Proceedings of the 2023 Winter Simulation Conference
C. G. Corlu, S. R. Hunter, H. Lam, B. S. Onggo, J. Shortle, and B. Biller, eds.

QUANTUM EMBEDDING FRAMEWORK OF INDUSTRIAL DATA FOR QUANTUM DEEP LEARNING

Hyunsoo Lee

School of Industrial Engineering
Kumoh National Institute of Technology
61 Daehak-ro, Gumi
Gyeongbuk, PO 39177, S. KOREA

Amarnath Banerjee

Department of Industrial & Systems Engineering
Texas A&M University
Emerging Technologies Building
College Station, TX 77841-3131, USA

ABSTRACT

Quantum computing is a contemporary engineering discipline that innovatively overcomes computational burdens. This study applies quantum computing techniques to data analyses with input data issues. When a dataset has insufficient attributes and uncertainties, quantum embedding techniques contribute to the dimensional expansion of input vectors and the quantification of uncertainties. The converted qubits are linked to subsequent deep learning modules, and this architecture is used for accurate data analysis. This study proposes a quantum embedding technique and a corresponding quantum neural network (QNN) to better understand these processes. In this QNN architecture, input data are converted into corresponding qubits, which are transformed with quantum phase-operating modules. The quantum features pass through subsequent deep learning layers for more accurate data analyses. To demonstrate the effectiveness of the proposed model, a process model and relevant analyses are presented and compared with existing deep learning methods.

1 INTRODUCTION

Advancements in machine learning approaches, including deep learning techniques, have contributed to the seamless execution of manufacturing processes and the prevention of possible abnormal conditions. In general, datasets comprising inputs and outputs are understood using corresponding deep learning architectures. While several effective applications have employed deep learning, critical issues include input data issues, such as insufficient data attributes and shortages from missing values, which result in the failure of well-trained deep learning machines and inaccurate decisions.

This study addresses the issue of an input dataset with comparatively fewer attributes for analysis. A quantum computing approach is introduced and applied to address this issue. Although the initial dataset had limited data attributes and uncertainties, each data point was quantified to the corresponding qubit using a quantum embedding technique. The proposed quantum embedding method contributed to the dimensional expansion of input vectors. These qubits were then transformed dynamically using relevant quantum computing dynamics. In this study, quantum phase operations were applied to various quantum computing operations. The transformed quantum features were input into subsequent deep neural network (DNN) layers. The proposed quantum embedding technique enables the expansion of features, thereby leading to a more accurate data analysis. This architecture is called a quantum neural network (QNN). To demonstrate the effectiveness of the proposed QNN, an engine dynamometer manufacturing process was modeled and simulated. The use of general DNN is not suitable as the dataset contained insufficient data attributes. The proposed QNN was applied to overcome this problem.

The following section provides relevant background information and a literature review. The quantum embedding technique and subsequent quantum operation modules are described in Section 3. Section 4 compares the manufacturing data model and analyses.

2 BACKGROUND AND LITERATURE REVIEWS

Process data analyses and predictions of key performance indices are crucial for increasing process productivity. Amongst the newer methods that have been proposed, deep learning-based machine learning methods have attracted much attention and have been used extensively in various applications. However, their successful deployment and continuous use must be guaranteed with supported data. For small datasets and the commonly encountered difficulties in identifying sufficient data attributes, the efficiencies of deep learning-based data analyses are reduced. Table 1 summarizes instances of several data imbalance issues in deep learning-based analyses.

Table 1: Data imbalance issues in deep learning data analyses.

Existing research studies	Applications	Data imbalance issues	Solutions
Zhang and Ling (2018)	Image processing in material science	Small datasets	Degree-of-Freedom based boosting
Hamad et al. (2020)	Smart home controls	Lack of human activity data	Temporal time windows technique
Hang et al. (2023)	Synthetic data analysis	Imbalance in classification label data	Under-bagging nearest neighbors (K-nn)
Sarafianos et al. (2018)	Image classification	Imbalanced attributes in classification data	Visual attention aggregation
Oh and Lee (2020)	Air pressure system	Data with a number of missing values	Gaussian Progress Regression-based GAN
Zhu et al. (2020)	Syntactic data analysis	Class imbalance	Interpolation-based oversampling
Li et al. (2018)	Syntactic data analysis	Small datasets	Fuzzy based data processing

As shown in Table 1, most previous studies used oversampling methods to compensate for small datasets and class imbalanced datasets. However, few studies have investigated small attributes in input datasets. For example, Lee et al. (2020) used a generative adversarial network (GAN) to secure sufficient data to diagnose train safety.

This study applies a quantum embedding method to increase the input data dimensions. Quantum embedding (Knizia and Chan. 2013; Gianai et al. 2022) is a quantum computing technique that represents a vector using the corresponding quantum state. When a vector is converted into a quantum state, the representation power of these quantum bits increases. In addition, several quantum operations can be applied to the converted qubits. The detailed method is described in Section 3.1.

The converted quantum states were integrated with subsequent deep learning layers as a hybrid deep learning model (Lee et al. 2020), which is a QNN architecture. QNN-based machine learning is a fully mature analytical method. However, several studies have proposed using the characteristics and advantages of quantum computing. Table 2 provides several existing quantum deep learning studies. Although several studies have proposed QNN-based data analysis applications, relatively few authors have investigated studies on input data issues. An effective quantum embedding method and subsequent QNN architecture are proposed here to investigate issues related to input data.

Table 2: Several research studies using quantum deep learning.

Research studies	Application	Characteristics of quantum machine learning
Javier et al. (2022)	Computer vision for steel quality control	DNN layers with rotation operations of qubits
Huang et al. (2021)	MNIST dataset classification	Quantum embedding and kernel projection
Batra et al. (2021)	Drug discovery	Quantum computing for molecular descriptors
Meyer et al. (2023)	Tic-tac-toe simulation/ vehicle classification	Variational quantum machine learning and symmetry techniques

3 QUANTUM EMBEDDING AND QUANTUM DEEP LEARNING FRAMEWORK FOR INDUSTRIAL DATA

3.1 Quantum Embedding Framework for Industrial Data

This section proposes a new and effective quantum embedding method for handling industrial datasets. As introduced in Section 2, quantum embedding converts a normal number ($x_{ij} \in N$) into the corresponding quantum state ($|\psi\rangle$). x_{ij} denotes the ith value of the jth attribute. While several quantum embedding techniques exist, the data range, variance and dependencies are not adequately captured during the conversion process. This study proposes a quantum embedding technique that retains the variance in the original dataset after measuring the converted quantum states. In the first stage, a value (x_{ij}) is transitioned and scaled down to (x_{ij}') using (1).

$$x'_{ij} = \frac{(x_{ij} - \min_j(x_{ij}))}{max_j(x_{ij}) - min_j(x_{ij})} \tag{1}$$

In (1), $\min_j(x_{ij})$ and $\max_j(x_{ij})$ indicate the minimum and maximum values of the jth attribute, respectively. Accordingly, x_{ij}' is located between 0 and 1. In the proposed embedding technique, 0 and 1 are used as quantum bases: $|0\rangle$ and $|1\rangle$. In order to maintain the variance of the original data after measurement, x'_{ij} is converted to the corresponding quantum bit $|\psi_{ij}\rangle$denoted by (2).

$$|\psi_{ij}\rangle = \sqrt{1 - x'_{ij}} \cdot |0\rangle + \sqrt{x'_{ij}} \cdot |1\rangle \tag{2}$$

Using Born's rule (McMahon 2020), qubit $|\psi_{ij}\rangle$ has an x'_{ij} probability of detecting 1. This indicates that the mean of $|\psi_{ij}\rangle$ after the measurement is the same as that of the normalized original data x'_{ij}, and the variance of the overall data is retained.

The advantages of the proposed embedding method are the expanded representational ability of the data and the use of subsequent quantum operators such as Pauli matrices (Hidary 2019) and quantum phase operators (Steeb and Hardy 2012). For instance, a quantum embedding vector changes its state ($|\psi_{ij}\rangle$) to $|\phi_{ij}\rangle$ using a rotation operator $R(\alpha, \beta, \gamma)$ with respect to the Bloch basis, shown in (3).

$$R(\alpha, \beta, \gamma): |\psi_{ij}\rangle \rightarrow |\phi_{ij}\rangle \tag{3}$$

In terms of the Bloch sphere-based rotation, (3) is interpreted using (4).

$$\begin{aligned} |\phi_{ij}\rangle &= R(\alpha, \beta, \gamma) \cdot |\psi_{ij}\rangle \\ &= R_z(\gamma) \cdot R_y(\beta) \cdot R_x(\alpha) \cdot |\psi_{ij}\rangle \end{aligned} \tag{4}$$

From (4), each rotation operator $R_k(\theta)$ is interpreted with $e^{(-i\cdot\theta\cdot k/2)}$, where k is one of the axes in the Bloch sphere. Thus, a data value is converted into a corresponding qubit, and the qubit is dynamically changed into its state. Figure 1 shows a conceptual example of quantum embedding and unitary operation-based dynamic changes from the original data value. The detailed procedures are provided in Section 3.2.

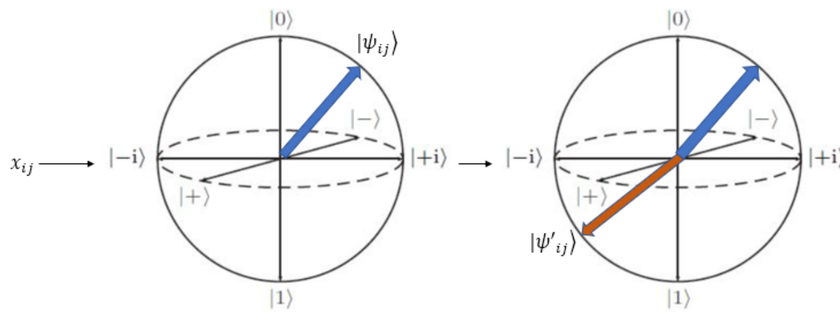


Figure 1: Quantum embedding and rotation processes from a data value.

Using this approach, one input value is expanded into a vector with two dimensions. When n sets of data are used for the input, $2 \cdot n$ dimensions are generated using these qubits.

The following section introduces a quantum deep learning architecture using quantum embedding.

3.2 Quantum Deep Learning Framework using Quantum Embedding Technique

As mentioned in the previous section, a data value is converted into a qubit with two bases: $|0\rangle$ and $|1\rangle$. Then, it changes to another state using rotation-based unitary operations. As a result, n input dimensions are expanded to $2 \cdot n$, and a more in-depth mapping can be achieved. As a subsequent relation mapping technique, this study applied deep learning to predict future values. Thus, the framework is a type of QNNs that incorporates a quantum embedding method. In general, a DNN has an architecture denoted by (5), where x_i is an input vector in the i^{th} layer and Y is the target output.

$$Y = \phi_l(\textstyle\sum_{m\in M}(w_{lm} \cdot (\phi_m \cdots (\sum_{i\in I} w_{ji} \cdot x_i)))) \tag{5}$$

w_{ji} is the weight from the i^{th} to j^{th} layers, and ϕ_l is the l^{th} activation function. Although a general DNN has a better performance than several traditional machine learning methods, it requires modifications to handle quantum embedded vectors. Thus, (5) can be expanded to (6).

$$Y = \phi_l(\textstyle\sum_{m\in M}(w_{lm} \cdot (\phi_m \cdots (\sum_{i\in I} w_{ji} \cdot |\phi_i\rangle)))) \tag{6}$$

Using equation (2), the input value x_i is changed to $|\phi_i\rangle$, as shown in equation (6). Accordingly, equation (4) is represented by (2), with the embedding vector $|\psi_i\rangle$ from the original data.

$$Y = \phi_l(\textstyle\sum_{m\in M}(w_{lm} \cdot (\phi_m \cdots (\sum_{i\in I} w_{ji} \cdot R_z(\gamma) \cdot R_y(\beta) \cdot R_x(\alpha) \cdot |\psi_{ij}\rangle)))) \tag{7}$$

Figure 2 shows the overall QNN architecture with the proposed embedding method.

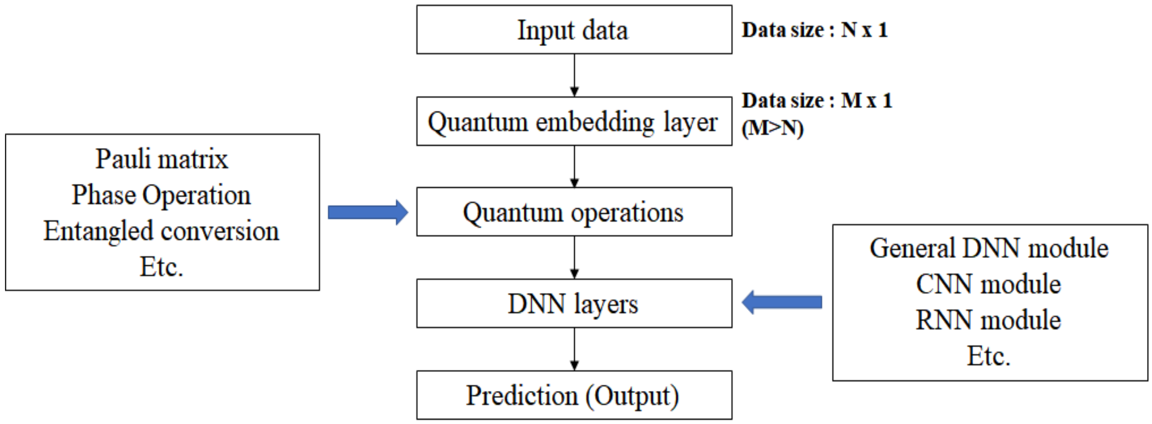


Figure 2: QNN architecture with quantum embedding modules.

As shown in Figure 2, the proposed QNN comprises of five parts: the input layer, quantum embedding layer, quantum operation layers, DNN layers, and the output layer. The learnable parameters are classified into two types: rotation parameters (α, β, and γ) and weights. In order to update these parameters, the energy function that was used is the same as that of the DNN, denoted by equation (8). Y_l is the l^{th} output value in the training data, and $\widehat{Y_l}$ is the output from the QNN machine.

$$\mathrm{E} = \frac{1}{2} \cdot \sum \left(Y_l - \widehat{Y_l}\right)^2 \tag{8}$$

Because the provided QNN has modules that are shared with a DNN, the updating policy of weight w_{ji} follows (9).

$$\mathrm{w'}_{\mathrm{ji}} = w_{ji} - \eta \cdot \frac{\partial E}{\partial v_l} \cdot \frac{\partial v_l}{\partial v_m} \cdots \frac{\partial v_j}{\partial w_{ji}} \tag{9}$$

$\mathrm{w'}_{\mathrm{ji}}$ is a newly updated value w_{ji} with a learning rate η. Each v_i is the sum ($\sum_{i\in I} w_{ji} \cdot x_i$) of the input vectors to the i^{th} layer and their corresponding weights. While the other parameters need updates for the minimization of (8), ∂E/ ∂γ in (10) needs an additional gradient with the accumulated backward errors.

$$\gamma' = \gamma - \eta \cdot \frac{\partial E}{\partial \gamma} \tag{10}$$

Accordingly, ∂E/ ∂γ is denoted by (11).

$$\frac{\partial E}{\partial \gamma} = \frac{\partial E}{\partial v_l} \cdot \frac{\partial v_l}{\partial v_m} \cdots \frac{\partial v_j}{\partial \gamma} \tag{11}$$

As $v_j = \phi_i(w_{ji} \cdot e^{-i\cdot\gamma\cdot k} \cdot I_i)$ where I_i is the input to the $R_z(\gamma)$ layer and tanh (·) is its activation function, the final term in (11) is derived to (12).

$$\frac{\partial v_j}{\partial \gamma} = -i \cdot \left(1 + v_j\right) \cdot \left(1 - v_j\right) \cdot w_{ji} \cdot I_i \cdot e^{-i\cdot\gamma\cdot k} \tag{12}$$

In this manner, the quantum phase parameters are updated, and the overall learnable parameters are learned in the QNN model. The provided framework supports a quantum embedding technique from a data to a qubit-embedding data, and explains how the uncertainties are quantified using the subsequent deep learning model. The following section presents numerical analyses and comparisons to demonstrate the effectiveness of the proposed model.

4 NUMERICAL ANALYSIS AND COMPARISON STUDIES FOR QUANTUM DEEP LEARNING WITH QUANTUM EMBEDDING MODULE

For the analysis process, the module production in an engine dynamometer was considered. As shown in Figure 3, the module gain (Y) affects the successful operation of the dynamometer. In order to predict Y during the production process, eight sensors ($s_{i,i\in[1,8]}$) were installed. The sensing values are simulated using dynamometer dynamics.

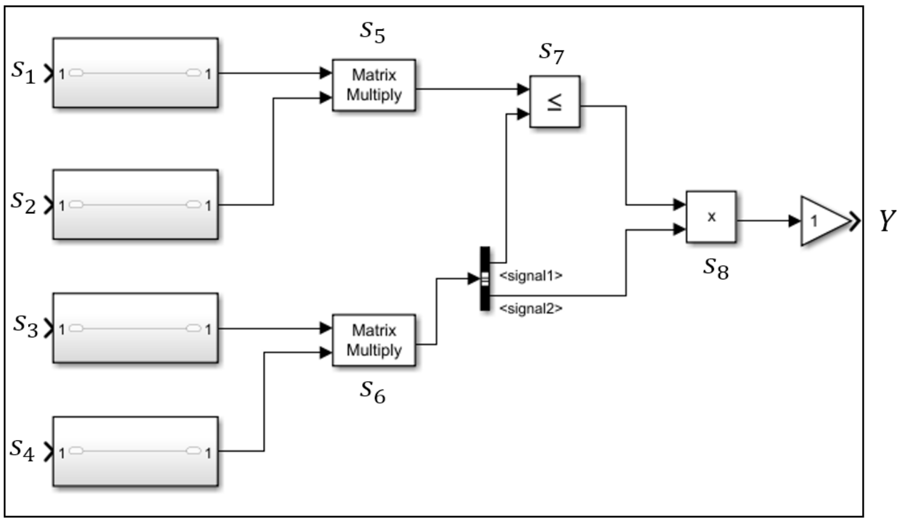


Figure 3: Installed sensors and a quality index for an engine dynamometer module.

Figure 4 shows the correlations among sensor signals and Y.

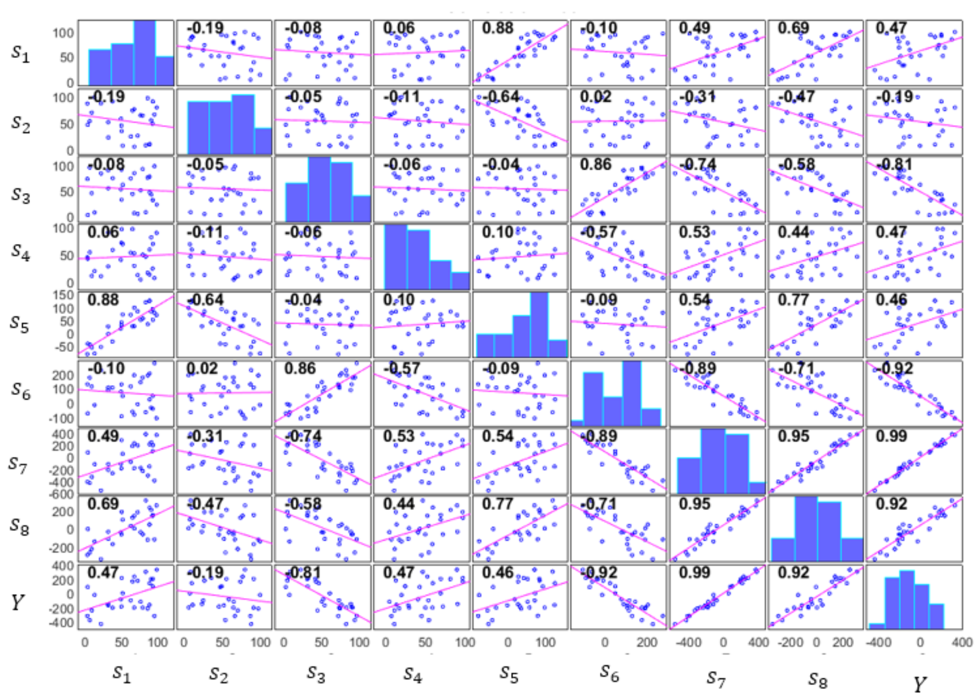


Figure 4: Correlation coefficient among each variable.

Each sensing value was used to detect an abnormal status of the module. As shown in Figure 4, several variables (s_6,s_7 and s_8) have strong correlations with Y, whereas s_2 has a comparatively weak correlation with the output. However, these datasets have a comparatively small number of input attributes. Thus, the effectiveness of the proposed framework is demonstrated, as it has the advantage of expanding the input dimensions using the quantum embedding technique. Figure 5 shows the proposed framework using the quantum embedding method.

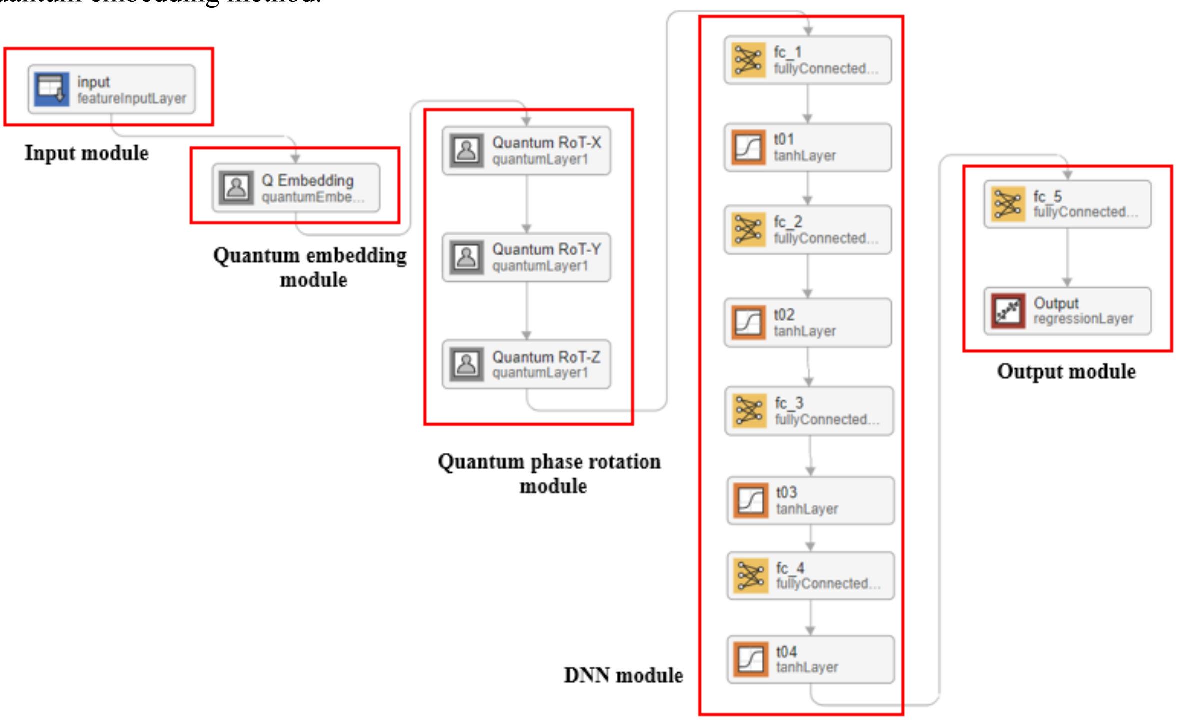


Figure 5: Quantum neural net incorporating quantum embedding.

As shown in Figure 5, the QNN consists of five parts: the input layer, quantum embedding layer, quantum phase operating layers, DNN layers and output layer. Table 3 provides the detailed network configuration.

Table 3: Configuration of the proposed QNN.

Part No.	Name	Configuration	Learnable parameters
1	Input	• Input layer: (8x1)	-
2	Quantum embedding	• Quantum embedding layer: (16x1)	-
3	Quantum Phase operation	• Three layers: Rot-X, Rot-Y, and Rot-Z	α,β and γ (phase rotation angle angles)
4	DNN layers	• 4 layers: each layer is a combination of a fully connected layer and an activation function (tanh)	Each weight and bias
5	Output	• One fully connected layer • One Output layer (1x1)	Weights / bias

To demonstrate the effectiveness of the proposed QNN, these data are compared with results obtained from three existing DNN machines. Figure 6 presents a comparison of the three DNN-based methods.

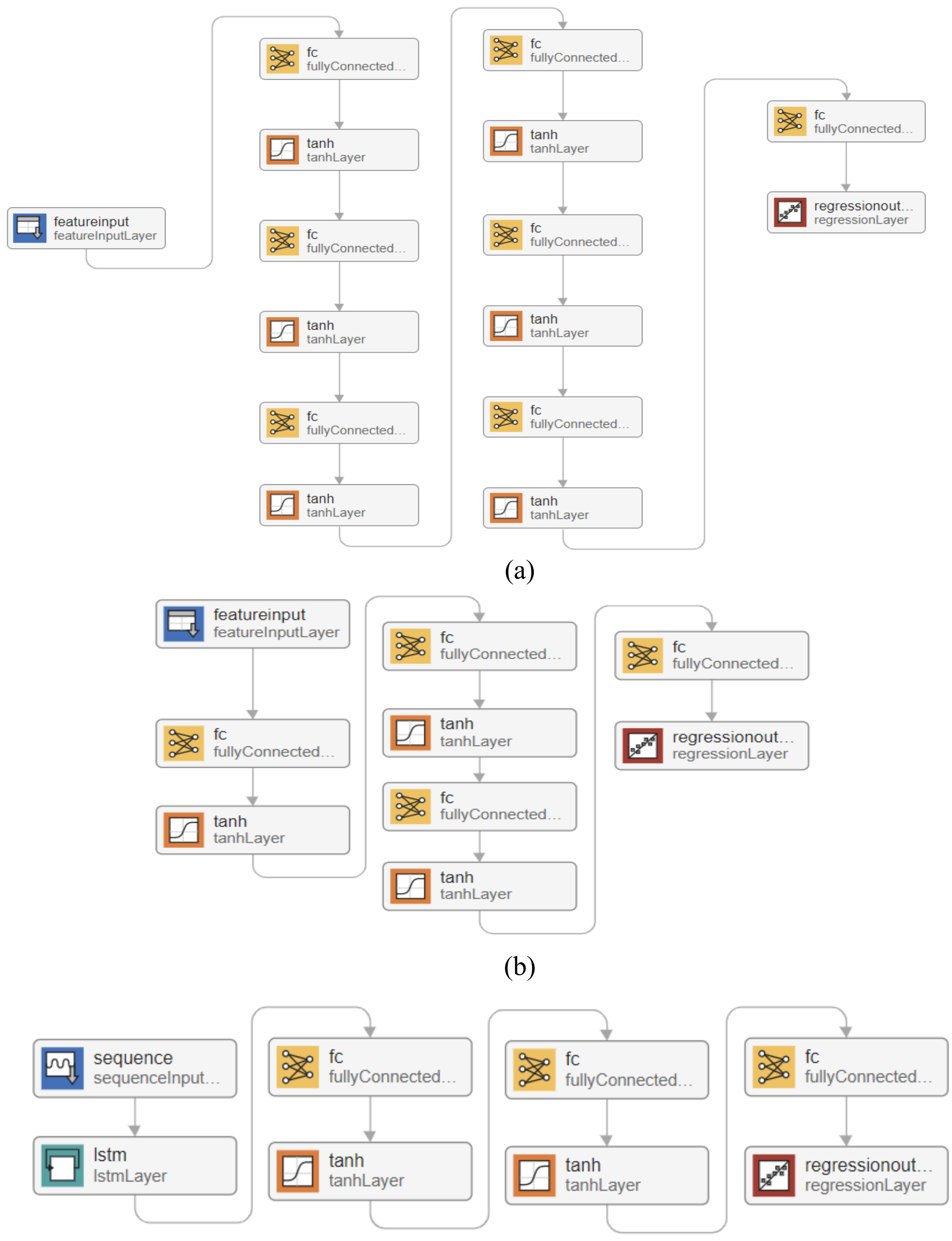
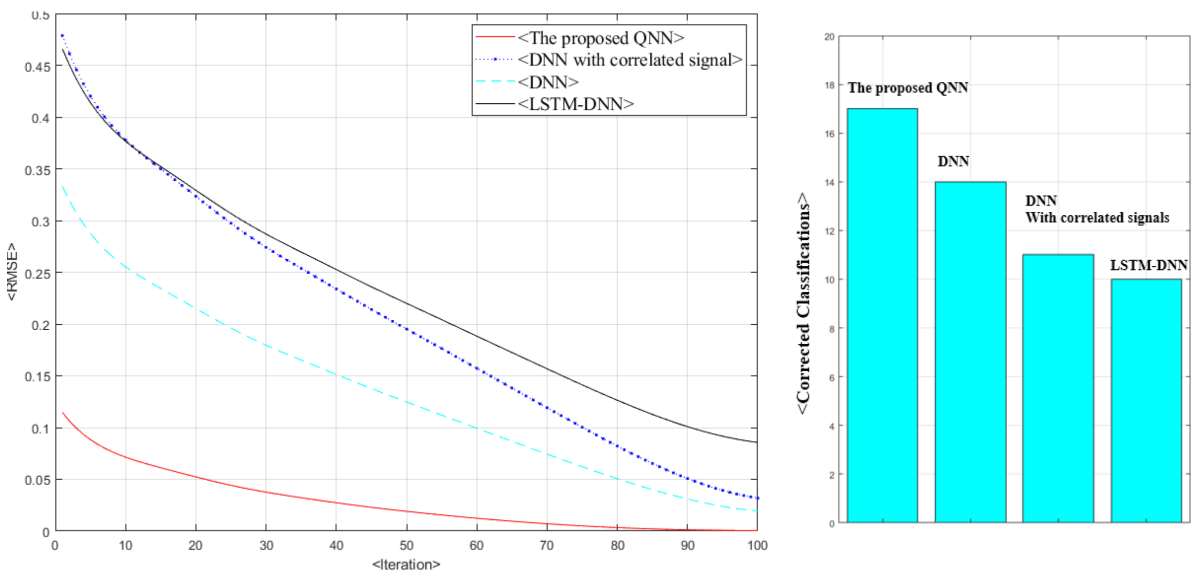


Figure 6: Comparison methods.

The first comparison model is a general DNN model (Figure 6(a)) with eight sensory data inputs. Although the proposed QNN used 16 expanded features for subsequent quantum phase operations and DNN layers, the comparison model used only eight signals. The second model (Figure 6(b)) used only highly correlated inputs (s_6,s_7 and s_8) as shown in Figure 4. The final comparison model was an LSTM-DNN model with eight signal inputs, as shown in Figure 6(c). Using the training set from the data, each configuration was determined to be an experimentally superior model. Among 100 datasets, 80% of the data were extracted randomly as a training set and the remainder was used as a test set. Figure 7(a) shows the training errors of the four methods, including that of the proposed framework.



(a) (b)

Figure 7: Training errors and test accuracies.

As shown in Figure 7(a), the proposed QNN has the lowest root mean square error (RMSE) among all four methods. This implies that the expanded signals obtained using the quantum embedding method are effective for feature extraction. The DNN model with only three signals and the LSTM-DNN models performed worse than the general DNN with eight signals. As shown in Figure 7(b), the proposed QNN correctly classified 17 of the 20 test data and outperformed the other comparison models. The proposed QNN expands the dimensions of the input data and uses subsequent quantum operators, therefore it is helpful in data analyses to overcome input data issues.

5 CONCLUSION AND FURTHER STUDY

Quantum computing is a contemporary engineering discipline that has attracted much attention in recent years. Because it has several advantages with respect to representing information, the ability to expand dimensions and quantum operations is considered a representative characteristic. This study applies these characteristics to data analysis with input data issues. When the number of input data attributes is comparatively small, it is difficult to analyze the dataset. In real applications, this is limited to securing sufficient input attributes and data for various reasons. The proposed framework is useful in this context. Limited input data are expanded using the quantum embedding, and are used to extract more meaningful features with subsequent quantum operations. This study integrates a quantum technique with a DNN machine. In the QNN model, the quantum layers are located between the input layer and subsequent DNN layers. The quantum embedding layer and quantum phase operation layers perform dimensional expansion and dynamic conversion. Then, additional learnable parameters are updated using quantum operations and backpropagation. The effectiveness of the proposed framework was experimentally studied by comparing it with several existing DNN machines.

This study considers a limited degree of quantum embedding and operations. Although there are numerous quantum dynamics and characteristics, more effective computations and analyses are required. In particular, in further studies, there is a need to consider entangled treatments among the correlated input attributes.

ACKNOWLEDGMENTS

This research study was supported by The Basic Science Research Program through National Research Foundation of Korea (NRF), funded by the Ministry of Education, S. Korea (grant number: NRF-2021R1A2C1008647).

REFERENCES

Batra, K., K.M. Zorn, D.H. Foil, E. Minerali, V.O. Gawriljuk, T.R. Lane, and S. Ekins. 2021. "Quantum Machine Learning Algorithms for Drug Discovery Applications" *J. Chem. Inf. Model*. 61(6): 2641-2647.

Gianani, I., I. Mastroserio, L. Buffoni, N. Bruno, L. Donati, V. Cimini, M. Barbieri, F.S. Cataliotti, and F. Caruso. 2022. "Experimental Quantum Embedding for Machine Learning". *Advanced Quantum Technologies* 5(8).

Hamad, R. A., M. Kimura, and L. Lundstrom. 2020, "Efficacy of Imbalanced Data Handling Methods on Deep Learning for Smart Homes Environments". *SN Computer Science* 1(204).

Hang, H., Y. Cai, H. Yang, and Z. Lin. 2023, "Under-bagging Nearest Neighbors for Imbalanced Classification". *Journal of Machine Learning Research* 23: 1-63.

Huang H., M. Broughton, M. Mohseni, R. Babbush, S. Boixo, H. Neven, and J. R. McClean. 2021. "Power of Data in Quantum Machine Learning". *Nature Communications* 12:2631.

Javier, V., O. Joaquin, G. Ana, and A.S. Larzabal. 2022. "Quantum Deep Learning for Steel Industry Computer Vision Quality Control". *IFAC-PapersOnLine* 55(2): 337-342.

Hidary, J. D. 2019, *Quantum Computing: An Applied Approach*.1st ed. Springer Nature Switzerland AG.

Knizia, G., and G. K. Chan. 2013. "Density Matrix Embedding: A Strong-coupling Quantum Embedding Theory". *Journal of Chem. Theory Computing* 9(3): 1428-1432.

Li, D., H. Chen, and Q. Shi. 2018. "Learning from Small Datasets Containing Nominal Attributes". *Nerocomputing* 291(24): 226-236.

Lee, H., S. Han, and K. Park. 2020, "Generative Adversarial Network-based Missing Data Handling and Remaining Useful Life Estimation for Smart Train Control and Monitoring Systems". *Journal of Advanced Transformation* 2020(1):1-15.

Lee, H., S. Han, K. Park, H. Lee, and T. Kwon. 2020, "Real-time Hybrid Deep Learning-based Training Learning Safety Prediction Framework of Railway Vehicle". *Machines* 9(7): 1-18.

McMahon, D. 2007, *Quantum Computing Explained*. 1st ed. Hoboken, New Jersey: Wiley-IEEE Computer Society, Pr.

Meyer, J.J., M. Mularski, E. Gil-Fuster, A.A. Mele, F. Arzani, A. Wilms, and J. Eisert. 2023. "Exploiting Symmetry in Variational Quantum Machine Learning". *PRX Quantum* 4:010328.

Oh, E., and H. Lee. 2020, "An Imbalanced Data Handling Framework for Industrial Big Data using Gaussian Process Regression-based Generative Adversarial Network". *Symmetry* 12(4) : 1-19.

Powell, J. H., and N. Mustafee. 2017. "Widening Requirements Capture with Soft Methods: An Investigation of Hybrid M&S Studies in Healthcare". *Journal of the Operational Research Society* 68(10):1211-1222.

Sarafianos, N., X. Xu, and I. A. Kakadiaris. 2018. "Deep Imbalanced Attribute Classification using Visual Attention Aggregation". In *Proceedings of the ECCV 2018 Conferences*, September 8th-14th, Munich, Germany, 1-18.

Steeb, W., and Y. Hardy. 2012, Problems and Solutions in Quantum Computing and Quantum Information, 3rd ed. Singapore, World Scientific.

Zhang, Y., and C. Ling. 2018. "A Strategy to Apply Machine Learning to Small Datasets in Materials Science". *npj Computational Materials* 4:25.

Zhu, T., Y. Lin, and Y. Liu. 2020. "Improving Interpolation-based oversampling for imbalanced data learning". *Knowledge-Based Systems* 187:104826.

AUTHOR BIOGRAPHIES

HYUNSOO LEE is an Professor in the School of Industrial Engineering at Kumoh National Institute of Technology, S. Korea. His primary research interest is in the area of nonlinear controls, simulation-based optimization handing various type uncertainties, deep learning-based hybrid simulation, and quantum computing. He is a senior member of IEEE. His email address is hsl@kumoh.ac.kr and his homepage is http://kit.kumoh.ac.kr/~hsl.

AMARNATH BANERJEE is a Professor in the Department of Industrial & Systems Engineering, Texas A&M University, College Station. His research interests include modeling, simulation and visualization, with applications in health care systems, information systems, manufacturing systems and Industry 4.0. His email address is banerjee@tamu.edu and his website is https://engineering.tamu.edu/industrial/profiles/abanerjee.html.