2024 IEEE INTERNATIONAL WORKSHOP ON MACHINE LEARNING FOR SIGNAL PROCESSING, SEPT. 22–25, 2024, LONDON, UK

QUANTUM POSITIVE UNLABELED LEARNING ALGORITHMS WITH APPLICATIONS TO ENERGY

Salil Naik[†]*, Glen Uehara*[*]*, Kristen Jaskie*[*]*, Leslie Miller*[*]*, Andreas Spanias*[*]

[†]SCAI at Arizona State University, [*]SenSIP Center, School of ECEE at Arizona State University

2024 IEEE 34th International Workshop on Machine Learning for Signal Processing (MLSP) | 979-8-3503-7225-0/24/$31.00 ©2024 IEEE | DOI: 10.1109/MLSP58920.2024.10734794

ABSTRACT

Positive unlabeled (PU) learning is a semi-supervised machine learning (ML) approach to a binary classification problem where a small subset of the data has positive labels, and the rest of the data is unlabeled. PU learning algorithms can be very efficient at reducing the computational cost of training ML classifiers because the labeling of data for supervised machine learning algorithms is time intensive. In this study, we develop new quantum PU (QPU) learning methods for photovoltaic (PV) fault detection and perform comparative analysis with classical PU learning methods. Quantum circuits for PU learning are designed and evaluated in this study. In addition, we examine various quantum machine learning (QML) and PU learning challenges including quantum measurement noise. As with many QML simulations, the presence of quantum noise and the heavy computational and memory requirements affect classification accuracy. Nevertheless, regardless of quantum noise effects and limited computational resources, our initial designs of quantum PU learning circuits achieved an accuracy score of 79.6%. This was achieved by implementing QPU learning with a quantum-classical hybrid neural network (QNN) architecture. In our study, we present simulation results with and without quantum noise models for comparison purposes.

***Index Terms*—** Quantum Computing, Machine Learning, Positive Unlabeled Learning, PV Arrays

1. INTRODUCTION

The development of quantum machine learning (QML) methods in various applications is a focus area in many federal and industrial research initiatives. In fact, quantum AI is a White House priority area [1] for research and workforce development. This paper provides new designs of QML circuits for use in various applications including solar energy monitoring. More specifically, quantum positive unlabeled (PU) machine learning algorithms are developed for photovoltaic (PV) fault detection. Previous work demonstrated the use of customized ML algorithms on classical computers for solar array or PV fault detection and the results have been reported in [2-6]. The overall concept of PV monitoring and fault detection using ML is captured in Figure 1. The choice of PV applications is driven by the fact that the new quantum positive unlabeled learning techniques that we propose are well suited for "binary" fault detection applications. In addition, our group has access to PV data from our own solar energy testbed [7] and from NREL [8]. Automatic PV fault detection is crucial in maintaining power efficiency and reducing the cost of maintenance relative to manual fault detection which is labor intensive. Preliminary studies of QML in PV fault detection reported limited accuracy [9] because of various effects including quantum noise and limited qubit precision. In addition, the authors of [9] reported that training QML algorithms on quantum simulators required very long training times with heavy computing and memory requirements.

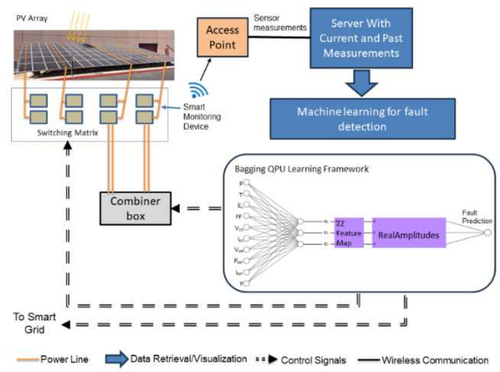

Fig. 1. *PV* Monitoring System and QPU based fault detection. Individual Smart Monitoring Devices are attached to each solar cell to collect sensor measurements which are passed through the QPU learning algorithm to perform fault detection.

In this study, we propose to leverage the power of PU learning to reduce labeling costs while maintaining accuracy. We design and evaluate custom quantum circuits for quantum PU (QPU) learning, and we apply our algorithms to PV fault detection. We also evaluate these circuits with quantum noise models and provide results in terms of detection accuracy and computational cost. We emphasize that this study is distinctly different than the previous QML effort [9], which used a simple hybrid quantum neural network, in that the proposed QPU learning includes new quantum circuit development. In addition, the proposed QPU method is different than previous PU learning approaches reported in [6] that were implemented in the classical domain.

The main contributions of this study are: a) the design of new QPU circuits, b) extensive simulations of PV fault detection using quantum simulators, c) comparison of quantum and classical detection algorithms with respect to performance and computational requirements, d) design and implementation of a bagging process for QPU-based labeling, e) tabulating classical vs quantum comparative results in terms of epochs, training times and testing accuracy, and f) examining the effects of quantum noise on QPU learning using extensive simulation studies based on two quantum noise models.

2. BACKGROUND AND RELATED WORK

2.1. PU Learning

Our study explores the design of quantum positive unlabeled learning circuits for automatic solar energy fault detection, with an emphasis on decreasing total computational costs for QPU training while maintaining a reasonable level of accuracy. Previous research into the application of classical machine learning algorithms in solar energy fault detection has yielded high accuracies. A classical feed-forward neural network trained on solar fault data achieved a detection accuracy in excess of 88% [1]. However, the study reported in [1] implemented a form of supervised deep learning with a large training database where all datapoints were labeled. For customized models and large datasets, fully supervised models are often not feasible as manual data labeling becomes prohibitively expensive. In these situations, classical forms of machine learning may not be the appropriate choice for fault detection. However, positive unlabeled (PU) learning [10] offers an efficient alternative in terms of a semi-supervised learning paradigm. In these algorithms, only a small subset of positive datapoints needs to be labeled, and the rest—both positive and negative can remain unlabeled. In accordance with the norms of PU learning research, our study will make the selected completely at random (SCAR) assumption [11], that the positive labeled datapoints are chosen independently of their features. Additionally, the algorithms developed in our study will assume an unknown class prior, where the ratio of positive to negative data in the dataset is not known.

2.2. Quantum Machine Learning and Quantum PU Learning

QML potentially has numerous advantages over classical ML computation. In QML, the principles of superposition and entanglement are leveraged to promise exponential improvements in speed. However, the current state of quantum computation hardware has not yet advanced far enough to realize these benefits. Nevertheless, several classical ML algorithms have been adapted into the current generation of noisy intermediate-scale quantum (NISQ) computers [12]. Some of the problems faced by these approaches include qubit noise and precision errors due to current insufficiencies of NISQ hardware. Yet, certain QML studies have demonstrated potential quantum advantage [13]. Motivated by the potential of QML speed and the efficiency of PU learning, in this paper we develop and evaluate Quantum PU algorithms. We also design and evaluate custom quantum circuits and provide promising preliminary results for PV fault detection. We note that the proposed methods can be extended to other applications including image and audio signal processing. Although the use of QPU faces the challenges of quantum noise and qubit precision tradeoffs, we show that our hybrid machine learning approach to QPU learning provides computational benefits due to the smaller size of the quantum circuit and thus potentially a reduced error rate.

2.3. Previous Signal Processing and Machine Learning Methods for PV Monitoring

Previous research in the field of automatic PV fault detection has proposed the installation of smart monitoring devices (SMDs) and customized signal processing to monitor and control individual solar cells within the array [3]. These SMDs can then contribute to a larger cyber-physical PV monitoring system for ML-based fault detection (Figure 1). Some classical ML methods in these applications include support vector machines (SVM) [4], neural networks [2], and even a graph-based semi-supervised algorithm [5]. Extensions of classical machine learning to PV soilage detection were also studied in [6]. The QPU learning algorithm studied here assumes the presence of measurements provided by SMDs, which are an integral part of our PV testbed. SMDs provide voltage, current, temperature and irradiance measurements for each panel. The PV testbed and SMDs are described in detail in [7], and real-time measurements from this testbed were used in [14].

Previous papers have found success in the applications of classical PU learning on solar fault data. A feedback modified logistic regression (MLRf) algorithm achieved an F-score of 0.886 in fault vs no fault classification with just 2% of the data being labeled [6]. Other studies have also explored the application of quantum classifiers in PV fault detection and have yielded good results. Of the various quantum-classical hybrid neural networks (QNNs) developed in [9], base case results approached the accuracy of a classical NN model but with excessive computational simulation cost—often requiring 4-5 days of QNN training time even with GPU based systems.

Motivated by the decreased labeling costs associated with PU learning and the advantages of quantum computing, we propose various QPU circuit designs and evaluate them in terms of accuracy and computational complexity.

The rest of the paper is organized as follows. Section 3 will describe the PV fault detection dataset used in this study. Section 4 discusses the design of the bagging PU learning framework. Then, section 5 will discuss the design of the QPU algorithm. Section 6 will discuss the simulation results. Section 7 will describe the QPU simulation results with quantum noise. Finally, section 8 will compare the results of the QPU algorithms to classical PU methods, and section 9 will state the conclusions of the study and future work.

3. PV DATASET

The data used in this QPU study was obtained through the National Renewable Energy Laboratory's (NREL) PVWatts Calculator [8]. We note that the NREL data set was previously used successfully in numerous studies and allows generation of data for various PV array configurations, fault types and shading profiles. In fact, the NREL utility estimates the cost and energy production of grid-connected solar energy systems worldwide. The dataset contains ten features: power output (P), temperature (T), irradiance (E_e), fill factor (FF), open circuit voltage (V_{OC}), short-circuit current (I_{SC}), maximum voltage (V_{MP}), maximum power (P_{MP}), maximum current (I_{MP}), and gamma (γ)—the ratio of power over irradiance. NREL provided datasets for adequate training of classical PU and QPU algorithms in this study that aligned well with our goals of fault detection for various conditions. The data was obtained from January to December 2006 with a sampling duration of one hour. The data typically contains 4 faults—shaded, soiled, short-circuit, and degraded—however, this study modifies the data for binary classification of fault vs standard test conditions (STC). After this transformation and data balancing, the final dataset contains 8,592 datapoints evenly distributed into positive and negative classes. In this study we define the positive class to signify a detected fault and the negative class to signify STC. These datapoints were then adapted to the PU learning problem by leaving only a small subset of positive datapoints labeled and stripping the labels of the rest. In this study, 25% of the positive datapoints—and therefore 12.5% of the total—were labeled, while the remaining 87.5% were stripped of their labels. This method of obtaining positive unlabeled data was favorable for the experimental conditions of our research. Since the prior label of each datapoint is known, the accuracy of the model can be better identified under development and testing circumstances.

4. PU LEARNING FRAMEWORK DESIGN

In our study, we built a bagging PU learning framework in which any QML algorithm can be quickly and easily inserted. This framework is heavily based on the ideas presented by [15]. In a bagging PU learning algorithm, a series of "bags" are chosen at random from the initial dataset. Each of these bags consists of the entire labeled data and a subset of the unlabeled data of the same size as the labeled dataset. The unlabeled data within the bag is temporarily assigned a negative label. The machine learning model is then trained on this bag and used to predict the labels for the rest of the unlabeled data outside the bag. After multiple iterations of this bagging and prediction process, the predictions from every iteration are averaged to yield a final predicted label for each unlabeled datapoint. These predictions are then used to support fully supervised learning of the dataset.

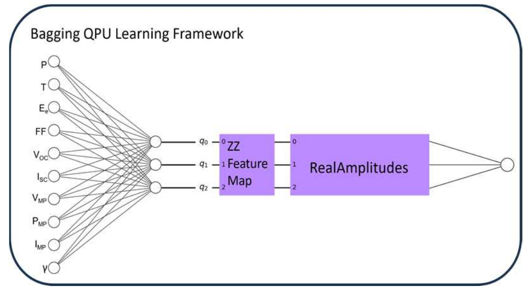

Fig. 2. Design of the hybrid QNN. A bagging QPU learning framework encapsulates a hybrid neural network to allow for QPU prediction.

Algorithm 1 describes the function of the PU learning framework designed for this study. Within the algorithm, s_t can be replaced with any classical or quantum machine learning algorithm that can classify datasets. This "hot swappable" design allowed for ease of use of this framework as well as flexibility when determining the algorithm with the highest accuracy. In pursuit of this same goal, the classical neural network *f* at the end of the algorithm was kept constant throughout the study. This network had 4 layers: an input layer with 10 neurons, two consecutive hidden layers with 20 neurons each, and finally the output layer with 1 neuron. Hyperparameters such as number of epochs, learning rate, activation functions, and loss functions were also kept constant. This ensured a fair comparison in this study.

Algorithm 1. Bagging QPU Learning Algorithm

INPUT: Unlabeled data, U
Labeled positive data, P
Size of P, K
Number of repetitions, R

OUTPUT: Trained model, $f(x)$
Initialize $\forall x \in U$, $n(x) \leftarrow 0$, $s(x) \leftarrow 0$, where $n(x)$ is the number of times x has been scored and $s(x)$ the collection of scores of x
Initialize $r \leftarrow 0$
while $\exists x \in U$ such that $n(x) = 0$ **do**
Select arbitrary $U_t \subset U$ of size K.
Train binary QML algorithm s_t to discriminate P against U_t.
For all $x \in U \setminus U_t$, update:
$s(x) \leftarrow s(x) + s_t(x)$,
$n(x) \leftarrow n(x) + 1$.
end while
Train classical neural network f with inputs $P \cup U$ and label $s(x)/n(x)$ for $x \in U$ and 1 for $x \in P$.
Return $f(x)$.

5. QPU ALGORITHM DESIGN

The QNN used in this study took the form of a 4-layer neural network including: an input layer with 10 neurons, one hidden layer with 3 neurons, a 3-qubit quantum layer, and finally the output layer with 1 neuron (Figure 2). The quantum layer consisted of a 3 qubit ZZFeatureMap to

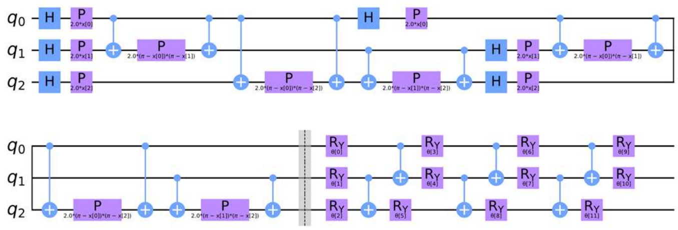

Fig. 3. Detailed view of the ZZFeatureMap and RealAmplitudes circuits used in the QNN.

encode the classical data into a quantum state, followed by a 3 qubit RealAmplitudes function to process the quantum state to support machine learning.

The detailed or decomposed quantum circuit for the QNN is shown in Figure 3. Different configurations of both the classical and quantum parts of this model were tested in pursuit of higher accuracies and this model architecture proved to be the most effective of those tested at this time. The neural network also implemented a learning rate scheduler to prevent overfitting of the data; the learning rate started at 0.07 and linearly trended towards 0 as the model trained. A standard binary cross entropy loss function was used in this implementation. However, this loss function comes with its own drawbacks, namely inherent biases in the context of a bagging PU learning model. Future work must be done to explore new asymmetric loss functions such as [16] for this application.

For comparison purposes, the quantum support vector machine (QSVM) [17] was also tested within the PU learning framework in this study. This model implemented Qiskit's [18] FidelityQuantumKernel and PegasosQSVC. While optimizing the simulations, runtime was the most difficult challenge to overcome. Even though the dataset had 10 features, using a 10-qubit QSVM was not feasible with a realistically sized dataset. To counteract this, some form of dimensionality reduction was needed to reduce the number of qubits. In this study, systems with two, three, and four qubits were examined; however, simulations with five qubits and beyond were still not feasible because of excessive training times and limited computational resources within time constraints.

6. SIMULATION RESULTS

The QNN has shown promising results, achieving a 79.6% single-run testing accuracy with 3000 epochs in the bagging step + 1000 final training epochs. However, some biases in the network can be seen in Figure 4 as the predicted labels skew more towards 0 (no fault) than 1 (fault). This bias is common in PU learning algorithms due to the inherent skewness of the training data and can be counteracted with asymmetric loss functions [16].

The distribution of the estimated sample labels roughly follows an inverse bell curve, with a majority of the points estimated close to 0 or 1. This shows that the model has extremely high confidence in estimating the value of these unlabeled points. Furthermore, the framework disregards any datapoint with an estimated value between 0.45 and 0.55; disregarding a few datapoints while training is less harmful to the final model than training with potentially inaccurate labels.

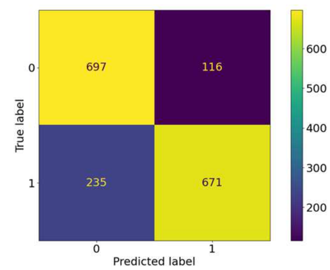

Fig. 4. Confusion matrix of quantum neural network in QPU learning framework (79.6% accuracy).

When performing the same QPU simulations with QSVM's, we found that learning from the data was minimal. The reasons for this are limitations of quantum computing simulators and excessive training time. For example, with a 10-qubit implementation of a QSVM with a reasonably sized dataset, model training times would be currently in the order of months. Even after implementing methods to increase the speed of the algorithm, the runtimes were still too high for the scope of this study. Additional work on QSVM's in the future will focus on reducing training time with a modest number of qubits.

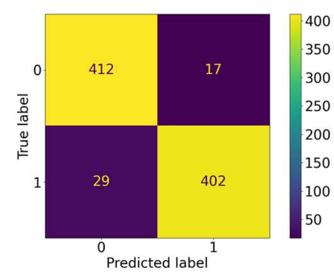

Fig. 5. Confusion matrix of classical neural network in PU learning framework (94.6% accuracy).

While the bagging NN achieved a higher final accuracy, it required more than 40 times the number of training epochs than the bagging QNN. However, due to the inherent inefficiencies of current quantum simulators, the training took significantly longer for each quantum algorithm. Advances in quantum computing simulators and hardware will considerably reduce training times in the future. Figure 5 corresponds to the highest single-run performance of the classical PU approach.

7. QPU PERFORMANCE WITH QUANTUM NOISE

We used the Qiskit simulator to simulate the bagging QNN with quantum noise effects. The simulation incorporated a generated noise model corresponding to the IBMQ Osaka backend [19-21]. The Osaka backend represents Eagle R3 quantum hardware, which consists of 127 qubits. By employing the noise model, coupling maps, and basis gates

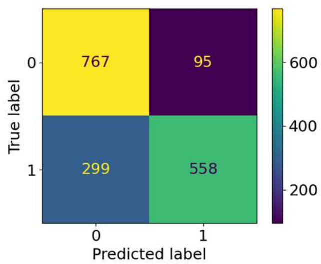

Fig. 6. Confusion matrix of quantum neural network with noise model in QPU learning framework (77.1% accuracy).

provided by the IBMQ Osaka backend, we could closely replicate the estimated noise effects of the Osaka quantum processor within our quantum simulation framework. The noise model simulates the error and noise characteristics of the hardware. The basis gates are a list of the supported quantum gates in the hardware while the coupling map shows the layout of the physical qubits in hardware and how they can be entangled with each other. The integration of all of these in the simulation creates a good estimate of the model's

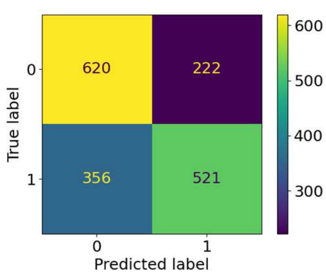

Fig. 7. Confusion matrix of quantum neural network with noise model, coupling map, and basis gates in QPU learning framework (66.4% accuracy).

performance in quantum hardware, and thus its performance in potential real-world applications. Figure 6 shows the confusion matrix from the highest single-run accuracy of the QPU approach using just the noise model from IBM Osaka while Figure 7 is from the QPU approach using the noise model, coupling map, and basis gates from IBM Osaka. Employing this model, our QNN achieved a classification accuracy of 66.4%. When simulating with just the noise model from the Osaka backend, we achieved an accuracy of 77.1%, which represents only a 2% loss in accuracy. Advancements in quantum noise mitigation have been reported recently in [22,23]. We note, however, that quantum noise mitigation is still an open problem.

We ran initial QNN Monte Carlo trials to estimate the average accuracy of the models. However, due to the very long runtimes of quantum algorithms, we could not perform these trials using the entire dataset. Rather, we segmented the existing dataset into five distinct parts and trained a classifier on each segment. With the unusually short dataset available and the exceedingly long training times for the Monte Carlo simulation, the overall accuracy was lower than the single-run simulation. However, we expect Monte Carlo trials with the full dataset to closely match the single-run accuracies.

8. COMPARISON OF QPU TO CLASSICAL PU LEARNING

To serve as a metric to compare the performance of the QPU learning algorithms, classical neural network PU learning classifiers were also trained using the same bagging framework. The highest single-run accuracy yielded by these classifiers was 94.6%. Table 1 shows the final testing accuracies of each algorithm and the number of epochs in both the bagging and final model training steps. We note that the bagging methods mentioned in Table 1 were used in several classical PU learning studies [24,25]. The QPU studies and accuracy figures reported in Table 1, however, are from our own simulations. We also note that Figures 4-7 show the confusion matrices associated with the simulations listed in Table 1.

Table 1. Comparison of various PU learning algorithms in PV fault detection in terms of performance and training time.

Fault Detection Algorithm	Epochs (Bagging + Final)	Training Time	Single-run Testing Accuracy
Bagging NN	160,000 + 1,000	6 min	94.6%
Bagging QNN	3,000 + 1,000	1.5 days	79.6%
Bagging QNN with Noise Model	3,000 + 1,000	4 days	77.1%
Bagging QNN with Noise Model, Coupling Map, and Basis Gates	3,000 + 1,000	4 days	66.4%

9. CONCLUSION AND FUTURE WORK

As noted in the introduction there is strong interest in the areas of quantum computing and quantum machine learning [26-28]. In this paper, we proposed a Quantum Positive Unlabeled learning algorithm, which we used in a PV fault detection application, however this algorithm can be generalized and applied to other application domains and ML algorithms. In this application of QPU learning, the quantum-classical hybrid neural networks achieve accuracies nearing 80%. Further optimization of the QPU algorithm explored in this study could yield slightly higher performance. The greatest challenge we faced was excessive quantum simulation time, particularly in terms of training the QML algorithms. In addition, although quantum noise resulted in only 2% reduction in accuracy, simulations with quantum noise that included coupling maps and basis gates reduced the accuracy considerably.

Current QPU algorithm design is based on a tradeoff between accuracy and algorithmic complexity. The QPU models achieved lower accuracies than the traditional PU models, however required significantly fewer epochs to train. This could potentially be beneficial in a real-world use case with large amounts of data—such as in a large solar farm. Integrating QPU learning would reduce the total computation costs when run on quantum hardware. In this use case, a lower prediction accuracy would only lead to reduced efficiency but would not lead to unacceptable results. In terms of broader impact, the QPU PV fault detection quantum circuit designs and results are expected to elevate interest in other application areas. In fact, future research in diverse applications of QPU in other fields is expected to yield results that will drive improved algorithm designs particularly with quantum noise. In summary, this study explored the design of new QPU circuits which reduced the costs associated with dataset labeling. Additional work is needed though to fully address the effects of quantum noise in QPU learning.

ACKNOWLEDGEMENT

This study was supported in part by the NSF IRES award 1854273 and by the ASU SenSIP consortium project on solar energy.

REFERENCES

[1] White House Publication on web: “Two Presidential Directives Advancing Quantum Technologies,” Web Publication on May 2022 (last accessed August 2, 2024) https://www.whitehouse.gov/briefing-room/statements-releases/2022/05/04/fact-sheet-president-biden-announces-two-presidential-directives-advancing-quantum-technologies/

[2] S. Rao, A. Spanias and C. Tepedelenlioglu, “Solar Array Fault Detection using Neural Networks,” *2019 IEEE International Conference on Industrial Cyber Physical Systems (ICPS)*, Taipei, Taiwan, 2019, pp. 196-200.

[3] H. Braun, S. T. Buddha, V. Krishnan, A. Spanias, C. Tepedelenlioglu, M. Banavar, “Signal Processing for Solar Array Monitoring Fault Detection and Optimization,” San Rafael, CA, USA: Morgan & Claypool, pp. 1-95, 2012.

[4] A. Mellit, G. M. Tina and S. A. Kalogirou, “Fault detection and diagnosis methods for photovoltaic systems: A review”, *Renewable and Sustainable Energy Reviews*, vol. 91, pp. 1-17, Aug. 2018.

[5] Y. Zhao, R. Ball, J. Mosesian, J. F. D. Palma and B. Lehman, “Graph-based semi-supervised learning for fault detection and classification in solar photovoltaic arrays”, *IEEE Trans. Power Elect*, no. 5, p. 2848, 2015.

[6] K. Jaskie, J. Martin, and A. Spanias, “PV Fault Detection Using Positive Unlabeled Learning,” *Applied Sciences*, vol. 11, no. 12, p. 5599, Jun. 2021.

[7] A. Spanias, C. Tepedelenlioglu, E. Kyriakides, D. Ramirez, S. Rao, H. Braun, J. Lee, D. Srinivasan, J. Frye, S. Koizumi, Y. Morimoto, “An 18 kW Solar Array Research Facility for Fault Detection Experiments,” *Proc. 18th IEEE MELECON*, Limassol, April 2016.

[8] A. P. Dobos, “PVWatts Ver. 1 technical reference”, 2013.

[9] G. Uehara, S. Rao, M. Dobson, C. Tepedelenlioglu and A. Spanias, “Quantum Neural Network Parameter Estimation for Photovoltaic Fault Detection,” *2021 12th IEEE IISA*, July 2021, pp. 1-7.

[10] K. Jaskie and A. Spanias, “Positive Unlabeled Learning,” Morgan & Claypool Publishers (now Springer nature), AI and Machine Learning Eds. R. Brachman and F, Rossi, 152 pages, ISBN 9781636393087,

[11] C. Elkan and K. Noto. (2008). Learning classifiers from only positive and unlabeled data. Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 213-220. 10.1145/1401890.1401920.

[12] G. Torlai and R. G. Melko. “Machine-learning quantum states in the NISQ era.” *Annual Review of Condensed Matter Physics* 11 (2020): 325-344.

[13] D. Ristè, M. P. Da Silva, C. A. Ryan, A. W. Cross, A. D. Córcoles, J. A. Smolin, J. M. Gambetta, J. M. Chow, and B. R. Johnson. “Demonstration of quantum advantage in machine learning.” *npj Quantum Information* 3, no. 1 (2017): 16.

[14] S. Rao, D. Pujara, C. Tepedelenlioglu, and D. Srinivasan. “Real-time Solar Array Data Acquisition and Fault Detection using Neural Networks.” In *2023 IEEE 6th (ICPS*, pp. 1-5. IEEE, 2023.

[15] F. Mordelet and J. P. Vert, “A bagging SVM to learn from positive and unlabeled examples”, *Pattern Recognition Letters*, vol. 37, pp. 201–209, 2014.

[16] C. Wang, J. Pu, Z. Xu and J. Zhang, “Asymmetric Loss for Positive-Unlabeled Learning,” *2021 IEEE ICME*, Shenzhen, China, 2021, pp. 1-6.

[17] P. Rebentrost, M. Mohseni, and S. Lloyd. "Quantum support vector machine for big data classification." *Physical review letters* 113, no. 13 (2014).

[18] R. Wille, R. Van Meter, and Y. Naveh (2019, March). IBM’s Qiskit tool chain: Working with and developing for real quantum computers. In *2019 Design, Automation & Test in Europe Conference & Exhibition,* pp. 1234-1240. IEEE.

[19] Qiskit Development Team, “NoiseModel — Qiskit Aer 0.13.3 documentation,” Qiskit Aer, 06-Feb-2024. (last accessed: August 2, 2024). https://qiskit.github.io/qiskit-aer/stubs/qiskit_aer.noise.NoiseModel.html#qiskit_aer.noise.NoiseModel.from_backend.

[20] Qiskit Development Team, “Noise Models (qiskit_aer.noise) — Qiskit Aer 0.13.3 documentation,” Qiskit Aer, (last accessed August 2, 2024). https://qiskit.github.io/qiskit-aer/apidocs/aer_noise.html.

[21] IBM Quantum Development Team, “FakeOsaka | IBM Quantum Documentation,” IBM Quantum, https://docs.quantum.ibm.com/api/qiskit-ibm-runtime/qiskit_ibm_runtime.fake_provider.FakeOsaka.

[22] Z. Cai, R. Babbush, Simon C. Benjamin, S. Endo, W. J. Huggins, Y. Li, J. R. McClean, and T. E. O’Brien (2023). Quantum error mitigation. *Reviews of Modern Physics*, *95*(4).

[23] A. Shaib, M. H. Naim, M. E. Fouda, R. Kanj, and F. Kurdahi. "Efficient noise mitigation technique for quantum computing." *Scientific Reports* 13, no. 1 (2023): 3912.

[24] S. De Block, and J. Bekker (2022). Bagging Propensity Weighting: A Robust method for biased PU Learning. 4th *Int. Workshop on Learning with Imbalanced Domains: Theory and Applications* (pp. 23-37). PMLR.

[25] L. Zhao, H. Ma, J. Dong, X. Wu, H. Xu, and R. Niu "A Comparative Study of Landslide Susceptibility Mapping Using Bagging PU Learning in Class-Prior Probability Shift Datasets." *Remote Sensing* 15, no. 23 (2023): 5547.

[26] P. H. Wang, J. H. Chen, Y. Y. Yang, C. Lee, and Y. J. Tseng (2023). Recent advances in quantum computing for drug discovery and development. *IEEE Nanotechnology Magazine*, *17*(2), 26-30.

[27] Z. Yang, Z. Maede, and J. Raj. "A survey of important issues in quantum computing and communications." IEEE Communications Surveys & Tutorials 25.2 (2023): 1059-1094.

[28] D. Pan, G. Long, L. Yin, Y. Sheng, D. Ruan, S. X. Ng, J. Lu, and L. Hanzo "The evolution of quantum secure direct communication: on the road to the Qinternet." IEEE Communications Surveys & Tutorials (2024).