Quantum Machine Learning based Wind Turbine Condition Monitoring: State of the Art and Future Prospects

Zhefeng Zhang, Yueqi Wu, Xiandong Ma*

School of Engineering, Lancaster University, Lancaster, LA1 4YW, United Kingdom

Abstract

Wind energy, as a popular renewable resource, has gained extensive development and application in recent decades. Effective condition monitoring and fault diagnosis are crucial for ensuring the reliable operation of wind turbines. While conventional machine learning methods have been widely used in wind turbine condition monitoring, these approaches often face challenges such as complex feature extraction, limited model generalization, and high computational costs when dealing with large-scale, high-dimensional, and complex datasets. The emergence of quantum computing has opened up a new paradigm of machine learning algorithms. Quantum machine learning combines the advantages of quantum computing and machine learning, with the potential to surpass classical computational capabilities. This paper firstly reviews applications and limitations of the state-of-the-art machine learning-based condition monitoring techniques for wind turbines. It then reviews the fundamentals of quantum computing, quantum machine learning algorithms and their applications, covering quantum-based feature extraction, classification and regression for fault detection and the use of quantum neural networks for predictive maintenance. Through comparison, it is observed that quantum machine learning methods, even without extensive optimization, can achieve accuracy levels comparable to those of optimized conventional machine learning approaches. The challenges of applying quantum machine learning are also addressed, along with the future research and development prospects. The objective of this review is to fill a gap in the published literature by providing a new paradigm approach for wind turbine condition monitoring. By promoting quantum machine learning in this field, the reliability and efficiency of wind power systems are ultimately sought to be enhanced.

Keywords: Condition Monitoring (CM), Wind Turbine (WT), Machine Learning (ML), Deep Learning (DL), Quantum Machine Learning (QML), Fault Detection, Fault Diagnosis, Fault Prognosis

Nomenclature

ALWM-ResNet Loss-Weighted Meta-Residual Network

ANFIS Adaptive Neuro-Fuzzy Inference System

CBAN Convolutional Block Attention Module

CG-CNN Correlation-Graph-CNN

CWT Continuous Wavelet Transform

DAN Deep Adaptive Networks

DeepFedWT Federated Deep Learning Framework

*Corresponding author

Email address: `xiandong.ma@lancaster.ac.uk` (Xiandong Ma)

DRDN Deep Residual Deformable Network

DTL Deep Transfer Learning

GAT Graph Attention Network

GL Graph Learning

HA Hybrid Attention

IPCA Incremental Principal Component Analysis

LMMD Local Maximum Mean Discrepancy

MAML Model-Agnostic Meta-Learning

MC Multi-Channel

MCA Multi-Channel Attention

MDA Mixture Discriminant Analysis

MIP-YOLO Multivariate Information Perception You Look Only Once

MSCNN Multi-Scale Convolutional Neural Network

MSRAN Multi-Scale Residual Attention Network

MSTFAN Multidirectional Spatial-Temporal Feature Aggregation Networks

PCC Pearson Correlation Coefficient

SETCN-MVC Spectrum-Embedded Temporal Convolutional Network Multivariate Coefficient of Variation

SMOTE Synthetic Minority Oversampling Technique

STAGNN Spatial–Temporal Autocorrelation Graph Neural Network

WPT Wavelet Packet Transformation

1. Introduction

Wind energy, as a clean and renewable source, has gained widespread attention globally due to its environmental and economic benefits. According to the Global Wind Energy Council, by the end of 2023, the global installed wind power capacity exceeded 906 GW, showing a significant growth compared to the past decades [1]. Figure 1 shows a continuing trend of growth in installed wind power capacity. As the key part in wind energy conversion and utilization, wind turbines (WTs) play a crucial role in ensuring the stability and efficiency of wind energy systems. Their stable and reliable operation not only supports a continuous electricity supply but also significantly impacts operation and maintenance (O&M) costs, as well as the lifespan of the equipment [2]. However, maintaining reliable operation of the WTs presents substantial challenges due to the harsh environmental conditions where they operate. Offshore WTs are particularly susceptible to high salinity, strong winds, humidity, and large waves, leading to accelerated equipment wear and increased maintenance demands [3, 4]. Onshore turbines, though not exposed to marine conditions, still experience high wind speeds, temperature fluctuations, sandstorms, and lightning strikes, all of which contribute to mechanical degradation and operational instability [5, 6]. These environmental factors complicate O&M activities, necessitating the use of effective condition monitoring (CM) systems to ensure turbine reliability and minimize unplanned downtimes [7].

The development of WT CM dates back to the earlier maintenance strategies, initially relying on periodic manual inspections and time-based preventive maintenance strategies. These methods are not only time-consuming and

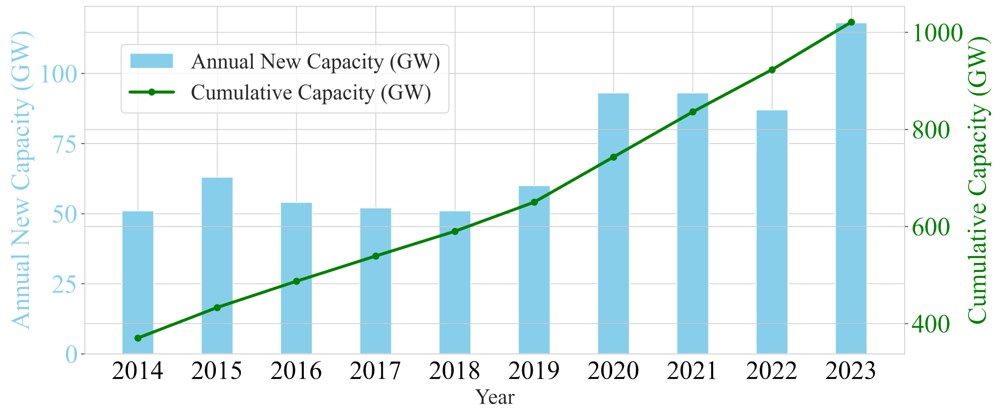

Figure 1: Cumulative wind power installations and yearly new installations [1]

labor-intensive but also often fail to capture the early fault signals in WTs, leading to high maintenance costs and reduced overall operational efficiency [8]. WT CM has gradually adopted more advanced techniques, including vibration analysis, acoustic monitoring, and oil analysis. These methods monitor physical changes in key turbine components, enabling a more accurate detection of mechanical faults. However, these traditional methods have limitations in handling complex data and predicting faults [9]. The development of sensor technology and data processing techniques has significantly enhanced the efficiency of WT CM. Sensors can collect real-time data on turbine operating conditions, including vibration, temperature, and pressure parameters, while data analysis can reveal potential fault risks earlier. Nevertheless, these methods still largely depend on expert knowledge and predefined fault models, making it challenging to address the complex and variable operating environments of WTs [10].

In recent years, machine learning (ML), a subset of artificial intelligence (AI), has emerged as a prominent research focus and mainstream direction in WT CM [11]. ML algorithms can automatically extract features, identify fault patterns, and predict fault trends from large volumes of monitoring data. Compared to traditional methods in the time and frequency analysis domains, ML offers significant advantages. Firstly, ML algorithms can process vast amounts of historical operational data, achieving fault detection, diagnosis, and prognosis through model training, thus reducing reliance on expert knowledge. Secondly, the automatic analysis and processing of real-time monitoring data enable early fault warnings, thus enhancing monitoring accuracy and reliability. Furthermore, ML models can be continuously optimized with data updates, improving their ability to recognize new fault patterns [12, 13].

Despite the successes of ML methods, conventional ML techniques face several challenges when applied to WT CM. One of the primary challenges is the complexity of feature extraction from high-dimensional and heterogeneous data [14]. WTs generate massive amounts of data from various sensors, and extracting meaningful features from data requires sophisticated algorithms and substantial computational resources. Moreover, the generalization capability of conventional ML models can be limited, particularly when dealing with unseen data, which refers to fault conditions or new operating environments that are not part of the training set and have not been encountered during model training [15]. Additionally, the high computational cost associated with training and deploying these models on large datasets poses a significant barrier to their widespread adoption [16]. The conventional ML methods appear to have reached a saturation point, with recent research focusing on stacking complex network structures but yielding very little performance gain.

On other hand, Quantum Machine Learning (QML), as an emerging technology that combines quantum computing and ML, has the potential to address these issues. Currently, QML algorithms, such as Quantum Support Vector Machine (QSVM), Quantum Boltzmann Machine (QBM) and Quantum Neural Network (QNN) have demonstrated a superior performance across various domains. For instance, in the financial sector, QSVM and QBM have outperformed conventional ML in tasks such as fraud detection and risk prediction [17]. Similarly, in the field of chemistry, QNNs have shown remarkable results in molecular structure exploration and drug synthesis pathway optimization [18]. The challenges in WT CM, such as processing high-dimensional data, align closely with those addressed by QML in fields like finance and chemistry. The success of QML in these areas provides valuable insights into potential applications of QML in WT CM.

Despite extensive research, no recent reviews provide a comprehensive analysis of advancements in WT CM, particularly with the role and potential of QML. Although [19] has attempted a broad review of QML applications in renewable energy, its focus deviates toward materials and chemistry. Additionally, other reviews in the energy sector primarily emphasize QML-based control and optimization, further highlighting the need for a more balanced and comprehensive evaluation. To address these gaps, our study systematically examines the transition from conventional ML-based methods to QML-based approaches to WT CM. The study presents the relevant QML frameworks applicable to WT CM and discusses the advantages of QML over the conventional ML. We also analyze key challenges in adopting QML for WT CM, including hardware constraints, algorithm scalability, and real-world implementation, and propose actionable recommendations for future research. In this manner, the main contributions of our study are:

- The first comprehensive review that bridges the gap between conventional ML techniques and QML-based methodologies in WT CM.
- Categories and comparisons of different QML algorithms applicable to WT CM, highlighting their strengths and limitations.
- Critical evaluation of the feasibility of QML adoption in real-world wind energy applications by considering trade-offs between quantum hardware constraints and algorithmic performance.
- A research roadmap identifying key challenges and future research for WT CM through QML, offering valuable insights which may lead to a paradigm shift approach to support WT CM activities.

Figure 2a shows the source distribution of the nearly 200 papers selected for this review, while Figure 2b shows the percentage of JCR (Journal Citation Reports) partitions for the selected articles. The JCR divides journals into four quartiles, Q1, Q2, Q3, and Q4, based on their impact factor rankings within their respective disciplines, with proportions of the top 25%, 26%-50%, 51%-75%, and 76%-100%, to evaluate their relative academic influence and quality.

Section 2 discusses conventional ML methods applied to WT CM, highlighting their applications and limitations. Section 3 presents the fundamentals of quantum computing and QML, explaining the basic principles and algorithms. Section 3 also explores the potential applications of QML in WT CM, including quantum-based feature extraction, classification, and predictive maintenance. Section 4 addresses the challenges and future development in applying QML to WT CM. Finally, Section 5 concludes the review by summarizing the key findings and contributions. Through this analysis, a comprehensive understanding of the current landscape and future trends in WT CM is provided, particularly concerning the development and application of QML to enhance the reliability and efficiency of wind energy systems.

2. Conventional Machine Learning-based Condition Monitoring

Based on a thorough review of recent literature, it is evident that ML has become the mainstream approach for WT CM. Figure 3 shows the results of a literature search on Scopus, indicating that ML-related methods have become increasingly prevalent in recent years, compared to the statistics-based CM. The data were retrieved and refined through searches using keyword combinations, including "wind turbine," "condition monitoring," "machine learning," "deep learning," "statistical," "fault detection," "fault diagnosis," and "prognosis". This shift is largely attributed to the ability of ML to handle vast amounts of data, adapt to changing operational conditions, and provide accurate and timely predictions of potential failures. The primary benefits of ML-based CM include higher accuracy in fault detection, reduced false alarms, and the capability to analyze complex data patterns from multiple sensors in real-time, thus enhancing maintenance scheduling and minimizing downtime. Numerous recent studies have demonstrated these advantages, showcasing the effectiveness of ML in improving the reliability and efficiency of WT operations. Ref [20] categorized ML-based WT CM techniques based on components and subsystems, while [21] reviewed WT CM methods from the perspective of data sources and employed techniques. Additionally, [22] focused on the development roadmap of explainable WT CM, highlighting advancements in interpretability.

Conventional ML algorithms can be categorized into classical ML and deep learning (DL) methods. In WT CM, commonly used classical ML approaches include support vector machine (SVM) [23], decision tree (DT) [24],

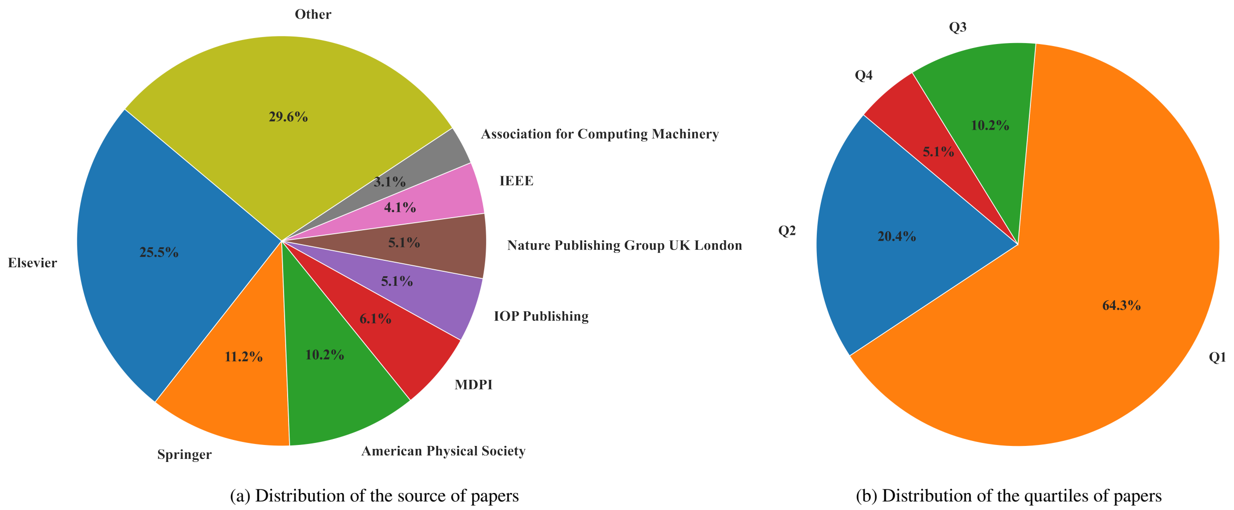

Figure 2: Distribution of selected papers

random forest (RF) [25], extreme gradient boosting (XGBoost) [26], and k-nearest neighbor (KNN) [27]. SVM algorithm utilizes kernel functions to map data into higher-dimensional spaces, enabling the construction of an optimal hyperplane for classification tasks. This makes them effective in detecting faults, where distinguishing between normal and faulty conditions requires clear decision boundaries. However, their performance depends on kernel selection and is computationally expensive when handling large datasets. DT algorithm classifies data by recursively partitioning the feature space using threshold-based splitting rules, offering an interpretable model structure for identifying failure conditions in the WT components. RF algorithm, an ensemble learning method, enhances classification stability by aggregating multiple DTs, reducing the likelihood of overfitting and improving model robustness against sensor noise. However, DT tends to overfit smaller datasets, while RF requires significant memory and computational resources as dataset size increases. XGBoost algorithm refines weak classifiers iteratively by adjusting model weights based on previous errors, making it particularly effective for health assessment tasks. However, tuning hyperparameters such as learning rate and tree depth is essential to prevent overfitting and ensure the model performance. KNN algorithm, a distance-based classifier, determines the categories of samples based on the majority label among its closest training examples. It is often applied in real-time aerodynamic monitoring of turbine blades, where the aerodynamic response of different blade positions can be clustered into distinct operational states. However, its computational cost increases significantly with dataset size, and its performance is sensitive to feature scaling. While these algorithms remain effective for small datasets, their generalization capabilities diminish as data scales increase. Recent studies therefore focus on enhancing feature engineering and refining threshold calculation [28], and the present review does not delve deeply into these classical approaches.

DL models include convolutional neural network (CNN) [29], recurrent neural network (RNN) [30], autoencoder (AE) [31], generative adversarial network (GAN) [32], and Transformer [33]. CNN extracts local features through convolution operations and progressively construct higher-level abstractions in deeper layers, enabling effective pattern recognition in WT operating states. Variants of CNN can capture both spatial and temporal features in data, improving fault detection accuracy. RNN processes sequential data by maintaining historical information through recurrent connections, however, standard RNN suffers from the vanishing gradient problem when learning long sequences, limiting its ability to retain long-range dependencies. Its variants such as long short-term memory (LSTM) network and gated recurrent unit (GRU) introduce gating mechanism to regulate information storage and forgetting, thus enhancing the ability to model long-term dependencies and improving performance trend forecasting in WT CM [34]. AE learns compact representations by encoding input data into a lower-dimensional space and reconstructing it through a decoder, where the reconstruction error serves as an indicator of deviations. This makes AE suitable for unsupervised fault detection, aiding in identifying anomalies in sensor data. GAN consists of a generator that learns the data distribution to produce samples similar to real data and a discriminator that distinguishes between real

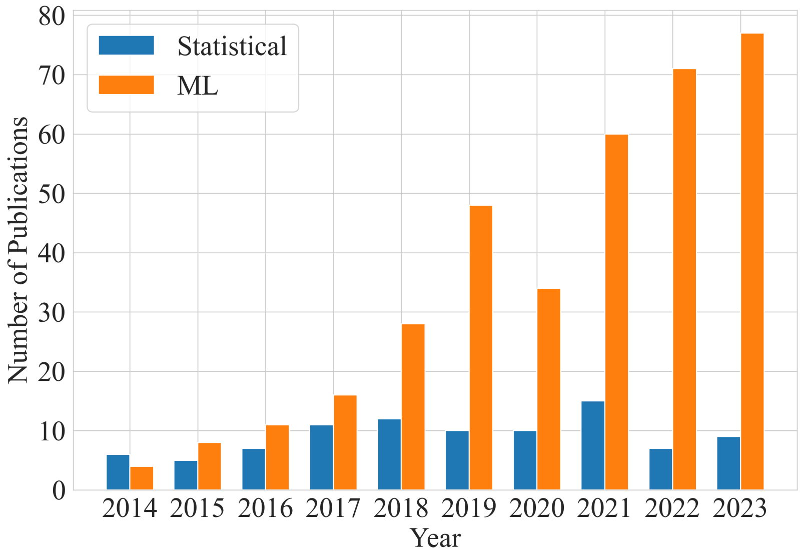

Figure 3: Yearly WT CM papers distributions

and synthetic data. The adversarial training mechanism enables data augmentation, thereby improving fault detection when failure samples are limited. Transformer models use a self-attention mechanism to capture dependencies across all elements in an input sequence, allowing it to model long-range relationships more effectively than RNN.

2.1. Machine Learning-based Condition Monitoring

WT CM tasks can be divided into fault detection, fault diagnosis, and fault prognosis. Fault detection identifies whether a WT has any anomalies or faults [35]. Fault diagnosis determines the type, location, and severity of faults[36]. Fault prognosis forecasts future fault trends or estimates the remaining useful life (RUL) of WTs [37]. The design and implementation of WT CM needs to be based on the domain knowledge of WT, especially the fault modes of WT. A WT consists of multiple key components and subsystems, each playing an important role. The blades capture wind energy and convert it into rotational mechanical energy, which drives the rotor. Common blade faults, including blade cracks, surface erosion, and imbalance, can lead to blade failure, and compromise the safety and performance of WTs [38]. The main shaft connects the hub, which links the blades to the rotor and to the gearbox. Faults in the main shaft such as fatigue, wear, poor lubrication, contamination, installation errors, or overload, can result in costly shutdowns and secondary damage to the gearbox and generator [39]. The main shaft bearing supports the shaft and reduces rotational friction. Failures such as insufficient lubrication, fatigue spalling, and corrosion can cause shaft instability, accelerated wear, or necessitate a full bearing replacement [40]. The gearbox converts low-speed, high-torque energy from the rotor into high-speed, low-torque energy to drive the generator. Gear wear, bearing failure and oil leakage are common faults that reduce transmission efficiency, increase mechanical vibrations, and cause further damage to connected components [41]. The generator transforms mechanical energy into electrical energy. Faults such as rotor damage, bearing wear, winding issues, cooling failures, and short circuits can lower power output or cause a complete turbine shutdown [42]. The power converter stabilizes the electrical output from the turbine, ensuring grid compatibility by regulating voltage, current, and frequency. Common faults, including power module failures, cooling issues, control circuit problems, and sensor malfunctions can lead to downtime, reduced power quality, and increased maintenance costs [43]. The yaw system adjusts the direction of WTs to align the blades with the wind. Common faults include mechanical wear, poor lubrication, sensor failures, and servo motor issues. Extreme weather conditions, such as icing, strong gusts, or lightning, can worsen wear, damage components, and disrupt normal operation [44]. The pitch system adjusts blade angles to optimize efficiency and protect against strong winds, using hydraulic or electric actuators. Faults in the pitch system such as wear, poor lubrication, and control failures can reduce power generation and damage components [45]. The common faults in the key components and subsystems and their proportions are shown in Table 1 and Figure 4, respectively.

Table 1: Common faults in WTs and their causes

Component	**Common Faults**	**Causes**
Rotor Blades	Blade Cracks	Fatigue due to cyclic loads, extreme weather (hail, lightning).
	Erosion	Exposure to high-speed particles (rain, sand).
	Imbalance	Accumulated dirt/ice or manufacturing defects.
Gearbox	Gear Wear	Insufficient lubrication, misalignment, or overloading.
	Bearing Failure	Contamination, overloading, or high temperatures.
	Oil Leakage	Seal degradation or improper maintenance.
Generator	Electrical Faults (short circuits, open circuits)	Insulation breakdown, wear of windings, or voltage surges.
	Overheating	Inefficient cooling or excessive load.
Control System	Communication Failures	Faulty sensors, wiring issues, or software errors.
	Pitch System Malfunctions	Hydraulic or electric actuator failures, sensor errors, or extreme conditions.
Tower	Structural Fatigue	Resonance or constant cyclic loading from wind forces.
	Corrosion	Exposure to saline or humid environments.
Yaw System	Motor Failure	Overloading, wear and tear of motors/gears.
	Misalignment	Faulty sensors or mechanical failure in yaw mechanism.
Electrical System	Cable Damage	Over-voltage, thermal stresses, or abrasion from vibration.
	Converter Failures	Thermal stress, voltage fluctuations, or poor-quality components.
Hydraulic System	Leakage	Seal degradation or wear of hydraulic lines.

Figure 5 illustrates the key steps involved in WT CM. The process begins with data sensing and acquisition, where sensors such as vibration sensors, torque sensors, acoustic emission sensors, oil debris sensors, thermal imaging cameras, and strain gauges are installed either intrusively or non-intrusively on WTs to collect high-sampling-rate data [47–51]. Additionally, data can be collected through the Supervisory Control and Data Acquisition (SCADA) system, which captures environmental conditions, electrical characteristics, controller data, and various temperature readings. Unlike sensor data, SCADA data typically has a lower sampling rate and consists of averaged sensor readings over fixed time intervals [47]. Once raw data is acquired, data preprocessing is essential, involving steps such as data cleaning, feature engineering, and data augmentation [52]. Data cleaning addresses issues such as missing values, outliers, noise, and dirty data [53]. Feature engineering techniques such as feature selection, feature extraction, feature reduction, data transform, normalization and standardization can shorten the model training time and improve the model performance [54]. Data augmentation can solve the issues such as over-sampling, under-sampling, and data imbalance that are common in WT data and improve model performance through techniques including sliding window, random noise injection [55]. Ref [56] provides a summary of data cleaning techniques. A detailed introduction to common feature engineering techniques can be found in [57], while additional details on data augmentation techniques are available in [58]. Fault detection models can be categorized into regression-based anomaly detection and labeled data-based binary classification, depending on the type of ML task [35]. Fault diagnosis models use multi-class classification to analyze labeled data, applying pattern recognition techniques to identify fault patterns [36]. Fault prognosis models use regression or time-series analysis to predict the future faults or to estimate RUL of WTs [37]. The performance of WT CM models can be evaluated by metrics such as accuracy (Acc), precision (Prec), recall (Rec), F1 score (F1), area under the curve (AUC), receiver operating characteristic (ROC), mean squared error (MSE), root mean squared error (RMSE), mean absolute error (MAE), R-squared (R^2 or R2), Matthews correlation coefficient (MCC), logarithmic loss (Log Loss), hinge loss (HL), Cohen's kappa (Kappa), normalized mutual information (NMI), adjusted Rand index (ARI), mean Average Precision (mAP) and envelope spectrum correlation kurtosis (ECK) [59].

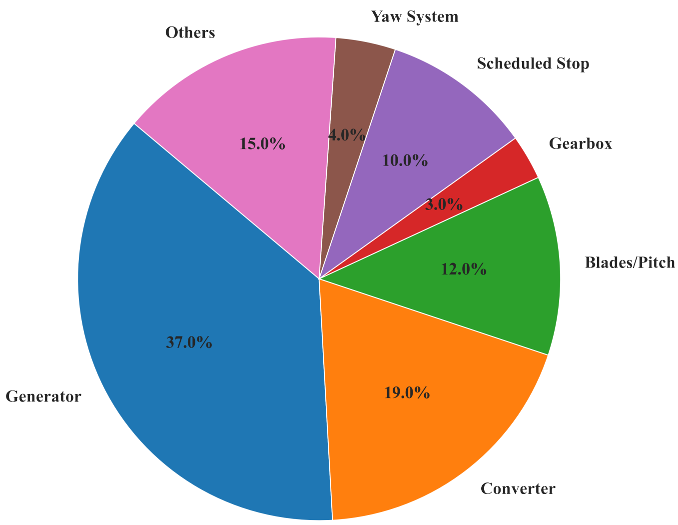

Figure 4: Common faults in WTs and their proportions [46]

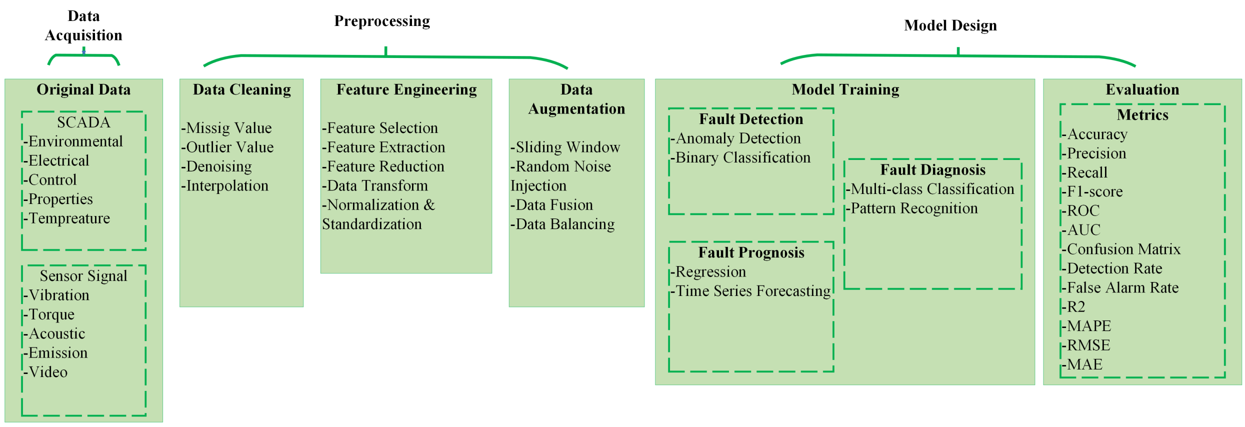

Figure 5: Process of ML-based WT CM

2.2. Machine Learning-Based Fault Detection

Recent studies in WT CM have predominantly focused on combining various CNN, RNN, transformer variants for compound fault detection tasks. Table 2 lists the cutting-edge research in fault detection. CNN has shown strong capabilities in extracting fault features, processing multi-channel and spatial-temporal data, and handling image analysis. In [73], a Multi-Channel CNN (MCNN) was applied to learn the data features of 3-Axis vibration sensor data under four states: normal state, blade angle anomaly, surface damage, and blade breakage, achieving an accuracy of 87.8%. In another study [74], a 3D squeeze-excitation CNN (3DSE-CNN) was used to detect compound fault. A sliding window was used to convert 1D SCADA data into 2D images; the attention mechanism in the SE module was utilized to extract spatial features from the images; and a 2DLSTM was employed to capture spatial-temporal fused features. This approach achieved an accuracy of 96.7% on a dataset containing compound faults such as feeder and excitation faults, as well as grid power and air-cooling faults. In [66], a 3D CNN was applied to hyperspectral imaging (HSI) data of WT blades, achieving 100% accuracy in detecting cracks, erosion, and ice accumulation. A deep residual network (DRN) was developed in [63], utilizing Convolutional Residual Building Blocks (CRBB) and Squeeze-and-Excitation (SE) units to enhance fault-related features in raw SCADA data. The DRN, requiring no preprocessing, achieved 99.68% accuracy in detecting compound faults containing both gearbox and generator faults.

Table 2: Fault Detection Related Articles

Ref	Year	Data	Component/Subsystem	Preprocessing	Model	Metrics
[60]	2021	SCADA	Gearbox Generator	Data cleaning Feature selection	CNN-LSTM-AM	RMSE: 1.005 MAE: 0.621 R^2: 0.988
[61]	2024	SCADA	Pitch system	Time series filling Z-score	Hierarchical STAGNN	F1: 91.83%
[62]	2024	SCADA	IGBT module	Data cleaning Z-score	Transformer	F1: 100%
[63]	2023	SCADA	Gearbox Generator	-	DRN	Acc: 99.68%
[64]	2022	SCADA	Main bearing Pitch system	Sliding window	MSTFAN	F1: 93.1% AUC: 98.1%
[65]	2022	SCADA	Gearbox	Resampling	LSTM-CNN-DAN	Acc: 91.21%
[66]	2024	Hyperspectral imaging	Blade	Noise removal IPCA	3D CNN	Acc: 100% Prec, Rec, F1: 100%
[67]	2021	SCADA	Gearbox Generator	Data cleaning Standardization	LSTM-KLD	Acc: 94% (gearbox) 92% (generator)
[68]	2022	SCADA	Blades Generator	Data normalization Downsampling	MSRAN DeepFedWT	Prec: 96.87% Rec: 99.87% F1: 0.984 AUC: 0.9998
[69]	2022	SCADA	Nacelle	Data cleaning Feature selection	SETCN-MCV	MAE: 0.014 MSE: 0.0007 R^2: 0.766
[70]	2023	SCADA	Generator	Data cleaning	1D-CNN-MAML	MAE: 1.374 RMSE: 1.692 R^2: 0.927 Fault detection time: 136.7 hours in advance
[71]	2024	Drone video	Blades	Data annotation	MIP-YOLO	Acc: 98.7% Detection mAP: 97.1% Segmentation mAP: 93.9%
[72]	2024	SCADA	Pitch bearing	Data cleaning	BATCN	Acc: 95.4%

WT operational data consist of time-series data, which makes algorithms such as RNN variants and Temporal Convolutional Network (TCN) particularly suitable for WT CM. In [67], LSTM was used for normal behavior modeling (NBM), with Kullback-Leibler divergence (KLD) being employed to detect faults by comparing probability distributions of normal and current states. The LSTM-KLD method achieved 94% and 92% accuracy in detecting gearbox and generator faults. In [72], a Bayesian-augmented TCN (BATCN) was designed to detect pitch bearing faults. The model used Bayesian augmentation to realize signal filtering, achieving automatically finding of the best patch length that influences fault signal extraction. TCN was used to capture temporal dependencies in non-stationary data, achieving better detection of pitch bearing failures than other methods. In [60], CNN was applied to extract the features and attention mechanisms (AM) was then utilized to re-allocate the weights of features in LSTM to improve fitting performance, reducing RMSE to 0.875 and achieving a 0.988 R2 on a dataset containing gearbox gear breakage fault. In [64], a Graph Attention Network (GAT) was applied to SCADA data to model spatial-temporal relationships, achieving an F1 score of 0.931 and an AUC of 0.981. This method outperformed other models like TCN and CNN-LSTM. Additionally, the Transformer-based methods have shown satisfactory performance under different feature selection algorithms, especially F1 and Rec scores [62].

Recent research on fault detection has primarily focused on utilizing SCADA data as the core data source, supplemented by HSI, drone-based video analysis, and multi-sensor fusion technologies to enhance the monitoring accuracy of critical components such as gearboxes, generators, pitch systems, and blades. With advancements in computational resources and the emergence of high-performance DL models, the research focus has shifted from complex data preprocessing to optimizing model architectures. Recent work has also prioritized to enhance the model architectures and feature extraction capabilities.

In terms of DL models, CNN variants remain the dominant choice for spatial feature extraction, while RNN-based architectures continue to play a key role in modeling temporal dependencies. In recent years, there has been a clear shift toward optimizing spatial-temporal dependency modeling, with an increasing integration of multiple techniques to improve the fault detection accuracy. The attention mechanism has become widely adopted, while hybrid architectures combining CNN, LSTM, TCN, and Transformer models are gaining traction due to their advantages in compound fault detection. This reflects a growing emphasis on designing models that can better adapt to complex operating conditions and real-world scenarios.

Additionally, emerging paradigms such as self-supervised learning and federated learning are being explored to reduce reliance on large-scale labeled datasets and improve generalization across different wind turbine models and operational environments. Meanwhile, the integration of explainable AI (XAI) is becoming an important research, addressing the need for model interpretability and decision reliability in high-risk WT fault detection tasks.

2.3. Machine Learning-based Fault Diagnosis

In recent research on WT fault diagnosis, a variety of advanced techniques have been applied to address challenges such as data imbalance, limited operational data, and the need for effective feature extraction. Table 3 contains a list of articles related to ML in WT fault diagnosis. WT operational data are predominantly normal, with only a small portion labeled as abnormal, creating a significant data imbalance for fault diagnosis models. GANs address this issue by generating sufficient fault data, balancing the dataset and improving model training. In [89], GANs were used to align faulty and normal data distributions, enhancing feature extraction and improving generalization on unseen data.

Table 3: Fault Diagnosis Related Articles

Ref	Year	Data	Component/Subsystem	Preprocessing	Model	Metrics
[75]	2021	SCADA	Bearings Gearbox Generator Rotor Blades	Time-domain to images	MC-CNN	Acc: 99.85%
[76]	2023	Vibration	Bearings	CWT	DRDSAN	Acc: 99.94%
[77]	2022	Vibration	Bearings	Weighted majority voting	MSCNN-BiLSTM	F1: 97.12%
[78]	2022	Vibration	Bearings gears	Signal compression	DTL-CNN	Acc: 97.73%
[79]	2019	SCADA vibration	Wind wheel Bearing	-	LSTM	99.8%
[80]	2023	Accelerometer sensor Acoustic emission sensor	Bearings	CWT	Coupled CNN	Acc: 98.18%
[81]	2020	Vibration	Generator bearings	-	Concurrent CNN	Acc: >98.5%
[82]	2023	Vibration	Drivetrain	CWT CBAM	Residual CNN-CBAM	Acc: >90%
[83]	2023	Vibration	Drivetrain	PCC Kalman filter	CGCNN	Acc: 96.7%
[84]	2023	SCADA	Pitch system	Data cleaning	MCA-LSTM	Acc: 98.1%
[85]	2021	Vibration	Gearbox	WPT	HA-ResNet	Acc: 98.79%
[86]	2021	Vibration	Gearbox	WPT	ALWM-ResNet	Acc: 97.1%
[87]	2021	SCADA	Drivetrain	SMOTE	CAE-TL	Acc: 92.5%
[88]	2021	SCADA	Drivetrain	ReliefF PCA	DNN	Acc: >97.82%

With only 30 labeled samples, the model achieved over 90% accuracy in diagnosing various bearing and alignment faults.

Many newly installed WTs face the challenge of insufficient operational data, making fault diagnosis difficult. In [87], a fault diagnosis method based on Transfer Learning (TL) and Convolutional Autoencoder (CAE) was proposed for small-scale datasets. The method used parameter-based TL, where a model trained on similar WT operational data (source domain) assisted in diagnosing faults in WTs with limited data (target domain). By sharing lower-level neural network parameters, TL facilitated the extraction of general fault features. CAE was employed for feature extraction, with its low-dimensional output used for fault classification. The model was trained on the data contains 7 kind of faults including power converter system, low yawing velocity, high temperature of gearbox, high temperature of generator, low pressure of hydraulic system, emergency stop of cabin and emergency stop of tower, achieved over 90% accuracy, outperforming traditional methods and AEs without TL.

The use of LSTM variants for classifying time-series data from WT sensors has proven highly effective. In [79], a wind tunnel platform was used to simulate 11 types of WT faults, including rotor, bearing, and shaft faults. The LSTM model achieved a classification accuracy of 87.58% on multi-sensor data, outperforming methods such as SVM, MLP, and CNN. In [84], an LSTM model enhanced with a multi-channel attention mechanism (MCA-LSTM) was applied for WT pitch system fault diagnosis. The attention mechanism fused time-series data across different channels, enabling key feature extraction at various scales. This model outperformed Support Vector Regression (SVR) and conventional LSTM in terms of RMSE and other metrics.

CNNs are highly effective for classification tasks. In [76], a deep residual deformable subdomain adaptation network (DRDSAN) was proposed, replacing standard convolutions with deformable modules to enhance feature extraction. This adaptation allowed the network to adjust convolutional kernels based on input features, significantly improving feature representation. DRDSAN achieved a diagnostic accuracy of over 99.86% and an F1 score of 1 in diagnosing WT bearing faults, demonstrating excellent performance. Similarly, transforming time-series data into images and applying CNN for feature learning is a powerful approach for fault diagnosis. In [75], a multi-channel CNN (MC-CNN) was used to process data from various WT sensors, converting time-series data into images for analysis. This method achieved 99.85% accuracy in classifying seven types of faults, automatically extracting effective features without human intervention, making it well-suited for real-time monitoring and diagnosis.

Building on this, combining LSTM with CNN, attention mechanisms (AM), and other techniques can further enhance performance. In [77], a multi-scale CNN (MSCNN) was used to extract multi-scale features from raw vibration signals, while Bi-LSTM captured semantic correlations. A weighted majority voting fusion mechanism, optimized via genetic algorithms, improved diagnostic performance, achieving an average F1 score of 97.12%. This method demonstrated strong fault diagnosis capabilities, particularly in noisy environments, due to its noise resistance and multi-sensor fusion.

Current research on fault diagnosis primarily focuses on precisely identifying fault types, addressing data imbalance, and improving small-sample learning capabilities. In terms of data sources, vibration signals, acoustic emission signals, and accelerometer data remain the primary inputs for fault diagnosis. Additionally, SCADA data, HSI, acoustic monitoring, and ultrasonic inspection are increasingly employed to enhance diagnostic accuracy for gearboxes, generators, bearings, and pitch systems.

From a methodological perspective, frequency-domain DL methods (e.g., wavelet transform combined with CNNs), TL, and GANs have been widely adopted to address challenges related to insufficient labeled data, data distribution shifts, and small-sample learning. Specifically, TL facilitates faster fault identification for newly deployed wind turbines, while GANs are used to generate synthetic fault samples, mitigating data imbalance and enhancing model robustness. Additionally, attention mechanisms, which have been extensively employed in fault detection, are increasingly applied to fault diagnosis, significantly improving feature extraction for complex fault patterns.

An emerging research focus is multi-sensor fusion, where integrating heterogeneous sensor data sources can improves diagnostic comprehensiveness and accuracy. Furthermore, XAI is also gaining prominence in fault diagnosis, aiming to enhance model transparency and decision reliability, particularly for high-stakes fault classification tasks. These trends indicate that future research will increasingly prioritize high-precision fault diagnosis, multimodal data fusion, and enhanced interpretability, aligning with the growing operational and maintenance requirements of the wind energy sector.

2.4. Machine Learning-based Fault Prognosis

Recent research on fault prognosis based on ML is relatively limited, and the performance of ML methods has not been as impressive as in the fields of detection and diagnosis. More studies have relied on statistical methods, which have yielded better results. Table 4 contains a list of articles related to ML in WT fault prognosis.

Table 4: Fault Prognosis Related Articles

Ref	Year	Data	Component/Subsystem	Preprocessing	Model	Horizon	Metrics
[90]	2017	Vibration	High-speed shaft bearings	Spectral kurtosis	SVR	-	Close to the actual RUL
[91]	2023	Vibration	High-speed shaft bearings	Degraded feature fusion	Self-constraint state-space estimator	-	RUL Predicted Error: 0.4823
[92]	2022	Stress	Blades	Zonotopic Kalman filter	Zonotopic Kalman filter	-	Close to the actual RUL
[93]	2020	Vibration	Blades	Histogram features	Lazy classifiers	-	Acc: 93.83% MAE: 0.0423 RMSE: 0.1344
[94]	2013	SCADA	Pitch system	Signal extraction	ANFIS	21 days	Acc: 88.3%
[95]	2018	-	Drivetrain	Sensor layout Euclidean distance calculation	Physical model Data clustering	65 hours	Acc: >0.7
[96]	2020	Vibration	Bearings	Wavelet transform	Bayesian framework Particle filter	-	Close to the actual RUL
[97]	2020	Vibration	Gearbox	Signal intensity estimator PCA	SVM ANN GP MDA DT	177 hours	Minimal RUL prediction error in GP model
[98]	2022	Vibration	Shaft bearings	Spectral shape factor	Elman neural network	35 days	RMSE: 1.6514e05
[99]	2023	SCADA Accelerometer data	Tower-transition piece interface	Data cleaning	Physics-guided ML ANN	9 months	Cumulative fatigue prediction error <3%
[100]	2023	SCADA	Drivetrain	Cosine similarity measure	GAT GL	Short-term	RMSE: 18.32-28.49 MAE: 6.36-11.49 R^2: 0.71-0.73
[101]	2024	IoT	Drivetrain	Data cleaning	PM-C-LSTM	-	Acc: 96.77% Prec: 98.98% F1: 86.29

In RUL prediction, statistical methods are often combined with ML techniques. In [95], both time-domain and frequency-domain features were extracted from the vibration data from the WT transmission system. Spearman rank correlation and hierarchical clustering were used for feature selection, and PCA was applied for dimensionality reduction. The principal components with the highest variance were used as a fused feature and health indicator. An exponential degradation model estimated the RUL, with a T-test detecting the first prediction. Results showed a high precision, particularly in later predictions, with a 95% confidence interval. In [98], vibration data from the high-speed shaft bearing (HSSB) were used to train an Elman Neural Network (ENN) for RUL prediction. The teager energy operator (TEO) and short-time Fourier transform (STFT) were applied to enhance impulsive components and calculate the spectral shape factor (SSF). The ENN outperformed other methods, achieving an RMSE of 0.0025. In [97], various models, including SVM, ANN, Mixture Discriminant Analysis (MDA), and Gaussian Process (GP), were trained on WT gear vibration data. The GP achieved the lowest prediction error, providing RUL estimates closest to actual values.

To predict potential failures, ML methods are often combined with statistical, physics-based, and signal processing techniques. In [94], an Adaptive Neuro-Fuzzy Inference System (ANFIS) was proposed for the fault prognosis of the WT pitch system, integrating ANN and Fuzzy Inference Systems (FIS) with a six-fault pitch feature database to enhance generalization. Tested on 26 WTs, the model predicted pitch faults 21 days in advance, achieving 85.9% accuracy, 62.2% recall, and 94.4% precision, significantly outperforming the SCADA alarm system. In [90], a SVR model was trained on vibration data from the high-speed shaft bearing (HSSB) to predict RUL. Spectral Kurtosis (SK) features, such as kurtosis, skewness, and peak value, were extracted to monitor early stage faults, showing better trendability and monotonicity compared to time-domain features. In [101], a Predictive Maintenance Convolutional LSTM (PM-C-LSTM) framework was introduced, combining CNNs for spatial feature extraction with LSTM networks for sequential data analysis. The model achieved an accuracy of 96.77% on data collected over 16 months from a wind farm, demonstrating its superior performance in predicting turbine failures compared to traditional models. For a more comprehensive review of prognosis techniques, [102] examined the current state and development trends of prognosis methods for low-speed bearings and gearboxes. Ref [103] focused on control strategies aimed at achieving the expected service life. Ref [104] reviewed recent advancements in health state management, while [105] provided a comprehensive summary of progress in structural health management.

Current research on fault prognosis remains relatively limited compared to fault detection and diagnosis, with statistical methods often outperforming ML approaches in RUL prediction and failure trend analysis. In terms of data usage, vibration signals remain the primary data source, particularly for drivetrain components such as bearings and gearboxes. Meanwhile, SCADA data, stress measurements, and physics-informed features are increasingly incorporated to enhance the accuracy of long-term degradation modeling. Compared to detection and diagnosis, prognosis focuses on predictive modeling rather than classification, imposing higher demands on the completeness and consistency of historical degradation data.

From a technical perspective, SVR, ANN, GP, and LSTM are the most commonly used ML methods, often combined with signal processing techniques and physics-based models to improve prediction reliability. In recent years, hybrid frameworks have demonstrated strong performance in fault prognosis, such as CNN-LSTM based spatiotemporal feature extraction models, ANFIS, and GAT, particularly in extracting spatial-temporal degradation patterns from WT operational data. A key research trend in fault prognosis is the shift toward "physics-guided machine learning", where data-driven models incorporate engineering constraints to improve generalization and interpretability.

Overall, current research on WT CM is evolving from single-source data analysis to multi-modal data fusion, from single-task learning to multi-task collaborative modeling, and from ML methods to more efficient and interpretable intelligent algorithms. However, as models become increasingly complex, their performance improvements have begun to plateau, indicating diminishing returns from merely increasing model depth or computational capacity. This limitation highlights the need for novel approaches to enhance model performance, efficiency, and robustness in WT CM. The boundaries between detection and diagnosis are becoming increasingly blurred, as researchers explore end-to-end WT CM frameworks to enable early anomaly detection, precise fault diagnosis, and reliable RUL through a fully integrated intelligent monitoring system. In the future, key research in WT CM will include physics-informed ML, self-supervised learning, TL and XAI, as well as edge computing and distributed intelligent maintenance, addressing the challenges of data heterogeneity and high-reliability O&M activities due to the large-scale wind farm deployment.

3. Quantum Machine Learning-based Condition Monitoring

QML is an emerging interdisciplinary field that combines the strengths of quantum computing and ML [106]. Quantum computing, based on fundamental principles of quantum mechanics such as superposition, entanglement, and interference, introduces a novel computational paradigm with significant parallel processing capabilities [107]. The foundational work in [108] outlined the theoretical potential of quantum computing, highlighting its anticipated advantages. Recent advancements, as reviewed in [109], have validated many of these early projections, demonstrating the tangible progress in quantum algorithms and hardware performance. Concurrently, ML, as a cornerstone of artificial intelligence, has achieved remarkable successes in areas such as image recognition, natural language processing (NLP), and predictive analytics by learning and discovering patterns from data [110]. QML aims to harness the formidable computational power of quantum computing to enhance ML tasks, with the expectation that quantum algorithms will significantly improve the performance and efficiency of ML models [111]. By merging the advance of ML and quantum computing, QML is poised to demonstrate unique advantages in WT CM.

3.1. Quantum Computing

In classical computer, the fundamental unit of information is the bit, which can exist in one of two states: 0 or 1 [112]. Classical computers perform data operations using different logic gate circuits such as AND, OR, and XOR, which manipulate these bits. A myriad of such circuits stacked together ultimately forms the modern general-purpose classical computer. In contrast, quantum computing uses quantum bits, or qubits [113]. Unlike classical bits, qubits can exist simultaneously in a superposition of states. This means a qubit can be in a state $|0\rangle$, a state $|1\rangle$, or any quantum superposition of these states [114]. Mathematically, the state of a qubit can be represented as equation (1), where α and β are complex numbers that satisfy the normalization condition equation (2). This property enables quantum computers to process a vast amount of information simultaneously, offering potential exponential speedups for certain computations. Similarly, in quantum computing, various logic gates such as the CNOT gate and the Pauli-X gate are employed to manipulate qubits. These quantum logic gates function in a manner analogous to classical logic gates in classical computers. By stacking and combining multiple quantum logic gates, quantum circuits can be constructed to perform specific quantum computing tasks, thereby enabling the execution of a complex quantum algorithm [115].

$$|\psi\rangle = \alpha |0\rangle + \beta |1\rangle \tag{1}$$

$$|\alpha|^2 + |\beta|^2 = 1 \tag{2}$$

To illustrate the principles of quantum computing, the exclusive OR (XOR) gate can be used as an example. In classical computing, the XOR gate outputs true if the inputs are different, and false if they are the same. In quantum computing, the functionality of the XOR gate can be implemented using the controlled-NOT (CNOT) gate. The CNOT gate takes two inputs: the control qubit and the target qubit, and its output is the updated state of the target qubit. When the control qubit is in the state $|1\rangle$, the target qubit flips its state. When the control qubit is in the state $|0\rangle$, the target qubit remains unchanged. Figure 6 shows the functional verification of the CNOT gate by testing it with different input states and comparing its outputs. The top section of each column depicts the quantum circuit. The bottom section illustrates the transformation of the ground state into the input state for the CNOT gate through quantum operations, followed by its transition into the output state. In the first column, the circuit contains two qubits, q_0 and q_1, and one classical bit, c. The quantum state of the quantum circuit is represented as $|q_1 q_0\rangle$. Both qubits are initialized to $|0\rangle$, as quantum computers typically prepare qubits in the $|0\rangle$ state. Therefore, the initial (ground) state of the circuit is $|00\rangle$. The blue plus symbol ('+') and the connected dot represent the CNOT gate, where q_0 acts as the control qubit and q_1 as the target qubit. Since the input state to the CNOT gate is $|00\rangle$ and the control qubit is in the $|0\rangle$ state, the target qubit remains unchanged, resulting in an output state of $|00\rangle$. The gray square in the circuit represents the measurement operation, which collapses the quantum state of the target qubit q_1 into classical bit c. In the second column, the blue box labeled 'X' represents the Pauli-X gate, which flips the state of the qubit it acts on. The circuit starts from the ground state $|00\rangle$, the Pauli-X gate is applied to q_1, flipping its state to $|1\rangle$. Thus, the input state to the CNOT gate becomes $|10\rangle$. As the control qubit is 0, the target qubit remains unchanged, resulting in an output state of $|10\rangle$. In the third column, the circuit also begins in the ground state $|00\rangle$. The Pauli-X gate is applied to q_0, flipping its state to $|1\rangle$, The resulting input state to the CNOT gate is $|01\rangle$. As the control qubit q_0 is now in the state $|1\rangle$, the target qubit flips its state, producing an output state of $|11\rangle$. In the fourth column, the circuit starts in the ground state $|00\rangle$. The Pauli-X gates are applied to both q_0 and q_1, flipping their states to $|1\rangle$. The input state to the CNOT gate is $|11\rangle$, resulting in an output state of $|01\rangle$. These results verify the implementation of the XOR operation by the CNOT gate. They demonstrate the capacity of quantum logic gates to replicate classical operations while utilizing quantum principles. This provides a foundation for the development of advanced quantum circuits and algorithms that exceed the limits of classical computing.

The continued breakthroughs in quantum algorithm research are making ML tasks increasingly feasible on quantum computers. Quantum Fourier Transform (QFT) [116] is a quantum analogue of the discrete Fourier transform, widely used in spectrum analysis for identifying frequency components in data, while Quantum Phase Estimation (QPE) [117] leverages QFT to extract eigenvalues, which are critical for feature extraction in quantum systems. The Shor algorithm [118], specifically designed for integer factorization and solving discrete logarithms, provides exponential speedups and has potential applications in certain optimization problems. The Grover algorithm [119] achieves quadratic speedups for unstructured search tasks, forming a foundational tool for optimization in QML. Variational algorithms are widely used in ML for probabilistic inference and optimization by approximating complex distributions or cost functions. In QML, Variational Quantum Algorithms (VQAs), such as the Variational Quantum Eigensolver (VQE) and Quantum Approximate Optimization Algorithm (QAOA), employ a hybrid quantum-classical optimization to solve the specific problems in quantum chemistry and combinatorial optimization [120]. VQAs leverage quantum properties such as superposition and entanglement to enhance certain optimization tasks, making them a potential bridge between quantum computing and ML-related applications. Quantum matrix operations, including matrix

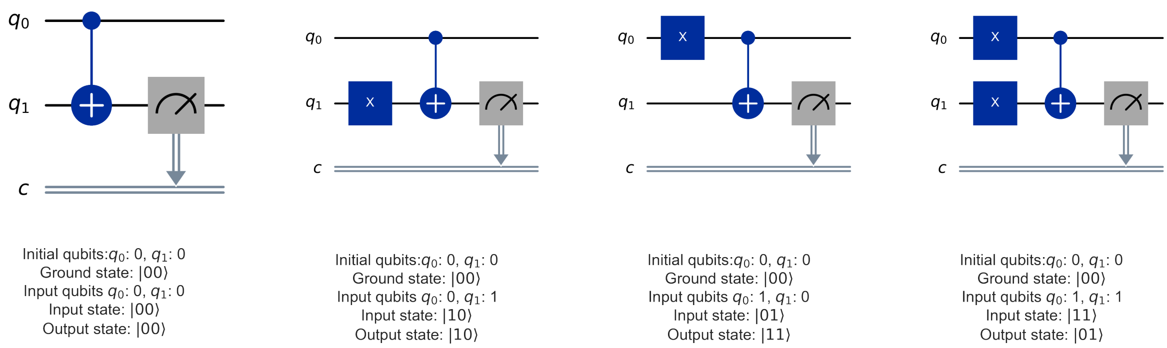

Figure 6: Functional verification of the CNOT gate

multiplication and inversion, provide a mathematical backbone for QML algorithms [121], with the Harrow-Hassidim-Lloyd (HHL) algorithm [122] solving linear equations under specific conditions, such as sparsity and well-conditioned matrices, thus achieving exponential speedup. Quantum Principal Component Analysis (QPCA) [123] accelerates dimensionality reduction and feature extraction, assuming efficient encoding of classical data into quantum states. Adiabatic Quantum Computing (AQC) [124], which finds near-optimal solutions by gradually evolving the quantum system, is particularly suitable for solving the complex combinatorial optimization problems. Quantum boosting algorithms [125] enhance QML performance by combining classical boosting techniques with quantum methods, such as leveraging quantum-enhanced weak learners. Lastly, Quantum Amplitude Amplification [126] generalizes Grover's algorithm to further improve search efficiency in applications where initial success probabilities exceed uniform distributions.

3.2. Quantum Machine Learning

QML tasks can be divided into two categories. One involves using QML algorithms to process quantum data, and the other involves using QML algorithms to process classical data [106]. Quantum data originates from quantum systems and is not applicable to the scenarios of this study, and therefore it will not be further discussed. From the implementation perspective, QML can be divided into pure QML and hybrid quantum-classical QML [127]. Pure QML relies entirely on quantum circuits, but its applicability is severely constrained by current quantum hardware, making it impractical for large-scale tasks such as WT CM. In contrast, hybrid quantum-classical methods incorporate quantum circuits into the conventional machine learning models by replacing key computational steps, neurons, or network layers to enable execution on existing quantum hardware. These circuits are known as parameterized quantum circuits (PQCs). For WT CM, the appropriate approach is to train the hybrid quantum-classical QML models using classical data.

As shown in Figure 7, the implementation of QML in WT CM follows a structured process that combines classical neural networks with quantum circuits. The overall workflow is similar to the conventional ML-based WT CM in Figure 5, with the main difference being the introduction of a hybrid quantum-classical QML model in the model design stage. Specifically, the preprocessed operational data from WTs is first fed into the classical neural network. Next, the classical data is encoded into quantum states through data encoding before being processed by the PQC. The encoded quantum states are then inputted into the PQC that implements the ML algorithm. Finally, the output of the hybrid neural network is used for WT CM tasks.

3.2.1. Data Encoding

Classical data can be used by QML algorithms after being processed by quantum circuits that implement encoding algorithms [128]. Encoding circuits usually contain quantum gates such as Hadamard gates that can excite classical data into quantum states, and common encoding methods include direct encoding, amplitude encoding, and density matrix encoding [129]. Direct encoding maps each data feature to a corresponding qubit, making the number of qubits

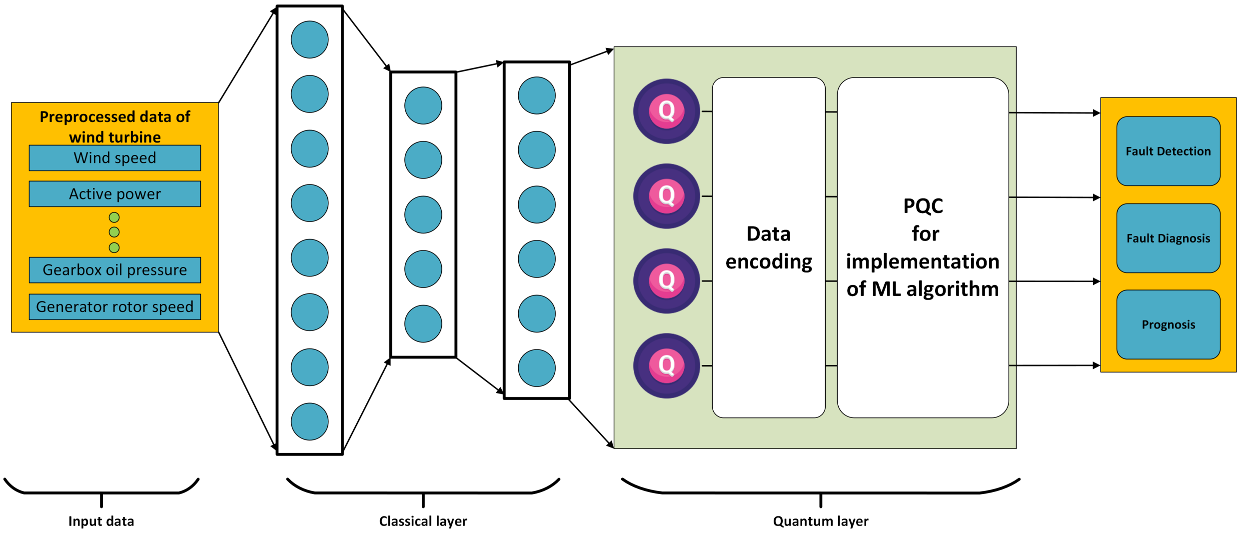

Figure 7: Hybrid quantum-classical QML architecture for WT CM

equal to the data dimensionality n, suitable for low-dimensional data. Amplitude encoding normalizes data features and maps them to the amplitudes of a quantum state, efficiently representing high-dimensional data, where the number of qubits required is the logarithm of the data dimensionality, $\log_2(n)$. Density matrix encoding maps data features to the density matrix of a quantum state, typically requiring more qubits than the data dimensionality, such that the number of qubits q satisfies $q > n$. Each encoding method imposes different requirements on the number of qubits and the input data dimensionality. Suitable encoding method can significantly affect the efficiency of the QML. More technical details on data encoding can be found in [130].

3.2.2. Parameterized Quantum Circuits

The PQC is used to implement ML algorithms and serve as a core component of QML. It has a layered structure similar to a neural network and consists of layers of parametrized quantum gates and measurement operations. The layers of parametrized quantum gates include rotation gates such as R_X, R_Y, R_Z, and controlled gates like CR_Z and CNOT. These gates function similarly to the parameters in a neural network, represented by values such as rotation angles. These parameters function similarly to weights in a neural network, allowing the PQC to learn and adapt to specific tasks. The measurement operations convert the quantum states in the PQC into classical data, which can then be fed into an optimizer. Through parameter optimization, the optimizer adjusts the parameters of the quantum gates to improve the performance of the PQC [131]. The operation of a PQC is illustrated in Figure 8. First, it is necessary to encode the classical data into quantum states and initialize the parameters. Next, the quantum circuit runs on a quantum computer with the current parameters to generate quantum states. Then, these quantum states are measured to obtain output results. The loss value of the cost function is calculated based on the measurement results. A optimization algorithm is employed to update the parameters of the quantum circuit. Finally, the above steps are repeated until the loss function converges to the desired value. Complex PQCs can be constructed by stacking multiple layers of parameterized quantum gates, enabling the implementation of different QML algorithms.

Figure 9 illustrates a PQC designed for binary classification, comprising a data encoder and two layers of parameterized quantum gates. The circuit operates on two qubits (q_0 and q_1), and uses two classical bits (c_0 and c_1) to store the measurement results. Initially, classical data is encoded into quantum states using R_Y gates applied to q_0 and q_1. A barrier is inserted to separate the encoding phase from the parameterized circuit, enhancing the readability of the circuit. The first layer of the parameterized circuit includes parameterized R_X and R_Y gates acting on both q_0 and q_1, followed by a CNOT gate that entangles the qubits. The second layer similarly applies parameterized R_X and R_Y gates and another CNOT gate. Finally, the quantum states are measured, with the results stored in classical bits c_0 and c_1. The R_X and R_Y gates represent rotations around the X and Y axes, respectively, while the CNOT gates (depicted by

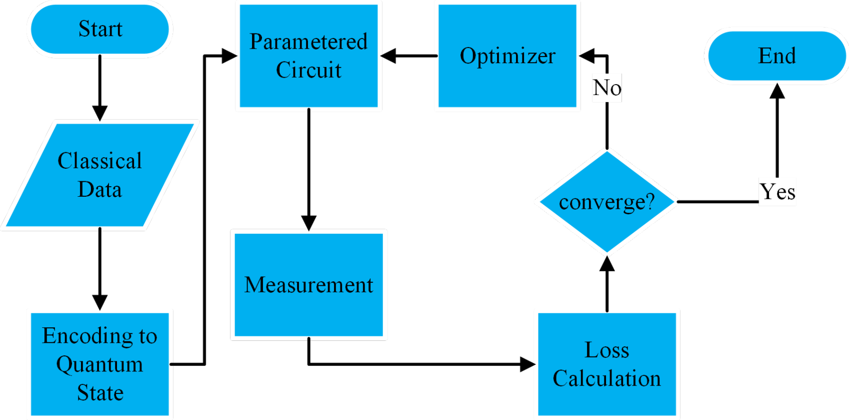

Figure 8: Flowchart of PQC

the blue plus symbol and connected dot) conditionally flip the target qubit based on the state of the control qubit. The measurement operations (grey squares) collapse the quantum states into classical bits to produce the final classification output.

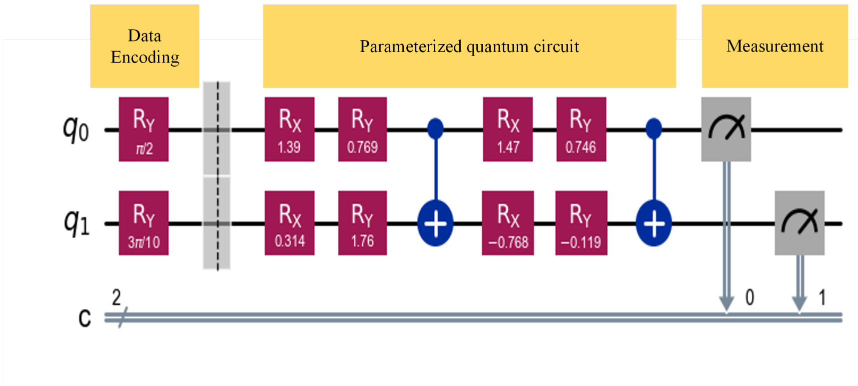

Figure 9: Quantum binary classifier circuit

PQCs offer a broad application prospect in QML. By combining the parallel computing capabilities of quantum computation with the classical optimization techniques, PQCs can demonstrate a superior performance in specific ML tasks. More details about PQC can be found in [132].

3.2.3. QML Algorithms

Currently, many conventional ML algorithms have been extended to their quantum counterparts, including both pure QML models and hybrid quantum-classical QML models. These algorithms have demonstrated their performance on various public datasets that is comparable to or even surpasses the performance of the conventional ML, while also exhibiting a greater potential in terms of explainability and model interpretability. This suggests that QML is already capable of being applied to WT CM to enhance wind energy efficiency.

Variational Quantum Classifier (VQC) is a classifier based on the VQA, which maps input data into a high-dimensional Hilbert space using PQC and obtains classification results through quantum measurement [133]. Compared to SVM and neural networks, VQC benefits from quantum superposition and entanglement, enabling stronger feature representation in high-dimensional nonlinear classification tasks, making it particularly suitable for learning complex data distributions. In terms of explainability, the measurability of quantum states allows VQC to provide a more intuitive interpretation of decision boundaries, making it more transparent than the deep neural networks. However, compared to classical SVM, its decision-making process remains relatively complex.

Quantum Markov Chain Monte Carlo (QMCMC) leverages quantum computation to accelerate MCMC sampling, improving the computational efficiency of Bayesian inference and probabilistic reasoning [134]. MCMC methods suffer from slow convergence in high-dimensional spaces, whereas QMCMC utilizes quantum superposition to parallelize multiple possible state transitions, reducing the mixing time of the Markov chain and thereby accelerating the sampling process. Compared to MCMC, QMCMC operates directly on probability distributions and provides statistical information during quantum measurement, making the inference process more transparent and facilitating a better understanding of model convergence and parameter estimation.

Quantum Boltzmann Machine (QBM) models probability distributions using the energy levels of quantum states, making it suitable for probabilistic generative modeling and feature learning [135]. Compared to the Boltzmann Machine (BM), QBM leverages quantum tunneling and quantum entanglement to escape local optima more efficiently, enhancing its ability to approximate complex energy distributions and thereby accelerating training and improving generalization performance. Parameter adjusting in QBM relies on quantum partition function computation, providing stronger physical interpretability than BM. However, due to the complex evolution of quantum states, its training process still requires further optimization to reduce computational overhead and mitigate error accumulation.

Quantum SVM (QSVM) utilizes Quantum Kernel Estimation (QKE) to classify high-dimensional data. Unlike SVM, which requires computing kernel function values for all pairs of data points, QSVM directly constructs quantum states in Hilbert space and measures their similarity using quantum computation, reducing the computational complexity from $O(N^2)$ to $O(logN)$ [136]. Since QSVM relies on quantum state similarity calculations, its decision boundaries can be derived through quantum measurements, making it more interpretable than the classical kernel SVM and facilitating a clearer understanding of the classification process.

The Quantum Gaussian Process (QGP) utilizes quantum kernel methods for Bayesian regression and has the potential to offer computational advantages over Gaussian Process Regression (GPR) in high-dimensional regression tasks [137]. GPR suffers from high computational complexity due to the need to invert the covariance matrix, making it inefficient for large datasets. In contrast, QGP leverages quantum computation to accelerate covariance matrix calculations, reducing computational complexity and improving regression efficiency. Since QGP follows a Bayesian inference framework, it provides a stronger interpretability in uncertainty estimation compared to deep learning models. However, the quantum computation component, which involves high-dimensional quantum mappings, may still introduce some degree of black-box behavior.

The Quantum Genetic Algorithm (QGA) integrates quantum computing with evolutionary optimization by encoding individuals as quantum states and applying quantum gate operations for mutation and evolutionary search, enabling population evolution to occur across a larger search space simultaneously, thereby enhancing global search capability [138]. Compared to the conventional Genetic Algorithm (GA), QGA has the potential to achieve a faster convergence in large-scale search problems and can mitigate the impact of local optima, particularly in complex optimization tasks.

Quantum Particle Swarm Optimization (QPSO) integrates quantum computing with particle swarm optimization (PSO) by encoding particle information in quantum superposition states and updating their positions using quantum gate operations, enabling a faster convergence during the search process and reduces the likelihood of getting trapped in local optima [139]. Compared to PSO, QPSO demonstrates an improved search efficiency in high-dimensional non-convex optimization problems.

Quantum Neural Network (QNN) is constructed using PQC. Unlike neural networks such as MLP and CNN, QNN leverages quantum superposition and entanglement to enhance information representation, potentially offering superior computational efficiency in high-dimensional data processing tasks [140]. Additionally, since the core computations of QNN rely on quantum gate operations, its computational complexity may be lower than that of classical neural networks for certain tasks. Compared to classical deep learning, QNN benefits from the transparency of quantum measurements, allowing feature extraction and decision-making processes to be analyzed through quantum state interpretation, thereby improving model explainability.

Quantum CNN (QCNN) utilizes Quantum Feature Mapping (QFM) [141] to project data into a high-dimensional Hilbert space and employ Quantum Pooling to reduce dimensionality, thereby lowering computational complexity while preserving key features [142]. Due to the parallelism inherent in quantum computing, QCNN can achieve the feature extraction performance comparable to classical CNN with fewer parameters. In terms of explainability, since QCNN performs feature extraction through quantum state transformations, its learning process can be analyzed via quantum measurements, making its feature representation and decision-making mechanism more interpretable than classical CNNs.

Hybrid QCNN combines the strengths of CNN and QCNN. A common implementation involves using a CNN for initial feature extraction, followed by a QNN for high-dimensional feature mapping and classification [143]. Compared to CNN, Hybrid QCNN may exhibit superior feature extraction performance in complex pattern recognition tasks due to the introduction of quantum computing. However, since hybrid QCNN still partially relies on classical computation, its overall computational cost may not necessarily be lower than that of CNN.

Quantum LSTM (QLSTM) utilizes PQC to implement the gating mechanisms of LSTM, enabling information storage and updates within quantum states while leveraging the parallelism of quantum computing to enhance computational efficiency [144]. Compared to LSTM, QLSTM benefits from the storage capacity of quantum states, allowing it to retain long-term dependencies more effectively and improve prediction accuracy. Since QLSTM stores information in quantum states, its temporal state transitions can be visualized through quantum measurements, making it more interpretable than LSTM in understanding state evolution.

Quantum GAN (QGAN) integrates a quantum generator and a quantum discriminator, making it well-suited for data generation tasks [145]. Compared to GAN, QGAN leverages quantum superposition to generate data, resulting in more diverse data distributions while mitigating the mode collapse problem, thereby improving data generation quality. Additionally, due to the advantages of quantum computing, QGAN exhibits better generalization in small-sample learning scenarios and has the potential for computational speedups in high-dimensional data generation tasks. Since quantum states can be directly measured and analyzed probabilistically, the generated data distribution can be explicitly interpreted through quantum measurements, offering improved explainability over GAN.

Quantum Reinforcement Learning (QRL) integrates reinforcement learning with quantum computing, leveraging quantum Q-learning and quantum policy gradient methods to enhance learning efficiency [146]. In reinforcement learning (RL), an agent needs extensive interactions and sample collections to optimize its policy, leading to high computational costs, particularly in high-dimensional state spaces. QRL utilizes quantum parallelism, allowing the agent to explore multiple states simultaneously in quantum state space, thereby improving policy optimization efficiency and reducing convergence time. This advantage is especially significant in sparse reward environments, where QRL may converge faster than RL. Compared to RL, QRL benefits from the measurability of quantum states, making the policy optimization process more transparent and improving its interpretability of decision-making.

3.2.4. QML vs. Conventional ML

In neural networks, the output of a neuron in the l-th layer is given by equation (3), where $a^{(l)}$ is the output of the l-th layer, $W^{(l)}$ is the weight matrix, $b^{(l)}$ is the bias vector, and f is the activation function. The loss function is typically defined as equation (4), where y_i is the true output, $\hat{y}_i$ is the predicted output, and ℓ is the loss function (e.g., cross-entropy loss).

$$a^{(l)} = f\left(W^{(l)} a^{(l-1)} + b^{(l)}\right) \tag{3}$$

$$L = \frac{1}{N} \sum_{i=1}^{N} \ell\left(y_i, \hat{y}_i\right) \tag{4}$$

In PQCs, the parameterized quantum state is given by equation (5), where $U(\theta)$ is the quantum circuit dependent on parameters θ, and $|\psi_0\rangle$ is the initial quantum state. The expectation value of a measurement operator O is defined by equation (6). The loss function can be defined by equation (7), where y_i are the true outputs and $p_i(\theta)$ are the probabilities predicted by the quantum circuit. Figure 10 briefly compares the differences between conventional ML and QML. A two-layer ANN is shown in the upper left corner of the figure, and a schematic of an LSTM is shown in the lower left corner. The top right corner is a QNN with the same function as the ANN, and the blue boxes in the QNN are the PQCs, where a layer of PQC can be regarded as a hidden layer. The lower right corner is a schematic diagram of QLSTM, where the gate computation in LSTM is replaced by PQC. This figure depicts the correspondence between QML and ML.

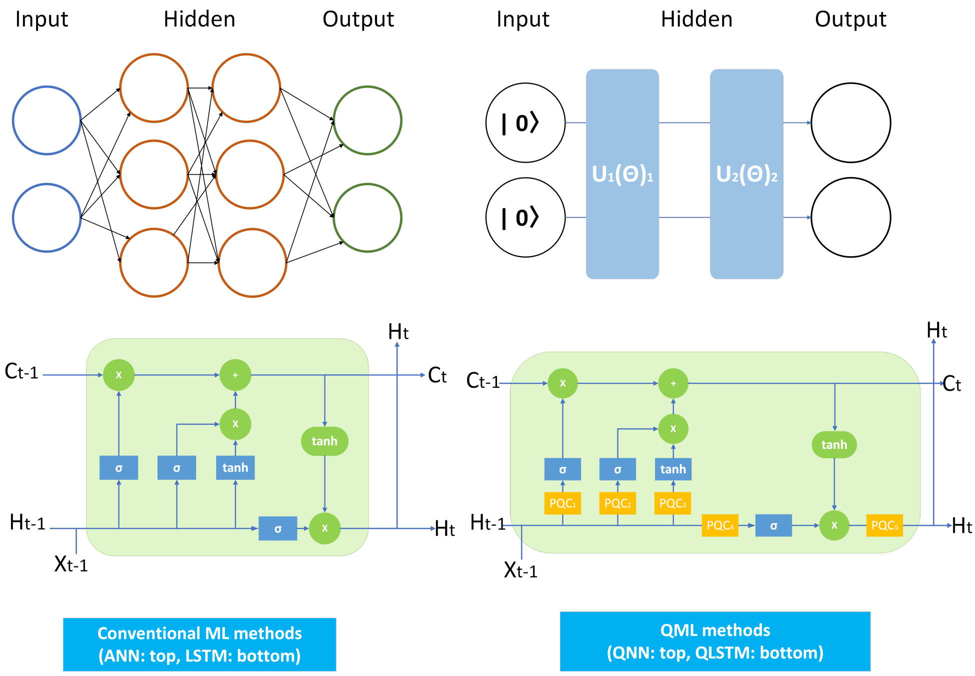

Figure 10: Comparison of conventional ML and QML

$$|\psi(\theta)\rangle = U(\theta)|\psi_0\rangle \tag{5}$$

$$\langle O\rangle_\theta = \langle\psi(\theta)|O|\psi(\theta)\rangle \tag{6}$$

$$L(\theta) = -\sum_i y_i \log(p_i(\theta)) \tag{7}$$

3.2.5. Running of Quantum Machine Learning

QML programs consist of classical computing platforms and quantum computing platforms. The classical computing platform includes development environments and ML frameworks. Development environments refer to integrated development environments (IDEs) such as PyCharm, Jupyter Notebook, and VS Code, as well as programming languages such as Python and C++. Quantum-specific programming languages, such as Q# and Cirq, are also used. ML frameworks include tools such as TensorFlow Quantum and PyTorch Quantum, which support quantum computing. The quantum computing platform includes quantum software development kits (SDKs), real quantum hardware, and quantum simulators. Quantum SDKs are based on classical or quantum-specific programming languages, which are used to organize and execute tasks on real quantum hardware or quantum simulators.

Qiskit, PennyLane, Braket, and Microsoft Quantum Development Kit are the current popular quantum SDKs. Qiskit is developed by IBM for designing, simulating, and executing quantum circuits on quantum computers of IBM, and provides a library of well-established QML algorithms, including QSVM, QNN, which can be seamlessly integrated with Pytorch and TensorFlow. PennyLane is well-suited for VQA research, but it can be cumbersome to construct the QNN. Braket supports various quantum hardware platforms and simulators, making it convenient for researchers to experiment with quantum algorithms in the cloud. The Microsoft Quantum Development Kit is based on Q#, and focuses on quantum algorithm development. It provides comprehensive libraries, simulators, and integration with Azure Quantum, making it ideal for large-scale quantum research.

To run a QML model on a real quantum computer needs to rely on a quantum computing resource provider on the cloud. Platforms providing quantum hardware computing resources include IBM Quantum, Google Quantum, Rigetti Computing, D-Wave Systems, IonQ, and Xanadu. These platforms operate their own quantum computers and make quantum computing resources available through cloud services. Google have recently introduced a quantum chip named Willow, equipped with 105 qubits [147]. Willow demonstrates exponential error reduction, meaning that as the number of qubits increases, errors decrease, and all error rates remain below the surface code threshold. Reducing error rates in quantum hardware and quantum algorithms presents significant challenges. [148] introduces the fundamental concepts and techniques of quantum error correction. [149] identifies error correction challenges associated with different quantum hardware materials and discusses potential solutions. [150] explores methods for mitigating quantum errors in current quantum hardware. [151] provides an outlook on how quantum error correction can drive the development of next-generation quantum hardware toward more general-purpose quantum computing. The most advanced quantum computer of IBM contains 1,000 qubits, with up to 156 qubits accessible via cloud services [152]. Amazon provides access to quantum computers with 84 qubits [153]. However, the implementation of fault-tolerant quantum circuits is extremely costly; therefore, quantum computing resource providers generally offer very limited free access. As a result, the availability of real quantum hardware for QML research is significantly constrained. When quantum computers are not yet fully developed or resources are limited, quantum simulators play a key role in QML research. These simulators replicate the behavior of quantum computers, accurately modeling the evolution of quantum states and the execution of quantum circuits. They are widely used for testing quantum algorithms, optimizing quantum circuits, and studying physical phenomena in quantum systems. Popular quantum simulators include Qiskit Aer, Cirq simulator, Amazon Braket simulator. SDKs such as Qiskit and PennyLane support switching the runtime environment of QML programs between different real quantum hardware and quantum simulators with minimal code.

3.3. QML Applications

3.3.1. Wide applications

QML has shown significant potential across various fields by processing large and complex datasets more effectively than conventional ML methods. The physics is one of the earliest fields to explore QML. In [154], the quantum variational classifier (QVC) was applied to datasets generated by the Large Hadron Collider to distinguish between signal and background, demonstrating a higher classification accuracy compared to SVM and RF. Similarly, QSVM has been proven to outperform SVM in signal and background classification tasks on quantum hardware provided by IBM, Google, and Amazon [155]. Ref [156] provides a comprehensive review of QML applications in high-energy physics.

In the field of NLP, QML has been widely discussed. The feasibility of applying QML to NLP was explored in [157]. Ref [158] discussed the potential of QML in translation, while [159] provided a comprehensive classification and analysis of QML applications in various NLP tasks. In [160], QLSTM networks have outperformed conventional LSTM networks in part-of-speech tagging tasks on social media code-mixed datasets, such as Twitter and Facebook. As NLP has evolved into large language model (LLM), the potential of QML in LLM has also been discussed. The feasibility of implementing LLM with QML was discussed in [161]. The prospects of QML-based LLM were explored in [162], while ref [163] investigated the design of QML algorithms tailored for LLM.

QML has also shown significant potential in chemistry and materials. For instance, a QNN combined with a quantitative structure-property relationship (QSPR) model was used to predict the corrosion inhibition efficiency of pyrimidine compounds [164]. Similarly, variational quantum circuits (VQC) were employed to predict the corrosion inhibition efficiency of pyridine-quinoline compounds [165]. In terms of more advanced applications, a framework named QMLMaterial was developed as a universal QML-based materials design and discovery tool [166]. The application of Quantum Machine Learning (QML) in compound space exploration was reviewed in [167]. Ref [168] focused on QML applications in medicinal chemistry, while a comprehensive review and outlook on QML in chemistry and materials science were provided in [169].

In medical and cancer research, QML offers new opportunities for improving diagnostics and treatments. By processing medical imaging and genomic data, QML has significantly advanced cancer diagnosis and therapeutic strategies [170]. Hybrid quantum-classical GANs have also been applied for image generation, showcasing QML's ability to learn discrete distributions effectively and its capabilities in image processing tasks [171]. Besides, QML has enhanced protein function prediction and drug design by accurately predicting protein PKA [172].

In fluid dynamics and hydrodynamics, QML has been applied to model complex processes, such as liquid-solid circulating fluidized bed risers, providing valuable insights for fluid dynamics research [173]. Contributions of the

QML to distributed computing and data security include secure implementations within distributed systems, enhancing both data privacy and big data analysis capabilities [174]. QML has also been used for health monitoring, such as drowsiness detection via EEG signals, offering efficient solutions for fatigue detection and health management [175].

In the broader context of ML, [176] systematically reviewed QML advancements, emphasizing its computational benefits and implementation strategies. However, the review largely focuses on theoretical developments, lacking critical discussion on the scalability and feasibility of these approaches in real-world applications. [177] explored the practical applicability of QML, yet failed to propose the fundamental challenges posed by current quantum hardware, such as noise sensitivity and the limited number of qubits, which significantly hinder its deployment beyond small-scale experimental settings. In image processing, [178] surveyed QML techniques for image classification, however, its discussion remains largely speculative, as existing quantum models struggle to handle high-dimensional data efficiently due to the limitations in quantum hardware. Similarly, [179] analyzed the role of QML in medical imaging but overlooked the critical issue of data availability, as medical datasets are often too large for current quantum devices to process, making the claimed advantages largely theoretical. The application of QML in hybrid quantum-classical communication networks was discussed in [180], yet the study lacks concrete evidence on how QML can outperform conventional ML in this domain, particularly in terms of reliability and scalability. [181] reviewed the HHL algorithm and its application in QML, however, as with many studies on quantum algorithms, it assumes idealized conditions that do not align with the constraints of near-term quantum hardware, raising questions about its practical relevance. Additionally, [182] provided a state-of-the-art review on hybrid quantum-classical ML but largely avoided discussing the lack of empirical validation, which remains a major bottleneck in this field. These studies collectively illustrate both the potential and the limitations of QML, yet they have not addressed the pressing issues of hardware scalability, error mitigation, and real-world implementation.

3.3.2. Applications in Industrial and Power Sector

In the industrial and power sectors, QML has been extensively studied for its potential to improve system monitoring and optimization. For example, QML was used to classify if a wave energy device has a negative impact on the environment, achieving accuracies of 98% with a QSVM and 87.5% with a conventional SVM [183]. In photovoltaic systems, QML has been used for fault detection by optimizing QNNs through parameters such as the number of qubits, encoding methods, and the depth of PQC, by considering various factors affecting QNN performance and conducting extensive evaluations. As a result, the QNN achieved a peak accuracy of 93.89% while requiring significantly fewer convergence steps compared to ANN, demonstrating the superiority of QNN over ANN in both accuracy and training efficiency [184]. To predict solar irradiance, two ANNs embedded with 1 and 2 QNN layers, respectively, were compared to a conventional ANN, with the ANN containing 2 QNN layers achieving the lowest error rate. In addition to outperforming the conventional ANN, the model with 2 QNN layers required significantly fewer parameters, demonstrating the efficiency advantage of QNN in reducing model complexity while maintaining superior performance [185]. QML has also been applied to transient stability assessment (TSA) in power systems, where a quantum gradient descent method has enhanced model accuracy and robustness [186]. Furthermore, a multi-agent grid control framework based on quantum Q-learning was proposed, where QNNs replaced the key computational units of Q-learning, significantly improving grid control performance and reducing carbon emissions compared to conventional algorithms [187]. In another study, a QML-based deep belief network (DBN) was employed to monitor the Continuous Stirred Tank Reactor (CSTR) and the Tennessee Eastman (TE) processes, achieving 79.2% accuracy for the CSTR and 99.39% for the TE process [188]. A hybrid QML-based deep learning framework was tested on a simulated power system with 30 bus bars, addressing substation and transmission line faults. By embedding a quantum computing unit into the Conditional Restricted Boltzmann Machine (CRBN), the framework achieved improved false alarm and missed detection rates compared to conventional models like ANN and DT [189].

For further insights into QML applications in the energy sector, readers may refer to studies as follows. [190] provided an overview of quantum computing applications in power system analysis, covering topics such as quantum optimization and quantum security. [191] discussed the potential of QML in smart grid control, highlighting their role in enhancing grid efficiency and security. [192] reviewed recent advances in QML for grid analytics and optimization. These studies indicate the growing interests in leveraging QML for improving power system performance. However, these studies primarily concentrate on control and optimization aspects.

3.3.3. Applications in Wind Turbine Condition Monitoring

QML has had many studies in fault detection and fault diagnosis. In [193], quantum genetic algorithm (QGA) was combined with least squares support vector machine (LSSVM) to detect the wind turbine gearbox fault. QGA-LSSVM gained a higher accuracy than GA-LSSVM and other methods. In [194], QVE was applied to detect multi-class faults for WTs with the available public datasets. Despite multiple adjustments to the structure of QVE, it consistently outperformed Multi-Layer Perceptron (MLP) models in accuracy, F1, particularly in heavyweight datasets characterized by large data volumes, high dimensionality, and numerous variables. Ref [195] employed a hybrid quantum-classical ML approach to analyze WT blade ultrasonography data for blade damage detection. The results demonstrated that QNNs achieved over a 20% accuracy improvement compared to ANNs. In [196], PCA and AE were used for feature reduction of SCADA data with pitch fault, and different dimensions were selected for model training respectively. This study compared the performance of SVMs with different kernels and QSVM, where the Gaussian SVM achieved an accuracy of 94.5%, while the QSVM achieved an accuracy of 92.5%. A Hybrid Quantum-Classical CNN (QC-CNN) integrated with Adaptive Moment Estimation (Adam) was used to detect high-temperature faults in wind turbine intermediate bearings, achieving an accuracy of 99.2%, outperforming the conventional CNN models [197]. These studies initially proved the preliminary feasibility of QML in WT CM as well as the research prospect in WT CM.

More advanced applications of QML in WT CM are associated with the control optimization. QNN was used for the optimization of maximum power point tracking (MPPT) control parameters, which captures the maximum wind energy compared to methods such as CNN [198]. In [199], a novel quantum parallel multi-layer Monte Carlo optimization algorithm (QPMMCOA) was proposed to optimize the rotor-side controller (RSC) parameters to maximize the power output, achieving better MPPT than methods such as GA. Ref. [200] proposed a Quantum Deep Reinforcement Learning (QDRL) algorithm for double-fed induction generator-based WTs. By integrating quantum layers with reinforcement learning, this approach enabled real-time control updates while avoiding offline computation. The QDRL model exhibited a superior control performance under varying wind speeds and voltage conditions, outperforming other techniques in terms of metrics such as Integrated Absolute Error (IAE), Integral Squared Error (ISE), and Integrated Time-Weighted Absolute Error (ITAE).

Methods such as WT layout optimization, wind speed forecasting, power prediction can improve the efficiency of wind farm [201–203]. Quantum annealing algorithm was used to find the best layout of wind farm to maximize the available wind speed of each WT in the wind farm [204]. In [205], CNN, LSTM, and QNN were concatenated to form a hybrid quantum-classical model to predict environmental data with seasonality, achieving an R2 that outperforms other methods in all seasons, ensuring that the WT operates at safe wind speeds. QGA was combined with fuzzy neural network to predict the power generation and gained an accuracy of 87.12%, improving the stability and reliability of grid-connected wind power [206]. In another study, QDRL was used to predict power output, achieving a 13% improvement in prediction accuracy and a 20% improvement in power generation [207].

Through a detailed view of the literature, it is evident that although the application of QML in WT CM is limited, it appears from the performance comparison that QML is of a great research value for WT CM. Table 5 presents the limits of the reviewed studies on conventional ML and QML applied to WT CM. Given that conventional ML and QML utilize different datasets, feature engineering methods, and network architectures in the current body of research, a direct performance comparison between the two is not entirely feasible. Conventional ML studies typically rely on highly optimized or stacked complex network structures combined with specific feature engineering techniques, whereas QML research often employs original network architectures and simpler datasets. Consequently, the performance differences between the two are influenced not only by the algorithms themselves but also by experimental conditions. Existing research demonstrates that QML has undergone initial validation on relatively simple datasets, showing comparable performance to the conventional ML in key metrics such as accuracy, precision, and F1 score. With its foundational performance established, QML holds a significant potential for further application to more complex datasets and tasks, suggesting that it may achieve even better performance in handling more intricate data. Additionally, QML offers several theoretical advantages, including lower dependency on feature engineering, stronger generalization ability, and reduced risk of overfitting. These characteristics make it potentially more efficient when handling complex and large-scale data. Although these theoretical advantages still require further validation in practical scenarios, as future research delves deeper, particularly under comparable data preprocessing methods, network architectures, and experimental conditions, QML is expected to demonstrate performance on par with, or even superior to the conventional

ML. Therefore, while conventional ML currently dominates in practical applications, the potential demonstrated by QML suggests it may lead to breakthrough developments in the future, particularly in optimizing resource utilization and addressing more complex tasks in the WT CM field.

Table 5: Comparison of current research in ML and QML

Criteria	Conventional ML				QML				
	Simple ML	ANN-Based	CNN-Based	RNN-Based	QVC	QSVM	QNN	QCNN	QLSTM
Accuracy	92%-96.9%	93.53%-99.68%	97%-99.85%	<=96.1%	<=98.47%	<=97.4%	<=97.4%	78.8%-97%	85.9%-95.12%
Precision		87.2%-90.85%		<=98.98%	<=98%				
F1 Score		70%-87.5%	80.2%-98%	86.29%-97.6%	<=97%				70.86%-87.67%
Feature Engineering	High	Moderate	Low	Low	Low	Low	Low	Low	Low
Overfitting Risk	Low	Low	High	High	Low	Low	Moderate	Moderate	Moderate
Generalization Ability	Low	Moderate	High	High	Low	Low	Moderate	High	High
Real-World Application	High	High	High	High	Research stage	Research stage	Research stage	Research stage	Research stage

4. Challenges and Future Prospects

4.1. Challenges

4.1.1. Limitations of Quantum Computing

QML in WT CM faces significant challenges due to limitations in current quantum hardware. These include unreliable qubits, hardware instability, shallow circuit depths, and limited resource availability [208]. For example, IBM's Condor chip, released in December 2023, features 1,121 qubits but suffers from a high error rate [209]. In contrast, the Willow chip from Google represents a more advanced solution with 105 qubits and lower error rates [210]. WT SCADA data often involves over 100 dimensions (variables), exceeding the current capabilities of quantum hardware. Most existing quantum computers fall under the category of Noisy Intermediate-Scale Quantum (NISQ) systems [211]. These systems are particularly vulnerable to quantum decoherence, noise, and computational errors, which lead to unstable results. Deeper quantum circuits further exacerbate these issues, increasing noise levels, decoherence, and computational errors, thus limiting both the performance and scalability of QML models [211]. Additionally, the high cost of quantum computing resources restricts access to real quantum hardware for extensive QML research. Researchers often resort to quantum simulators, which have limitations in scalability and efficiency. Integrating quantum computing components into existing classical computing infrastructure for WT CM also remains a significant technical challenge.

4.1.2. Limitations of QML Algorithm

Research on QML-based WT CM is constrained by high learning costs, long model training times, and low training efficiency. These challenges arise due to the complex nature of QML, which combines advanced mathematics, quantum physics, artificial intelligence, and engineering [212]. This interdisciplinary nature would make QML far more difficult to implement compared to conventional algorithms and increase the steep learning curve for researchers. For instance, there is currently no open-source framework for QLSTM-based time-series analysis. As a result, implementing QLSTM for WT CM requires extensive time for quantum circuit design and adaptation. Furthermore, due to the high cost and limited availability of quantum hardware, most QML models are trained on quantum simulators. These simulators significantly prolong training times, particularly as model complexity increases and more qubits are required [213]. Simulators also introduce noise models to simulate real hardware conditions, which coupled with long training durations, frequently result in training failures and low efficiency [214]. Addressing these challenges requires substantial optimization efforts to improve QML algorithms, reduce computational demands, and streamline the training process.

4.2. *Future Prospects*

4.2.1. *Benchmark Framework*

The research on QML in WT CM is still in the early stages, with studies primarily focusing on validating feasibility through simple algorithms. Key algorithms such as QNN, QCNN, QLSTM, QTransformer, and QGAN hold a significant promise for WT CM, however, their effectiveness must be scientifically validated under controlled conditions. For example, a comparative study between LSTM and QLSTM needs to be conducted under identical network architectures for NBM of SCADA data, allowing a precise analysis of their fitting performance. Similarly, a structured comparison between QGAN and GAN should be performed to determine whether QGAN can effectively address data imbalance in WT datasets, thereby improving fault diagnosis accuracy. Beyond algorithmic evaluation, multi-modal data fusion and end-to-end WT CM techniques need to be investigated to facilitate the scalable deployment of QML for condition monitoring of large-scale wind farms.

To systematically and rigorously assess the potential of QML in WT CM, a benchmark framework is recommended to develop, which is driven and evaluated by task-specific goals. This framework will leverage unified datasets, standardized interfaces, and a consistent evaluation protocol, thus enabling a fair and comprehensive assessment of various QML algorithms across different WT CM tasks. This initiative could bridge the gap between theoretical feasibility and practical implementation of the QML-based WT CM. The authors of this review paper are currently working on this innovative research.

Beyond algorithmic evaluation, this framework will also support research on multi-modal data fusion and end-to-end WT CM techniques, facilitating the scalable deployment of QML for large-scale wind farm condition monitoring. By establishing a scientifically rigorous, comprehensive, and task-driven benchmark, this initiative aims to provide the wind energy sector with a reliable foundation for the advancement and practical adoption of QML-based WT CM.

4.2.2. *QML algorithm Optimization*

Optimizing QML algorithms to meet the specific demands of WT CM is critical due to the limitations of current quantum hardware and algorithms. Key research should focus on encoding optimization, circuit optimization, and algorithm refinement. Efficient encoding methods can transform SCADA data into quantum states while reducing qubit requirements. This ensures same or better performance with fewer computational resources [215]. For example, encoding methods such as direct encoding, amplitude encoding, and qubit-efficient algorithms like "Word2Ket" [216] tailored for WT CM, could be explored. These encoding approaches can be combined with QSVM, QNN, and QLSTM models to evaluate whether they reduce qubit usage in WT CM without compromising, or even improving the model performance. Reducing circuit depth and gate complexity can improve model convergence, decrease computational costs, and accelerate inference speeds [217]. The most challenging yet impactful prospect is the optimization of QML algorithm structures. The goal is to design simpler and more efficient QML models capable of delivering equivalent or superior functionality. For instance, with advancements in quantum computing technology, the proportion of quantum circuits in previously mentioned hybrid quantum-classical neural networks can be gradually increased to investigate potential improvements in model performance. By addressing these areas, researchers can significantly enhance the efficiency, scalability, and practical applicability of QML-based WT CM models.

5. Conclusion

This review explores the progress of WT CM, tracing the transition from conventional ML approaches to QML. While ML techniques have achieved significant success in fault detection, diagnosis, and prognosis, their generalization capabilities remain limited. Additionally, the increasing complexity of ML models has led to diminishing returns in performance improvements. QML represents a promising direction by combining quantum computing with ML, offering potential advantages in improving model performance and scalability. Preliminary studies suggest its strong feasibility in WT CM. However, QML applications remain constrained due to the limitations of current quantum hardware and the algorithm development is presently still at the early stage.

This study highlights several unresolved challenges that must be addressed to unlock the full potential of QML for WT CM. Key areas of focus include the scalability of QML algorithms, integration with real-world energy systems, and the development of robust and interpretable models. Practical implementation also demands further advancements in quantum hardware, efficient data encoding techniques, and hybrid quantum-classical frameworks to manage the computational loads of large-scale applications.

Future research should focus on creating efficient and scalable QML algorithms tailored to specific WT CM tasks. Key priorities are suggested to include optimizing data encoding, reducing quantum circuit depth, and improving quantum-classical hybrid models. Comparative studies between QML and state-of-the-art ML methods using large-scale datasets are essential to validate the advantages of QML in practical applications.

The findings of this review should have broader implications for policymakers and industry stakeholders. To enhance WT CM systems enabled by QML presents a strong potential to significantly improve the reliability of wind energy, reduce operation and maintenance costs, and support global sustainability goals. Policymakers can play a pivotal role by spurring and investing in innovation in quantum technology development and fostering collaborations between academia and industry. Overcoming the current limitations of QML will pave the way for offering a paradigm shift approach to wind turbine condition monitoring activities.

Credit Authorship Contributions

Zhefeng Zhang: Conceptualization, Data curation, Formal analysis, Writing – original draft. **Yueqi Wu:** Conceptualization, Writing – review and editing. **Xiandong Ma:** Funding acquisition, Supervision, Investigation, Writing – review and editing.

Funding Sources

This work was supported by a scholarship from the Engineering Department of Lancaster University.

References

[1] G. W. E. Council, Global wind report 2024 (2024).
URL https://gwec.net/global-wind-report-2024/

[2] J. Liu, D. Song, Q. Li, J. Yang, Y. Hu, F. Fang, Y. Hoon Joo, Life cycle cost modelling and economic analysis of wind power: A state of art review, Energy Conversion and Management 277 (2023) 116628. doi:https://doi.org/10.1016/j.enconman.2022.116628.
URL https://www.sciencedirect.com/science/article/pii/S0196890422014066

[3] M. B. A. Bashir, Principle parameters and environmental impacts that affect the performance of wind turbine: an overview, Arabian Journal for Science and Engineering 47 (7) (2022) 7891–7909. doi:10.1007/s13369-021-06357-1.

[4] C. A. Walford, Wind turbine reliability :understanding and minimizing wind turbine operation and maintenance costs., SANDIA Report (3 2006). doi:10.2172/882048.
URL https://www.osti.gov/biblio/882048

[5] P. Tavner, D. M. Greenwood, M. W. G. Whittle, R. Gindele, S. Faulstich, B. Hahn, Study of weather and location effects on wind turbine failure rates, Wind Energy 16 (2) (2013) 175–187. arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/we.538, doi:https://doi.org/10.1002/we.538.
URL https://onlinelibrary.wiley.com/doi/abs/10.1002/we.538

[6] X. Wang, X. Zeng, J. Li, X. Yang, H. Wang, A review on recent advancements of substructures for offshore wind turbines, Energy Conversion and Management 158 (2018) 103–119. doi:https://doi.org/10.1016/j.enconman.2017.12.061.
URL https://www.sciencedirect.com/science/article/pii/S019689041731213X

[7] Z. Hameed, S. Ahn, Y. Cho, Practical aspects of a condition monitoring system for a wind turbine with emphasis on its design, system architecture, testing and installation, Renewable Energy 35 (5) (2010) 879–894. doi:https://doi.org/10.1016/j.renene.2009.10.031.
URL https://www.sciencedirect.com/science/article/pii/S0960148109004704

[8] C. Choe Wei Chang, T. Jian Ding, W. Han, C. Chong Chai, M. A. S. Bhuiyan, H. Choon-Yian, T. Chuan Song, Recent advancements in condition monitoring systems for wind turbines: A review, Energy Reports 9 (2023) 22–27, proceedings of 2022 7th International Conference on Renewable Energy and Conservation. doi:https://doi.org/10.1016/j.egyr.2023.08.061.
URL https://www.sciencedirect.com/science/article/pii/S2352484723012015

[9] M. Benbouzid, T. Berghout, N. Sarma, S. Djurović, Y. Wu, X. Ma, Intelligent condition monitoring of wind power systems: State of the art review, Energies 14 (18) (2021). doi:10.3390/en14185967.
URL https://www.mdpi.com/1996-1073/14/18/5967

[10] R. J. A. Vieira, M. A. Sanz-Bobi, Condition monitoring and maintenance methods in wind turbines, in: Use, Operation and Maintenance of Renewable Energy Systems: Experiences and Future Approaches, Springer International Publishing, Cham, 2014, pp. 3–59. doi:10.1007/978-3-319-03224-5_1.
URL https://doi.org/10.1007/978-3-319-03224-5_1

[11] M. Shoaei, Y. Noorollahi, A. Hajinezhad, S. F. Moosavian, A review of the applications of artificial intelligence in renewable energy systems: An approach-based study, Energy Conversion and Management 306 (2024) 118207. doi:https://doi.org/10.1016/j.enconman.2024.118207.
URL https://www.sciencedirect.com/science/article/pii/S0196890424001481

[12] W. Qiao, D. Lu, A survey on wind turbine condition monitoring and fault diagnosis—part i: Components and subsystems, IEEE Transactions on Industrial Electronics 62 (10) (2015) 6536–6545. doi:10.1109/TIE.2015.2422112.

[13] R. Pandit, D. Astolfi, J. Hong, D. Infield, M. Santos, Scada data for wind turbine data-driven condition/performance monitoring: A review on state-of-art, challenges and future trends, Wind Engineering 47 (2) (2023) 422–441. arXiv:https://doi.org/10.1177/0309524X221124031, doi:10.1177/0309524X221124031.
URL https://doi.org/10.1177/0309524X221124031

[14] K. Dineva, T. Atanasova, Systematic look at machine learning algorithms–advantages, disadvantages and practical applications, International Multidisciplinary Scientific GeoConference: SGEM 20 (2.1) (2020) 317–324. doi:10.5593/sgem2020/2T/s07.041.

[15] A. Schueller, C. Saldaña, Generalizability analysis of tool condition monitoring ensemble machine learning models, Journal of Manufacturing Processes 84 (2022) 1064–1075. doi:https://doi.org/10.1016/j.jmapro.2022.10.064.
URL https://www.sciencedirect.com/science/article/pii/S1526612522007575

[16] K. C. A. Khanzode, R. D. Sarode, Advantages and disadvantages of artificial intelligence and machine learning: A literature review, International Journal of Library & Information Science (IJLIS) 9 (1) (2020) 3.

[17] M. Pistoia, S. F. Ahmad, A. Ajagekar, A. Buts, S. Chakrabarti, D. Herman, S. Hu, A. Jena, P. Minssen, P. Niroula, A. Rattew, Y. Sun, R. Yalovetzky, Quantum machine learning for finance iccad special session paper, in: 2021 IEEE/ACM International Conference On Computer Aided Design (ICCAD), 2021, pp. 1–9. doi:10.1109/ICCAD51958.2021.9643469.

[18] M. Sajjan, J. Li, R. Selvarajan, S. H. Sureshbabu, S. S. Kale, R. Gupta, V. Singh, S. Kais, Quantum machine learning for chemistry and physics, Chemical Society Reviews 51 (15) (2022) 6475–6573.

[19] A. Ajagekar, F. You, Quantum computing and quantum artificial intelligence for renewable and sustainable energy: A emerging prospect towards climate neutrality, Renewable and Sustainable Energy Reviews 165 (2022) 112493. doi:https://doi.org/10.1016/j.rser.2022.112493.
URL https://www.sciencedirect.com/science/article/pii/S1364032122003975

[20] A. Stetco, F. Dinmohammadi, X. Zhao, V. Robu, D. Flynn, M. Barnes, J. Keane, G. Nenadic, Machine learning methods for wind turbine condition monitoring: A review, Renewable Energy 133 (2019) 620–635. doi:https://doi.org/10.1016/j.renene.2018.10.047.
URL https://www.sciencedirect.com/science/article/pii/S096014811831231X

[21] M. R. Innes Murdo Black, A. Kolios, Condition monitoring systems: a systematic literature review on machine-learning methods improving offshore-wind turbine operational management, International Journal of Sustainable Energy 40 (10) (2021) 923–946. arXiv:https://doi.org/10.1080/14786451.2021.1890736, doi:10.1080/14786451.2021.1890736.
URL https://doi.org/10.1080/14786451.2021.1890736

[22] D. Astolfi, F. De Caro, A. Vaccaro, Condition monitoring of wind turbine systems by explainable artificial intelligence techniques, Sensors 23 (12) (2023). doi:10.3390/s23125376.
URL https://www.mdpi.com/1424-8220/23/12/5376

[23] D. M. Abdullah, A. M. Abdulazeez, Machine learning applications based on svm classification a review, Qubahan Academic Journal 1 (2) (2021) 81–90. doi:10.48161/qaj.v1n2a50.

[24] B. de Ville, Decision trees, WIREs Computational Statistics 5 (6) (2013) 448–455. arXiv:https://wires.onlinelibrary.wiley.com/doi/pdf/10.1002/wics.1278, doi:https://doi.org/10.1002/wics.1278.
URL https://wires.onlinelibrary.wiley.com/doi/abs/10.1002/wics.1278

[25] K. Balachandar, R. Jegadeeshwaran, Friction stir welding tool condition monitoring using vibration signals and random forest algorithm – a machine learning approach, Materials Today: Proceedings 46 (2021) 1174–1180, 28th International Conference on Processing and Fabrication of Advanced Materials (PFAM28). doi:https://doi.org/10.1016/j.matpr.2021.02.061.
URL https://www.sciencedirect.com/science/article/pii/S2214785321011214

[26] T. Chen, C. Guestrin, Xgboost: A scalable tree boosting system, in: Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining, 2016, pp. 785–794. doi:10.1145/2939672.2939785.

[27] C. Bao, T. Zhang, Z. Hu, W. Feng, R. Liu, Wind turbine condition monitoring based on improved active learning strategy and knn algorithm, IEEE Access 11 (2023) 13545–13553. doi:10.1109/ACCESS.2023.3243625.

[28] G. Hübner, H. Pinheiro, C. de Souza, C. Franchi, L. da Rosa, J. Dias, Detection of mass imbalance in the rotor of wind turbines using support vector machine, Renewable Energy 170 (2021) 49–59. doi:https://doi.org/10.1016/j.renene.2021.01.080.
URL https://www.sciencedirect.com/science/article/pii/S0960148121000872

[29] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (2015) 436–444. doi:10.1038/nature14539.

[30] W. Fang, Y. Chen, Q. Xue, Survey on research of rnn-based spatio-temporal sequence prediction algorithms, Journal on Big Data 3 (3) (2021) 97.

[31] Y. Wang, H. Yao, S. Zhao, Auto-encoder based dimensionality reduction, Neurocomputing 184 (2016) 232–242.

[32] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial networks, Commun. ACM 63 (11) (2020) 139–144. doi:10.1145/3422622.
URL https://doi.org/10.1145/3422622

[33] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, Advances in Neural Information Processing Systems 30 (2017). doi:10.48550/arXiv.1706.03762.

[34] S. Zargar, Introduction to sequence learning models: Rnn, lstm, gru, Department of Mechanical and Aerospace Engineering, North Carolina State University (2021).

[35] K. Abid, M. Sayed Mouchaweh, L. Cornez, Fault prognostics for the predictive maintenance of wind turbines: State of the art, in: ECML PKDD 2018 Workshops: DMLE 2018 and IoTStream 2018, Dublin, Ireland, September 10-14, 2018, Revised Selected Papers 18, Springer,

2019, pp. 113–125. doi:10.1007/978-3-030-14880-5_10.
[36] O. E. Hairech, A. Lyhyaoui, Fault detection and diagnosis in condition-based predictive maintenance, in: International Conference on Advanced Intelligent Systems for Sustainable Development, Springer, 2022, pp. 296–301. doi:10.1007/978-3-031-35251-5_28.
[37] F. P. G. Márquez, Fault Detection, Diagnosis and Prognosis, IntechOpen, Rijeka, 2020. doi:10.5772/intechopen.77564.
URL https://doi.org/10.5772/intechopen.77564
[38] P. Das, R. Amano, Comparative analysis of three wind turbine blades, in: 50th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition, 2012, p. 519. arXiv:https://arc.aiaa.org/doi/pdf/10.2514/6.2012-519, doi:10.2514/6.2012-519.
URL https://arc.aiaa.org/doi/abs/10.2514/6.2012-519
[39] Z. Zhang, Z. Yin, T. Han, A. C. Tan, Fracture analysis of wind turbine main shaft, Engineering Failure Analysis 34 (2013) 129–139. doi:https://doi.org/10.1016/j.engfailanal.2013.07.014.
URL https://www.sciencedirect.com/science/article/pii/S1350630713002422
[40] E. Hart, B. Clarke, G. Nicholas, A. Kazemi Amiri, J. Stirling, J. Carroll, R. Dwyer-Joyce, A. McDonald, H. Long, A review of wind turbine main bearings: design, operation, modelling, damage mechanisms and fault detection, Wind Energy Science 5 (1) (2020) 105–124. doi:10.5194/wes-5-105-2020.
URL https://wes.copernicus.org/articles/5/105/2020/
[41] J. P. Salameh, S. Cauet, E. Etien, A. Sakout, L. Rambault, Gearbox condition monitoring in wind turbines: A review, Mechanical Systems and Signal Processing 111 (2018) 251–264. doi:https://doi.org/10.1016/j.ymssp.2018.03.052.
URL https://www.sciencedirect.com/science/article/pii/S0888327018301791
[42] A. G. Olabi, T. Wilberforce, K. Elsaid, E. T. Sayed, T. Salameh, M. A. Abdelkareem, A. Baroutaji, A review on failure modes of wind turbine components, Energies 14 (17) (2021). doi:10.3390/en14175241.
URL https://www.mdpi.com/1996-1073/14/17/5241
[43] C. Xiao, Z. Liu, T. Zhang, X. Zhang, Deep learning method for fault detection of wind turbine converter, Applied Sciences 11 (3) (2021). doi:10.3390/app11031280.
URL https://www.mdpi.com/2076-3417/11/3/1280
[44] M. Kim, P. Dalhoff, Yaw systems for wind turbines–overview of concepts, current challenges and design methods, Journal of Physics: Conference Series 524 (1) (2014) 012086. doi:10.1088/1742-6596/524/1/012086.
URL https://iopscience.iop.org/article/10.1088/1742-6596/524/1/012086/meta
[45] R. Gao, Z. Gao, Pitch control for wind turbine systems using optimization, estimation and compensation, Renewable Energy 91 (2016) 501–515. doi:https://doi.org/10.1016/j.renene.2016.01.057.
URL https://www.sciencedirect.com/science/article/pii/S096014811630057X
[46] Y. Zhao, D. Li, A. Dong, D. Kang, Q. Lv, L. Shang, Fault prediction and diagnosis of wind turbine generators using scada data, Energies 10 (8) (2017). doi:10.3390/en10081210.
URL https://www.mdpi.com/1996-1073/10/8/1210
[47] J. Maldonado-Correa, S. Martín-Martínez, E. Artigao, E. Gómez-Lázaro, Using scada data for wind turbine condition monitoring: A systematic literature review, Energies 13 (12) (2020). doi:10.3390/en13123132.
URL https://www.mdpi.com/1996-1073/13/12/3132
[48] M. L. Hossain, A. Abu-Siada, S. M. Muyeen, Methods for advanced wind turbine condition monitoring and early diagnosis: A literature review, Energies 11 (5) (2018) 1309. doi:10.3390/en11051309.
[49] D. Breteler, C. Kaidis, T. Tinga, R. Loendersloot, Physics based methodology for wind turbine failure detection, diagnostics & prognostics, in: A. Rosmi (Ed.), EWEA 2015, European Wind Energy Association, 2015, pp. 1–9, european Wind Energy Association Annual Conference and Exhibition 2015, EWEA 2015, EWEA 2015 ; Conference date: 17-11-2015 Through 20-11-2015.
[50] M. Schlechtingen, I. F. Santos, S. Achiche, Wind turbine condition monitoring based on scada data using normal behavior models. part 1: System description, Applied Soft Computing 13 (1) (2013) 259–270. doi:https://doi.org/10.1016/j.asoc.2012.08.033.
URL https://www.sciencedirect.com/science/article/pii/S1568494612003821
[51] J. M. Pinar Pérez, F. P. García Márquez, A. Tobias, M. Papaelias, Wind turbine reliability analysis, Renewable and Sustainable Energy Reviews 23 (2013) 463–472. doi:https://doi.org/10.1016/j.rser.2013.03.018.
URL https://www.sciencedirect.com/science/article/pii/S1364032113001779
[52] S. García, J. Luengo, F. Herrera, et al., Data preprocessing in data mining, Vol. 72, Springer, 2015. doi:https://doi.org/10.1007/978-3-319-10247-4.
[53] X. Chu, I. F. Ilyas, S. Krishnan, J. Wang, Data cleaning: Overview and emerging challenges, in: Proceedings of the 2016 International Conference on Management of Data, SIGMOD '16, Association for Computing Machinery, New York, NY, USA, 2016, p. 2201–2206. doi:10.1145/2882903.2912574.
URL https://doi.org/10.1145/2882903.2912574
[54] G. Dong, H. Liu, Feature engineering for machine learning and data analytics, CRC press, 2018.
[55] Q. Xie, Z. Dai, E. Hovy, T. Luong, Q. Le, Unsupervised data augmentation for consistency training, Advances in neural information processing systems 33 (2020) 6256–6268.
[56] S. A. Alasadi, W. S. Bhaya, Review of data preprocessing techniques in data mining, Journal of Engineering and Applied Sciences 12 (16) (2017) 4102–4107.
[57] A. Zheng, A. Casari, Feature engineering for machine learning: principles and techniques for data scientists, " O'Reilly Media, Inc.", 2018.
[58] A. Mumuni, F. Mumuni, Data augmentation: A comprehensive survey of modern approaches, Array 16 (2022) 100258. doi:https://doi.org/10.1016/j.array.2022.100258.
URL https://www.sciencedirect.com/science/article/pii/S2590005622000911
[59] B. J. Erickson, F. Kitamura, Magician's corner: 9. performance metrics for machine learning models, Radiology: Artificial Intelligence 3 (3) (2021) e200126. arXiv:https://doi.org/10.1148/ryai.2021200126, doi:10.1148/ryai.2021200126.

URL https://doi.org/10.1148/ryai.2021200126
[60] L. Xiang, P. Wang, X. Yang, A. Hu, H. Su, Fault detection of wind turbine based on scada data analysis using cnn and lstm with attention mechanism, Measurement 175 (2021) 109094. doi:10.1016/j.measurement.2021.109094.
[61] Y. Zheng, C. Wang, C. Huang, K. Li, J. Yang, N. Xie, B. Liu, Y. Zhang, Hierarchical spatial-temporal autocorrelation graph neural network for online wind turbine pitch system fault detection, Neurocomputing 586 (2024) 127574. doi:10.1016/j.neucom.2024.127574.
[62] J. Maldonado-Correa, J. Torres-Cabrera, S. Martín-Martínez, E. Artigao, E. Gómez-Lázaro, Wind turbine fault detection based on the transformer model using scada data, Engineering Failure Analysis 162 (2024) 108354. doi:10.1016/j.engfailanal.2024.108354.
[63] J. Liu, X. Wang, S. Wu, L. Wan, F. Xie, Wind turbine fault detection based on deep residual networks, Expert Systems with Applications 213 (2023) 119102. doi:10.1016/j.eswa.2022.119102.
[64] J. Zhan, C. Wu, C. Yang, Q. Miao, S. Wang, X. Ma, Condition monitoring of wind turbines based on spatial-temporal feature aggregation networks, Renewable Energy 200 (2022) 751–766. doi:10.1016/j.renene.2022.09.102.
[65] Y. Zhu, C. Zhu, J. Tan, C. Song, D. Chen, J. Zheng, Fault detection of offshore wind turbine gearboxes based on deep adaptive networks via considering spatio-temporal fusion, Renewable Energy 200 (2022) 1023–1036. doi:10.1016/j.renene.2022.10.018.
[66] P. Rizk, F. Rizk, S. S. Karganroudi, A. Ilinca, R. Younes, J. Khoder, Advanced wind turbine blade inspection with hyperspectral imaging and 3d convolutional neural networks for damage detection, Energy and AI 16 (2024) 100366. doi:10.1016/j.egyai.2024.100366.
[67] Y. Wu, X. Ma, A hybrid lstm-kld approach to condition monitoring of operational wind turbines, Renewable Energy 181 (2022) 554–566. doi:10.1016/j.renene.2021.09.067.
[68] G. Jiang, W. Fan, W. Li, L. Wang, Q. He, P. Xie, X. Li, Deepfedwt: A federated deep learning framework for fault detection of wind turbines, Measurement 199 (2022) 111529. doi:10.1016/j.measurement.2022.111529.
[69] J. Zhan, C. Wu, X. Ma, C. Yang, Q. Miao, S. Wang, Abnormal vibration detection of wind turbine based on temporal convolution network and multivariate coefficient of variation, Mechanical Systems and Signal Processing 174 (2022) 109082. doi:10.1016/j.ymssp.2022.109082.
[70] L. Qiao, Y. Zhang, Q. Wang, Fault detection in wind turbine generators using a meta-learning-based convolutional neural network, Mechanical Systems and Signal Processing 200 (2023) 110528. doi:10.1016/j.ymssp.2023.110528.
[71] X. Hang, X. Zhu, X. Gao, Y. Wang, L. Liu, Study on crack monitoring method of wind turbine blade based on ai model: Integration of classification, detection, segmentation and fault level evaluation, Renewable Energy 224 (2024) 120152. doi:10.1016/j.renene.2024.120152.
[72] C. Zhang, L. Zhang, Wind turbine pitch bearing fault detection with bayesian augmented temporal convolutional networks, Structural Health Monitoring 23 (2) (2024) 1089–1106. doi:10.1177/14759217231175886.
[73] M.-H. Wang, S.-D. Lu, C.-C. Hsieh, C.-C. Hung, Fault detection of wind turbine blades using multi-channel cnn, Sustainability 14 (3) (2022) 1781. doi:10.3390/su14031781.
[74] T. Wang, L. Yin, A hybrid 3dse-cnn-2dlstm model for compound fault detection of wind turbines, Expert Systems with Applications 242 (2024) 122776. doi:10.1016/j.eswa.2023.122776.
[75] S. Zare, M. Ayati, Simultaneous fault diagnosis of wind turbine using multichannel convolutional neural networks, ISA Transactions 108 (2021) 230–239. doi:10.1016/j.isatra.2020.08.021.
[76] P. Liang, B. Wang, G. Jiang, N. Li, L. Zhang, Unsupervised fault diagnosis of wind turbine bearing via a deep residual deformable convolution network based on subdomain adaptation under time-varying speeds, Engineering Applications of Artificial Intelligence 118 (2023) 105656. doi:10.1016/j.engappai.2022.105656.
[77] Z. Xu, X. Mei, X. Wang, M. Yue, J. Jin, Y. Yang, C. Li, Fault diagnosis of wind turbine bearing using a multi-scale convolutional neural network with bidirectional long short term memory and weighted majority voting for multi-sensors, Renewable Energy 182 (2022) 615–626. doi:10.1016/j.renene.2021.10.024.
[78] Y. Zhang, W. Liu, X. Wang, H. Gu, A novel wind turbine fault diagnosis method based on compressed sensing and dtl-cnn, Renewable Energy 194 (2022) 249–258. doi:10.1016/j.renene.2022.05.085.
[79] J. Lei, C. Liu, D. Jiang, Fault diagnosis of wind turbine based on long short-term memory networks, Renewable Energy 133 (2019) 422–432. doi:10.1016/j.renene.2018.10.031.
[80] Y. Zhang, K. Yu, Z. Lei, J. Ge, Y. Xu, Z. Li, Z. Ren, K. Feng, Integrated intelligent fault diagnosis approach of offshore wind turbine bearing based on information stream fusion and semi-supervised learning, Expert Systems with Applications 232 (2023) 120854. doi:10.1016/j.eswa.2023.120854.
[81] Y. Chang, J. Chen, C. Qu, T. Pan, Intelligent fault diagnosis of wind turbines via a deep learning network using parallel convolution layers with multi-scale kernels, Renewable Energy 153 (2020) 205–213. doi:10.1016/j.renene.2020.02.004.
[82] Y. Zhu, Y. Pei, A. Wang, B. Xie, Z. Qian, A partial domain adaptation scheme based on weighted adversarial nets with improved cbam for fault diagnosis of wind turbine gearbox, Engineering Applications of Artificial Intelligence 125 (2023) 106674. doi:10.1016/j.engappai.2023.106674.
[83] D. Wang, C. Cao, N. Chen, W. Pan, H. Li, X. Wang, A correlation-graph-cnn method for fault diagnosis of wind turbine based on state tracking and data driving model, Sustainable Energy Technologies and Assessments 56 (2023) 102995. doi:10.1016/j.seta.2022.102995.
[84] S. Qin, J. Tao, Z. Zhao, Fault diagnosis of wind turbine pitch system based on lstm with multi-channel attention mechanism, Energy Reports 10 (2023) 4087–4096. doi:10.1016/j.egyr.2023.10.076.
[85] K. Zhang, B. Tang, L. Deng, X. Liu, A hybrid attention improved resnet based fault diagnosis method of wind turbines gearbox, Measurement 179 (2021) 109491. doi:10.1016/j.measurement.2021.109491.
[86] K. Zhang, B. Tang, L. Deng, Q. Tan, H. Yu, A fault diagnosis method for wind turbines gearbox based on adaptive loss weighted meta-resnet under noisy labels, Mechanical Systems and Signal Processing 161 (2021) 107963. doi:10.1016/j.ymssp.2021.107963.
[87] Y. Li, W. Jiang, G. Zhang, L. Shu, Wind turbine fault diagnosis based on transfer learning and convolutional autoencoder with small-scale data, Renewable Energy 171 (2021) 103–115. doi:10.1016/j.renene.2021.01.143.
[88] X. Wen, Z. Xu, Wind turbine fault diagnosis based on relieff-pca and dnn, Expert Systems with Applications 178 (2021) 115016. doi:10.1016/j.eswa.2021.115016.
[89] T. Han, W. Xie, Z. Pei, Semi-supervised adversarial discriminative learning approach for intelligent fault diagnosis of wind turbine,

Information Sciences 648 (2023) 119496. doi:10.1016/j.ins.2023.119496.

[90] L. Saidi, J. B. Ali, E. Bechhoefer, M. Benbouzid, Wind turbine high-speed shaft bearings health prognosis through a spectral kurtosis-derived indices and svr, Applied Acoustics 120 (2017) 1–8. doi:10.1016/j.apacoust.2017.01.005.

[91] X. Li, W. Teng, D. Peng, T. Ma, X. Wu, Y. Liu, Feature fusion model based health indicator construction and self-constraint state-space estimator for remaining useful life prediction of bearings in wind turbines, Reliability Engineering & System Safety 233 (2023) 109124. doi:10.1016/j.ress.2023.109124.

[92] K. Boutrous, V. c. Puig, F. Nejjari, A set-based prognostics approach for wind turbine blade health monitoring, IFAC-PapersOnLine 55 (6) (2022) 402–407. doi:10.1016/j.ifacol.2022.07.162.

[93] A. Joshuva, V. Sugumaran, A lazy learning approach for condition monitoring of wind turbine blade using vibration signals and histogram features, Measurement 152 (2020) 107295. doi:10.1016/j.measurement.2019.107295.

[94] B. Chen, P. C. Matthews, P. J. Tavner, Wind turbine pitch faults prognosis using a-priori knowledge-based anfis, Expert Systems with Applications 40 (17) (2013) 6863–6876. doi:10.1016/j.eswa.2013.06.018.

[95] M. A. Djeziri, S. Benmoussa, R. Sanchez, Hybrid method for remaining useful life prediction in wind turbine systems, Renewable Energy 116 (2018) 173–187. doi:10.1016/j.renene.2017.05.020.

[96] J. Wang, Y. Liang, Y. Zheng, R. X. Gao, F. Zhang, An integrated fault diagnosis and prognosis approach for predictive maintenance of wind turbine bearing with limited samples, Renewable Energy 145 (2020) 642–650. doi:10.1016/j.renene.2019.06.103.

[97] M. Elforjani, Diagnosis and prognosis of real world wind turbine gears, Renewable Energy 147 (2020) 1676–1693. doi:10.1016/j.renene.2019.09.109.

[98] B. Merainani, S. Laddada, E. Bechhoefer, M. A. A. Chikh, D. Benazzouz, An integrated methodology for estimating the remaining useful life of high-speed wind turbine shaft bearings with limited samples, Renewable Energy 182 (2022) 1141–1151. doi:10.1016/j.renene.2021.10.062.

[99] F. d. N. Santos, P. D'Antuono, K. Robbelein, N. Noppe, W. Weijtjens, C. Devriendt, Long-term fatigue estimation on offshore wind turbines interface loads through loss function physics-guided learning of neural networks, Renewable Energy 205 (2023) 461–474. doi:10.1016/j.renene.2023.01.093.

[100] F. Wang, Y. Liu, Z. Zou, J. Jiang, Y. Xu, Z. Liu, Gwtsp: A multi-state prediction method for short-term wind turbines based on gat and gl, Procedia Computer Science 221 (2023) 963–970. doi:10.1016/j.procs.2023.08.075.

[101] L. Gong, Y. Chen, Machine learning-enhanced lot and wireless sensor networks for predictive analysis and maintenance in wind turbine systems, International Journal of Intelligent Networks 5 (2024) 133–144. doi:10.1016/j.ijin.2024.02.002.

[102] S. T. Kandukuri, A. Klausen, H. R. Karimi, K. G. Robbersmyr, A review of diagnostics and prognostics of low-speed machinery towards wind turbine farm-level health management, Renewable and Sustainable Energy Reviews 53 (2016) 697–708. doi:10.1016/j.rser.2015.08.061.

[103] J. G. Njiri, N. Beganovic, M. H. Do, D. Söffker, Consideration of lifetime and fatigue load in wind turbine control, Renewable Energy 131 (2019) 818–828. doi:10.1016/j.renene.2018.07.109.

[104] M. H. Do, D. Söffker, State-of-the-art in integrated prognostics and health management control for utility-scale wind turbines, Renewable and Sustainable Energy Reviews 145 (2021) 111102. doi:10.1016/j.rser.2021.111102.

[105] N. Beganovic, D. Söffker, Structural health management utilization for lifetime prognosis and advanced control strategy deployment of wind turbines: An overview and outlook concerning actual methods, tools, and obtained results, Renewable and Sustainable Energy Reviews 64 (2016) 68–83. doi:10.1016/j.rser.2016.05.083.

[106] I. S. Maria Schuld, F. Petruccione, An introduction to quantum machine learning, Contemporary Physics 56 (2) (2015) 172–185. arXiv:https://doi.org/10.1080/00107514.2014.964942, doi:10.1080/00107514.2014.964942.
URL https://doi.org/10.1080/00107514.2014.964942

[107] R. LaPierre, Introduction to quantum computing, Springer Nature, 2021.

[108] A. Steane, Quantum computing, Reports on Progress in Physics 61 (2) (1998) 117. doi:10.1088/0034-4885/61/2/002.
URL https://dx.doi.org/10.1088/0034-4885/61/2/002

[109] T. Proctor, K. Young, A. D. Baczewski, R. Blume-Kohout, Benchmarking quantum computers, Nature Reviews Physics (2025) 1–14doi:10.1038/s42254-024-00796-z.

[110] I. El Naqa, M. J. Murphy, What is machine learning?, Springer, 2015. doi:https://doi.org/10.1007/978-3-319-18305-3_1.

[111] P. Wittek, Quantum machine learning: what quantum computing means to data mining, Academic Press, 2014.

[112] D. A. Patterson, F. P. Brooks Jr, I. E. Sutherland, C. P. Thacker, Computer architecture, Elsevier Science, 2011.

[113] S. G. Philips, M. T. Mądzik, S. V. Amitonov, S. L. de Snoo, M. Russ, N. Kalhor, C. Volk, W. I. Lawrie, D. Brousse, L. Tryputen, et al., Universal control of a six-qubit quantum processor in silicon, Nature 609 (7929) (2022) 919–924.

[114] T. D. Ladd, F. Jelezko, R. Laflamme, Y. Nakamura, C. Monroe, J. L. O'Brien, Quantum computers, nature 464 (7285) (2010) 45–53. doi:https://doi.org/10.1038/nature08812.

[115] G. Chiribella, G. M. D'Ariano, P. Perinotti, Quantum circuit architecture, Physical review letters 101 (6) (2008) 060401.

[116] Y. S. Weinstein, M. Pravia, E. Fortunato, S. Lloyd, D. G. Cory, Implementation of the quantum fourier transform, Physical review letters 86 (9) (2001) 1889. doi:10.1103/PhysRevLett.86.1889.

[117] U. Dorner, R. Demkowicz-Dobrzanski, B. J. Smith, J. S. Lundeen, W. Wasilewski, K. Banaszek, I. A. Walmsley, Optimal quantum phase estimation, Physical review letters 102 (4) (2009) 040403. doi:10.1103/PhysRevLett.102.040403.

[118] P. W. Shor, Algorithms for quantum computation: discrete logarithms and factoring, in: Proceedings 35th annual symposium on foundations of computer science, IEEE, 1994, pp. 124–134. doi:10.1109/SFCS.1994.365700.

[119] L. K. Grover, A fast quantum mechanical algorithm for database search, in: Proceedings of the twenty-eighth annual ACM symposium on Theory of computing, 1996, pp. 212–219. doi:10.1145/237814.237866.

[120] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio, et al., Variational quantum algorithms, Nature Reviews Physics 3 (9) (2021) 625–644. doi:10.1038/s42254-021-00348-9.

[121] M. Nakahara, T. Ohmi, Quantum computing: from linear algebra to physical realizations, CRC press, 2008. doi:10.1201/9781420012293.

[122] A. W. Harrow, A. Hassidim, S. Lloyd, Quantum algorithm for linear systems of equations, Physical review letters 103 (15) (2009) 150502. doi:10.1103/PhysRevLett.103.150502.
[123] S. Lloyd, M. Mohseni, P. Rebentrost, Quantum principal component analysis, Nature Physics 10 (9) (2014) 631–633. doi:10.1038/nphys3029.
[124] T. Albash, D. A. Lidar, Adiabatic quantum computation, Reviews of Modern Physics 90 (1) (2018) 015002. doi:10.1103/RevModPhys.90.015002.
[125] S. Arunachalam, R. Maity, Quantum boosting, in: H. D. III, A. Singh (Eds.), Proceedings of the 37th International Conference on Machine Learning, Vol. 119 of Proceedings of Machine Learning Research, PMLR, 2020, pp. 377–387.
URL https://proceedings.mlr.press/v119/arunachalam20a.html
[126] S. Lomonaco, H. Brandt, Quantum Computation and Information: AMS Special Session Quantum Computation and Information, January 19-21, 2000, Washington, Contemporary mathematics - American Mathematical Society, American Mathematical Society, 2002.
URL https://books.google.co.uk/books?id=o2QbCAAAQBAJ
[127] T. M. Khan, A. Robles-Kelly, Machine learning: Quantum vs classical, IEEE Access 8 (2020) 219275–219294. doi:10.1109/ACCESS.2020.3041719.
[128] M. Weigold, J. Barzen, F. Leymann, M. Salm, Data encoding patterns for quantum computing, in: Proceedings of the 27th Conference on Pattern Languages of Programs, PLoP '20, The Hillside Group, USA, 2022, pp. 1–11.
[129] M. Schuld, R. Sweke, J. J. Meyer, Effect of data encoding on the expressive power of variational quantum-machine-learning models, Physical Review A 103 (3) (2021) 032430. doi:10.1103/PhysRevA.103.032430.
[130] D. Ranga, A. Rana, S. Prajapat, P. Kumar, K. Kumar, A. V. Vasilakos, Quantum machine learning: Exploring the role of data encoding techniques, challenges, and future directions, Mathematics 12 (21) (2024). doi:10.3390/math12213318.
URL https://www.mdpi.com/2227-7390/12/21/3318
[131] K. Mitarai, M. Negoro, M. Kitagawa, K. Fujii, Quantum circuit learning, Physical Review A 98 (3) (2018) 032309. doi:10.1103/PhysRevA.98.032309.
[132] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, S. Lloyd, Quantum machine learning, Nature 549 (7671) (2017) 195–202. doi:10.1038/nature23474.
[133] Y. Du, T. Huang, S. You, M.-H. Hsieh, D. Tao, Quantum circuit architecture search for variational quantum algorithms, npj Quantum Information 8 (1) (2022) 62. doi:10.1038/s41534-022-00570-y.
[134] D. Layden, G. Mazzola, R. V. Mishmash, M. Motta, P. Wocjan, J.-S. Kim, S. Sheldon, Quantum-enhanced markov chain monte carlo, Nature 619 (7969) (2023) 282–287.
[135] C. Zoufal, A. Lucchi, S. Woerner, Variational quantum boltzmann machines, Quantum Machine Intelligence 3 (1) (2021) 7. doi:10.1007/s42484-020-00033-7.
URL https://doi.org/10.1007/s42484-020-00033-7
[136] S. L. Wu, S. Sun, W. Guan, C. Zhou, J. Chan, C. L. Cheng, T. Pham, Y. Qian, A. Z. Wang, R. Zhang, M. Livny, J. Glick, P. K. Barkoutsos, S. Woerner, I. Tavernelli, F. Carminati, A. Di Meglio, A. C. Y. Li, J. Lykken, P. Spentzouris, S. Y.-C. Chen, S. Yoo, T.-C. Wei, Application of quantum machine learning using the quantum kernel algorithm on high energy physics analysis at the lhc, Physical Review Research 3 (3) (Sep. 2021). doi:10.1103/physrevresearch.3.033221.
URL http://dx.doi.org/10.1103/PhysRevResearch.3.033221
[137] Z. Zhao, J. K. Fitzsimons, J. F. Fitzsimons, Quantum-assisted gaussian process regression, Physical Review A 99 (5) (2019) 052331.
[138] Z. Laboudi, S. Chikhi, Comparison of genetic algorithm and quantum genetic algorithm., Int. Arab J. Inf. Technol. 9 (3) (2012) 243–249.
[139] Z. Xin-gang, L. Ji, M. Jin, Z. Ying, An improved quantum particle swarm optimization algorithm for environmental economic dispatch, Expert Systems with Applications 152 (2020) 113370.
[140] B.-Q. Chen, X.-F. Niu, Quantum neural network with improved quantum learning algorithm, International Journal of Theoretical Physics 59 (7) (2020) 1978 – 1991, cited by: 12. doi:10.1007/s10773-020-04470-9.
URL https://www.scopus.com/inward/record.uri?eid=2-s2.0-85084992265&doi=10.1007%2fs10773-020-04470-9&partnerID=40&md5=9fa4483e56f70205b498437983f36c8a
[141] S. Altares-López, A. Ribeiro, J. J. García-Ripoll, Automatic design of quantum feature maps, Quantum Science and Technology 6 (4) (2021) 045015.
[142] I. Cong, S. Choi, M. D. Lukin, Quantum convolutional neural networks, Nature Physics 15 (12) (2019) 1273–1278. doi:10.1038/s41567-019-0648-8.
URL https://doi.org/10.1038/s41567-019-0648-8
[143] J. Liu, K. H. Lim, K. L. Wood, W. Huang, C. Guo, H.-L. Huang, Hybrid quantum-classical convolutional neural networks, Science China Physics, Mechanics & Astronomy 64 (9) (2021) 290311. doi:10.1007/s11433-021-1734-3.
URL https://doi.org/10.1007/s11433-021-1734-3
[144] S. Y.-C. Chen, S. Yoo, Y.-L. L. Fang, Quantum long short-term memory, in: ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2022, pp. 8622–8626. doi:10.1109/ICASSP43922.2022.9747369.
[145] S. Lloyd, C. Weedbrook, Quantum generative adversarial learning, Phys. Rev. Lett. 121 (2018) 040502. doi:10.1103/PhysRevLett.121.040502.
URL https://link.aps.org/doi/10.1103/PhysRevLett.121.040502
[146] V. Dunjko, J. M. Taylor, H. J. Briegel, Advances in quantum reinforcement learning, in: 2017 IEEE international conference on systems, man, and cybernetics (SMC), IEEE, 2017, pp. 282–287.
[147] R. Acharya, D. A. Abanin, L. Aghababaie-Beni, I. Aleiner, T. I. Andersen, M. Ansmann, F. Arute, K. Arya, A. Asfaw, N. Astrakhantsev, J. Atalaya, R. Babbush, D. Bacon, B. Ballard, J. C. Bardin, J. Bausch, A. Bengtsson, A. Bilmes, S. Blackwell, S. Boixo, G. Bortoli, A. Bourassa, J. Bovaird, L. Brill, M. Broughton, D. A. Browne, B. Buchea, B. B. Buckley, D. A. Buell, T. Burger, B. Burkett, N. Bushnell, A. Cabrera, J. Campero, H.-S. Chang, Y. Chen, Z. Chen, B. Chiaro, D. Chik, C. Chou, J. Claes, A. Y. Cleland, J. Cogan, R. Collins, P. Conner, W. Courtney, A. L. Crook, B. Curtin, S. Das, A. Davies, L. De Lorenzo, D. M. Debroy, S. Demura, M. Devoret, A. Di Paolo, P. Donohoe,

I. Drozdov, A. Dunsworth, C. Earle, T. Edlich, A. Eickbusch, A. M. Elbag, M. Elzouka, C. Erickson, L. Faoro, E. Farhi, V. S. Ferreira, L. F. Burgos, E. Forati, A. G. Fowler, B. Foxen, S. Ganjam, G. Garcia, R. Gasca, E. Genois, W. Giang, C. Gidney, D. Gilboa, R. Gosula, A. G. Dau, D. Graumann, A. Greene, J. A. Gross, S. Habegger, J. Hall, M. C. Hamilton, M. Hansen, M. P. Harrigan, S. D. Harrington, F. J. H. Heras, S. Heslin, P. Heu, O. Higgott, G. Hill, J. Hilton, G. Holland, S. Hong, H.-Y. Huang, A. Huff, W. J. Huggins, L. B. Ioffe, S. V. Isakov, J. Iveland, E. Jeffrey, Z. Jiang, C. Jones, S. Jordan, C. Joshi, P. Juhas, D. Kafri, H. Kang, A. H. Karamlou, K. Kechedzhi, J. Kelly, T. Khaire, T. Khattar, M. Khezri, S. Kim, P. V. Klimov, A. R. Klots, B. Kobrin, P. Kohli, A. N. Korotkov, F. Kostritsa, R. Kothari, B. Kozlovskii, J. M. Kreikebaum, V. D. Kurilovich, N. Lacroix, D. Landhuis, T. Lange-Dei, B. W. Langley, P. Laptev, K.-M. Lau, L. Le Guevel, J. Ledford, J. Lee, K. Lee, Y. D. Lensky, S. Leon, B. J. Lester, W. Y. Li, Y. Li, A. T. Lill, W. Liu, W. P. Livingston, A. Locharla, E. Lucero, D. Lundahl, A. Lunt, S. Madhuk, F. D. Malone, A. Maloney, S. Mandrà, J. Manyika, L. S. Martin, O. Martin, S. Martin, C. Maxfield, J. R. McClean, M. McEwen, S. Meeks, A. Megrant, X. Mi, K. C. Miao, A. Mieszala, R. Molavi, S. Molina, S. Montazeri, A. Morvan, R. Movassagh, W. Mruczkiewicz, O. Naaman, M. Neeley, C. Neill, A. Nersisyan, H. Neven, M. Newman, J. H. Ng, A. Nguyen, M. Nguyen, C.-H. Ni, M. Y. Niu, T. E. O'Brien, W. D. Oliver, A. Opremcak, K. Ottosson, A. Petukhov, A. Pizzuto, J. Platt, R. Potter, O. Pritchard, L. P. Pryadko, C. Quintana, G. Ramachandran, M. J. Reagor, J. Redding, D. M. Rhodes, G. Roberts, E. Rosenberg, E. Rosenfeld, P. Roushan, N. C. Rubin, N. Saei, D. Sank, K. Sankaragomathi, K. J. Satzinger, H. F. Schurkus, C. Schuster, A. W. Senior, M. J. Shearn, A. Shorter, N. Shutty, V. Shvarts, S. Singh, V. Sivak, J. Skruzny, S. Small, V. Smelyanskiy, W. C. Smith, R. D. Somma, S. Springer, G. Sterling, D. Strain, J. Suchard, A. Szasz, A. Sztein, D. Thor, A. Torres, M. M. Torunbalci, A. Vaishnav, J. Vargas, S. Vdovichev, G. Vidal, B. Villalonga, C. V. Heidweiller, S. Waltman, S. X. Wang, B. Ware, K. Weber, T. Weidel, T. White, K. Wong, B. W. K. Woo, C. Xing, Z. J. Yao, P. Yeh, B. Ying, J. Yoo, N. Yosri, G. Young, A. Zalcman, Y. Zhang, N. Zhu, N. Zobrist, G. Q. AI, Collaborators, Quantum error correction below the surface code threshold, Nature (2024). doi:10.1038/s41586-024-08449-y.
URL https://doi.org/10.1038/s41586-024-08449-y

[148] S. J. Devitt, W. J. Munro, K. Nemoto, Quantum error correction for beginners, Reports on Progress in Physics 76 (7) (2013) 076001. doi:10.1088/0034-4885/76/7/076001.

[149] N. P. de Leon, K. M. Itoh, D. Kim, K. K. Mehta, T. E. Northup, H. Paik, B. S. Palmer, N. Samarth, S. Sangtawesin, D. W. Steuerman, Materials challenges and opportunities for quantum computing hardware, Science 372 (6539) (2021) eabb2823. arXiv:https://www.science.org/doi/pdf/10.1126/science.abb2823, doi:10.1126/science.abb2823.
URL https://www.science.org/doi/abs/10.1126/science.abb2823

[150] S. Endo, Z. Cai, S. C. Benjamin, X. Yuan, Hybrid quantum-classical algorithms and quantum error mitigation, Journal of the Physical Society of Japan 90 (3) (2021) 032001.

[151] K. Bertels, A. Sarkar, I. Ashraf, Quantum computing—from nisq to pisq, IEEE Micro 41 (5) (2021) 24–32. doi:10.1109/MM.2021.3099195.

[152] M. AbuGhanem, Ibm quantum computers: Evolution, performance, and future directions, arXiv preprint arXiv:2410.00916 (2024). doi:https://doi.org/10.48550/arXiv.2410.00916.

[153] S. S. Rosales-Alvarado, O. Montiel, U. Orozco-Rosas, J. J. Tapia, Developing a quantum genetic algorithm in matlab using a quantum device on aws, in: New Directions on Hybrid Intelligent Systems Based on Neural Networks, Fuzzy Logic, and Optimization Algorithms, Springer, 2024, pp. 111–127. doi:https://doi.org/10.1007/978-3-031-53713-4_10.

[154] S. L. Wu, J. Chan, W. Guan, S. Sun, A. Wang, C. Zhou, M. Livny, F. Carminati, A. D. Meglio, A. C. Y. Li, J. Lykken, P. Spentzouris, S. Y.-C. Chen, S. Yoo, T.-C. Wei, Application of quantum machine learning using the quantum variational classifier method to high energy physics analysis at the lhc on ibm quantum computer simulator and hardware with 10 qubits, Journal of Physics G: Nuclear and Particle Physics 48 (12) (2021) 125003. doi:10.1088/1361-6471/ac1391.
URL https://dx.doi.org/10.1088/1361-6471/ac1391

[155] S. L. Wu, S. Sun, W. Guan, C. Zhou, J. Chan, C. L. Cheng, T. Pham, Y. Qian, A. Z. Wang, R. Zhang, M. Livny, J. Glick, P. K. Barkoutsos, S. Woerner, I. Tavernelli, F. Carminati, A. Di Meglio, A. C. Y. Li, J. Lykken, P. Spentzouris, S. Y.-C. Chen, S. Yoo, T.-C. Wei, Application of quantum machine learning using the quantum kernel algorithm on high energy physics analysis at the lhc, Phys. Rev. Res. 3 (2021) 033221. doi:10.1103/PhysRevResearch.3.033221.
URL https://link.aps.org/doi/10.1103/PhysRevResearch.3.033221

[156] W. Guan, G. Perdue, A. Pesah, M. Schuld, K. Terashi, S. Vallecorsa, J.-R. Vlimant, Quantum machine learning in high energy physics, Machine Learning: Science and Technology 2 (1) (2021) 011003. doi:10.1088/2632-2153/abc17d.
URL https://dx.doi.org/10.1088/2632-2153/abc17d

[157] K. Meichanetzidis, S. Gogioso, G. De Felice, N. Chiappori, A. Toumi, B. Coecke, Quantum natural language processing on near-term quantum computers, arXiv preprint arXiv:2005.04147 (2020).

[158] M. Abbaszade, V. Salari, S. S. Mousavi, M. Zomorodi, X. Zhou, Application of quantum natural language processing for language translation, IEEE Access 9 (2021) 130434–130448. doi:10.1109/ACCESS.2021.3108768.

[159] S. Wu, J. Li, P. Zhang, Y. Zhang, Natural language processing meets quantum physics: A survey and categorization, in: M.-F. Moens, X. Huang, L. Specia, S. W.-t. Yih (Eds.), Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Online and Punta Cana, Dominican Republic, 2021, pp. 3172–3182. doi:10.18653/v1/2021.emnlp-main.254.
URL https://aclanthology.org/2021.emnlp-main.254

[160] S. Pandey, N. J. Basisth, T. Sachan, N. Kumari, P. Pakray, Quantum machine learning for natural language processing application, Physica A: Statistical Mechanics and its Applications 627 (2023) 129123. doi:https://doi.org/10.1016/j.physa.2023.129123.
URL https://www.sciencedirect.com/science/article/pii/S0378437123006787

[161] S. Khurana, et al., Quantum machine learning: Unraveling a new paradigm in computational intelligence, Quantum 74 (2024) 1.

[162] J. Liu, M. Liu, J.-P. Liu, Z. Ye, Y. Wang, Y. Alexeev, J. Eisert, L. Jiang, Towards provably efficient quantum algorithms for large-scale machine-learning models, Nature Communications 15 (1) (2024) 434. doi:10.1038/s41467-023-43957-x.

[163] Z. Liang, J. Cheng, R. Yang, H. Ren, Z. Song, D. Wu, X. Qian, T. Li, Y. Shi, Unleashing the potential of llms for quantum computing: A study in quantum architecture design, arXiv preprint arXiv:2307.08191 (2023).

[164] M. Akrom, S. Rustad, H. K. Dipojono, Development of quantum machine learning to evaluate the corrosion inhibition capability of pyrimidine compounds, Materials Today Communications 39 (2024) 108758. doi:https://doi.org/10.1016/j.mtcomm.2024.108758.
URL https://www.sciencedirect.com/science/article/pii/S2352492824007396

[165] M. Akrom, S. Rustad, H. K. Dipojono, Variational quantum circuit-based quantum machine learning approach for predicting corrosion inhibition efficiency of pyridine-quinoline compounds, Materials Today Quantum 2 (2024) 100007. doi:https://doi.org/10.1016/j.mtquan.2024.100007.
URL https://www.sciencedirect.com/science/article/pii/S2950257824000076

[166] M. P. Lourenço, L. B. Herrera, J. Hostaš, P. Calaminici, A. M. Köster, A. Tchagang, D. R. Salahub, Qmlmaterial-a quantum machine learning software for material design and discovery, Journal of Chemical Theory and Computation 19 (17) (2023) 5999–6010, pMID: 37581570. arXiv:https://doi.org/10.1021/acs.jctc.3c00566, doi:10.1021/acs.jctc.3c00566.
URL https://doi.org/10.1021/acs.jctc.3c00566

[167] O. A. von Lilienfeld, Quantum machine learning in chemical compound space, Angewandte Chemie International Edition 57 (16) (2018) 4164–4169. arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/anie.201709686, doi:https://doi.org/10.1002/anie.201709686.
URL https://onlinelibrary.wiley.com/doi/abs/10.1002/anie.201709686

[168] K. Atz, C. Isert, M. N. Böcker, J. Jiménez-Luna, G. Schneider, δ-quantum machine-learning for medicinal chemistry, Physical Chemistry Chemical Physics 24 (18) (2022) 10775–10783.

[169] B. Huang, N. O. Symonds, O. A. von Lilienfeld, Quantum machine learning in chemistry and materials, Handbook of Materials Modeling: Methods: Theory and Modeling (2020) 1883–1909.

[170] J. Geraci, Shattering cancer with quantum machine learning: A preview, Patterns 2 (6) (2021) 100281. doi:https://doi.org/10.1016/j.patter.2021.100281.
URL https://www.sciencedirect.com/science/article/pii/S2666389921001148

[171] N.-R. Zhou, T.-F. Zhang, X.-W. Xie, J.-Y. Wu, Hybrid quantum–classical generative adversarial networks for image generation via learning discrete distribution, Signal Processing: Image Communication 110 (2023) 116891. doi:https://doi.org/10.1016/j.image.2022.116891.
URL https://www.sciencedirect.com/science/article/pii/S0923596522001709

[172] P. Hunt, L. Hosseini-Gerami, T. Chrien, M. Segall, P125 - predicting pka using a combination of quantum and machine learning methods, Drug Metabolism and Pharmacokinetics 35 (1, Supplement) (2020) S60, aBSTRACTS FROM THE 12TH INTERNATIONAL ISSX MEETING 28–31 July 2019. doi:https://doi.org/10.1016/j.dmpk.2020.04.126.
URL https://www.sciencedirect.com/science/article/pii/S1347436720301518

[173] S. A. Razzak, O. H. Alkhalaf, S. M. Rahman, J. Zhu, Quantum machine learning – a novel approach for hydrodynamics analysis and modeling of liquid-solid circulating fluidized bed risers, Chemical Engineering Science 282 (2023) 119310. doi:https://doi.org/10.1016/j.ces.2023.119310.
URL https://www.sciencedirect.com/science/article/pii/S0009250923008667

[174] Y.-B. Sheng, L. Zhou, Distributed secure quantum machine learning, Science Bulletin 62 (14) (2017) 1025–1029. doi:https://doi.org/10.1016/j.scib.2017.06.007.
URL https://www.sciencedirect.com/science/article/pii/S2095927317303250

[175] I. D. Lins, L. M. M. Araújo, C. B. S. Maior, P. M. da Silva Ramos, M. J. das Chagas Moura, A. J. Ferreira-Martins, R. Chaves, A. Canabarro, Quantum machine learning for drowsiness detection with eeg signals, Process Safety and Environmental Protection 186 (2024) 1197–1213. doi:https://doi.org/10.1016/j.psep.2024.04.032.
URL https://www.sciencedirect.com/science/article/pii/S0957582024003847

[176] D. Peral-García, J. Cruz-Benito, F. J. García-Peñalvo, Systematic literature review: Quantum machine learning and its applications, Computer Science Review 51 (2024) 100619. doi:https://doi.org/10.1016/j.cosrev.2024.100619.
URL https://www.sciencedirect.com/science/article/pii/S1574013724000030

[177] A. Jadhav, A. Rasool, M. Gyanchandani, Quantum machine learning: Scope for real-world problems, Procedia Computer Science 218 (2023) 2612–2625, international Conference on Machine Learning and Data Engineering. doi:https://doi.org/10.1016/j.procs.2023.01.235.
URL https://www.sciencedirect.com/science/article/pii/S1877050923002351

[178] R. Kharsa, A. Bouridane, A. Amira, Advances in quantum machine learning and deep learning for image classification: A survey, Neurocomputing 560 (2023) 126843. doi:https://doi.org/10.1016/j.neucom.2023.126843.
URL https://www.sciencedirect.com/science/article/pii/S0925231223009669

[179] L. Wei, H. Liu, J. Xu, L. Shi, Z. Shan, B. Zhao, Y. Gao, Quantum machine learning in medical image analysis: A survey, Neurocomputing 525 (2023) 42–53. doi:https://doi.org/10.1016/j.neucom.2023.01.049.
URL https://www.sciencedirect.com/science/article/pii/S0925231223000589

[180] S. M, Machine learning and quantum computing for 5g/6g communication networks - a survey, International Journal of Intelligent Networks 3 (2022) 197–203. doi:https://doi.org/10.1016/j.ijin.2022.11.004.
URL https://www.sciencedirect.com/science/article/pii/S2666603022000240

[181] B. Duan, J. Yuan, C.-H. Yu, J. Huang, C.-Y. Hsieh, A survey on hhl algorithm: From theory to application in quantum machine learning, Physics Letters A 384 (24) (2020) 126595. doi:https://doi.org/10.1016/j.physleta.2020.126595.
URL https://www.sciencedirect.com/science/article/pii/S037596012030462X

[182] E. H. Houssein, Z. Abohashima, M. Elhoseny, W. M. Mohamed, Machine learning in the quantum realm: The state-of-the-art, challenges, and future vision, Expert Systems with Applications 194 (2022) 116512. doi:https://doi.org/10.1016/j.eswa.2022.116512.
URL https://www.sciencedirect.com/science/article/pii/S0957417422000136

[183] T. Rezaei, A. Javadi, Environmental impact assessment of ocean energy converters using quantum machine learning, Journal of Environmental Management 362 (2024) 121275. doi:https://doi.org/10.1016/j.jenvman.2024.121275.

URL https://www.sciencedirect.com/science/article/pii/S0301479724012611
[184] G. Uehara, S. Rao, M. Dobson, C. Tepedelenlioglu, A. Spanias, Quantum neural network parameter estimation for photovoltaic fault detection, in: 2021 12th International Conference on Information, Intelligence, Systems & Applications (IISA), 2021, pp. 1–7. doi:10.1109/IISA52424.2021.9555558.
[185] M. M. Sushmit, I. M. Mahbubul, Forecasting solar irradiance with hybrid classical–quantum models: A comprehensive evaluation of deep learning and quantum-enhanced techniques, Energy Conversion and Management 294 (2023) 117555. doi:https://doi.org/10.1016/j.enconman.2023.117555.
URL https://www.sciencedirect.com/science/article/pii/S0196890423009019
[186] Y. Zhou, P. Zhang, Noise-resilient quantum machine learning for stability assessment of power systems, IEEE Transactions on Power Systems 38 (1) (2023) 475–487. doi:10.1109/TPWRS.2022.3160384.
[187] D. Liu, Y. Wu, Y. Kang, L. Yin, X. Ji, X. Cao, C. Li, Multi-agent quantum-inspired deep reinforcement learning for real-time distributed generation control of 100% renewable energy systems, Engineering Applications of Artificial Intelligence 119 (2023) 105787. doi:https://doi.org/10.1016/j.engappai.2022.105787.
URL https://www.sciencedirect.com/science/article/pii/S0952197622007771
[188] A. Ajagekar, F. You, Quantum computing assisted deep learning for fault detection and diagnosis in industrial process systems, Computers & Chemical Engineering 143 (2020) 107119. doi:10.1016/j.compchemeng.2020.107119.
[189] A. Ajagekar, F. You, Quantum computing based hybrid deep learning for fault diagnosis in electrical power systems, Applied Energy 303 (2021) 117628. doi:10.1016/j.apenergy.2021.117628.
[190] Y. Zhou, Z. Tang, N. Nikmehr, P. Babahajiani, F. Feng, T.-C. Wei, H. Zheng, P. Zhang, Quantum computing in power systems, iEnergy 1 (2) (2022) 170–187. doi:10.23919/IEN.2022.0021.
[191] A. Safari, A. A. Ghavifekr, Quantum technology & quantum neural networks in smart grids control: Premier perspectives, in: 2022 8th International Conference on Control, Instrumentation and Automation (ICCIA), 2022, pp. 1–6. doi:10.1109/ICCIA54998.2022.9737203.
[192] S. Golestan, M. Habibi, S. Mousazadeh Mousavi, J. Guerrero, J. Vasquez, Quantum computation in power systems: An overview of recent advances, Energy Reports 9 (2023) 584–596. doi:https://doi.org/10.1016/j.egyr.2022.11.185.
URL https://www.sciencedirect.com/science/article/pii/S2352484722025720
[193] B. Jiao, Z. W. Gao, The fault diagnosis of wind turbine gearbox based on qga—lssvm, Applied Mechanics and Materials 543 (2014) 950–955. doi:https://doi.org/10.4028/www.scientific.net/AMM.543-547.950.
[194] C. B. S. Maior, L. M. M. Araújo, I. D. Lins, M. D. C. Moura, E. L. Droguett, Prognostics and health management of rotating machinery via quantum machine learning, IEEE Access 11 (2023) 25132–25151. doi:10.1109/ACCESS.2023.3255417.
[195] V. Yaghoubi, Quantum machine learning for structural health monitoring, Journal of Physics: Conference Series 2647 (18) (2024) 182013. doi:10.1088/1742-6596/2647/18/182013.
URL https://dx.doi.org/10.1088/1742-6596/2647/18/182013
[196] C. Correa-Jullian, S. Cofre-Martel, G. San Martin, E. Lopez Droguett, G. De Novaes Pires Leite, A. Costa, Exploring quantum machine learning and feature reduction techniques for wind turbine pitch fault detection, Energies 15 (8) (2022) 2792. doi:10.3390/en15082792.
[197] S. M. GBASHI, O. O. OLATUNJI, P. A. ADEDEJI, N. MADUSHELE, Hybrid quantum convolutional neural network for defect detection in a wind turbine gearbox, in: 2024 IEEE PES/IAS PowerAfrica, 2024, pp. 01–06. doi:10.1109/PowerAfrica61624.2024.10759407.
[198] S. Ganjefar, A. A. Ghassemi, M. M. Ahmadi, Improving efficiency of two-type maximum power point tracking methods of tip-speed ratio and optimum torque in wind turbine system using a quantum neural network, Energy 67 (2014) 444–453. doi:https://doi.org/10.1016/j.energy.2014.02.023.
URL https://www.sciencedirect.com/science/article/pii/S0360544214001571
[199] K. Han, T. Huang, L. Yin, Quantum parallel multi-layer monte carlo optimization algorithm for controller parameters optimization of doubly-fed induction generator-based wind turbines, Applied Soft Computing 112 (2021) 107813. doi:https://doi.org/10.1016/j.asoc.2021.107813.
URL https://www.sciencedirect.com/science/article/pii/S1568494621007341
[200] L. Yin, L. Chen, D. Liu, X. Huang, F. Gao, Quantum deep reinforcement learning for rotor side converter control of double-fed induction generator-based wind turbines, Engineering Applications of Artificial Intelligence 106 (2021) 104451. doi:10.1016/j.engappai.2021.104451.
[201] L. Sagbansua, F. Balo, Decision making model development in increasing wind farm energy efficiency, Renewable Energy 109 (2017) 354–362. doi:https://doi.org/10.1016/j.renene.2017.03.045.
[202] Y. Yu, J. Wang, Z. Liu, W. Zhao, A combined forecasting strategy for the improvement of operational efficiency in wind farm, Journal of Renewable and Sustainable Energy 13 (6) (2021). doi:https://doi.org/10.1063/5.0065937.
[203] X. Ju, F. Liu, Wind farm layout optimization using self-informed genetic algorithm with information guided exploitation, Applied Energy 248 (2019) 429–445. doi:https://doi.org/10.1016/j.apenergy.2019.04.084.
URL https://www.sciencedirect.com/science/article/pii/S0306261919307482
[204] H. Kagemoto, Possible application of quantum computing in the field of ocean engineering: optimization of an offshore wind farm layout with the ising model, Journal of Ocean Engineering and Marine Energy (2024) 1–10doi:https://doi.org/10.1007/s40722-024-00345-x.
[205] Y.-Y. Hong, C. L. P. P. Rioflorido, W. Zhang, Hybrid deep learning and quantum-inspired neural network for day-ahead spatiotemporal wind speed forecasting, Expert Systems with Applications 241 (2024) 122645. doi:https://doi.org/10.1016/j.eswa.2023.122645.
URL https://www.sciencedirect.com/science/article/pii/S0957417423031470
[206] Z. Kou, T. Liu, J. Zhao, Generation prediction of ultra-short-term wind farm based on quantum genetic algorithm and fuzzy neural network, in: 2020 39th Chinese Control Conference (CCC), 2020, pp. 4318–4321. doi:10.23919/CCC50068.2020.9188568.
[207] W. G. Jency, J. E. Judith, Homogenized point mutual information and deep quantum reinforced wind power prediction, International Transactions on Electrical Energy Systems 2022 (1) (2022) 3686786. arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1155/2022/3686786, doi:https://doi.org/10.1155/2022/3686786.

URL https://onlinelibrary.wiley.com/doi/abs/10.1155/2022/3686786
[208] Y. Zhou, E. M. Stoudenmire, X. Waintal, What limits the simulation of quantum computers?, Physical Review X 10 (4) (2020) 041038. doi:10.1103/PhysRevX.10.041038.
[209] J. Gambetta, Ibm quantum system two: the era of quantum utility is here, accessed: 2024-12-12 (December 2023).
URL https://www.ibm.com/quantum/blog/quantum-roadmap-2033
[210] H. Neven, Meet willow, our state-of-the-art quantum chip, accessed: 2024-12-12 (December 2024).
URL https://blog.google/technology/research/google-willow-quantum-chip/
[211] E. Pelofske, A. Bärtschi, S. Eidenbenz, Quantum volume in practice: What users can expect from nisq devices, IEEE Transactions on Quantum Engineering 3 (2022) 1–19. doi:10.1109/TQE.2022.3184764.
[212] E. F. Combarro, S. González-Castillo, A. Di Meglio, A Practical Guide to Quantum Machine Learning and Quantum Optimization: Hands-on Approach to Modern Quantum Algorithms, Packt Publishing Ltd, 2023.
[213] A. Luongo, Quantum algorithms for data analysis (2020).
[214] H.-L. Huang, X.-Y. Xu, C. Guo, G. Tian, S.-J. Wei, X. Sun, W.-S. Bao, G.-L. Long, Near-term quantum computing techniques: Variational quantum algorithms, error mitigation, circuit compilation, benchmarking and classical simulation, Science China Physics, Mechanics & Astronomy 66 (5) (2023) 250302.
[215] N. Roshan Ahmed, S. Shridevi, Deepket- quantum space-efficient word embedding layer for steganalysis, in: 2024 3rd International Conference on Artificial Intelligence For Internet of Things (AIIoT), 2024, pp. 1–6. doi:10.1109/AIIoT58432.2024.10574718.
[216] A. Panahi, S. Saeedi, T. Arodz, word2ket: Space-efficient word embeddings inspired by quantum entanglement, arXiv preprint arXiv:1911.04975 (2019).
[217] M. Amy, D. Maslov, M. Mosca, M. Roetteler, A meet-in-the-middle algorithm for fast synthesis of depth-optimal quantum circuits, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 32 (6) (2013) 818–830. doi:10.1109/TCAD.2013.2244643.