W1, W2, W3, and W4 denote four well boreholes. The blue curves present the synthetic seismograms.

are highly related to the key parameters of the formation [14], [15]. Fig. 11 shows part of well log curves extracted from W1, i.e., the training datasets used in this study. It should be noted that the sampling interval of each well log curve is 0.125 m. The right image in Fig. 11 denotes the corresponding manual lithology interpretation result, i.e., the training labels (mudstone, sand-mudstone, and sandstone) used in this study. Afterward, we use well log curves and corresponding labels of W1 and W4 to train the LSTM, CNN, and QEDL models, and then apply three well-trained models to predict the lithologies of W2 and W3.

B. Hyperparameters of Different Deep-Learning Models

For the CNN model, the sizes of the convolution layers are 32, 64, 64 in Fig. 1. For the proposed QEDL model, we introduce a quantum layer to replace the convolution layers in the CNN model. It should be noticed that we adopt the FC layers for the CNN and QEDL models, which combine all the features extracted by the former layers and then classify. The two models in this study are both built with the TensorFlow on Python 3.6. The learning rate is initiated at 0.001. The models are both trained with a batch size of 20. It should be noted that the details of the hyperparameters of two models

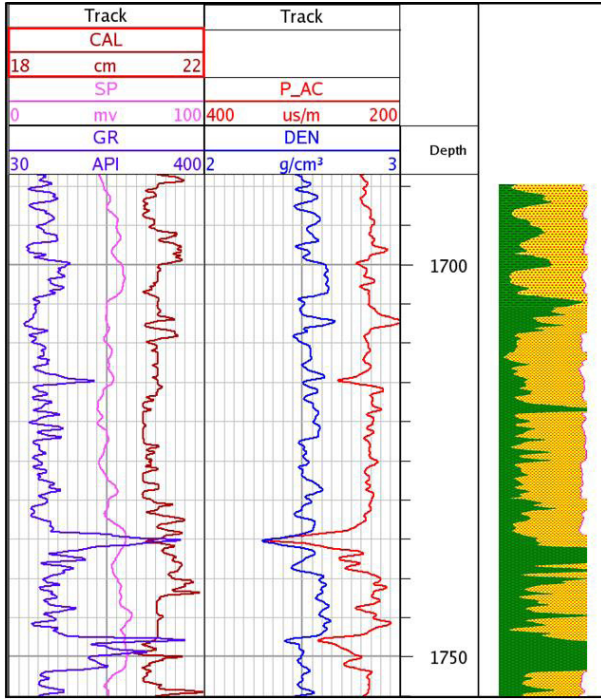


Fig. 11. Cropped five well logs (AC, CAL, DEN, GR, and SP) and its corresponding manual lithology interpretation (right image).

TABLE I
PARAMETERS OF THE CNN AND QEDL ARCHITECTURES

Network name	Number of Quantum filters	Number of Conv filters
CNN	\	32+64+64
QEDL	5	\

are given in Table I. Moreover, it should be noticed that the quantum filters are with five qubits in this study. In addition, we introduce an LSTM network as a comparison. More details about the architecture and training strategy of the used LSTM model can be found in [15].

To build the training, validation, and prediction datasets, we first select two wells (e.g., W1 and W4) from these four wells as the training and validation dataset, whereas the other two (e.g., W2 and W3) are selected as the testing (prediction) dataset. Moreover, we repeat this process for six times, i.e., selecting two wells as the training and validation datasets and the other two wells as the testing (prediction) dataset. It should be noticed that the LSTM, CNN, and QEDL networks are initialized randomly for each process. Then, we divide the selected two well logs (e.g., W1 and W4) as the training and validation datasets. The ratio of the training dataset to the total dataset is 0.7 and the rest (0.3) is selected as the validation dataset. The training and validation datasets are used to train and validate the LSTM, CNN, and QEDL models. The other two well logs (e.g., W2 and W3) are selected as the blind testing dataset to evaluate the performance of three well-trained models, especially the generalization capacity. It should be noticed that we have also implemented the k -fold ($k = 10$ in our study) cross validation [10], [23] to test the robustness and avoid the data bias. k -fold cross validation

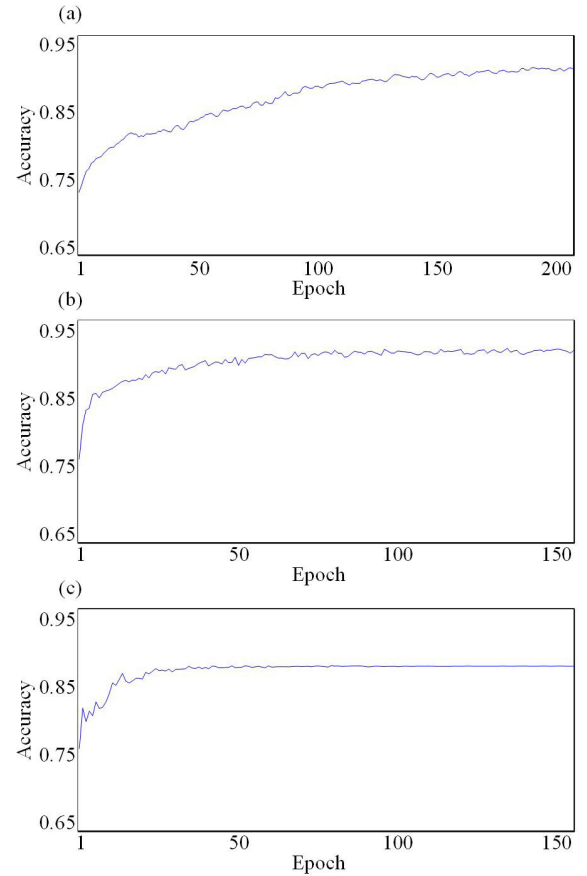


Fig. 12. Validation accuracy curves provided by (a) LSTM-, (b) CNN-, and (c) QEDL-based models, respectively.

TABLE II
DEFINITION OF THE CONFUSION MATRIX

		Predicted	
		Negative	Positive
Actual	Negative	True Negative (TN)	False Positive (FP)
	Positive	False Negative (FN)	True Positive (TP)

is a standard procedure for evaluating the performance of the classification methods [70]–[72]. In this study, for each fold, we randomly divide the dataset (e.g., W1 and W4) into the training and validation datasets by a fixed ratio as 7:3. For example, for each fold, we compute a validation accuracy curve by randomly selecting the training and validation datasets by a fixed ratio as 7:3. After obtaining 6*10 validation accuracy curves, we can easily calculate an average validation accuracy curve, shown in Fig. 12. Moreover, we compute the quantitative evaluation parameters in Tables III–V by following this way:

C. Well Log Data Preprocessing

The input data of the LSTM, CNN, and QEDL models need to be standardized. In this study, we introduce the Max-Min standardization method to standardize the well log

TABLE III

Precisions OF THE TRAINING/VALIDATION/PREDICTION DATASETS

Models	training	validation	prediction
LSTM	0.907	0.873	0.865
CNN	0.903	0.871	0.874
QEDL	0.866	0.858	0.863

TABLE IV

Precision, Recall, AND $F1$ Score OF THE VALIDATION DATASET

Models	precision	recall	$F1$ score
LSTM	0.870	0.840	0.854
CNN	0.852	0.847	0.853
QEDL	0.864	0.853	0.861

TABLE V

Precision, Recall, AND $F1$ Score OF THE PREDICTION DATASET

Models	precision	recall	$F1$ score
LSTM	0.873	0.870	0.872
CNN	0.874	0.872	0.873
QEDL	0.876	0.874	0.877

data, defined as

$$\hat{x}_i = \frac{x_i - \min_{1 \leq j \leq n} \{x_j\}}{\max_{1 \leq j \leq n} \{x_j\} - \min_{1 \leq j \leq n} \{x_j\}} \quad (9)$$

where x_j and \hat{x}_i are well log data before, and after the standardization, $\min_{1 \leq j \leq n} \{x_j\}$ and $\max_{1 \leq j \leq n} \{x_j\}$ denote the minimum and maximum values of the processed well log data x_j , and n is the length of the processed well log data x_j .

D. Model Evaluation

After training a model, it is important to evaluate its performance [51], which helps in predicting its future accuracy. In this study, we adopt the *precision*, *recall*, and *F1 score* to evaluate the performance of a trained model. To define these three parameters, we first introduce the confusion matrix in Table II. The true negative (TN), false negative (FN), true positive (TP), and false positive (FP) are defined in Table II based on the predicted results.

Then, the *precision*, *recall*, and *F1 score* are defined in (10)–(12), respectively. It should be noted that the higher the *precision* is, the stronger the ability of the model to recognize the negative samples. This means that the *precision* reflects the ability of the trained model to recognize the negative samples. Moreover, the higher the *recall* is, the stronger the ability of the model to recognize the positive samples. That is, the *recall* reflects the ability of the model to distinguish the positive samples [55]. The *F1 score* combines the *precision* and *recall*, which would be used for evaluating the trained models in this study. The higher the *F1 score* is, the more robust the

trained model.

$$\text{precision} = \frac{TP}{TP + FP} \quad (10)$$

$$\text{recall} = \frac{TP}{TP + FN} \quad (11)$$

$$F1 = 2 \cdot \frac{1}{\frac{1}{\text{recall}} + \frac{1}{\text{precision}}} = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \quad (12)$$

E. Results and Discussions

We first show the validation accuracy curves provided by the LSTM-, CNN-, and QEDL-based models, as shown in Fig. 12. Although the LSTM and CNN models achieve higher validation accuracy values than the QEDL model, all three models achieve high validation accuracy values. This seems that we obtain three well-trained models provided by the LSTM-, CNN-, and QEDL-based networks. Nonetheless, it should be noted that the QEDL model turns to be stable at about 40 epochs, whereas the LSTM and CNN models do not tend to be stable, even at 140 epochs. This means that it takes fewer epochs to obtain a well-trained QEDL model than well-trained LSTM and CNN models.

Then, we make detailed quantitative comparisons between the LSTM, CNN, and QEDL models. We first compare the *precisions* of the training and validation datasets calculated by using the LSTM, CNN, and QEDL models, presented in Table III. Moreover, we calculate the prediction *precisions* of the prediction dataset (i.e., the well logs of W2 and W3) by using three well-trained models. By comparing the *precisions* in Table III, we have two main observations. First, the LSTM and CNN models achieve higher training and validation *precisions* than the QEDL model. This seems that the LSTM and CNN models are two better-trained models than the QEDL model. Second, we find that the training *precisions* of the LSTM and CNN models are distinctly higher than their validation and prediction *precisions*. This means that the LSTM and CNN models may be severely overfitting. Although the training and validation *precisions* of the proposed model is lower than those of the LSTM and CNN models, the prediction *precision* of the QEDL model is closer to its training and validation *precisions*. It can be concluded that our QEDL model achieves better performance than the LSTM and CNN models for the lithology interpretation in terms of the generalization, especially for avoiding the overfitting, which benefits when a well-trained model needs to be applied to new data.

In this study, the lithology interpretation task is regarded as a three-classification issue. Hence, when retrieving the multiple class labels, we should average the evaluation measures to give a view on the general results. There are two approaches for averaging the results, i.e., macro- and microaveraging. The former adopts equal weights to the general scores generated from each individual category, whereas the latter is mainly dominated by the categories with more positive training samples [73]. We adopt microaveraging to calculate the scores for quantitative comparisons in this study. Afterward, we compare

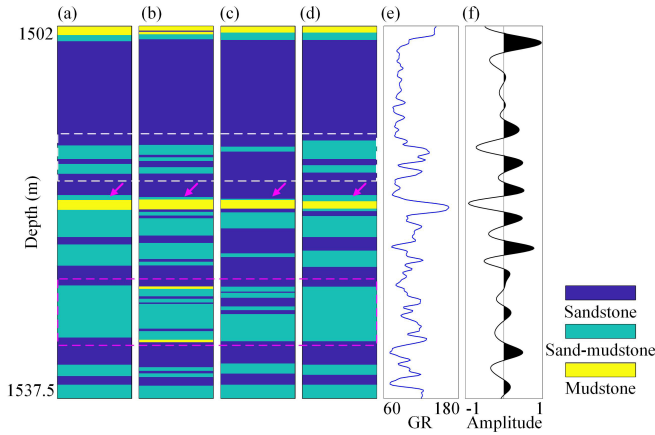


Fig. 13. (a) True lithology; interpreted lithology predicted by (b) LSTM-, (c) CNN-, and (d) QEDL-based models; and (e) and (f) denote the GR curve and normalized borehole-side seismic trace, respectively.

the *precision*, *recall*, and *F1 score* of the validation dataset calculated by using microaveraging, summarized in Table IV. It can be noticed that the *precision* provided by the QEDL model is higher than the one of the CNN but lower than the LSTM model, which demonstrates that the trained QEDL model achieves a comparable accuracy. In Table IV, the *recall* and *F1 score* of the QEDL model are also a little higher than those of the LSTM and CNN models. According to the definitions in Section IV-D, the proposed QEDL model can better distinguish the positive samples and has a more robust performance. To sum up, based on Tables III and IV, the proposed QEDL model shows comparable accuracy and *precision* performances with better generalization ability, as well as better *recall* and *F1 score* performances, compared with CNN and LSTM models.

After training and validating the LSTM, CNN, and QEDL models by using the well logs of W1 and W4, we then use the well logs of W2 and W3 to evaluate the validity and generalization performances of the trained models. Fig. 13(a) denotes the true lithology, which is a target part at this study area and extracted from W2, whereas Fig. 13(b)–(d) is the interpreted lithology predicted by the LSTM, CNN, and QEDL models. Based on the images in Fig. 13, we have three main observations. First, although all three trained models can predict the lithology by using the well logs, the proposed model interprets the thick layers more accurately, denoted by the pink and white rectangles. Second, the proposed model can also predict the thin layers more precisely than the LSTM and CNN models, presented by the pink arrows. Third, Fig. 13(e) and (f) denotes the GR curve and normalized borehole-side seismic trace. Apparently, the lithology result interpreted by using the proposed model matches with the GR curve and borehole-side seismic trace better than that computed by using the LSTM and CNN models.

Moreover, we show another deep predicted lithology in Fig. 14, which is another target part and extracted from W3. There are two main observations by comparing the images in Fig. 14. First, the QEDL model is obviously with a strong

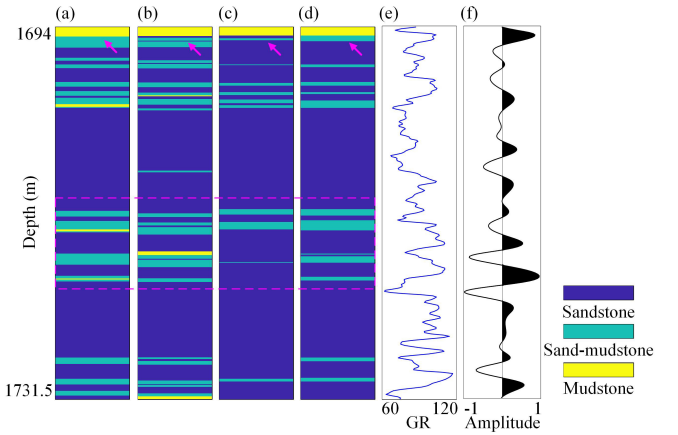


Fig. 14. (a) True lithology; interpreted lithology predicted by (b) LSTM-, (c) CNN-, and (d) QEDL-based models; and (e) and (f) denote the GR curve and normalized borehole-side seismic trace, respectively.

TABLE VI
PARAMETER NUMBER OF LSTM, CNN, AND QEDL

Models	parameter number
LSTM	18115
CNN	39811
QEDL	4567

generalization ability, which can achieve an accurate lithology interpretation at the boundary of the predicted lithology, presented by the pink arrows. However, the LSTM and CNN models obtain an inaccurate interpretation at the boundary of the predicted lithologies. Second, the QEDL model obtains a more precise interpretation for the thin layers than the LSTM and CNN models, indicated by the pink rectangle. The images in Figs. 13 and 14 lead us to the conclusion that the QEDL model is a more effective tool with a stronger generalization ability than the LSTM and CNN models for interpreting both thin and thick lithology layers. In addition, we calculate the *precision*, *recall*, and *F1 score* of the prediction dataset, indicated in Table V. Similar to the evaluation values in Table IV, three evaluation values provided by the QEDL model are higher than those of the LSTM and CNN models. We can easily draw a conclusion that the QEDL model achieves an accurate lithology interpretation result and improves the generalization power by introducing the quantum computing.

Finally, Table VI summarizes the parameter numbers of the LSTM, CNN, and QEDL models. It should be noted that the QEDL model needs much less model parameters than the LSTM and CNN models, e.g., the parameter number of the QEDL model in our study can be approximately 75% less than that of the LSTM model and 89% less than that of the CNN model, maybe because of the qubit inherent properties discussed in Appendix VI. Using less parameters indicates that the proposed QEDL model also has an advanced representation capacity, which means that the QEDL architecture accurately extracts the useful information from the underlying

features in the training dataset by using certain amount of the model's parameters. Need to mention, the model capacity (model parameter number) has a certain level of impact on the model generalization ability. So, it seems normal that the CNN and LSTM models have worse generalization as they have more parameters. However, we cannot only consider the model capacity and generalization. The model performances in terms of the *precision*, *recall*, and *F1 score* should be considered together. It is a comprehensive determination about the model performance. For example, if we reduce the model capacity of the CNN and LSTM models, the generalizations of those models can be improved, but the *precision*, *recall*, and *F1 score* will be worse because of the simpler models. In other words, the generalization itself is not important, but it is important when other model metrics are good enough. Therefore, our QEDL model has better generalization with less model parameter number and comparable *precision* as well as better *recall* and *F1 score*, which makes it promising.

V. DISCUSSION

In this study, we propose a QEDL model to automatically interpret lithology. It should be noted that we only introduce five well log curves to automatically interpret lithology, i.e., AC, CAL, DEN, GR, and SP. In the future work, we would select appropriate well log curves for automatic lithology interpretation based on the lithology features of the study survey. In addition, because of the limit of the number of well log data, we do not transfer the proposed QEDL model to the well log data at other seismic surveys. In future studies, we would further test the generalization property of the proposed model on well log data at other seismic survey, even on more and different formations.

It should be noted that the generalization power is important for a deep-learning-based seismic data processing. There are two main deep-learning-based ways to enhance the generalization power of a trained model. One is introducing the transfer-learning-based methods. More specifically, even for the transfer learning, the training data with corresponding labels of the new dataset are required for fine-tuning the trained model, which are often difficult and time consuming to realize in field data applications. Another way is adopting the deep-learning-based algorithms with a powerful generalization property, such as the quantum computing. By using the superposition and entanglement quantum systems, the quantum computing could potentially endow the generalization power to a deep-learning model. Therefore, we can adopt the proposed QEDL model to solve other seismic inversion issues in the future, such as the impedance and parameter inversion, and high-resolution processing.

VI. CONCLUSION

We propose a QEDL model for automatic lithology interpretation using well logs. It should be noticed that we adopt the proposed QEDL model for automatically interpreting sandstone, sand-mudstone, and mudstone in this study. By including other types of lithologies into the training dataset, such as shale, gypsum, and dolomite, we trust the proposed model

could also automatically recognize them with a generalization power. By utilizing the superposition and entanglement quantum systems, the quantum computing could potentially endow the generalization power to the deep-learning model. After training the proposed model, we test the model performances on field well log data from different wells and make qualitative and quantitative comparisons with the state-of-the-art LSTM and CNN models. Compared with the benchmark LSTM and CNN models, the QEDL model achieves a comparable accuracy with an improved generalization power for interpreting both the thin and thick lithology layers. Moreover, benefiting from the quantum circuit structure, the QEDL model needs much less model parameters than the LSTM and CNN models.

APPENDIX A QUBIT

The basic unit of the quantum computing is *qubit*, defined as $|\psi\rangle$, which is the probability combination (*superposition*) of classical states: the zero state $|0\rangle$ and the one state $|1\rangle$

$$|\psi\rangle := \alpha |0\rangle + \beta |1\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}; \quad |\alpha|^2 + |\beta|^2 = 1 \quad (\text{VI.1})$$

where α and β are the quantum amplitudes indicates the quantum collapse probability of zero state is $|\alpha|^2$, whereas $|\beta|^2$ for one state. Because 1 qubit can have two states, n entangled qubits carry 2^n thanks to the superposition, which is why the quantum computing can deal with the complicated computation tasks with small amount of qubits. It should be noted that the observation causes a collapse of states resulting the final measurement. In addition, using the Dirac notation "bra-ket," we can define the basis states as

$$|0\rangle := \begin{pmatrix} 1 \\ 0 \end{pmatrix}; \quad |1\rangle := \begin{pmatrix} 0 \\ 1 \end{pmatrix}. \quad (\text{VI.2})$$

APPENDIX B QUANTUM OPERATIONS

A. Representation

The networks of the quantum logic gates denoted by the unitary matrices are typically used to represent the quantum computation models [74]. With the same number of the input and output qubits, a quantum logic gate on n qubits uses a $2^n \times 2^n$ unitary matrix as the representation. Here, we use the tensor or kronecker product to represent the most common quantum gates operated on two qubits:

$$\begin{aligned} |ab\rangle &= |a\rangle \otimes |b\rangle \\ &= v_{00}|00\rangle + v_{01}|01\rangle \\ &\quad + v_{10}|10\rangle + v_{11}|11\rangle \rightarrow \begin{bmatrix} v_{00} \\ v_{01} \\ v_{10} \\ v_{11} \end{bmatrix} \end{aligned} \quad (\text{VI.1})$$

where \otimes denotes the tensor product.

The result state $|\psi_2\rangle$ of a gate action can be expressed as

$$|\psi_2\rangle = U|\psi_1\rangle \quad (\text{VI.2})$$

where $|\psi_1\rangle$ indicates the original state, whereas U is the gate.

B. Quantum Gates

Here, we introduce some common quantum gates. Typically, the unitary matrices, a type of square matrices that have the same inverse and complex conjugate, are used to represent the quantum gates.

1) *NOT Gate*: As a logical negation operation, the NOT gate X is used via the matrix multiplication as $X|0\rangle = |1\rangle$ and $X|1\rangle = |0\rangle$. Using a matrix, the NOT gate is defined as

$$X := \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}. \quad (\text{VI.3})$$

2) *Hadamard (H) Gate*: According to the Hadamard matrix

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (\text{VI.4})$$

the Hadamard (H) gate is a unitary matrix, mapping the basis states $|0\rangle$ and $|1\rangle$ on a single qubit to $((|0\rangle + |1\rangle)/\sqrt{2})$ and $((|0\rangle - |1\rangle)/\sqrt{2})$.

3) *Pauli Gates*: Pauli-X, Pauli-Y, and Pauli-Z gates ($\{\sigma_x, \sigma_y, \sigma_z\}$) are represented by using three 2×2 Hermitian and unitary matrices

$$\sigma_x = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad \sigma_y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}, \quad \sigma_z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}. \quad (\text{VI.5})$$

4) *Controlled Gates*: The controlled gates are applied on two or more qubits, where one or more qubits control the consequential operations. For example, suppose U is a gate on single qubits

$$U = \begin{bmatrix} u_{00} & u_{01} \\ u_{10} & u_{11} \end{bmatrix}. \quad (\text{VI.6})$$

Then, the controlled- U gate on two qubits uses the first qubit as the control of the other qubit, following:

$$\begin{aligned} |00\rangle &\mapsto |00\rangle \\ |01\rangle &\mapsto |01\rangle \\ |10\rangle &\mapsto |1\rangle \otimes U|0\rangle = |1\rangle \otimes (u_{00}|0\rangle + u_{10}|1\rangle) \\ |11\rangle &\mapsto |1\rangle \otimes U|1\rangle = |1\rangle \otimes (u_{01}|0\rangle + u_{11}|1\rangle). \end{aligned} \quad (\text{VI.7})$$

And the matrix expression is

$$C(U) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & u_{00} & u_{01} \\ 0 & 0 & u_{10} & u_{11} \end{bmatrix}. \quad (\text{VI.8})$$

C. Summary

The combination of the quantum logic gates and measurements describes a typical quantum computation. However, a quantum computation can also be expressed as a quantum logic gate-embedded network, because the measurement operations can be postponed to the end. Following this way, a measurement X is a summation of nonnegative probability coefficients p_i :

$$\begin{aligned} \langle X \rangle &= \sum_i p_i \langle \psi_i | X | \psi_i \rangle \\ &= \sum_i p_i \text{tr}(|\psi_i\rangle\langle\psi_i| X) \\ &= \text{tr}(\rho X) \end{aligned} \quad (\text{VI.9})$$

where $\text{tr}(\cdot)$ indicates the matrix trace. ρ is a density operator representing a mixed state as $\rho = \sum_{i=0}^{2^d} p_i |\psi_i\rangle\langle\psi_i|$, where $\{\psi_i\}$ represent the computational bases of the \mathcal{H}^{2^n} Hilbert space, the coefficients p_i add up to 1, and $|\psi\rangle\langle\psi|$ is an outer product written in bra-ket notation.

APPENDIX C QUANTUM NEURAL NETWORKS

In general, the QNN can be described as the product of the layers, which are represented as the unitaries

$$\begin{aligned} \hat{U}(\theta) &= \prod_{\ell=1}^L \left[\hat{V}_\ell \bigotimes_{j=1}^{M_\ell} \hat{U}_{\ell_j}(\theta_{\ell_j}^\ell) \right] \\ &= \prod_{\ell=1}^L \left[\hat{V}_\ell \bigotimes_{j=1}^{M_\ell} \prod_k e^{-i\theta_{\ell_j}^\ell \beta_k^{\ell_j} \hat{P}_k} \right] \end{aligned} \quad (\text{VI.1})$$

where \hat{V}^ℓ is a nonparametric unitary on the ℓ th layer, where the multiple variational unitaries $\{\hat{U}_{\ell_j}(\theta_{\ell_j}^\ell)\}_{j=1}^{M_\ell}$ exist. $\beta_k^{j\ell} \in \mathbb{R}$ is an operator for all k, j, ℓ , and \mathcal{P} denotes the Pauli's on n -qubits.

ACKNOWLEDGMENTS

The authors would like to thank Google for sharing the TensorFlow Quantum (TFQ) (<https://www.tensorflow.org/quantum>) and also would like to thank Dr. Hui Li, Daoyu Chen, and Jing Lin from Xi'an Jiaotong University for their beneficial suggestions. They deeply appreciate the valuable comments from Dr. S. H. Yueh from California Institute of Technology, the Associate Editor, and another three anonymous reviewers.

REFERENCES

- [1] R. P. Feynman, "Simulating physics with computers," *Int. J. Theor. Phys.*, vol. 21, nos. 6–7, pp. 467–488, Jun. 1982.
- [2] M. Schuld, I. Sinayskiy, and F. Petruccione, "The quest for a quantum neural network," *Quantum Inf. Process.*, vol. 13, no. 11, pp. 2567–2586, Nov. 2014.
- [3] A. Daskin, "A simple quantum neural net with a periodic activation function," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, Oct. 2018, pp. 2887–2891.
- [4] C. Shao, "A quantum model for multilayer perceptron," 2018, *arXiv:1808.10561*. [Online]. Available: <http://arxiv.org/abs/1808.10561>
- [5] K. Beer *et al.*, "Training deep quantum neural networks," *Nature Commun.*, vol. 11, no. 1, pp. 1–6, 2020.
- [6] M. Broughton *et al.*, "TensorFlow quantum: A software framework for quantum machine learning," 2020, *arXiv:2003.02989*. [Online]. Available: <http://arxiv.org/abs/2003.02989>
- [7] Google. (2018). *Cirq: A Python Framework for Creating, Editing, and Invoking Noisy Intermediate Scale Quantum Circuits*. [Online]. Available: <https://github.com/google/cirq>
- [8] I. M. Mohamed *et al.*, "Formation lithology classification: Insights into machine learning methods," in *Proc. SPE Annu. Tech. Conf. Exhib.*, London, U.K.: Society of Petroleum Engineers, 2019, pp. 1–21.
- [9] Z. Xu, B. Zhang, F. Li, G. Cao, and Y. Liu, "Well-log decomposition using variational mode decomposition in assisting the sequence stratigraphy analysis of a conglomerate reservoir," *Geophysics*, vol. 83, no. 4, pp. B221–B228, Jul. 2018.
- [10] F. Li, R. Xie, W.-Z. Song, and H. Chen, "Optimal seismic reflectivity inversion: Data-driven ℓ_p -loss- ℓ_q -regularization sparse regression," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 5, pp. 806–810, May 2019.
- [11] D. L. Xu, T. Li, B. H. Huang, and N. LI, "Research on the identification of the lithology and fluid type of foreign M oilfield by using the crossplot method," *Prog. Geophys.*, vol. 27, no. 3, pp. 1123–1132, 2012.

- [12] W. Liu and X. Zhu, "Identifying lithology of natural gas pool by fuzzy mathematics in Ordos basin," *Fault-Block Oil Gas Field*, vol. 12, no. 5, pp. 7–9, 2005.
- [13] J. M. Busch, W. G. Fortney, and L. N. Berry, "Determination of lithology from well logs by statistical analysis," *SPE Formation Eval.*, vol. 2, no. 4, pp. 412–418, Dec. 1987.
- [14] R. Malcol, *The Geological Interpretation of Well Logs*, vol. 1. New York, NY, USA: Ryder-French Consult Ltd, 2002, p. 32.
- [15] J. Lin, H. Li, N. Liu, J. Gao, and Z. Li, "Automatic lithology identification by applying LSTM to logging data: A case study in X tight rock reservoirs," *IEEE Geosci. Remote Sens. Lett.*, early access, Jun. 19, 2020, doi: 10.1109/LGRS.2020.3001282.
- [16] J. R. Harris and E. C. Grunsky, "Predictive lithological mapping of Canada's north using random forest classification applied to geophysical and geochemical data," *Comput. Geosci.*, vol. 80, pp. 9–25, Jul. 2015.
- [17] A. Al-Anazi and I. D. Gates, "On the capability of support vector machines to classify lithology from well logs," *Natural Resour. Res.*, vol. 19, no. 2, pp. 125–139, Jun. 2010.
- [18] S. J. Rogers, J. Fang, C. Karr, and D. Stanley, "Determination of lithology from well logs using a neural network (1)," *AAPG Bull.*, vol. 76, no. 5, pp. 731–739, 1992.
- [19] A. Mohebbi, R. Kamalpour, K. Keyvanloo, and A. Sarrafi, "The prediction of permeability from well logging data based on reservoir zoning, using artificial neural networks in one of an Iranian heterogeneous oil reservoir," *Petroleum Sci. Technol.*, vol. 30, no. 19, pp. 1998–2007, Jul. 2012.
- [20] W. Chen, L. Yang, B. Zha, M. Zhang, and Y. Chen, "Deep learning reservoir porosity prediction based on multilayer long short-term memory network," *Geophysics*, vol. 85, no. 4, pp. 1–69, 2020.
- [21] H. Wu, B. Zhang, T. Lin, F. Li, and N. Liu, "White noise attenuation of seismic trace by integrating variational mode decomposition with convolutional neural network," *Geophysics*, vol. 84, no. 5, pp. V307–V317, Sep. 2019.
- [22] X. Wu, Z. Geng, Y. Shi, N. Pham, S. Fomel, and G. Caumon, "Building realistic structure models to train convolutional neural networks for seismic structural interpretation," *Geophysics*, vol. 85, no. 4, pp. WA27–WA39, Jul. 2020.
- [23] F. Li, H. Zhou, Z. Wang, and X. Wu, "ADDCNN: An attention-based deep dilated convolutional neural network for seismic facies analysis with interpretable Spatial-Spectral maps," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 2, pp. 1733–1744, Feb. 2021.
- [24] Y. Imamverdiyev and L. Sukhostat, "Lithological facies classification using deep convolutional neural network," *J. Petroleum Sci. Eng.*, vol. 174, pp. 216–228, Mar. 2019.
- [25] M. B. Valentin *et al.*, "A deep residual convolutional neural network for automatic lithological facies identification in Brazilian pre-salt oilfield wellbore image logs," *J. Petroleum Sci. Eng.*, vol. 179, pp. 474–503, Aug. 2019.
- [26] J. Zhang, J. Li, X. Chen, and Y. Li, "Seismic lithology/fluid prediction via a hybrid ISD-CNN," *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 1, pp. 13–17, Jan. 2021.
- [27] H. Wu, B. Zhang, F. Li, and N. Liu, "Semiautomatic first-arrival picking of microseismic events by using the pixel-wise convolutional image segmentation method," *Geophysics*, vol. 84, no. 3, pp. V143–V155, May 2019.
- [28] N. Liu, T. He, Y. Tian, B. Wu, J. Gao, and Z. Xu, "Common-azimuth seismic data fault analysis using residual UNet," *Interpretation*, vol. 8, no. 3, pp. SM25–SM37, Aug. 2020.
- [29] H. Hedayati, B. J. McGuinness, M. J. Cree, and J. A. Perrone, "Generalization approach for CNN-based object detection in unconstrained outdoor environments," in *Proc. Int. Conf. Image Vis. Comput. New Zealand (IVCNZ)*, Dec. 2019, pp. 1–6.
- [30] Z. Zhu, G. Peng, Y. Chen, and H. Gao, "A convolutional neural network based on a capsule network with strong generalization for bearing fault diagnosis," *Neurocomputing*, vol. 323, pp. 62–75, Jan. 2019.
- [31] M. I. Jordan, "Serial order: A parallel distributed processing approach," in *Advances in Psychology*, vol. 121. Amsterdam, The Netherlands: Elsevier, 1997, pp. 471–495.
- [32] A. Graves, M. Liwicki, S. Fernandez, R. Bertolami, H. Bunke, and J. Schmidhuber, "A novel connectionist system for unconstrained handwriting recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 5, pp. 855–868, May 2009.
- [33] F. Li *et al.*, "Detection and diagnosis of data integrity attacks in solar farms based on multilayer long short-term memory network," *IEEE Trans. Power Electron.*, vol. 36, no. 3, pp. 2495–2498, Mar. 2021.
- [34] A. Chen, W. Song, F. Li, and J. M. Velni, "Distributed cooperative energy management in smart microgrids with solar energy prediction," in *Proc. IEEE Int. Conf. Commun., Control, Comput. Technol. Smart Grids (SmartGridComm)*, Oct. 2018, pp. 1–6.
- [35] A. Graves, A.-R. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, May 2013, pp. 6645–6649.
- [36] P. Liu, X. Qiu, and X. Huang, "Recurrent neural network for text classification with multi-task learning," 2016, *arXiv:1605.05101*. [Online]. Available: <http://arxiv.org/abs/1605.05101>
- [37] M. Alfarraj and G. AlRegib, "Semisupervised sequence modeling forelastic impedance inversion," *Interpretation*, vol. 7, no. 3, pp. SE237–SE249, 2019.
- [38] J. Zheng, J. Lu, S. Peng, and T. Jiang, "An automatic microseismic or acoustic emission arrival identification scheme with deep recurrent neural networks," *Geophys. J. Int.*, vol. 212, no. 2, pp. 1389–1397, Feb. 2018.
- [39] M. Tian, H. Omre, and H. Xu, "Inversion of well logs into lithology classes accounting for spatial dependencies by using hidden Markov models and recurrent neural networks," *J. Petroleum Sci. Eng.*, vol. 196, Jan. 2021, Art. no. 107598.
- [40] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Trans. Neural Netw.*, vol. 5, no. 2, pp. 157–166, Mar. 1994.
- [41] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," 2014, *arXiv:1412.3555*. [Online]. Available: <http://arxiv.org/abs/1412.3555>
- [42] L. Zeng, W. Ren, and L. Shan, "Attention-based bidirectional gated recurrent unit neural networks for well logs prediction and lithology identification," *Neurocomputing*, vol. 414, pp. 153–171, Nov. 2020.
- [43] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [44] Y. Wu, Y. Lin, Z. Zhou, D. C. Bolton, J. Liu, and P. Johnson, "DeepDetect: A cascaded region-based densely connected network for seismic event detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 1, pp. 62–75, Jan. 2019.
- [45] V. Vapnik, "The support vector method of function estimation," in *Nonlinear Modeling*. Boston, MA, USA: Springer, 1998, pp. 55–85.
- [46] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.
- [47] L. Zhang, W. Zhou, and L. Jiao, "Wavelet support vector machine," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 34, no. 1, pp. 34–39, Feb. 2004.
- [48] X. Liu, X. Chen, J. Li, X. Zhou, and Y. Chen, "Facies identification based on multikernel relevance vector machine," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 10, pp. 7269–7282, Oct. 2020.
- [49] W. S. Noble, "What is a support vector machine?" *Nature Biotechnol.*, vol. 24, no. 12, pp. 1565–1567, Dec. 2006.
- [50] L. H. An, "Neutral network in lithology determination," *J. Comput. Sci. Cybern.*, vol. 16, no. 2, pp. 59–62, Apr. 2013.
- [51] Y. Xie, C. Zhu, W. Zhou, Z. Li, X. Liu, and M. Tu, "Evaluation of machine learning methods for formation lithology identification: A comparison of tuning processes and model performances," *J. Petroleum Sci. Eng.*, vol. 160, pp. 182–193, Jan. 2018.
- [52] J. Hou, C. Huang, Y. Zhang, and J. Guo, "On the value of available MODIS and Landsat8 OLI image pairs for MODIS fractional snow cover mapping based on an artificial neural network," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 6, pp. 4319–4334, Jun. 2020.
- [53] M. A. Shahin, M. B. Jaksa, and H. R. Maier, "Artificial neural network applications in geotechnical engineering," *Austral. Geomech.*, vol. 36, no. 1, pp. 49–62, 2001.
- [54] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Netw.*, vol. 61, pp. 85–117, Jan. 2015.
- [55] G. Zhang, Z. Wang, and Y. Chen, "Deep learning for seismic lithology prediction," *Geophys. J. Int.*, vol. 215, pp. 1368–1387, Aug. 2018.
- [56] X. Wu, Y. Shi, S. Fomel, L. Liang, Q. Zhang, and A. Z. Yusifov, "FaultNet3D: Predicting fault probabilities, strikes, and dips with a single convolutional neural network," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 11, pp. 9138–9155, Nov. 2019.
- [57] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, p. 436, 2015.
- [58] Y. LeCun and Y. Bengio, "Convolutional networks for images, speech, and time series," in *The Handbook of Brain Theory and Neural Networks*, vol. 3361, no. 10. Cambridge, MA, USA: MIT Press, 1995, p. 1995.
- [59] B. Ricks and D. Ventura, "Training a quantum neural network," in *Proc. Adv. Neural Inf. Process. Syst.*, 2004, pp. 1019–1026.

- [60] J. Romero, J. P. Olson, and A. Aspuru-Guzik, "Quantum autoencoders for efficient compression of quantum data," *Quantum Sci. Technol.*, vol. 2, no. 4, Dec. 2017, Art. no. 045001.
- [61] J. Allcock, C.-Y. Hsieh, I. Kerenidis, and S. Zhang, "Quantum algorithms for feedforward neural networks," 2018, *arXiv:1812.03089*. [Online]. Available: <http://arxiv.org/abs/1812.03089>
- [62] K. Mitarai, M. Negoro, M. Kitagawa, and K. Fujii, "Quantum circuit learning," *Phys. Rev. A, Gen. Phys.*, vol. 98, no. 3, Sep. 2018, Art. no. 032309.
- [63] C. Zhao and X.-S. Gao, "QDNN: DNN with quantum neural network layers," 2019, *arXiv:1912.12660*. [Online]. Available: <http://arxiv.org/abs/1912.12660>
- [64] S. Garg and G. Ramakrishnan, "Advances in quantum deep learning: An overview," 2020, *arXiv:2005.04316*. [Online]. Available: <http://arxiv.org/abs/2005.04316>
- [65] A. Muthukrishnan, "Classical and quantum logic gates: An introduction to quantum computing quantum information seminar," Tech. Rep., 1999. [Online]. Available: <http://www2.optics.rochester.edu/~stroud/presentations/muthukrishnan991/LogicGates.pdf>
- [66] H. J. Briegel and R. Raussendorf, "Persistent entanglement in arrays of interacting particles," *Phys. Rev. Lett.*, vol. 86, no. 5, p. 910, 2001.
- [67] B. Wu, D. Meng, L. Wang, N. Liu, and Y. Wang, "Seismic impedance inversion using fully convolutional residual network and transfer learning," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 12, pp. 2140–2144, Dec. 2020.
- [68] Z. Wang, J. Gao, D. Wang, and Q. Wei, "3D seismic attributes for a tight gas sand reservoir characterization of the eastern Sulige gas field, Ordos Basin, China," *Geophysics*, vol. 80, no. 2, pp. B35–B43, Mar. 2015.
- [69] H. Wu, B. Zhang, and D. Cao, "The determination of the constant phase of seismic wavelet using automatic seismic-well tying," *Explor. Geophys.*, vol. 50, no. 3, pp. 245–254, May 2019.
- [70] R. Kohavi *et al.*, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *Proc. Int. Joint Conf. AI*, vol. 14, no. 2, Aug. 1995, pp. 1137–1145.
- [71] D. Anguita, S. Ridella, and F. Rivieccio, "K-fold generalization capability assessment for support vector classifiers," in *Proc. IEEE Int. Joint Conf. Neural Netw.*, Jul. 2005, pp. 855–858.
- [72] T.-T. Wong and N.-Y. Yang, "Dependency analysis of accuracy estimates in k-fold cross validation," *IEEE Trans. Knowl. Data Eng.*, vol. 29, no. 11, pp. 2417–2427, Nov. 2017.
- [73] Y. Liu, H. T. Loh, Y.-T. Kamal, and S. B. Tor, "Handling of imbalanced data in text classification: Category-based term weights," in *Natural Language Processing and Text Mining*. London, U.K.: Springer, 2007, pp. 171–192.
- [74] M. A. Nielsen and I. Chuang, *Quantum Computation and Quantum Information*. Cambridge, U.K.: Cambridge Univ. Press, 2002. [Online]. Available: <https://www.amazon.com/Quantum-Computation-Information-10th-Anniversary/dp/1107002176>



Naihao Liu (Member, IEEE) received the B.S. degree in communication engineering from Jilin University, Changchun, China, in 2012, and the Ph.D. degree in information and communication engineering from Xi'an Jiaotong University, Xi'an, China, in 2018.

From 2017 to 2018, he visited the Department of Geological Sciences, The University of Alabama, Tuscaloosa, AL, USA. He is with the School of Information and Communications Engineering, Xi'an Jiaotong University. His research interests

include seismic time-frequency analysis, attribute analysis and parameter inversion, machine learning, and reservoir characterization.



Teng Huang received the B.S. degree in information engineering from Xi'an Jiaotong University, Xi'an, China, in 2019, where he is pursuing the master's degree with the School of Information and Communications Engineering.

His research interests include machine learning, signal processing, and seismic interpretation.



Jinghui Gao (Member, IEEE) received the M.S. degree in applied geophysics from Chang'an University, Xi'an, China, in 1991, and the Ph.D. degree in electromagnetic field and microwave technology from Xi'an Jiaotong University, Xi'an, in 1997.

From 1997 to 2000, he was a Post-Doctoral Researcher with the Institute of Geology and Geophysics, Chinese Academy of Sciences, Beijing, China. In 1999, he was a Visiting Scientist with the Modeling and Imaging Laboratory, University of California at Santa Cruz, Santa Cruz, CA, USA.

He is an Associate Director with the National Engineering Laboratory for Offshore Oil Exploration, Xi'an Jiaotong University. He is a Project Leader of the Fundamental Theory and Method for Geophysical Exploration and Development of Unconventional Oil and Gas, Xi'an Jiaotong University, which is a major program of the National Natural Science Foundation of China under Grant 41390450. He is a Professor with the School of Electronic and Information Engineering, School of Mathematics and Statistics, Xi'an Jiaotong University. His research interests include seismic wave propagation and imaging theory, seismic reservoir and fluid identification, and seismic inverse problem theory and method.

Dr. Gao was a recipient of the Chen Zongqi Geophysical Best Paper Award in 2013. He is an Editorial Board Member of the *Chinese Journal of Geophysics*, *Journal of Applied Geophysics*, and *Chinese Science Bulletin*.



Zongben Xu (Member, IEEE) received the Ph.D. degree in mathematics from Xi'an Jiaotong University, Xi'an, China, in 1987.

His research interests include intelligent information processing and applied mathematics.

Dr. Xu was a recipient of the National Natural Science Award of China in 2007 and the CSIAM Su Buchin Applied Mathematics Prize in 2008. He serves as the Chief Scientist of the National Basic Research Program of China (973 Project) and the Director of the Institute for Information and System

Sciences, Xi'an Jiaotong University, where he is a member of the Chinese Academy of Sciences, Beijing, China. He delivered a 45-minute talk on the International Congress of Mathematicians in 2010.



Daxing Wang received the B.S. and M.S. degrees in geophysical exploration from Southwest Petroleum University, Chengdu, China, in 1983 and 1995, respectively, and the Ph.D. degree in solid geophysics from the Institute of Geology and Geophysics, Chinese Academy of Sciences, Beijing, China, in 2005.

He is a Professor-Level Senior Engineer with the Exploration and Development Research Institute of PetroChina Changqing Oilfield Company, Xi'an, China. His research interests include reservoir characterization and hydrocarbon detection.

Dr. Wang is an Editorial Board Member of the *Oil Geophysical Prospecting*, a member of Key Laboratory of Geophysical Prospecting, PetroChina, China, Society of Petroleum Geophysicists, and Chinese Petroleum Society.



Fangyu Li (Member, IEEE) received the bachelor's degree in electrical engineering from Beihang University, Beijing, China, in 2009, the master's degree in electrical engineering from Tsinghua University, Beijing, in 2013, and the Ph.D. degree in geophysics from The University of Oklahoma, Norman, OK, USA, in 2017.

He was a Post-Doctoral Fellow with the College of Engineering, University of Georgia, Athens, GA, USA. He is an Assistant Professor with the Department of Electrical and Computer Engineering,

Kennesaw State University (KSU), Kennesaw, Georgia. His research interests include seismic signal processing, seismic interpretation, subsurface imaging, seismic imaging, machine learning, deep learning, distributed computing, Internet of Things (IoT), and cyber-physical systems (CPS).

Dr. Li was a recipient of the J. Clarence Karcher Award for his contribution to exploration geophysics from the Society of Exploration Geophysicists (SEG) in 2020.