1. INTRODUCTION

A. IMPORTANCE OF WIND ENERGY FORECASTING

The 2030 Agenda for Sustainable Development, popularly called the ‘17 Sustainable Development Goals’, officially came into force on January 1, 2016. Goal 7.2 concerns the importance of increasing renewable energy in the global energy mix. Over the years, the use of wind energy has grown to supply almost 5% of global electricity . However, owing to its uncontrollable and intermittent nature, integrating wind turbines into energy systems poses stability and energy management problems. The volatility of wind power generation must be considered to ensure that the spinning reserve of power systems suffices for unit commitment (UC).

To take uncertain wind speeds into account, various forecasting methods are continuously being used and improved. Currently, the five classes of methods are as follows. (i) The persistence method assumes that the wind speed at time‘t+1t’ equals that at time ‘t’. (ii) The physical approach uses weather observations and a mathematical computer model of the atmosphere to generate forecasts. (iii) Statistical approaches, like autoregressive integrated moving average (ARIMA), estimate statistical relationships among input data. (iv) Artificial intelligence (AI) does not require predefined mathematical models. Examples include random forest (RF) , support vector machines , support vector regression (SVR) , extreme gradient boost (XG Boost) regression , nonlinear autoregressive neural networks (NAR) , and deep neural networks (DNN). The several types of DNN include autoencoders , the deep belief network (DBN) , the deep Boltzmann machine, the recurrent neural network (RNN) , long short-term memory (LSTM) , and the convolutional neural network (CNN) . (v) Hybrid structures combine two or more of the aforementioned and provide favorable outcomes by combining the advantages of two models. Some recent works with outstanding results have involved a combination of CNN and Light GBM , and a novel deep convolutional recurrent network to forecast wind power .

B. RISE OF QUANTUM MACHINE LEARNING

Recently, the field of quantum computing has been attracting increasing attention as research has demonstrated unparalleled quantum advantages over classical computing. These have led, in particular, to increased interest in Quantum Machine Learning (QML). Initially, the development of QML was mainly motivated by a desire to investigate quantum algorithms to accelerate classical training processes . This paved the way to quantum equivalents of classical machine learning methods, such as (i) the Quantum Principal Component Analysis (QPCA) , in which a 4-qubit nuclear magnetic resonance (NMR) quantum processor was trained, resulting in the accurate recognition of all test images, (ii) the Quantum Support Vector Machine (QSVM) , in which IBM superconducting quantum computers were used to show that QSVM outperforms classical SVM for some datasets. Subsequently, with the continued rise of Noisy Intermediate-Scale Quantum (NISQ) processors, the field shifted towards Quantum Neural Networks (QNNs), which are the quantum versions of deep neural networks. In QNNs, parameterized quantum circuits (PQCs) or variational circuits act as neurons with parameters that are adapted to minimize the objective loss function. Killoran et al. used the Strawberry Fields quantum simulator to perform binary classification using QNN. Results show a receiver operating characteristic (ROC) curve with an area of 0.945, opposed to an ideal value of 1. QNNs have also been applied to other classification tasks like the popular MNIST database, and to regression problems especially in the field of finance. Pistoia et al. , used Google's Cirq to simulate PQCs, and demonstrated that it outperformed classical BILSTM neural networks whenever the noise coefficient was high, and was comparable otherwise.

C. HYBRID CLASSICAL-QUANTUM ALGORITHMS

Noisy quantum computers, with over 100 qub its, have recently been developed and shown to perform tasks better than current supercomputers . While the present is an exciting time to investigate and explore quantum algorithms and other applications, noiseless quantum computers with thousands of qubits are required in order to fully exploit the advantages of major algorithms like Shor's algorithm and Grover's algorithm. Hence, researchers are leaning toward hybrid classical-quantum algorithms as applications of quantum computing to machine learning, with the general idea of combining quantum and classical computers. Endo et al. used IBM's superconducting quantum computer to review the results for hybrid quantum-classical algorithms and quantum error mitigation techniques, and determined that future work on error mitigation would be extremely helpful since the use of NISQ devices is limited by large errors.

D. LIMITATIONS OF EXISTING METHODS

The aforementioned methods have one or more limitations with regard to wind speed forecasting and general system structure as follows.

Physical methods use large mathematical models and require meteorological data on humidity, terrain structure, pressure, and other variables, to obtain accurate results. They are very costly and complicated and are generally not used
for short-term forecasting.

Persistence and statistical methods yield accurate results but their accuracies quickly decline as the forecasting horizon increases .

Many machine learning models are used for forecasting and may encounter difficulties when used with data with large standard deviations . Autoencoders and DBN require large amount of clean data to generate useful results .

Solving the model selection problem for (quantum) deep learning is time-consuming because the structure parameters and hyper parameters are obtained by trial-anderror thus, the robustness of (quantum) deep learning models cannot be guaranteed.

E. MOTIVATIONS OF THIS WORK

Overcoming the above limitations would improve the accuracy of the predictions. This paper proposes a novel method, combining a deep learning LSTM model and QNN model, for 24h ahead wind speed forecasting using historical wind speeds at seven sites (in Taiwan, China, South Korea, and the Philippines). The motivation behind this combination is rooted in the potential to harness quantum computing capabilities to enhance and potentially revolutionize machine learning and neural network-based tasks. There are potential advantages associated with QNNs: (i) Quantum parallelism: QNN can process multiple computations simultaneously through superposition, allowing QNNs to potentially evaluate multiple input states in parallel, making QNNs attractive for certain machine learning tasks. (ii) Quantum feature mapping: quantum feature mapping allows for efficient encoding of classical data into quantum states, providing a more expressive representation of data. (iii) Quantum entanglement: Quantum entanglement enables strong correlations among qubits, which can be exploited in QNNs to capture complex relationships between features in the data. On the other hand, LSTM's ability to capture long-term dependencies, handle sequential data efficiently, and mitigate the vanishing gradient problem makes it a powerful choice for a wide range of sequential data processing tasks in machine learning. Based on these reasons, the proposed model integrates LSTM with QNN.

F. NOVELTIES AND CONTRIBUTIONS

The novelties of this paper are as follows:

(1) A hybrid classical and quantum model is proposed. The LSTM is augmented by a deep QNN, allowing for the seamless fusion of classical and quantum neural networks. This fusion facilitates the transfer of knowledge gained from the classical layers to the quantum layers, empowering the QNN to refine predictions through the principles of the quantum mechanism, resulting in highly accurate 24-hour ahead wind speed forecasts. Consequently, the limitations mentioned earlier (1) and (2) can be effectively overcome.

(2) The hybrid classical and quantum model leverages the complementary learning capabilities of classical and quantum layers. LSTM is known for its proficiency in handling time-series data, while the QNN extends the learning capacity through quantum entanglement. This unique combination enables the network to recognize more complex patterns, which could be challenging for classical models to achieve alone.

(3) Originally, QML was applied mainly to classification problems. This paper aims to extend its application to regression problems, specifically wind speed forecasting.

The contributions of this paper are as follows.

(1) Historical wind speed data from the other ancillary sites are used to forecast wind speeds at a target location in Taiwan. Spearman correlation analysis is used to determine the time lags to be inputted to the LSTM. This aims to overcome limitations (3) and (4) listed above.

(2) The structure hyper parameters of the LSTM and QNN are determined without trial-and-error but using a robust design, specifically, the Taguchi method. The LSTM hyper parameters, considered as design factors, are the number of neurons, number of hidden LSTM, and dropout rate. The QNN hyper parameters are the type of embedding and the circuit depth. This aims to improve limitation (5).

(3) The hybrid LSTM-QNN offers a practical solution for wind speed forecasting, as it harnesses quantum advantages without necessitating fully quantum algorithms. Its seamless integration with the existing classical infrastructure makes it more feasible to implement in real-world scenarios. This contribution opens up new possibilities for leveraging cutting-edge technologies to optimize renewable energy resources and decision-making processes.

2. ANALYSIS

Existing system

Killoran et al. [20] used the Strawberry Fields quantum simulator to perform binary

classification using QNN. Results show a receiver operating characteristic (ROC) curve with an area of 0.945, opposed to an ideal value of 1. QNNs have also been applied to other classification tasks like the popular MNIST database, and to regression problems especially in the field of finance. Pistoia et al. [21], used Google's Cirq to simulate PQCs, and demonstrated that it outperformed classical BiLSTM neural networks whenever the noise coefficient was high, and was comparable otherwise.

Noisy quantum computers, with over 100 qubits, have recently been developed and shown to perform tasks better than current supercomputers [22]. While the present is an exciting time to investigate and explore quantum algorithms and other applications, noiseless quantum computers with thousands of qubits are required in order to fully exploit the advantages of major algorithms like Shor's algorithm and Grover's algorithm. Hence, researchers are leaning toward hybrid classical-quantum algorithms as applications of quantum computing to machine learning, with the general idea of combining quantum and classical computers.

Endo et al. [23] used IBM's superconducting quantum computer to review the results for hybrid quantum-classical algorithms and quantum error mitigation techniques, and determined that future work on error mitigation would be extremely helpful since the use of NISQ devices is limited by large errors.

Disadvantages

(1) Physical methods use large mathematical models and require meteorological data on humidity, terrain structure, pressure, and other variables, to obtain accurate results. They are very costly and complicated [2] and are generally not used

for short-term forecasting.

(2) Persistence and statistical methods yield accurate results but their accuracies quickly decline as the forecasting horizon increases.

(3) Many machine learning models are used for forecasting and may encounter difficulties when used with data with large standard deviations.

(4) Autoencoders and DBN require large amount of clean data to generate useful results.

(5) Solving the model selection problem for (quantum) deep learning is time-consuming because the structure parameters and hyperparameters are obtained by trial-anderror thus, the robustness of (quantum) deep learning models cannot be guaranteed.

Proposed System

(1) A hybrid classical and quantum model is proposed. The LSTM is augmented by a deep QNN, allowing for the seamless fusion of classical and quantum neural networks. This fusion facilitates the transfer of knowledge gained from the classical layers to the quantum layers, empowering the QNN to refine predictions through the principles of the quantum mechanism, resulting in highly accurate 24-hour ahead wind speed forecasts. Consequently, the limitations mentioned earlier (1) and (2) can be effectively overcome.

(2) The hybrid classical and quantum model leverages the complementary learning capabilities of classical and quantum layers. LSTM is known for its proficiency in handling time-series data, while the QNN extends the learning capacity through quantum entanglement. This unique combination enables the network to recognize more complex patterns, which could be challenging for classical models to achieve alone.

(3) Originally, QML was applied mainly to classification problems. This paper aims to extend its application to regression problems, specifically wind speed forecasting.

Advantages

(i) Quantum parallelism: QNN can process multiple computations simultaneously

through superposition, allowing QNNs to potentially evaluate multiple input states in parallel, making QNNs attractive for certain machine learning tasks. (ii) Quantum feature mapping:

quantum feature mapping allows for efficient encoding of classical data into quantum states, providing a more expressive representation of data. (iii) Quantum entanglement: Quantum entanglement enables strong correlations among qubits, which can be exploited in QNNs to capture complex relationships between features in the data.

On the other hand, LSTM's ability to capture long-term dependencies, handle sequential data efficiently, and mitigate the vanishing gradient problem makes it a powerful choice for a wide range of sequential data processing tasks in machine

learning. Based on these reasons, the proposed model integrates LSTM with QNN.

FEASIBILITY ANALYSIS

An important outcome of preliminary investigation is the determination that the system request is feasible. This is possible only if it is feasible within limited resource and time. The different feasibilities that have to be analyzed are

- **Operational Feasibility**
- **Economic Feasibility**
- **Technical Feasibility**

Operational Feasibility

Operational Feasibility deals with the study of prospects of the system to be developed. This system operationally eliminates all the tensions of the Admin and helps him in effectively tracking the project progress. This kind of automation will surely reduce the time and energy, which previously consumed in manual work. Based on the study, the system is proved to be operationally feasible.

Economic Feasibility

Economic Feasibility or Cost-benefit is an assessment of the economic justification for a computer based project. As hardware was installed from the beginning & for lots of purposes thus the cost on project of hardware is low. Since the system is a network based, any number of employees connected to the LAN within that organization can use this tool from at anytime. The Virtual Private Network is to be developed using the existing resources of the organization. So the project is economically feasible.

Technical Feasibility

According to Roger S. Pressman, Technical Feasibility is the assessment of the technical resources of the organization. The organization needs IBM compatible machines with a graphical web browser connected to the Internet and Intranet. The system is developed for platform Independent environment. Java Server Pages, JavaScript, HTML, SQL server and WebLogic Server are used to develop the system. The technical feasibility has been carried out. The system is technically feasible for development and can be developed with the existing facility.

REQUEST APPROVAL

Not all request projects are desirable or feasible. Some organization receives so many project requests from client users that only few of them are pursued. However, those projects that are both feasible and desirable should be put into schedule. After a project request is approved, it cost, priority, completion time and personnel requirement is estimated and used to determine where to add it to any project list. Truly speaking, the approval of those above factors, development works can be launched.

3. REQUIREMENTS

3.1 SOFTWARE REQUIREMENTS

- **Operating system** : Windows 7 Ultimate.
- **Coding Language** : Python.
- **Front-End** : Python.
- **Back-End** : Django-ORM
- **Designing** : Html, css, javascript.
- **Data Base** : MySQL (WAMP Server).

3.2 HARDWARE REQUIREMENTS

- Processor - Pentium – IV
- RAM - 4 GB (min)
- Hard Disk - 20 GB
- Key Board - Standard Windows Keyboard
- Mouse - Two or Three Button Mouse
- Monitor - SVGA

4. DESIGN

4.1.1 INPUT DESIGN

Input Design plays a vital role in the life cycle of software development, it requires very careful attention of developers. The input design is to feed data to the application as accurate as possible. So inputs are supposed to be designed effectively so that the errors occurring while feeding are minimized. According to Software Engineering Concepts, the input forms or screens are designed to provide to have a validation control over the input limit, range and other related validations.

This system has input screens in almost all the modules. Error messages are developed to alert the user whenever he commits some mistakes and guides him in the right way so that invalid entries are not made. Let us see deeply about this under module design.

Input design is the process of converting the user created input into a computer-based format. The goal of the input design is to make the data entry logical and free from errors. The error is in the input are controlled by the input design. The application has been developed in user-friendly manner. The forms have been designed in such a way during the processing the cursor is placed in the position where must be entered. The user is also provided with in an option to select an appropriate input from various alternatives related to the field in certain cases.

Validations are required for each data entered. Whenever a user enters an erroneous data, error message is displayed and the user can move on to the subsequent pages after completing all the entries in the current page.

4.1.2 OUTPUT DESIGN

The Output from the computer is required to mainly create an efficient method of communication within the company primarily among the project leader and his team members, in other words, the administrator and the clients. The output of VPN is the system which allows the project leader to manage his clients in terms of creating new clients and assigning new projects to them, maintaining a record of the project validity and providing folder level access to each client on the user side depending on the projects allotted to him. After completion of a project, a new project may be assigned to the client. User authentication procedures are maintained at the initial stages itself. A new user may be created by the administrator himself or a user can himself register as a new user but the task of assigning projects and validating a new user rests with the administrator only.

The application starts running when it is executed for the first time. The server has to be started and then the internet explorer in used as the browser. The project will run on the local area network so the server machine will serve as the administrator while the other connected systems can act as the clients. The developed system is highly user friendly and can be easily understood by anyone using it even for the first time.

4.1.3 MODULES

Service Provider

In this module, the Service Provider has to login by using valid user name and password. After login successful he can do some operations such as Browse and Train & Test Data Sets, View Trained and Tested Accuracy in Bar Chart, View Trained and Tested Accuracy Results, View Prediction Of Wind Speed Forecasting Status, View Wind Speed Forecasting Detection Ratio, Download Predicted Data Sets, View Wind Speed Forecasting Detection Ratio Results, View All Remote Users

View and Authorize Users

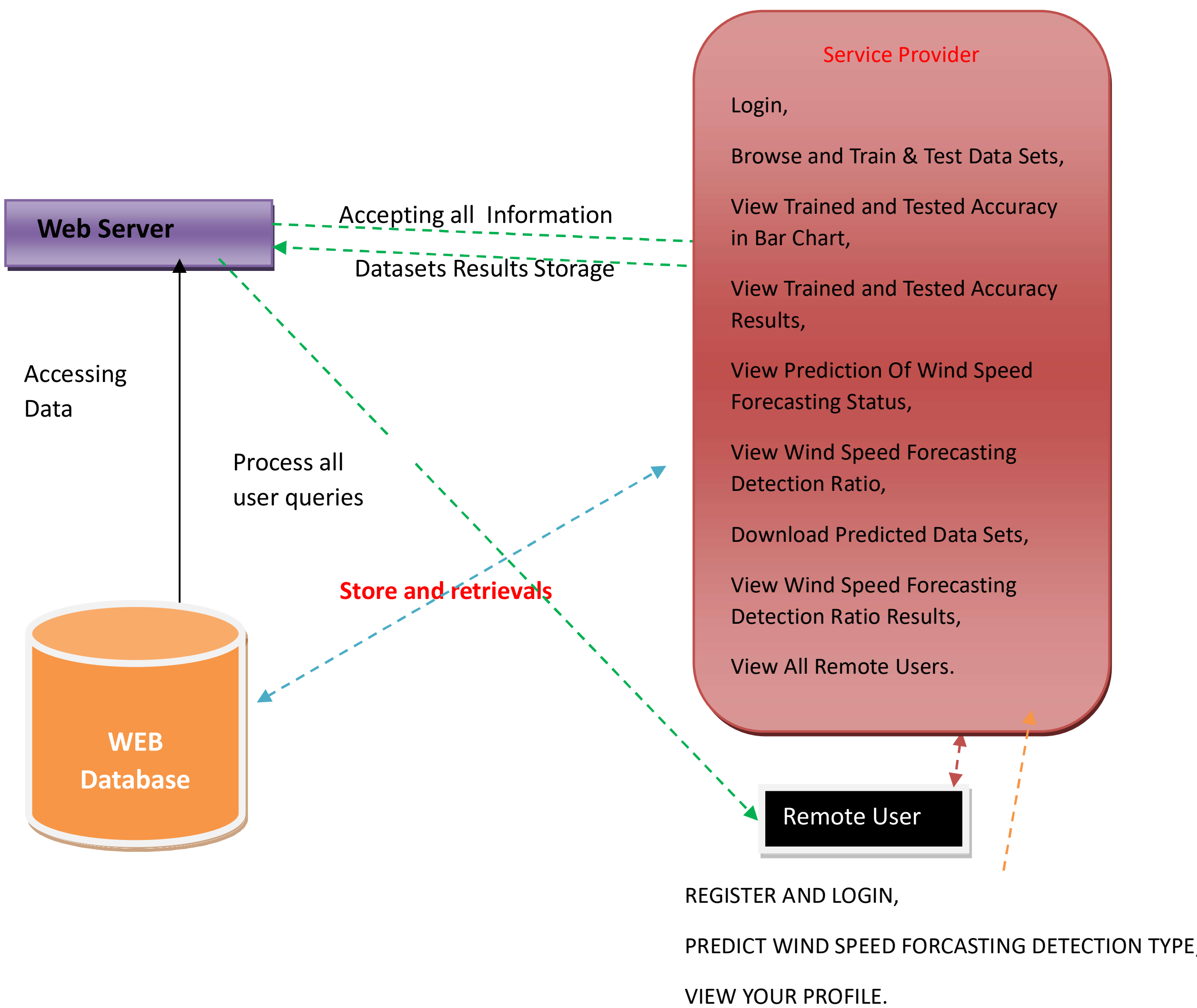
In this module, the admin can view the list of users who all registered. In this, the admin can view the user’s details such as, user name, email, address and admin authorizes the users.

Remote User

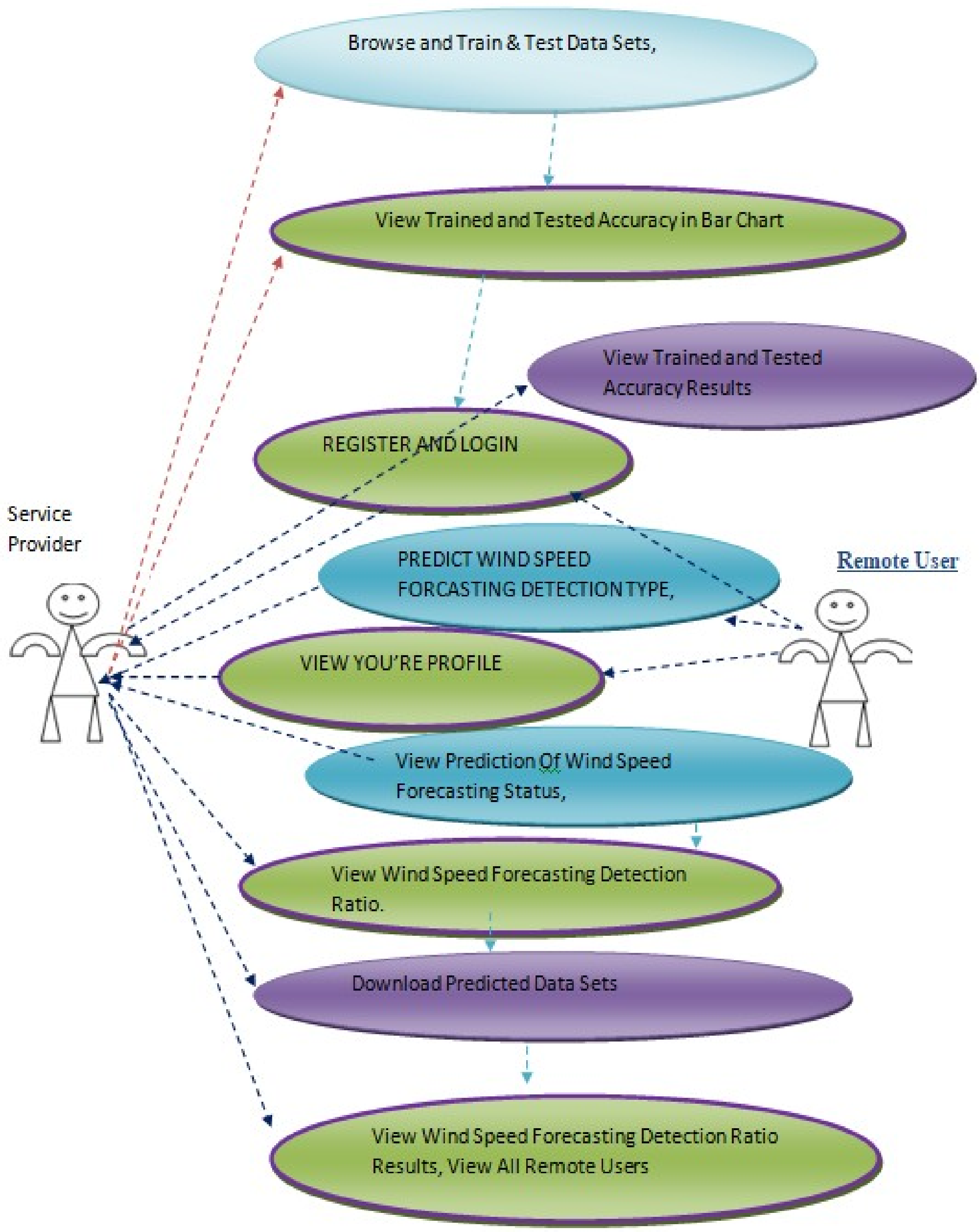
In this module, there are n numbers of users are present. User should register before doing any operations. Once user registers, their details will be stored to the database. After registration successful, he has to login by using authorized user name and password. Once Login is successful user will do some operations like REGISTER AND LOGIN, PREDICT WIND SPEED FORCASTING DETECTION TYPE, VIEW YOUR PROFILE.

UML DIAGRAMS

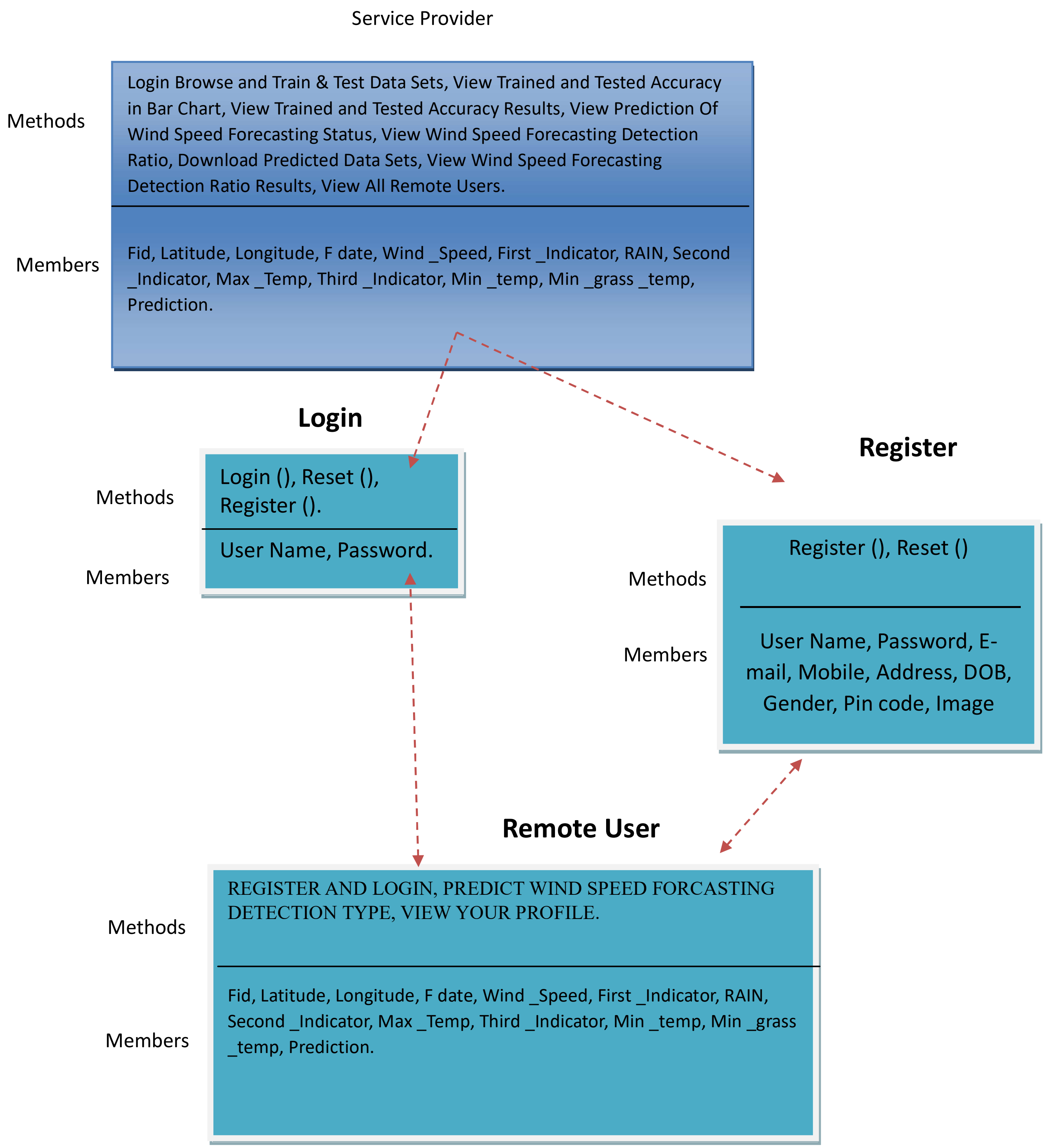
4. 2.1 Architecture Diagram



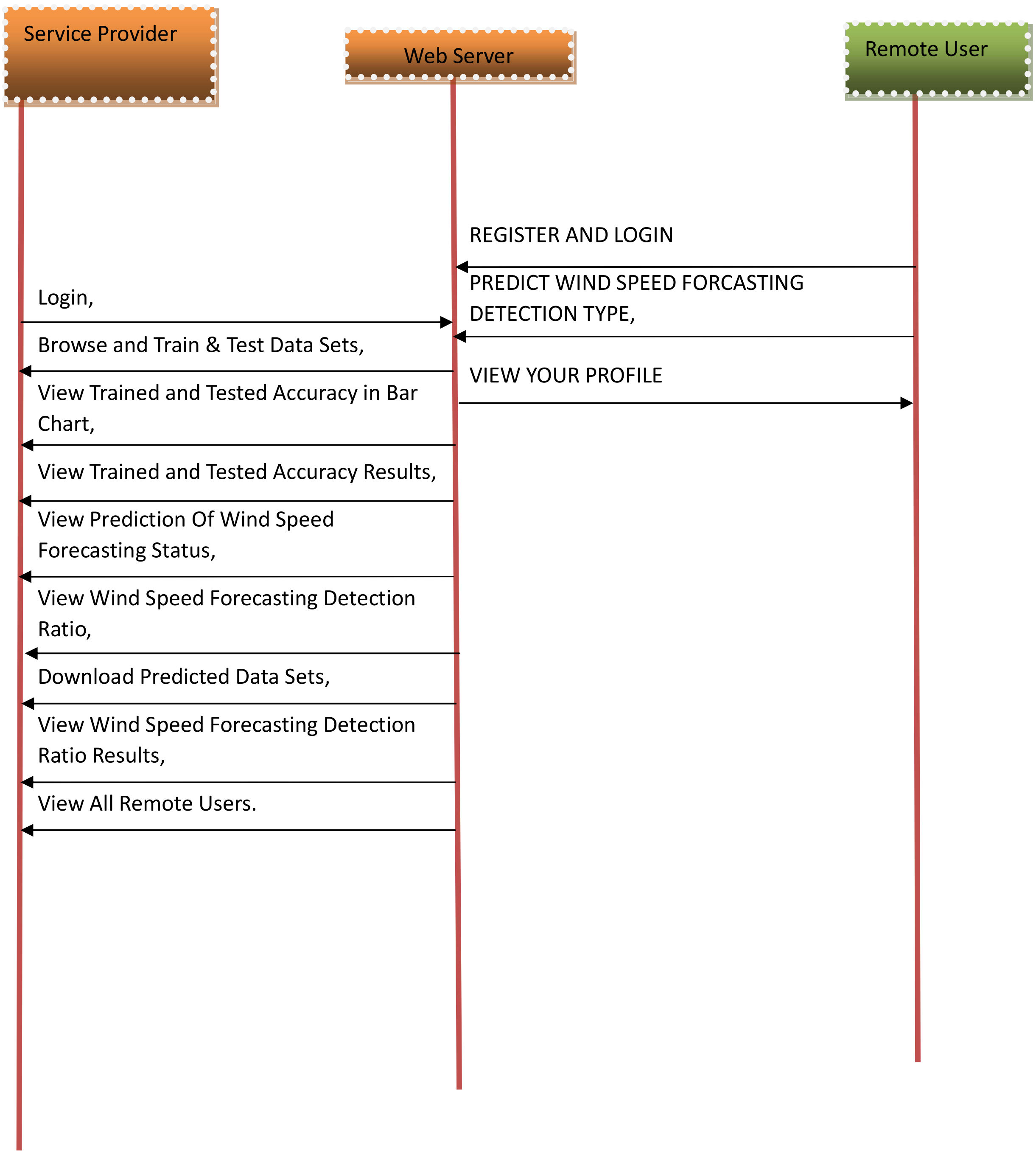
4.2.2 Use Case Diagram



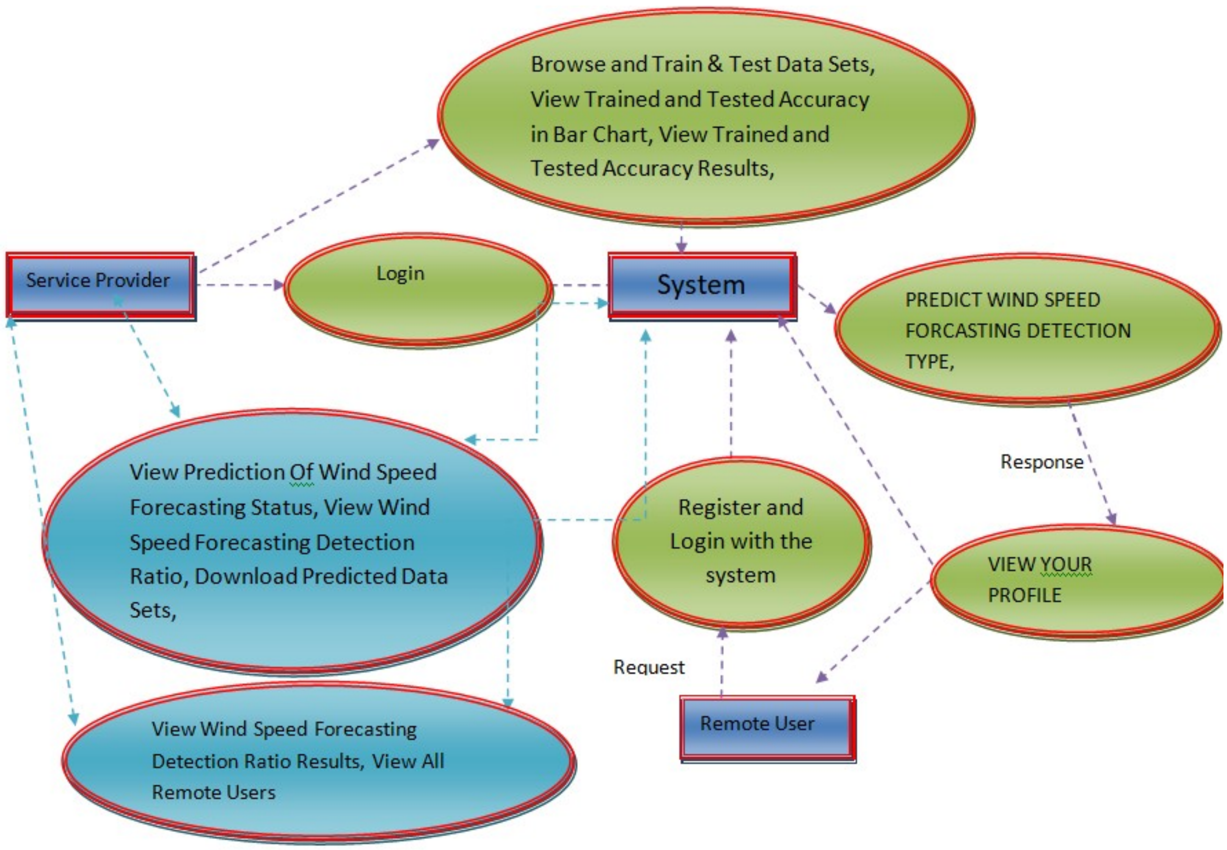
4.2.3 Class Diagram



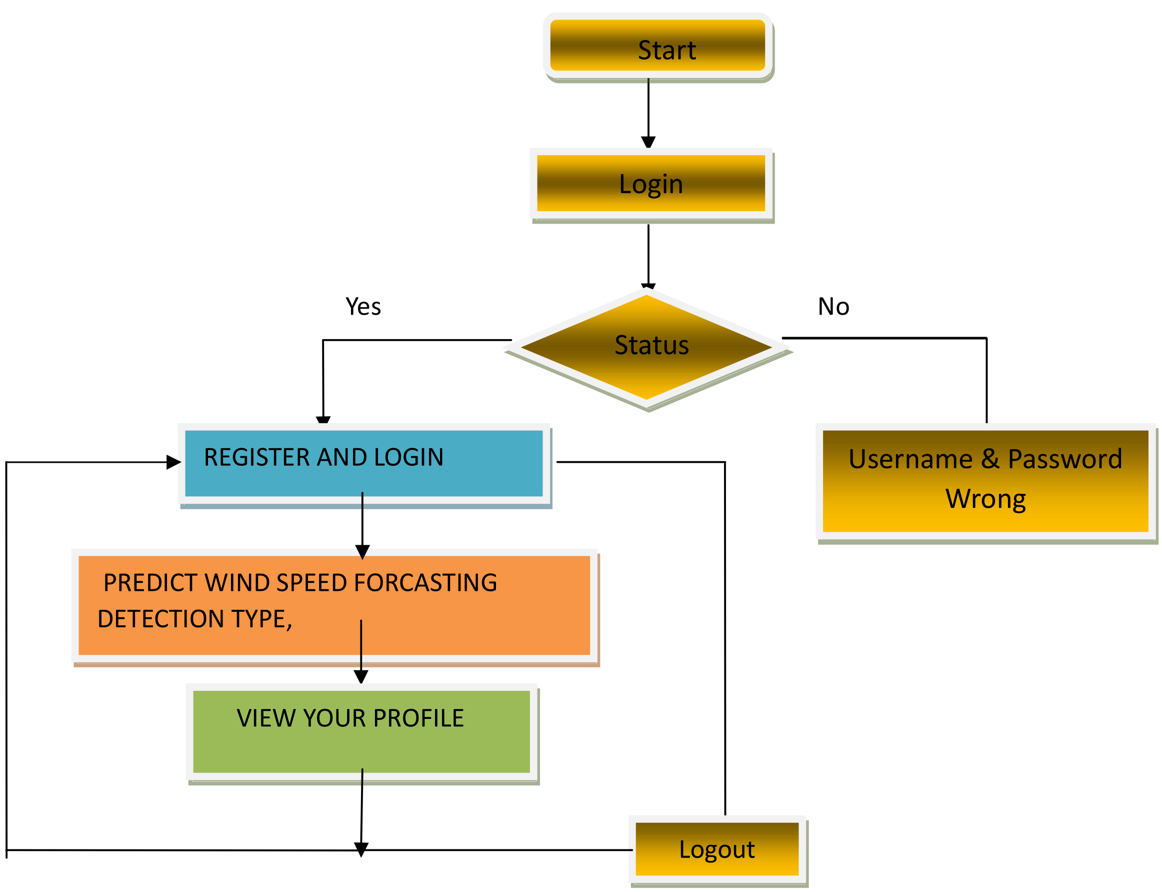
4.2.4 Sequence Diagram



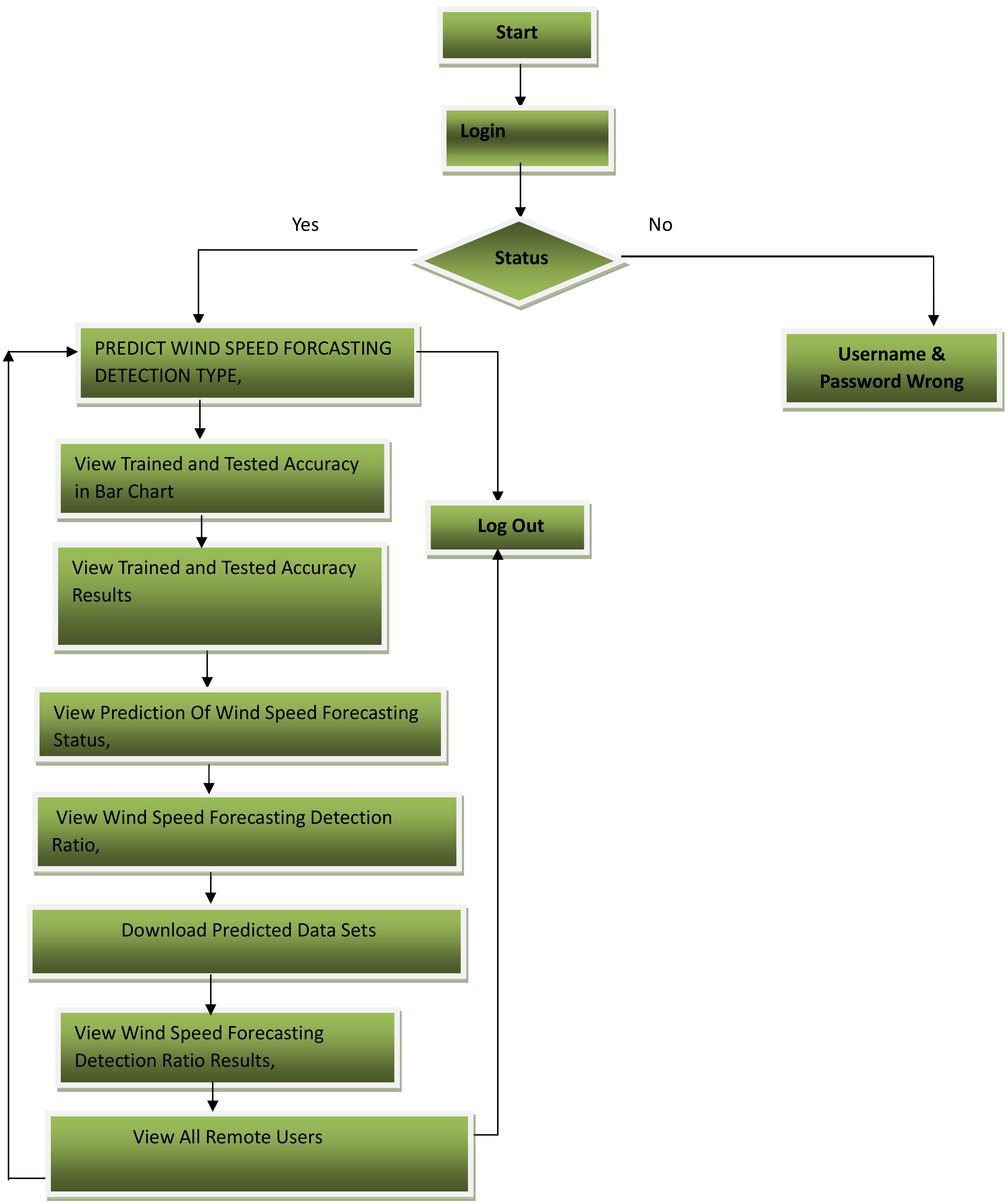
4.2.5 Data Flow Diagram



4.2.6 Flow Chart : Remote User



4.2.7 Flow Chart : Service Provider

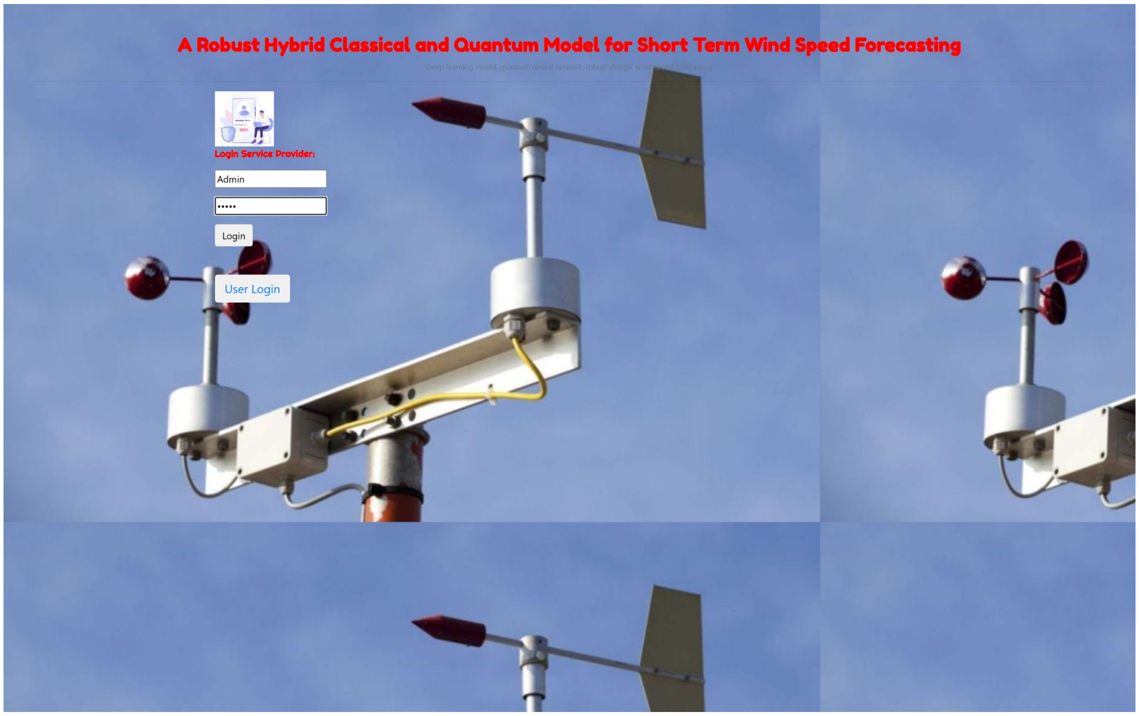


5. IMPLEMENTATION

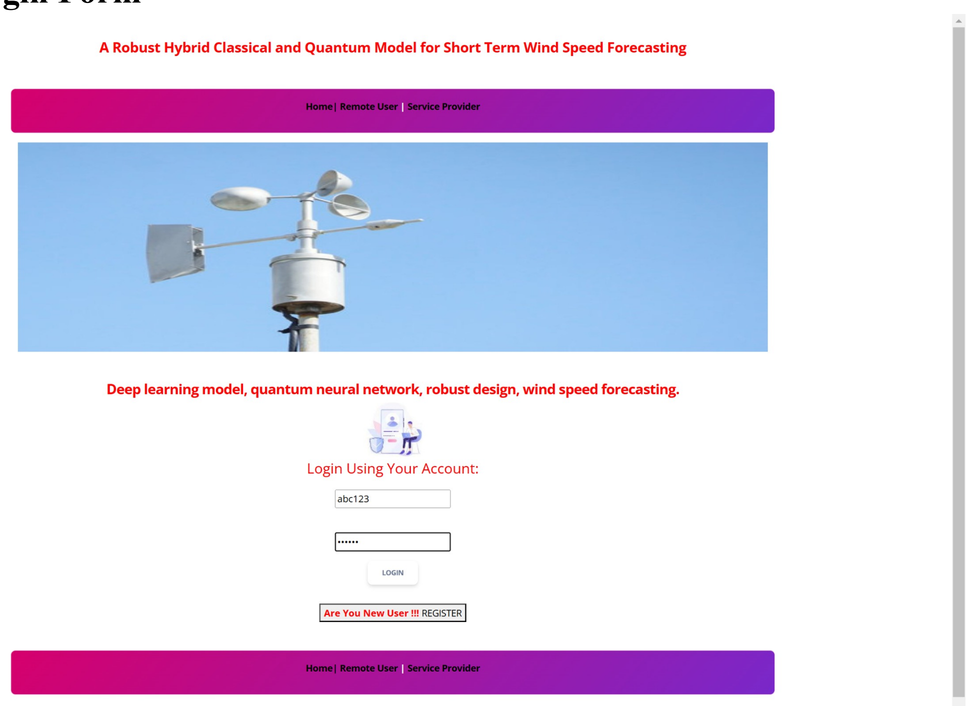
5.1 Home Page

5.2 User Register Form

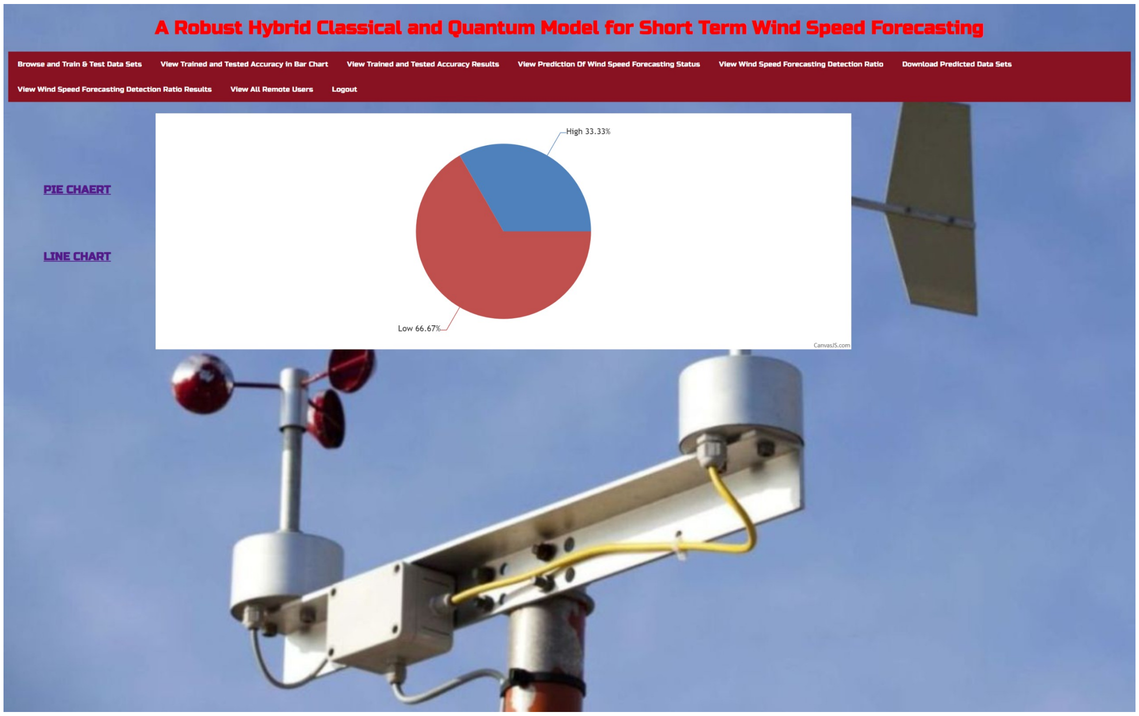
5.3 User Admin Form



5.4 User Login Form



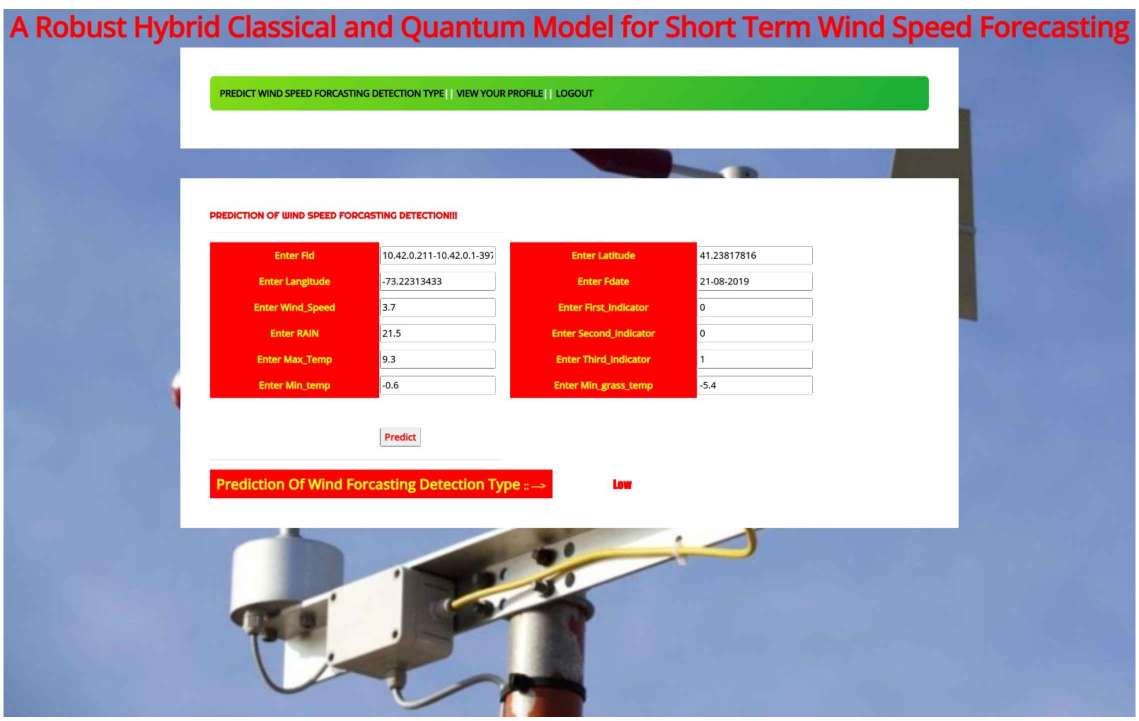
5.6 Wind Speed Forecasting Pie Chart



5.7 Applied Algorithms Scores

5.8 Result


A Robust Hybrid Classical and Quantum Model for Short Term Wind Speed Forecasting
Browse and Train & Test Data Sets
View Trained and Tested Accuracy in Bar Chart
View Trained and Tested Accuracy Results
View Prediction Of Wind Speed Forecasting Status
View Wind Speed Forecasting Detection Ratio
Download Predicted Data Sets
View Wind Speed Forecasting Detection Ratio Results
View All Remote Users
Logout
Wind Speed Forecasting Type Ratio Details
Wind Speed Forecasting Type
Ratio
High
33.33333333333333
Low
66.66666666666666


A Robust Hybrid Classical and Quantum Model for Short Term Wind Speed Forecasting
PREDICT WIND SPEED FORCASTING DETECTION TYPE || VIEW YOUR PROFILE || LOGOUT
PREDICTION OF WIND SPEED FORCASTING DETECTION!!!
Enter Fid
10.42.0.211-10.42.0.1-397
Enter Langitude
-73.22313433
Enter Wind_Speed
3.7
Enter RAIN
21.5
Enter Max_Temp
9.3
Enter Min_temp
-0.6
Enter Latitude
41.23817816
Enter Fdate
21-08-2019
Enter First_Indicator
0
Enter Second_Indicator
0
Enter Third_Indicator
1
Enter Min_grass_temp
-5.4
Predict
Prediction Of Wind Forcasting Detection Type :: -->
Low

6. TESTING

TESTING METHODOLOGIES

The following are the Testing Methodologies:

- **Unit Testing.**
- **Integration Testing.**
- **User Acceptance Testing.**
- **Output Testing.**
- **Validation Testing.**

6.1.1 Unit Testing

Unit testing focuses verification effort on the smallest unit of Software design that is the module. Unit testing exercises specific paths in a module's control structure to

ensure complete coverage and maximum error detection. This test focuses on each module individually, ensuring that it functions properly as a unit. Hence, the naming is Unit Testing.

During this testing, each module is tested individually and the module interfaces are verified for the consistency with design specification. All important processing path are tested for the expected results. All error handling paths are also tested.

6.1.2 Integration Testing

Integration testing addresses the issues associated with the dual problems of verification and program construction. After the software has been integrated a set of high order tests are conducted. The main objective in this testing process is to take unit tested modules and builds a program structure that has been dictated by design.

The following are the types of Integration Testing:

1. **Top Down Integration**

This method is an incremental approach to the construction of program structure. Modules are integrated by moving downward through the control hierarchy, beginning with the main program module. The module subordinates to the main program module are incorporated into the structure in either a depth first or breadth first manner.

In this method, the software is tested from main module and individual stubs are replaced when the test proceeds downwards.

2. Bottom-up Integration

This method begins the construction and testing with the modules at the lowest level in the program structure. Since the modules are integrated from the bottom up, processing required for modules subordinate to a given level is always available and the need for stubs is eliminated. The bottom up integration strategy may be implemented with the following steps:

- The low-level modules are combined into clusters into clusters that perform a specific Software sub-function.
- A driver (i.e.) the control program for testing is written to coordinate test case input and output.
- The cluster is tested.
- Drivers are removed and clusters are combined moving upward in the program structure

The bottom up approaches tests each module individually and then each module is module is integrated with a main module and tested for functionality.

6.1.3 User Acceptance Testing

User Acceptance of a system is the key factor for the success of any system. The system under consideration is tested for user acceptance by constantly keeping in touch with the prospective system users at the time of developing and making changes wherever required. The system developed provides a friendly user interface that can easily be understood even by a person who is new to the system.

6.1.4 Output Testing

After performing the validation testing, the next step is output testing of the proposed system, since no system could be useful if it does not produce the required output in the specified format. Asking the users about the format required by them tests the outputs generated or displayed by the system under consideration. Hence the output format is considered in 2 ways – one is on screen and another in printed format.

6.1.5 Validation Checking

Validation checks are performed on the following fields.

Text Field:

The text field can contain only the number of characters lesser than or equal to its size. The text fields are alphanumeric in some tables and alphabetic in other tables. Incorrect entry always flashes and error message.

Numeric Field:

The numeric field can contain only numbers from 0 to 9. An entry of any character flashes an error messages. The individual modules are checked for accuracy and what it has to perform. Each module is subjected to test run along with sample data. The individually tested modules are integrated into a single system. Testing involves executing the real data information is used in the program the existence of any program defect is inferred from the output. The testing should be planned so that all the requirements are individually tested.

A successful test is one that gives out the defects for the inappropriate data and produces and output revealing the errors in the system.

Preparation of Test Data

Taking various kinds of test data does the above testing. Preparation of test data plays a vital role in the system testing. After preparing the test data the system under study is tested using that test data. While testing the system by using test data errors are again uncovered and corrected by using above testing steps and corrections are also noted for future use.

Using Live Test Data:

Live test data are those that are actually extracted from organization files. After a system is partially constructed, programmers or analysts often ask users to key in a set of data from their normal activities. Then, the systems person uses this data as a way to partially test the system. In other instances, programmers or analysts extract a set of live data from the files and have them entered themselves.

It is difficult to obtain live data in sufficient amounts to conduct extensive testing. And, although it is realistic data that will show how the system will perform for the typical processing requirement, assuming that the live data entered are in fact typical, such data generally will not test all combinations or formats that can enter the system. This bias toward typical values then does not provide a true systems test and in fact ignores the cases most likely to cause system failure.

Using Artificial Test Data:

Artificial test data are created solely for test purposes, since they can be generated to test all combinations of formats and values. In other words, the artificial data, which can quickly be prepared by a data generating utility program in the information systems department, make possible the testing of all login and control paths through the program.

The most effective test programs use artificial test data generated by persons other than those who wrote the programs. Often, an independent team of testers formulates a testing plan, using the systems specifications.

The package “Virtual Private Network” has satisfied all the requirements specified as per software requirement specification and was accepted.

6.2.1 USER TRAINING

Whenever a new system is developed, user training is required to educate them about the working of the system so that it can be put to efficient use by those for whom the system has been primarily designed. For this purpose the normal working of the project was demonstrated to the prospective users. Its working is easily understandable and since the expected users are people who have good knowledge of computers, the use of this system is very easy.

6.2.3 MAINTAINENCE

This covers a wide range of activities including correcting code and design errors. To reduce the need for maintenance in the long run, we have more accurately defined the user’s requirements during the process of system development. Depending on the requirements, this system has been developed to satisfy the needs to the largest possible extent. With development in technology, it may be possible to add many more features based on the requirements in future. The coding and designing is simple and easy to understand which will make maintenance easier.

TESTING STRATEGY :

A strategy for system testing integrates system test cases and design techniques into a well planned series of steps that results in the successful construction of software. The testing strategy must co-operate test planning, test case design, test execution, and the resultant data collection and evaluation .A strategy for software testing must accommodate low-level tests that are necessary to verify that a small source code segment has been correctly implemented as well as high level tests that validate major system functions against user requirements.

Software testing is a critical element of software quality assurance and represents the ultimate review of specification design and coding. Testing represents an interesting anomaly for the software. Thus, a series of testing are performed for the proposed system before the system is ready for user acceptance testing.

SYSTEM TESTING:

Software once validated must be combined with other system elements (e.g. Hardware, people, database). System testing verifies that all the elements are proper and that overall system function performance is

achieved. It also tests to find discrepancies between the system and its original objective, current specifications and system documentation.

7. CONCLUSION

This study proposes a robust hybrid classical-quantum model for 24-hour ahead wind speed forecasting. The goal is to exploit the advantages of two promising methods, classical deep LSTM and quantum machine learning. The innovations/ findings are summarized as follows.

LSTM is combined with a QNN, which leverages quantum mechanical principles, to provide an unparalleled quantum advantage. A QNN includes an embedding layer that maps classical data into a quantum state, which is then processed by a variation layer with trainable parameters, and then a measurement is made to obtain a classical output.

The robust design of this proposed hybrid model is achieved by performing Taguchi orthogonal experiments, considering characteristics of various seasonal datasets. The hyper parameters (such as the number of hidden LSTM layers, number of hidden LSTM cells, dropout rate, kind of embedding, and quantum circuit depth) of the proposed hybrid model are determined systematically without trial-and-error.

The templates that are used to build the proposed QNN can be made more expressive by repeating each one and cascading them, effectively creating a deep QNN. This results from the quantum principles - entanglement, which states that two or more entangled qubits share information, and interference, which denotes a single quantum state interfering or combining with other quantum states of itself. In this investigation, two repetitions (or depth = 2) is found to yield the best results as a result of constructive interference of the probability of obtaining the correct state, and the destructive interference of that of the incorrect state.

Experiments reveal that the proposed hybrid model consistently produces more optimal outcomes than its counterpart classical model and other models, such as RF, XG Boost, SVR, NAR, LSTM, and LSTM autoencoder. Specifically, while the performance of classical models maybe comparable for some seasons, a notable decrease is observed during summer – the dataset with different statistical characteristics. The results obtained from this robust design, which integrates a classical LSTM with an expressive quantum circuit, demonstrate its great promise for near-term quantum devices.

Conducting the L18 experiments, evaluating the SNR, and performing an analysis of means, revealed that IQP embedding works best with 'Strongly Entangling Layers' variational circuit. Moreover, the QNN has a depth of two circuit repetitions, making it more expressive through interference. This design is robust and the least sensitive to variations in the four seasonal data sets.

The promising results and consistently accurate predictions amidst different seasonal datasets demonstrated by this study lead to a more optimistic outlook on the potential expansion of quantum computing applications in areas such as solar irradiance forecasting and renewable energy hosting capacity optimization involving quantum computing. Specifically, QC and QNN may contribute in the areas of simulation, scheduling and dispatch, and even reliability analyses in the renewable energy industry.

This work did not take into account the effects of noise on the performance of the quantum circuit, as a perfect quantum computer was assumed in the numerical simulations. Future research may explore the impacts of qubit decoherence and loss, incorporating error correction and noise models on prediction accuracy. Additionally, various hybrid architectures and more sophisticated quantum ersatzes may also be employed.

8. REFERENCES

1. P. Veers, K. Dykes, E. Lantz, S. Barth, C. L. Bottasso, O. Carlson,A. Clifton, J. Green, P. Green, H. Holttinen, D. Laird, V. Lehtomäki,J. K. Lundquist, J. Manwell, M. Marquis, C. Meneveau, P. Moriarty,X. Munduate, M. Muskulus, J. Naughton, L. Pao, J. Paquette, J. Peinke A. Robertson, J. Sanz Rodrigo, A. M. Sempreviva, J. C. Smith, A. Tuohy,and R. Wiser, "Grand challenges in the science of wind energy," *Science*,vol. 366, no. 6464, Oct. 2019, Art. no. eaau2027, doi: 10.1126/science.aau2027.

2. N. Chen, Z. Qian, I. T. Nabney, and X. Meng, "Wind power forecastsusing Gaussian processes and numerical weather prediction,"*IEEE Trans. Power Syst.*, vol. 29, no. 2, pp. 656–665, Mar. 2014, doi:10.1109/TPWRS.2013.2282366.

3. K. Yunus, T. Thiringer, and P. Chen, "ARIMA-based frequencydecomposedmodeling of wind speed time series," *IEEE Trans.Power Syst.*, vol. 31, no. 4, pp. 2546–2556, Jul. 2016, doi:10.1109/TPWRS.2015.2468586.

4. Z.-H. Zhou. *Machine Learning*. Berlin, Germany: Springer, 2021, doi:10.1007/978-981-15-1967-3.

5. H. Demolli, A. S. Dokuz, A. Ecemis, and M. Gokcek, "Wind powerforecasting based on daily wind speed data using machine learning algorithms,"*Energy Convers. Manag.*, vol. 198, Oct. 2019, Art. no. 111823,doi: 10.1016/J.ENCONMAN.2019.111823.

6. S. Nurunnahar, D. B. Talukdar, R. I. Rasel, and N. Sultana, "A short termwind speed forcasting using SVR and BP-ANN: A comparative analysis,"in *Proc. 20th Int. Conf. Comput. Inf. Technol. (ICCIT)*, Dec. 2017, pp. 1–6,doi: 10.1109/ICCITECHN.2017.8281802.

7. B. Bochenek, J. Jurasz, A. Jaczewski, G. Stachura, P. Sekuła, T. Strzyzewski, M. Wdowikowski, and M. Figurski, "Day-ahead wind power forecasting in Poland based on numerical weather prediction," *Energies*, vol. 14, no. 8, p. 2164, Apr. 2021.

8. M. Morina, F. Grimaccia, S. Leva, and M. Mussetta, "Hybrid weatherbased ANN for forecasting the production of a real wind power plant," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2016, pp. 4999–5005, doi: 10.1109/IJCNN.2016.7727858.

9. J. Yan, H. Zhang, Y. Liu, S. Han, L. Li, and Z. Lu, "Forecasting the high penetration of wind power on multiple scales using multi-to-multi mapping," *IEEE Trans. Power Syst.*, vol. 33, no. 3, pp. 3276–3284, May 2018,

10. doi: 10.1109/TPWRS.2017.2787667.

11. M. Khodayar, J.Wang, and M. Manthouri, "Interval deep generative neuralnetwork for wind speed forecasting," *IEEE Trans. Smart Grid*, vol. 10, no. 4, pp. 3974–3989, Jul. 2019, doi: 10.1109/TSG.2018.2847223.

12. C.-Y. Zhang, C. L. P. Chen, M. Gan, and L. Chen, "Predictive deep Boltzmann machine for multiperiod wind speed forecasting," *IEEE*

Trans. Sustain. Energy, vol. 6, no. 4, pp. 1416–1425, Oct. 2015, doi:10.1109/TSTE.2015.2434387.

13. J.-F. Toubeau, J. Bottieau, F. Vallée, and Z. De Grève, “Deep learningbasedmultivariate probabilistic forecasting for short-term scheduling in power markets,” *IEEE Trans. Power Syst.*, vol. 34, no. 2, pp. 1203–1215, Mar. 2019, doi: 10.1109/TPWRS.2018.2870041.

14. U. Cali and V. Sharma, “Short-term wind power forecasting using longshortterm memory based recurrent neural network model and variableselection,” *Int. J. Smart Grid Clean Energy*, vol. 8, no. 2, pp. 103–110,2019. [Online]. Available: http://www.ijsgce.com/index.php?m=content&c=index&a=show&catid=78&id=418

15. Y.-Y. Hong and T. R. A. Satriani, “Day-ahead spatiotemporal wind speedforecasting using robust design-based deep learning neural network,”*Energy*, vol. 209, Oct. 2020, Art. no. 118441.

16. Y. Ju, G. Sun, Q. Chen, M. Zhang, H. Zhu, and M. U. Rehman, “A model combining convolutional neural network and LightGBM algorithm for ultra-short-term wind power forecasting,” *IEEE Access*,vol. 7, pp. 28309–28318, 2019, doi: 10.1109/ACCESS.2019.2901920.

17. X. Liu, L. Yang, and Z. Zhang, “Short-term multi-step ahead windpower predictions based on a novel deep convolutional recurrent networkmethod,” *IEEE Trans. Sustain. Energy*, vol. 12, no. 3, pp. 1820–1833, Jul. 2021, doi: 10.1109/TSTE.2021.3067436.

18. J. Biamonte, P.Wittek, N. Pancotti, P. Rebentrost, N.Wiebe, and S. Lloyd, “Quantum machine learning,” *Nature*, vol. 549, pp. 195–202, Sep. 2017.

19. T. Xin, L. Che, C. Xi, A. Singh, X. Nie, J. Li, Y. Dong, and D. Lu,“Experimental quantum principal component analysis via parametrized quantum circuits,” *Phys. Rev. Lett.*, vol. 126, no. 11, Mar. 2021,Art. no. 110502.

20. J.-E. Park, B. Quanz, S.Wood, H. Higgins, and R. Harishankar, “Practical application improvement to quantum SVM: Theory to practice,” 2020, *arXiv:2012.07725*.

21. N. Killoran, T. R. Bromley, J. M. Arrazola, M. Schuld, N. Quesada, and S. Lloyd, “Continuous-variable quantum neural networks,” *Phys. Rev. Res.*, vol. 1, no. 3, Oct. 2019, Art. no. 033063, doi: 10.1103/PhysRevResearch.1.033063.

22. 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,and R. Yalovetzky, “Quantum machine learning for finance,” 2021, *arXiv:2109.04298*.

23. J. Preskill, “Quantum computing in the NISQ era and beyond,” Q*uantum*, vol. 2, p. 79, Aug. 2018. Endo, Z. Cai, S. C. Benjamin, and X. Yuan, “Hybrid quantum-classical

24. algorithms and quantum error mitigation,” *J. Phys. Soc. Jpn.*, vol. 90, no. Mar. 2021, Art. no. 032001.

25. *IBM Quantum*. Accessed: Feb. 2, 2023. [Online]. Available: https://quantum-computing.ibm.com/*Simulation: CIRQ: Google Quantum AI*. Google Quantum AI. Accessed: Feb. 2, 2023. [Online]. Available: https://quantumai.google/cirq/simulate/simulation