Machine Learning-Based Time Series Modelling with Applications for Forecasting Regional Wind Power and Air Quality Index

by

Hanin Alkabbani

A thesis

presented to the University of Waterloo

in fulfillment of the

thesis requirement for the degree of

Master of Applied Science

in

Chemical Engineering

Waterloo, Ontario, Canada, 2021

AUTHOR'S DECLARATION

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Recently, time series forecasting has acquired considerable academic and industrial interest in various areas for different applications. Machine learning (ML) algorithms are known for their ability to capture the chaotic temporal non-linear relations in time series data. This research employs various ML concepts and algorithms into two different case studies of time series forecasting: 1-Regional wind power forecasting and 2-Air quality index (AQI) forecasting.

The first case study is conducted to focus on regional wind power forecasting comprehensively from different perspectives. First, the meteorological and spatial parameters with seasonal and temporal features were filtered and selected by a proposed deep feature selection approach consisting of series of steps. Later, multiple ML algorithms, including artificial neural network (ANN), deep neural network (DNN), long short-term memory (LSTM), bagging tree (BT), and support vector machine/regression (SVM/SVR), were used for training one-step-ahead forecasting models. Lastly, an assessment of the constructed models was conducted based on different error criteria metrics. The final comparative discussion concluded that the SVR-based model provided accurate generalized performance when tested on unseen data and surpassed other models, including LSTM. However, when constructing the multi-step ahead forecasting models, the predictions obtained from the multi-input multi-output (MIMO) LSTM approach were reliable with higher accuracies. Overall, for multi-step forecasting, it was concluded that the performance of the MIMO multi-step strategy was superior to the direct multi-step forecasting method, especially by employing algorithms with recursive properties.

It is also essential to mention that chapter 2 of this thesis is a comprehensive literature review of machine learning and metaheuristics methodologies of renewable power forecasting. This review can guide scientists and engineers in analyzing and selecting the appropriate prediction approaches based on the different circumstances and applications.

The second case proposes a comprehensive method to forecast AQI. The proposed methodology was tested on ambient air quality observations at Al-Jahra, a major city in Kuwait. The hourly levels of the six criteria pollutants (O_3, SO_2, NO_2, CO, PM_{10}, and $PM_{2.5}$) were predicted using artificial neural networks, which then fed into the process of estimating AQI. The prediction of the AQI does not only require the selection of a robust forecasting model, rather it heavily relies on a sequence of pre-processing steps to select predictors and handle different issues in data. One major problem that commonly appears in ambient air quality datasets is data gaps. The presented method dealt with this by imputing missing entries using miss-forest; a machine learning-based imputation technique. The

effectiveness of this imputation method was examined against the linear imputation method for the six criteria pollutants and the AQI. Results obtained showed that models trained using miss-forest imputed data could generalize AQI forecasting and with a prediction accuracy of 92.41% when tested on new unseen data.

Acknowledgements

I would like to thank my esteemed supervisors Professor Ali Elkamel and Qinqin Zhu, for their invaluable supervision, support and tutelage during my research and degree. Additionally, I would like to express gratitude to Dr. Ashraf Ramadan, who provided guidance, insight, and data to develop a complete chapter in this thesis. A special thanks to my best friend, engineer Alghalyah Almashaan for her continuous support.

Dedication

Dedicated to my beloved supportive family and friends.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

Abbreviation	Explanation
ACF	Autocorrelation Function
ADALINE	Adaptive Linear Element
AI	Artificial Intelligence
ANFIS	Adaptive Neuro-Fuzzy Inference System
ANN	Artificial Neural Network
AQI	Air Quality Index
AR	Autoregressive
ARFIMA	Autoregressive Fractionally Integrated Moving Average
ARIMA	Autoregressive Integrated Moving Average
ARMA	Autoregressive Moving Average
BT	Bagging Tree
DBSCAN	Density-based Spatial Clustering of Applications with Noise
DE	Differential Evolution
DL	Deep Learning
DNN	Deep Neural Network
ELM	Extreme Learning Machines
EPA	Environmental Protection Agency
FFBP-ANN	Feed Forward Back Propagation Neural Network
GCA	Grey Correlation Analysis
GHI	Global Horizontal Irradiance
GRU	Gated Recurrent Unit
LSSVM	Least Square Support Vector Machine
LSTM	Long Short-Term Memory
MA	Moving Average
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
MARS	Multivariate Adaptive Regression
MIMO	Multiple-Input Multiple-Output
ML	Machine Learning

Abbreviation	Explanation
MTGP	Multitasking Gaussian Process Regression
NAR	Non-linear Autoregressive
NWP	Numerical Weather Predictions
PACF	Partial Autocorrelation Function
PFLR	Partial Functional Linear Regression
PSO	Particle Swarm Optimization
PV	Photovoltaic
RBF	Radial Basis Function
RF	Random Forest
RFR	Random Forest Regression
RNN	Recurrent Neural Network
RSM	Response Surface Method
SVR	Support Vector Regression
SVM	Support Vector Machine
WNN	Wavelet Neural Networks

Chapter 1
Introduction

In this chapter, the motivation of this study is clearly defined, and the outline of the overall represented work in this thesis is also described.

1.1 Overview

Time series data is timely sequenced data that have high temporal dependencies. Time series analysis and forecasting aim to find and capture the trends in these data to build reliable models capable of representing these data. Models that provide timely, reliable predictions can then be used for different applications related to optimizing processes, scheduling and managing, waste prevention, and many other applications. Time series forecasting models can be built using only endogenous predictors, in other words, by only using the past observations of the target itself to forecast its future trends. While other forecasting models can be built by considering the impact of additional different time series independent factors, this type of forecasting is known as exogenous forecasting. Statistical and persistence models are the classical prediction models that provided reliable predictions in various applications. Although these models' primary focus is on linear relationships, these models are robust and perform adequately only if the employed data are pre-processed and appropriately prepared. On the other hand, the machine learning (ML)-based forecasting models recently proved their ability to capture and map non-linear relationships for different applications in industry, finance, and supply chain management.

The main focus of this study is on these robust recent ML algorithms for time series forecasting. The motivation of this work is clearly defined in the following section.

1.2 Motivation

With the recent constant development and growth of the artificial intelligence (AI) discipline in general and ML in specific, wide ML applications have been proposed in the literature for different objectives. As mentioned before, time series forecasting is one of these objectives that has broad applications and implementations. Thus, our primary goal in this study is to assess ML learning algorithms by considering two different case studies:

1- Regional wind power forecasting,

2- Air quality index (AQI) forecasting.

``

Firstly, in each case study, a problem is clearly specified and motivated, then tackled by proposing systematic ML-based modelling consisting of all required pre-processing, feature selection, and dimensionality reduction steps to report results, finally, compare them, and recommend approaches, and provide guidance for future work.

It is essential to mention that the motivation of the two case studies is not only restricted to the ML methodologies implementation and testing, but the aim is also to provide reliable models that can contribute to industrial and environmental issues. For example, the regional wind power forecasting case study would provide a reliable, tested and verified model that can provide accurate regional wind power predictions. These predictions would be a part of the optimal scheduling and managing of the regional electrical grid, reducing the surplus energy production and increasing dependence on environmentally friendly resources. On the other hand, the AQI forecasting case study would help inhabitants prevent exposure to polluted low-quality air and help decision-makers cut off or reduce polluting actions at predicted peak hours. The more detailed, clear motivations for each case study will be presented later in chapters 4 and 5.

1.3 Outline

This thesis is compiled as follows:

- Chapter 2: Literature Review:

  The first section in Chapter 2 includes a comprehensive, in-depth literature review of ML applications for renewable power forecasting (including wind and solar power).

  The other section of Chapter 2 briefly reviews the previous work represented for AQI forecasting.

- Chapter 3: Concepts of Machine Learning Forecasting Models

  This chapter presents a conceptual and mathematical description of forecasting and evaluation methods utilized in the two studies.

- Chapter 4: Regional Wind Power Forecasting

  A complete study is conducted in this chapter, evaluated, and concluded for wind power forecasting in Ontario using different machine learning methods.

``

- Chapter 5: Air Quality Index Forecasting.

  This chapter proposes an approach to forecast the AQI by applying different ML algorithms for different purposes to build a forecasting model, evaluate the proposed approach, and provide guidance for future work.

- Chapter 6: Thesis Conclusion and Future Work

  This chapter concludes the presented work and provides guidance and recommendations for future work.

``

Chapter 2
Literature Review

This chapter is divided into two sections as follows:

- Section 2.1 includes a comprehensive, in-depth literature review of ML applications for renewable power forecasting (including wind and solar power).
- Section 2.2 reviews the previous work represented for air quality index forecasting.

2.1 Machine Learning and Metaheuristic Methods for Renewable Power Forecasting

This section represents a comprehensive review of the recently published and proposed wind and solar power forecasting (ML)-based models. Comparing to the existing studies on the same topic, the contributions of this conducted study:

1. A broad review of ML-based renewable power prediction methodologies and the metaheuristic optimizers of these methodologies is for the first time performed from a categorization viewpoint. Categorization is achieved by systematically allocating the ML prediction approaches and optimizers based on their similarities and differences and the type of forecasted renewable energy. This will provide an analytical review of the current renewable power forecasting studies based on renewable energy sort (wind or solar).
2. Comparative evaluations of the ML-based renewable prediction methods and their metaheuristics optimizers are carried out. The drawn-out results would help other scholars decide on the appropriate ML-predictors and metaheuristic optimizers for various forecasting situations and purposes.
3. Highlighting the ML applications' challenges for renewable power forecasting and providing key directions that would guide other scholars to focus on the potential issues that have not been resolved yet.

In summary, despite the flourishing of related studies conducted to propose ML-based forecasting models, a review that summarizes these renewables' forecasting models and analytically evaluates their performance from a categorization perspective has not been investigated yet.

``

Therefore, this section analyses renewable power ML prediction tools and optimizers of these tools, emphasizes their weaknesses and strengths and underlines the challenges they accompanied to direct researchers on the issues that have not been settled yet.

2.1.1 Introduction

Governments and policymakers have promoted renewable energies' penetration into the electricity production sector to respond to the environmental crisis and reduce greenhouse gas emissions. According to the international renewable energy agency report, renewable energy resources contribution to electricity generation is projected to reach 85% by 2050, mainly due to the growth of solar and wind-produced power[1]. Although renewables are highly efficient, pollutant-free, and inexpensive to produce and distribute, they lack consistency. Unlike conventional resources (coal and fossil fuels), which can be generated according to the consumption and at specific, accurate schedules, renewable energies' production is variable; they rely on seasonal and weather conditions (eg., temperature, pressure, wind speed, visibility, etc.) [2]. These chaotic conditions can change dramatically from time to time, enforcing difficulties in the optimal electricity generation scheduling and managing and imposes concerns regarding electricity quality and stability[2,3]. In fact, if the integration of renewable energy into the electricity sector is not handled and controlled adequately, it could cause imbalanced and excess power production, which may increase the government's expenses instead of reducing them [4,5]. Moreover, this unpredictable stochastic nature of renewables resulted in serious unit commitment issues[6]. Therefore, accurate prediction of renewables has become an enduring worldwide interest in the literature.

Thus far, various research studies have been employed to tackle the problem of unreliable and inaccurate renewable power forecasting models. These include persistence models, physical models, statistical models, AI models, and hybrid models consisting of a combination of two or more of these models. Lately, amongst these forecasting methodologies, AI-based models, particularly ML models, have gained researchers' interest. Unlike most of the traditional forecasting models, ML techniques can mainly capture the nonlinearity in power data. They can be applied for several purposes with only minor modifications. Therefore, because of their flexibility and compatibility, ML models could outperform and alternate the conventional ones [6] [7]. Moreover, these methods can significantly benefit the availability of large datasets to improve the forecasting performances, unlike the statistical models that typically are not expected to improve prediction with larger datasets.

``

2.1.2 An Overview of Renewable Power Forecasting

In this section, recent structures in the literature for renewable power forecasting are reviewed. Various schemes and methodologies plus AI tools are discussed and described.

Renewable power term generally encompasses all types of power gathered and generated from carbon-free renewable resources such as wind, sunlight, rainfall, and waves. Particularly wind and solar energies are fluctuating resources because their production rates depend on intermittent, unpredictable weather conditions (wind speed and directions and solar irradiation, respectively). Thereby, the renewable power forecasting-related research studies consider the wind and solar power outputs themselves and the wind speed and solar irradiation. From that perspective, renewable power forecasting methodologies will be reviewed, including wind speed or/and power and solar irradiance/power.

Forecasting methodologies

Figure 1 illustrates the differences between forecasting horizons and their applications in the electricity sector [7]. Including AI approaches, researchers proposed different forecasting structures by various methods and from multiple perspectives. These approaches and related research work for renewables forecasting (mainly wind and solar power) are reviewed in the following sections.

Figure 1: Forecasting time horizons [7]

``

Persistence methodologies

These methodologies simply assume that power data values at the next time step are similar to the values of the current time step. Although these methodologies are not very practical for long-term forecasting, they perform well in very short-term and short-term forecasting(from a few seconds to 6 hours-ahead) [8].

Physical methodologies

In addition to geographical locations and physical characteristics and layouts of wind turbines or solar panels, these methodologies depend on numerical weather predictions (NWP) such as (temperature, pressure, wind speed, wind density, roughness, turbulence intensity, etc.) [9]. Although these methodologies are reliable for medium and long-term forecasting, they cannot perform accurately for short-term forecasting [10]. Besides, they fail to adopt interferences, are computationally expensive, and require advanced computing machines[11]. Ref [12] comprehensively reviewed published studies tackling short-term forecasting by NWP models. This study was the last to summarize NWP-based models' applications because these models' implication was no longer attractive for researchers and many other recent methodologies started flourishing and outperforming physical and NWP-based models [9].

Statistical methodologies

Statistical-based forecasting models are mathematical models that attempt to map and recognize the relationship between timer series historical data and target outputs [13]. They can clearly describe the data's linear relationship with basic simple mathematical equations [7,8]. Furthermore, since they can be formulated easily, they can deliver timely predictions. Thus in the literature, these methods are mainly used for short-term forecasting [16].

A comprehensive literature review on statistical approaches for time series renewable energy forecasting was presented by Ghofrani et al. in Ref [17]. Autoregressive (AR) and moving average (MA) models are well-known examples of statistical forecasting systems [15]. The hybrid integration of these two techniques is known as the autoregressive moving average (ARMA). ARMA is widely used for forecasting and provides models with high accuracy for different applications. The work in [18] compared four other ARMA-based models for wind speed and direction forecasting. In another article, Gomes et al. presented a comparative study between ARMA and artificial neural networks

``

(ANN) for wind speed and power prediction. They concluded that both approaches result in similar results; however, the ARMA performance is slightly better [19]. In Ref [20], a non-linear autoregressive (NAR) model was suggested for short-term PV (Photovoltaic) power forecasting utilizing only historical data of the PV power (without using the NWP data). When comparing the NAR model's performance with the auto-regressive with exogenous input (ARX) model, it was determined that NAR gives better results than ARX. This conclusion contrasts with the result obtained in Ref [21], where ARX performed better.

Another robust approach known as ARIMA (auto-regressive integrated moving average) is widely employed for different purposes in the literature until the date. For example, [22] used the ARIMA approach to predict daily solar energy production. Note that the application of ARIMA models requires the utilized data to be stationary; therefore, in their work, they transformed the non-stationary seasonal data into stationary ones. For longer-term forecasting, Ref [23] used the ARIMA model for one year ahead of wind speed and temperature forecasting. According to their conclusion, this generated model is generic, and with some minor modifications like increasing the input data size, this model can be applied for two-years ahead forecasting.

A particular parsimonious type of ARIMA known as fractional-ARIMA was studied in Ref [24] for wind forecasting. Fractional-ARIMA is computationally simple and can capture time-series relations for both the long and short-term forecasting horizons. This paper employed this model for predicting hourly wind speed and up to two days ahead. The results were promising and showed that this simple model could improve the forecasting accuracy by 42 % compared to persistence forecasting models.

In general, statistical models are considered attractive to researchers until the date because they are inexpensive and straightforward to apply. They presented acceptable accurate results for short-term horizons up to 2 days; however, they fail in forecasting and result in very unstable predictions for longer-term horizons [16]. Besides, they require pre-processing time-series data (mostly when the data is discontinuous) to perform reliably and provide accurate prediction models. This pre-processing could cause issues and requires expensive computation machines. Thus, researchers started to use a hybrid combination of these statistical models with AI methods to resolve the pre-processing issues.

``

Regression methodologies

This type of model aims to find the best mathematical representation that relates independent variables (generally NWP and some physical properties and operation conditions of the turbines or solar panels) to the dependent variables (wind or solar power) through curve fitting hyperparameter optimization techniques. Multilinear regression models are the simplest case of regression where the forecast variable is related to the predictors by a simple linear relationship [13]. For example, in [25], Abuella et al. utilized multilinear regression to build a solar power probabilistic forecasting model. Besides, simple linear quantile regression was used in Ref [26] to create three different probabilistic models within the day (1-6 hours ahead) solar irradiation prediction. For building the three models, authors utilized historical data of solar irradiance as endogenous inputs and day-ahead NWP of irradiance as exogenous inputs. The obtained results showed that the presence of NWP as exogenous inputs improved the prediction results. However, similar to the results obtained in Ref [25], the comparative study of the model's performance in two different sites showed that the probabilistic models highly depend on the regional sky conditions. In the work done in Ref [27], the multilinear adaptive regression spline method was used with small training samples and a limited number of features to define a day-ahead solar power forecasting model. This proposed regression model used historical power output data and weather forecasts.

In another article, Wang et al. in Ref [28] proposed a novel partial functional linear regression (PFLR) model to forecast a PV system's daily output energy. PFLR is similar to the multilinear regression, but it can also represent the nonlinearity structure in solar power data. Unlike statistical models that focus on utilizing historical data and underestimate the importance of the renewables data within the day pattern, PFLR incorporates the intra-day pattern of data and extracts valuable information from them. This work showed that this novel model that involves a few parameter estimates outperformed the ANN models and the regular multilinear regression. Another regression technique, known as multitasking Gaussian process regression (MTGP), was used in Ref [29] as a post-processing step to improve the NWP of wind speed. This additional step tackled the unreliable predictions that yield from the NWP when the wind speed data's behavior is very complex and intermittent. The MTGP technique in this paper improved the forecasting accuracy of long-term forecasts and shorter-term forecasts. This improvement of NWP resulted in superior prediction results compared to the statistical predictors that are well known for their accuracy for short-term forecasting. Authors of Ref [30] performed a comparative study to compare four different heuristic regression techniques, including Kriging,

``

response surface method (RSM), multivariate adaptive regression (MARS), and M5 model tree (M5 Tree) for solar irradiation modeling. Comparative results showed that Kriging executed a better performance compared to the other three methods.

Overall, although regression forecasting methodologies are simple and performed promisingly in some applications, they lack generalization and highly depend on the input data. Besides increasing their accuracy, many explanatory variables are required, which is considered a limitation of this method [31]. Not to forget to mention that all the linear regression models share the assumption of a linear relationship between the independent and dependent variables, which is rare in most renewable power applications, especially in wind power-related problems.

Machine learning forecasting methodologies

AI is a subfield of computer science; in AI, intelligent machines or artifacts are designed and trained to function like humans by following specific commands in computer programming systems. AI-based forecasting models accelerate decision-making, data mining, and clustering problems because they can robustly handle big data fitting and develop good representations. Besides, they can employ too complex tasks with moderately short time and without being explicitly programmed. Thereby, AI methodologies have been used for various prediction applications in different areas of engineering, medicine, economy, and agriculture [32]. Thus, the focus on proposing AI-based forecasting models grew a lot in the past few years and even started to alternate the conventional known prediction models [33].

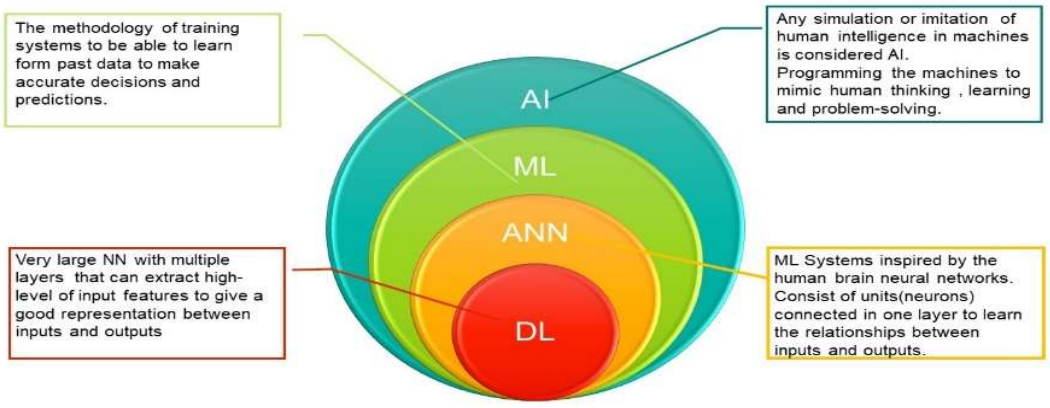

Figure 2: Relationship between artificial intelligence, machine learning, deep learning, and artificial neural networks [36]

``

ML, ANN, and deep learning (DL) are all subsets of AI Figure 2 illustrates the differences and relationships between these subsets [34]. The following sections will review the recent research routes of renewable power forecasting (both wind and solar power) based on the used machine learning algorithms for forecasting.

ML is an approach for data analysis, which gives computer systems the power to learn from data through experience. ML techniques can generally capture the nonlinearity and adapt instability in data, resulting in more reliable predictions [15].

Therefore, in the past few decades, ML tools were employed for forecasting various problems, such as renewable energy forecasting. According to our survey ANN, recurrent neural network (RNN), support vector machine (SVM), and extreme learning machine (ELM) are the most used ML techniques for renewable energy forecasting.

2.1.3 Artificial Neural Networks-Based Methodologies

All types of ANN have layers of neurons; The input layer is the layer where the network receives the input features; each neuron in this layer takes an input feature. The output layer is where the final targets are estimated. The hidden layer is the connection between the input and the output layer, where most of the required computational operations occur; Figure 3 represents the hidden layers' node structure. As shown in Figure 4, the outputs of the nodes of an ANN are determined by passing the input features multiplied by their corresponding weights to an activation function in the hidden layer's nodes. There are several types of ANN; Figure 4 represents the classical structure of an ANN. In this section, the applications of ANNs for wind and solar power forecasting are reviewed.

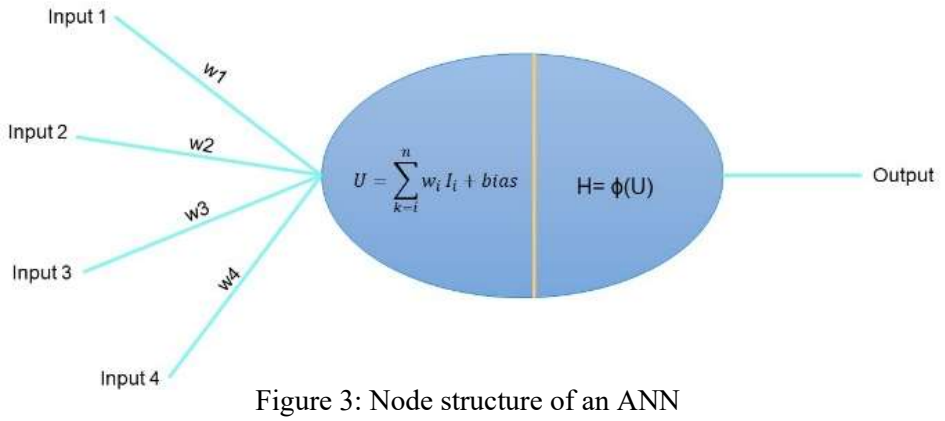

Figure 3: Node structure of an ANN

``

.

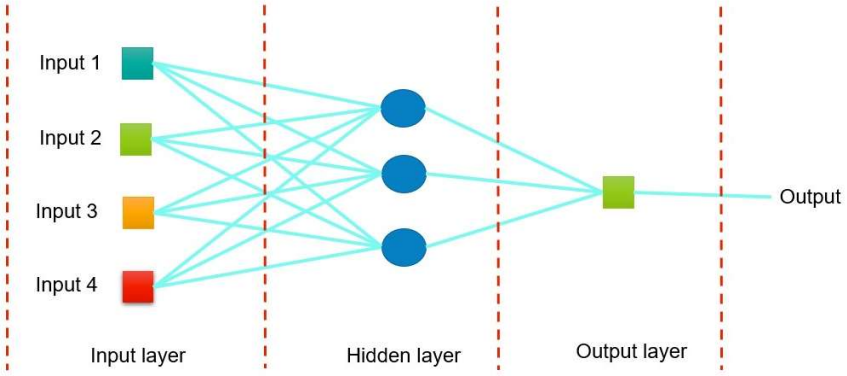

Figure 4: Schematic diagram of ANN structure

ANN for wind power forecasting

The systematic literature review for wind power forecasting in [35] confirmed that ANNs are considered the most frequently applied intelligence models in the literature for wind power forecasting in the past five years. These networks provided adequate results because of their ability to capture nonlinearity in wind patterns, especially for short and medium-term forecasting [35], [36]. The simplest type of ANN is the feedforward NN (FFNN); in [37], this network was used to predict a 2.5 MW wind turbine's monthly energy production. To train this network and increase forecasting accuracy, Nielson et al. selected wind speed and density incorporated with the atmospheric stability (represented in turbulence intensity, Richardson number, and wind shear) as input features to this network. This proposed approach reduced the mean absolute error (MAE) of wind power estimation by 59% compared to the standard estimation method.

On the other hand, the authors of [38] compared the performance of FFBP-ANN with the other two ANN types, namely adaptive linear element (ADALINE)NN and radial basis function ANN(RBF-ANN) for wind speed forecasting. According to the evaluation metrics, none of the three networks showed universally superior performance to the other. Nevertheless, RBF-ANN resulted in favorably accurate predictions when utilized in the forecasting stage in the hybrid wind forecasting model in Ref [39].

Although the logarithmic sigmoid function is the most commonly exploited transfer function, the work in [40] showed that building an ANN with two hidden layers with two different activation functions, hyperbolic tangent transfer function in the first hidden layer and sigmoid transfer function

``

in the second hidden layer could improve wind energy prediction accuracy and adequately map the data's features. It is essential to mention that, in this work, the monthly maintenance hours were used with metrological data as inputs to this ANN. Simulation results demonstrated that considering maintenance hours as an input improved the model reliability since they are inconsistent from month to month and directly affect the power production. Furthermore, another work incorporated the differential polynomial function in the ANN to build a wind speed correction model. This work's findings illustrated that the differential polynomial function could model an existing complex system by solving and forming differential equations. On the other side, wavelet neural networks (WNN) are also well-known powerful prediction tools when high accuracy predictions and fast convergence are needed [41]. For instance, the sine activation function was incorporated with the rough concept to build a rough sinusoidal ANN in [42]. This work showed that the rough sinusoidal function handled the dramatic changes and the erratic stochastic behaviors in wind speed, especially at the peaks.

The fuzzy concept also showed powerful, promising performance in wind prediction approaches. Although training an adaptive neuro-fuzzy inference system (ANFIS) is time-consuming and considered complex, it is a universal estimator that lowers convergence errors [43]. For instance, Liu et al. in Ref [44] employed a hybrid ANFIS approach for 48-hour-ahead short-term wind power forecasting. This approach combines the predicted power by three different forecasting models and outputs the final forecasted power. By comprehensive performance comparison between the hybrid proposed model and three individual forecasting models, namely RBF-ANN, BPNN, and least square support vector machine (LSSVM), the authors demonstrated that their hybrid methodology has superior performance with respect to reliance and accuracy. In addition, unlike the three models that their accuracy differs from season to season, the ANFIS model significantly improved the forecasting data throughout different seasons.

ANN for solar power forecasting

Similar to wind forecasting, solar forecasting is widely achieved by the different types of ANN approaches. This section will review some proposed ANN-based methodologies for forecasting solar irradiation and power.

The work done in [45] showed that the 14 input FFNN outperformed the well-known multilinear regression methodology for hourly solar power forecasting. Despite the importance of normalization of input data that is always discussed in the literature, this paper's analysis showed that the normalized

``

input data does not significantly improve the forecasted data's accuracy. Nevertheless, their investigations revealed that data preparation and cleansing significantly affect the results and the ease of training the ANN. Moreover, the findings showed that eliminating the night hours from the input data could slightly improve the performance, and as expected, the predictions for clear sky hours and days were more reliable than cloudy and rainy days. To overcome the issue related to sky conditions for solar forecasting, O'Leary et al. in Ref [46] suggest using the input masking technique based on the error clustering in the time domain. They categorized time frames into four categories (Night, Sunrise, Day: when solar energy is consistent (on sunny days), and Sunset). Simulation results showed that input masking could improve the prediction outputs of the ANN by 1.3 %. They suggest performing the same input masking for different environments and scenarios to confirm the importance of masking.

The correlation factor of monthly solar energy prediction was enhanced by 9 % in Ref [47] when the ANN was hybridized into the non-linear autoregressive method. Besides improving accuracy, this hybridization reduced the inputs' size to the non-linear autoregressive approach, saving memory. The technique was simulated using data from various sites with different climates in Nigeria to guarantee and prove the prediction generalization. This model generally showed adequate results for longer-term forecasting, which is considered essential for planning and scheduling solar power applications.

With all of the proposed forecasting techniques in the literature, choosing the most reliable prediction method became challenging. To discourse this issue, authors of reference [48] raised an essential question on how to perform a fair comparison that reflects the models' actual superiority concerning the data's nature. This question was addressed by comparing 68 ML and statistical techniques for 1-hour ahead global horizontal irradiance (GHI) forecasting, using data from 7 stations in 5 different climate zones in the US. This finding of this work contributes to suggesting the most appropriate prediction methodology for each specific climate zone.

Authors of Ref [49] reinforced the dramatic influence of feature selection of inputs on ML-based methodologies. They used neighborhood component analysis to select the appropriate inputs from a pool of 85 different inputs in their work. Their selection was based on regularization and minimization of a specific objective function that gives the most reliable daily solar irradiation forecasting results. The analysis results showed that evaporation rate, maximum air temperatures, albedo, cloud cover, relative humidity at maximum temperature, and specific humidity at 1000 hPa are the inputs that resulted in more accurate predictions. Afterward, they compared the performance of different ML

``

techniques, including SVM, process gaussian, and ANN. According to statistical evaluation metrics, a feedforward backpropagation ANN with Levenberg Marquardt as a training function shows significantly superior performance in the five different sites in Queensland in Australia.

2.1.4 Recurrent Neural Networks-Based Methodologies

Although the FFNN, as discussed before, is adequate for presenting the pattern that relates specific output into a set of inputs, it learns the pattern of the targets independently without having any context or memory of the previous targets [50]. To tackle this issue, (RNN) was introduced and used for time series forecasting. RNN is a subset of ANN, and it shows a robust performance when the order or the sequence of events or data matters and affects the following predictions [51]. Unlike the ANN, as shown in Figure 5, the RNN considers features from the current timestep inputs (x_t) and features from the previous hidden step (h_{t-1}). Figure 6 shows the simple structure of RNN with respect to node connections where the hidden neurons take two input sets, one from the input layer and the other from the hidden layer's output of the previous step. Holding and using information from the past time is considered a memory that relates the prior knowledge to the current one.

Nevertheless, RNN suffers from short-term memory, i.e., it cannot learn properly to preserve important information for long time sequences[52]. Moreover, during the training process of RNN, the error gradient starts to exponentially fall until it vanishes, interrupting the training process at early stages [53].

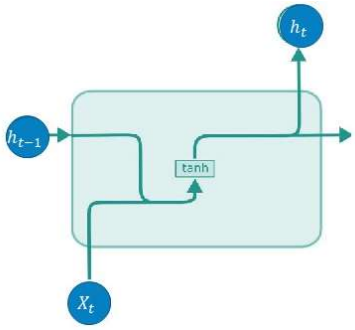

Figure 5: RNN cell

``

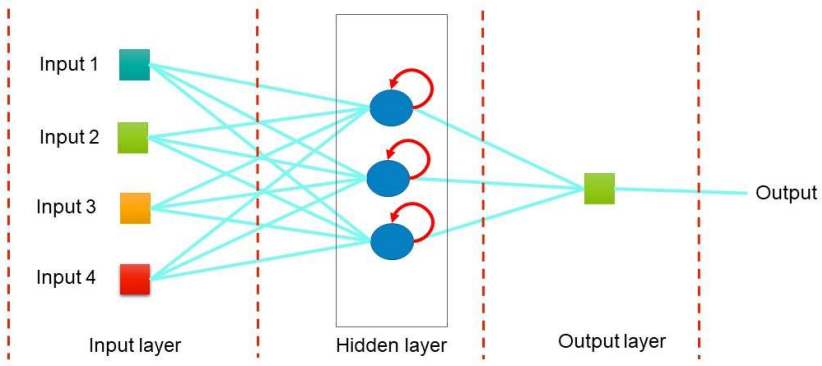

Figure 6: Schematic diagram of RNN

Two improved types of RNN nodes were proposed to overcome these issues, namely gated recurrent unit (GRU) and long short-term memory unit (LSTM). These two units have inner gates that can control the contribution of information from previous and current timesteps. By this, they pass significant attributes to long series sequences to predict and ignore un-significant information[52]. The GRU inputs are similar to the RNN ones; however, the mathematical operation that happens inside the GRU gates is slightly different. As shown in Figure 8, GRU's structure includes two gates, the update and the rest gate. The update gate decides on what previously stored information to remove and what new information to add. In comparison, the rest gate decides how much of previous attributes to overlook and forget.

On the other hand, as illustrated in Figure 7, the LSTM has four different gates (forget, input gate, cell state, and output gate). The forget gate is similar to the update gate in the GRU. The input gate takes the same inputs as the forget gate and processes them into sigmoid and tanh functions. The sigmoid function decides what information should be updated, and the tanh bounds the information between -1 and 1 to regulate the information's flow. Next, the outputs of the sigmoid and tanh functions are multiplied to generate the input gate's output. Afterward, the input gate outputs and the forget gate outputs are added to give the new cell state. Finally, the result passed to the output gate, which calculates the following hidden state.

The following section will review some published literature papers proposing wind and solar forecasting models utilizing RNN concepts.

``

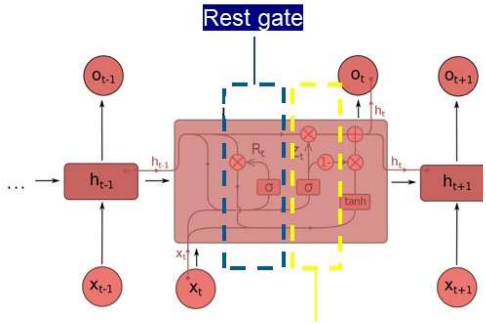

Figure 8: Structure of GRU

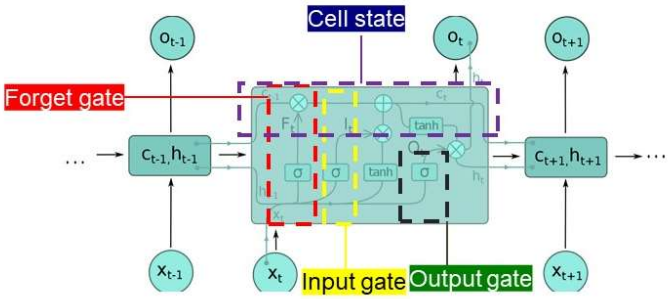

Figure 7: structure of LSTM unit

RNN for wind power forecasting

Since the recursive structure of the RNN can handle the complex nonlinearity in time series wind data, RNN has been employed in various references to manage the forecasting of wind power and speed. This section will review different classes of RNN proposed for wind forecasting.

Authors in [54] performed an ultra-short-term (15 minutes ahead) wind speed forecasting utilizing the GRU network. To determine the optimal input size required for training the GRU models, various input sizes were used. Results showed a considerable drop in the MAE when the input is the previous 30 timesteps. Nevertheless, MAE and RSME values start to fluctuate after the 30-input length. Thus, the 30 previous timesteps were considered adequate for forecasting in this paper. Afterward, to validate the model's accuracy, its performance was compared to the simple RNN and LSTM. Although LSTM was always known for its robust execution for time series forecasting, it did not perform better than the GRU approach. In fact, the GRU requires less parameter tuning and can be trained in a considerably shorter time. Besides, as expected, the simple RNN with the fastest training time performed poorly, especially at peaks where wind speed changes severely.

Consequently, it is more reasonable to consider using GRU when both the performance and training time are essential for forecasting wind speed. In fact, also for wind power forecasting, the same results confirming that the implementation of GRU is similar to the LSTM with faster convergence and less tuning were obtained in Ref [53]. To speed up the convergence, i.e., the training time of the LSTM, authors in [55] proposed an enhancement technique known as LSTM- enhancement forget gate (LSTM-EFG). In this approach, four modifications on the classical LSTM are performed: 1) two peepholes are added, 2) the tanh function is changed into softsign, 3) the input gate is completely removed, 4) the data

``

update value is determined by subtracting the output of the forget gate from one matrix. These modifications directly affect the forget-gate, which in its role accelerates the convergence. It is also important to mention that in order to maximize the LSTM-EFG approach's execution, a clustering technique combined with a temporal feature extraction methodology was incorporated into the system.

The work's conclusions verified the surpassing performance of the LSTM-EFG compared to the classical one and other benchmarking models. On the other hand, authors in Ref [56] claimed that the single wind power and speed predictions in some cases fail to be sufficient for electricity grid managing and scheduling. From this perspective, they suggested a multiple-input multiple-output (MIMO) model that forecasts wind power at different time horizons by a one-step simulation. In this model, an attention mechanism GRU coupled with a sequence-to-sequence technique is employed to select features. Unlike the classical feature selectors, which are applied once to discover the target's dependency on the inputs, the attention mechanism estimates all the inputs' relevance to the target wind power outputs and creates weights representing these dependencies. Besides that, the GRU blocks' hidden activations can extract both the spatial and temporal features for each time step, which contributes to improving the accuracy. Conclusions drawn from simulations confirmed that these two proposed strategies enhance the stability and accuracy of forecasting wind power simultaneously at different time horizons. Besides, the attention mechanism GRU lessened the error accumulation problem that always couple to the recursive forecasting models. In general, this proposed model resulted in the competitive performance of the LSTM with faster convergence.

RNN for solar power forecasting

Since the recursive construction of the RNN validated its ability to learn the patterns of time sequence data with seasonal and unstable trends, utilizing RNN for solar power/ irradiance forecasting also recently attracted the researchers' interest [57]. For instance, the comparative study in Ref [58] was carried out to compare different methodologies for long-term solar radiation forecasting (1-year interval). The simple RNN network and the RNN with GRU and LSTM units proved their effectiveness in learning temporal dynamic behavior between the inputs and outputs for this case study. Comparison results showed that these methods could accurately generate highly accurate outcomes with low means squared error (MSE) compared to the traditional forecasting techniques, i.e., random forest regression (RFR) and the conventional shallow FFNN. Another recurrent network, Elman-RNN, was trained by the cooperative neuro-evolution algorithm in Ref [59] to forecast the half-hourly PV power output. In

``

this paper, the suggested approach considered both univariate and multivariate models. The evaluation results, as expected, highlighted the improvement of the accuracy when training a multivariate model and verified the effectiveness of the proposed model by comparing it to three different persistence forecasting methodologies. Internal memory in this network that can deal with the variability of the PV data is considered a direct result of this promising performance.

Authors of [60] chose to utilize the recurrent networks, namely GRU and LSTM, to compare the univariate and multivariate approaches for direct normal irradiance hourly forecasting. They detected that computational-wise, GRU exhibited better performance than the LSTM because LSTM is computationally time-consuming with no significant superiority, especially for the multivariate approaches. Besides, to confirm the importance of incorporating the wind speed and direction and the cloud coverage data to the networks' input layer, they trained the networks with and without these inputs and compared the model's accuracy. The comparison results reinforced the significance and effectiveness of incorporating these inputs for irradiance forecasting, where the accuracy increased by 23.32 % and 8.91% for simulations with the wind and cloud coverage data, respectively.

Commonly the meteorological stations categorically report the daily sky condition without considering the variations from area to area throughout the day. These data, when used for forecasting solar power they negatively affect the accuracy of the forecasts. Aiming to address this issue and increasing the reliance on solar power forecasts, Hossain et al. [61] proposed an approach based on LSTM-RNN as the forecasting step. In their model, after performing a statistical correlation analysis to choose the most suitable predictors for the LSTM, a k-mean analysis approach was used to tackle the sky type issue. In this approach, solar irradiance was dynamically clustered for each hour of the day according to the sky type. These clusters create an hourly numerical approximation of the solar irradiances. Unlike classical sky type information represented for the entire day, the clustering technique makes an hourly synthetic weather forecast. These synthetic data are coupled with weather variables such as humidity, temperature, wind speed, and historical PV data fed as input to the deep-LSTM. When constructing a comparison simulation between the LSTM with the proposed approach and another two LSTM networks with hourly and daily categorical sky type data, findings verified the proposed approach's effectiveness to increase forecasting precision. Finally, to verify the LSTM's promising performance, it was compared to simple RNN, generalized regression NN, and ELM, all with the same synthetic input data. The LSTM followed by RNN outperformed the other two methodologies; this also supports the utilization of recursive forecasting structures.

``

2.1.5 Support Vector Machines-Based Methodologies

SVM is a powerful supervised machine learning technique based on a kernel-learning method that resolves the local minima issue that appears when training ANN [62]. The input datasets are mapped into linear features with a higher-dimensional space through a kernel function in SVM. This data mapping gives the SVM the ability to capture the nonlinearity in data and accurately predict erratic estimates such as wind and solar power [62]. In general, SVM is highly efficient in high dimensional spaces, comparatively memory effective, and resolves the local optimization problems in training ANN. However, in addition to its poor performance when the training data sets are relatively large, SVM's constrained optimization is computationally expensive. To overcome these drawbacks, (LSSVM) was recently introduced as a type of SVM with a loss function incorporating the summation of squared error (SSE) and transforming the inequality constraints to equality ones. This particular loss function of the LSSVM speeds up the training process and reduces the SVM's computational complexity [63]. Considering the appropriate kernel function has a significant impact on the performance of both the SVM and LSSVM. Linear kernel function, polynomial kernel function, radial basis kernel function, and wavelet kernel function are the most commonly employed functions in assembling the SVM.

SVM for wind power forecasting

As mentioned before, choosing the proper kernel function and tuning its parameters is a significant key when employing SVM models for forecasting. Thus, the authors of [62] suggested a new kernel function that can be incorporated into the SVM by holding the SVM's advantages and at the same time improving its accuracy in forecasting. This hybrid kernel function is a combination of a wavelet kernel function and a polynomial kernel function. Authors claim that this combined kernel function will preserve the good local interpolation ability in the wavelet function and, at the time, improve its extrapolation by combining it with a polynomial function. The claim was verified by training the SVM for ultra-short-term wind speed forecasting using this integrated kernel function. The cross-validation technique evaluated the performance, and the results showed that the hybrid function reduced the mean error by 3.94 %. In this paper, the input's dimensionality to SVM was reduced using the PCA approach, and the historical wind data were clustered according to their trend. This pre-processing step also contributed to improving the reliability of the proposed method. From the same perspective, in the study conducted in [64], the density-based spatial clustering of applications with noise (DBSCAN) clustering technique was employed after dimensionality reduction before using the SVM for wind speed

``

forecasting. This study also highlighted and reinforced the importance of clustering the data before implementing the SVM for forecasting, where the clustering decreased the MAE by 54 %.

Because the SVM models provide the generalization, utilizing SVM for wind speed and power forecasting attracted scholars' interest. Nevertheless, optimizing the performance and tuning the SVM parameters remains challenging, and no specific optimization algorithm has been highlighted to have superiority over the other. Accordingly, SVM models are accompanied mainly by various optimization techniques creating hybrid forecasting models. More SVM forecasting approaches hybridized with optimization procedures will be reviewed later in Section 2.17.

SVM for solar power forecasting

In general, SVM models can positively tolerate the noise and the volatility in data, and they can, in most cases, outperform the other ML techniques [65]. This superiority was also proven in the recent work in [66], where SVM was compared to ANFIS and ANN for estimating global solar irradiance in humid areas. The solar irradiance in damp locations is very chaotic and affected by the cloud coverage and the rainfall, and in fact, it is not tackled enough in literature. Therefore, the research conducted in [66]incorporated the rainfall as input to the three different ML techniques and tested their performance. As mentioned before, the results confirmed the superiority of SVM and illustrated the importance of considering the rain precipitation when forecasting the irradiance in such humid areas. The performance of ANFIS and ANN was almost similar, and no considerable supremacy was investigated.

To reduce the uncertainty in PV power generation forecasting and maintain the appropriate unit commitment in power plants, researchers in [67] suggested four different SVM forecasting models. Based on the seasons, four SVM models were trained to predict power generation and PV module parameters independently. Weather and PV power historical data were used as inputs to the SVM models. RBF kernel and the polynomial kernel were tested to determine the suitable kernel function for each model. According to accuracy, reported simulations in this paper and comparisons showed that the RBF kernel performs better for PV module parameters forecasting than the polynomial kernel. In contrast, the polynomial kernel resulted in lower MSE and MAE for PV power production forecasting. In fact, the work in this paper can provide beneficial guidance for future work related to PV power plant management and scheduling.

As mentioned before, the time horizon of forecasting can also affect the accuracy of ML models. For example, Fahteem et al. applied LSSVM with RBF kernel to forecast the solar irradiance at different

``

time horizons [68]. Among different inputs, they utilized sunshine durations and other weather data as inputs to build the models. Results showed that LSSVM performs better for short-term forecasting, and the accuracy of models decreases for longer-term forecasting. In fact, these results go along with the conclusions obtained in [44] where the LSSVM did not result in an adequate model for 48 hours ahead of forecasting. The authors of [69] addressed the weakness of LSSVM in longer-term forecasting by hybridizing it with the 3-D wavelet transform for 24-hours ahead PV power forecast. Their proposed approach handled and reduced the high dimensionality of the inputs to the LSSVM and considered the Spatio-temporal features, improving the long-term forecasting results.

2.1.6 Extreme Learning Machines-Based Methodologies

ELM is a particular type of single-layer FFNN that does not require the backpropagation algorithm for training and weights update. Instead, the ELM uses the Moore-Penrose generalized inverse for estimating the target outputs [31]. Unlike the FFNN, this unique ELM structure reduces computational complexity and cuts the need for manually optimizing and tuning multiple parameters [70]. Nevertheless, since ELM's loss function is based on second-order statistics, it fails to perform with non-linear or non-gaussian data. Most of the wind and solar power-related forecasting models are built based on chaotic and non-linear data. Therefore, an individual ELM approach for both wind and solar power forecasting is limited literature. Generally, when the ELM models are used, an optimization algorithm or other forecasting technique is combined with ELM to improve the prediction models' reliance and increase their accuracy.

ELM for wind power forecasting

To improve ELM's ability to capture the non-linear pattern in data and increase the forecasting model accuracy, the authors of [70] proposed a wind power forecasting model based on ELM with a modified loss function. They incorporated kernel mean p-power error loss instead of the classical MAE loss function in ELM. When authors conducted comparative experiments, they concluded that from a performance perspective, this adjustment in loss function improved the accuracy and provided reliable results compared to the classical ELM. Nevertheless, it resulted in losing the extreme computational speed of the ELM, which is considered a primary advantage when using the ELM. Therefore, as mentioned before, generally in the literature, to preserve the benefit of rapid learning in ELM and at the same time generate reliable models hybridizing the ELM with an optimization algorithm is necessary.

``

ELM for solar power forecasting

Hossain et al. in [71] conducted a comparative study for hourly and daily PV power forecasting of three different girds using various ML techniques. Solar radiation, wind speed ambient, module temperature, and PV power output data were used to train the models. RBF kernel SVM, sigmoid ANN trained with Levenberg–Marquardt algorithm, and the ELM were all trained and evaluated. Reported experimental simulations illustrated that ELM could perform better for longer-term forecasting and has the highest learning speed than the other two ML techniques. Nevertheless, the authors highlighted that this ELM model could not tolerate exogenous input data and suggested addressing this issue in future work.

2.1.7 Metaheuristic Optimized Machine Learning Forecasting Methodologies

Generally, metaheuristics algorithms are implemented as a search guide to find the near-optimal approximate solutions that can improve specific systems' performance with moderate computational costs [74]. Based on the search strategy, metaheuristic algorithms are mainly classified into two main algorithm classes 1) trajectory-based algorithms and 2) population-based algorithms.

According to what has been discussed and reviewed in the previous sections, it is clear that although single ML models can be trained to forecast renewable power, in some cases ML models, are inadequate to fulfill the accuracy required for electricity sector applications. For example, these models can easily fall into optimal local values issues and fail to generate generalized forecasting models. Besides, determining the networks' optimal structure and tuning their parameters can be time-consuming and requires an enormous number of trial-and-error experiments [33]. Therefore, scholars supplemented various ML approaches and metaheuristics optimization techniques together to build computationally inexpensive effective ML networks and reliable prediction results [73].

The metaheuristics are used with ML networks for two different purposes: 1) tuning and estimating the model parameters during the training process 2) tuning hyperparameters related to the network's structure [72]. The following section will briefly review the proposed metaheuristics techniques for tuning the hyperparameters of different ML methods.

2.1.8 Metaheuristic Optimization of the Network Hyperparameters of the ML Systems

The ML networks' hyperparameters are the variables that are set to construct the network structure. Tuning these parameters is essential since they can directly affect the training algorithm's performance,

``

which will eventually have a crucial control on the precision of the prediction model that is being learned and trained [73]. These parameters generally obtain the networks' structures (i.e., the numbers of units in the hidden layer, the type of the activation functions) and the initializing schemes weights and biases based on the selected activation function [73]. Unlike the learning-related parameters, the network's structure hyperparameter tuning is mainly achievable through Grid search, random search, and Bayesian optimization[73].

The applications of metaheuristics for hyperparameters tuning are not notably found in the literature. Only a few scholars reported their application of metaheuristics for networks' hyperparameters tuning. For example, for wind power forecasting systems, the authors of [74] conducted a study to validate the importance of tuning the number of hidden neurons of a wind power ANN forecaster through the swarm and evolutionary optimization algorithms. This study defines the ANN prediction system's structure as applying particle swarm optimization (PSO) and differential evolution (DE) algorithms through an automated selection approach. The proposed models were tested for predicting the wind power of 10 different wind power stations in Germany. Reported results illustrated that the proposed automated system through the two optimization approaches reduced the prediction error for most power stations compared to the manually tuned ANN . The PSO tuning approach enhanced the prediction by 9.6% and the DE approach by 6.8. Another application route of this ML- metaheuristics integration can be found in [42], where authors employed the GA to determine the optimal configuration of a stacked denoising autoencoder that was used as a pre-wind speed forecasting approach to denoise the data before processing them into forecasting network.

2.1.9 Comparative Discussion of Machine Learning and Metaheuristic Methodologies

Machine learning techniques (ANN, RNN, SVM, and ELM) have been successfully utilized for renewable power forecasting. In many references, according to statistical evaluation metrics such as MAE, MSE, root mean squared error (RMSE), and correlation coefficient (R^2), those techniques confirmed their ability. They surpassed various traditional forecasting approaches, especially for short and medium-term forecasting.

With simple structures, ANN can capture the non-linear and chaotic features in data and generate reliable, accurate predictions, especially for short and medium-term forecasting horizons [76]. BPFF-ANN is a robust ANN known for its ability to map non-linear patterns usually found in solar and wind power. Nevertheless, this type sometimes fails to tolerate oscillations and quickly falls in the local

``

minima [75]. Besides, it suffers from a low convergence rate [76].On the other hand, the RBF-ANN is usually introduced for renewable power forecasting problems because it is faster in learning, and it is not computationally expensive compared to the regular BP-ANN [59].

Nevertheless, several parameters related to the training process or the network structure directly affect the models' reliability [77]. Tuning these parameters requires an integration of different optimization algorithms that are considered time-consuming in some cases. Besides, sufficiently large historical data is needed to train the networks. ANN-based models based on different time horizons and approaches to renewable power forecasting in recent literature are summarized in Table A.1 (Appendix A).

RNNs are special types of ANN that can preserve and utilize the features from previous time steps, making them able to learn to attain the temporal relations between data [57]. Although RNNs can generate accurate forecasting models, short-memory problems associated with them cause immature training issues. GRU and LSTM are special nodes introduced to overcome the RNN drawbacks; these nodes process data in different mathematical activation functions to benefit from the previous timesteps attributes with longer memory terms. They actively confirmed their superiority for time series forecasting with moderately short training times. However, this recursive mechanism in all types of RNN results in error accumulation, which causes exploding gradient concerns that affect the networks' training process [56]. Table A.2 (Appendix A) summarizes some studies that mitigated these issues, particularly for renewable power forecasting, and provided consistent, accurate results.

SVM approaches are also powerful ML techniques that are well-known for their global approximation abilities. They can simplify complex mathematical computations, and unlike the ANN, they can learn patterns with a moderately small size of datasets with little dependence on prior knowledge[78]. Nevertheless, their performance highly depends on the kernel function parameters, which requires the incorporation of optimization algorithms for tuning and training [79]. Furthermore, their prediction stability diminishes for longer forecasting horizons and when the extensive training dataset [73]. Furthermore, the overfitting issue also comes with the SVM training process, necessitating different resolutions during the training process[31].

System optimization requirement also appears when utilizing the ELM tools to estimate the appropriate weights and biases [69]. Although the convergence is quickly achieved for ELM training, this convergence could be premature in some case. Therefore, the created model fails to be generalized,

``

and forecasting precision becomes insufficient in some cases. In fact, this encouraged considering deep learning concepts with ELM approaches [33]. Furthermore, their applications in some cases are restricted to linear relationships and fail to present complex nonlinear patterns. Table A.2 (Appendix A). summarizes some recent papers utilizing SVM and ELM tools for wind and solar power forecasting.

The hybridized ML approaches with metaheuristics algorithms are recommended solutions to increase ML models' reliability and resolve their limitations. The metaheuristic approaches are used to tune the ML model's parameters or/and the networks' structure. Incorporating these metaheuristics aims to achieve adequate convergence, resulting in higher prediction accuracy than standalone ML methodologies. Tuning the hyperparameters related to the ML network structure is another challenge when using ML approaches. This challenge is regularly tackled through search grid, random grid, and Bayesian optimization. However, in some cases, it is a time-consuming process that some researchers prefer to depend on previous knowledge and experience to tune these hyperparameters.

Finally, Based on our investigations in this paper, to improve the ML-based forecasting techniques, the following steps are usually recommended: 1- increasing the dataset size, especially for ANN; 2- pre-processing and analyzing the data to detect and filter the outliers and missing data is essential and affects the prediction results; 3- the presence of NWP as input feature to the ML networks is crucial and improves forecasting results; 4- shorter-term forecasting horizons are preferable when using the ML techniques to ensure higher accuracies; 5-hybridizing the ML models with optimization techniques improves the outcomes, but might decelerate the training process in some case; therefore, it remains a trade-off process that scholars need to consider based on the forecasting applications; 6-hybridizing the ML approaches with metaheuristics improves the results of the multi-step prediction.

The following section will also highlight some vital challenges of utilizing ML methods for renewable power forecasting to direct scholars to the problems requiring higher focus and considerations in future studies.

2.1.10 Challenges and Future Directions

Even though many research studies are conducted on the ML methodologies for renewable power, some remaining significant questions and problems have not been efficiently tackled:

1- Minimal studies have been conducted for regional wind or solar power forecasting; most studies consider single locations or stations. However, regional electrical grid optimal scheduling and

``

managing would be achieved by constructing a model that forecasts the solar or wind power for multiple locations in a specific region. Hence, constructing a precise regional wind or solar power forecasting model is one of the critical problems to be tackled in the future.

2- Probabilistic prediction of wind and solar energies is not adequately considered in the literature. These predictions can quantify the changes in renewable energies' resources. This could improve the scheduling of the electricity networks based on the estimated odd operating conditions. Therefore, focusing on probabilistic forecasting of renewable is a future key direction for researchers.

3- While the one-step-ahead forecasting has been extensively studied and tested, the multi-step ahead forecasting proposed models remain a complex task that is not considered adequately in literature and needs to become more encountered by researchers.

4- Currently, most of the published studies do not look at the problem of renewable power forecasting through the core structure of the ML model; The mathematical correlations between the input features and the renewable power prediction targets are not fully systematically disclosed and explained. Moreover, the input attributes that majorly affect the forecasting behaviors and precision are not entirely unambiguously indicated. In other words, the appropriate mathematical way to describe the renewable power forecasting model needs to be seen by scholars in the future.

5- From the forecasting horizon perspective, it was investigated that the proposed ML methodologies in the literature mainly focused on very short and short-term forecasting. Although these time horizons of forecasting have various vital applications related to maintaining the microgrid's stability, medium, and long-term forecasting horizons are also essential for studying the economic feasibility of the renewable power integration to the electricity sector. Thus, a higher focus on longer-term forecasting is expected and needed and could improve the incorporation of renewables into electricity networks.

``

2.2 Machine Learning Models for Air Quality Index Forecasting

This section presents a review of basic definitions of the air quality index with a brief review of the proposed air quality index forecasting approaches. This section aims to give a brief insight into the tackled problem in chapter 5.

2.2.1 Background

In the past decades, urban cities' economic and technological growth triggered different severe pollution problems, including air pollution. Besides the harmful impact of air pollution on the individual's health, it damages the ecosystem and harms forest life, plants, and marine life[80]. Poor air quality is mainly caused by burning fossil fuels for power production, oil and petroleum processing exhausts, vehicle vents, and other human habits[81]. Air pollutants include various types of gases and particulate matter. Air pollution is reported whenever high levels of different pollutants are detected in the air. US Environmental Protection Agency (EPA) defined six criteria pollutants that dangerously affect the respiratory system of humans[82]. Sulfur Dioxide (SO_2), Nitrogen Dioxide (NO_2), ground-level Ozone (O_3), Carbon Monoxide (CO), and Particulate Matter (PM) are all considered criteria pollutants. EPA reported specific guidelines and standards of the acceptable levels of these criteria pollutants[83].

AQI quantifies the healthiness and harmfulness of the air based on the concentrations of the pollutants and reports the health effects associated with these concentrations. The AQI standard ranges specified by EPA are shown in Table 1 [83]. Appendix B presents the official agreed-on procedure of calculating AQI.

Forecasting the criteria pollutants levels in the air to report the predicted AQI of the following hours would help the inhabitants prevent hazardous and low-quality air exposure. Besides, it could help decision-makers plan future operating conditions and/or cut off certain polluting activities at predicted peak pollution hours. Thus, developing accurate levels forecasting models hourly to report the AQI could provide reliable pollution alerts, protect the populations' health, and improve the ambient air quality.

``

2.2.2 Related Work

Based on the prediction approaches, AQI forecasting can be divided into two main classes: statistical and (ML) based.

Statistical forecasting methods build data-driven mathematical models to map the relationship between the time-series historical data and target data. With simple mathematical formulation, these methods can provide timely, accurate predictions. ARIMA is a well-known statistical forecasting approach that is usually employed for short-term forecasting. For example, ARIMA and auto-regressive fractionally integrated moving average (ARFIMA) statistical methods were used by [84] to forecast the monthly values of AQI in Malaysia.

Table 1: EPA's definition of the six AQI categories.

Index Value	Level of health concern	Description
0-50	Good	Air quality is satisfactory.
51-100	Moderate	Air quality is acceptable; however, there may be moderate health concerns for groups with unusual sensitivity to air pollution for some pollutants.
101-150	Unhealthy for sensitive groups	Only sensitive groups may experience health effects.
151-200	Unhealthy	All individuals may start to experience health effects. Sensitive groups may experience more severe effects.
201-300	Very unhealthy	Health alert: everyone may experience serious health effects.
301-500	Hazardous	Health warning for emergency conditions

The built models were able to predict the AQI with 95 % confidence. For shorter-term forecasting, ARIMA was utilized in [85] to forecast the daily values of AQI and the Holt exponential smoothing model. More recently, authors in [86] proposed a survey comparing classical statistical models for AQI

``

forecasting and concluded that ARIMA models have the superiority in mapping the trends and predict with the lowest RMSE comparing it to other statistical models. Nevertheless, these statistical-based models consider only the previous recorded level to forecast the following ones without accounting for the effect of other atmospheric variables and conditions. Moreover, unlike ML models, the statistical models require computationally expensive data pre-processing, especially in the case of discontinuity in historical data[87].

On the other hand, ML algorithms, with their proven superiority and effectiveness in various forecasting problems, integrating their applications into environmental-related issues has become more attractive to researchers. For example, Wang et al. in [88] applied a radial basis neural network to forecast the SO_2 levels and concluded that the achieved results could be promising for forecasting AQI in future researches. Similarly, [89] showed that a feedforward neural network was superior to multilinear regression in predicting different pollutant concentrations. The work proposed in [90] utilized another robust ML algorithm known as SVM for predicting the pollutants concentrations required to estimate the hourly AQI in California. The built models were able to predict the AQI with 94.1% accuracy with the testing unseen data. An alternative approach was proposed in [91] to forecast AQI directly through ensemble learning. In this study, the predicted AQI outputs from five different ML and regression models were further processed and fed to ensemble models to increase the forecasting accuracy.

From the brief review above, it can be seen that the accurate forecasting of AQI mainly relies on the forecasted pollutants concentrations in the ambient air. One significant issue associated with predicting pollutants concentrations is gaps in the ambient air quality and meteorological data. As a result of temporal dependencies between the two datasets, discarding observation with missing variables for training or building prediction models is generally impractical and affects the model's ability to capture the time relations between data accurately. Furthermore, imputing missing observations with mean or median values or any other single imputation approaches could fail to map extreme or abnormal behaviors in the data. Therefore, assigning values to these missing incidents with the consideration of other factors is essential, especially for the case of AQI predictions where the extreme and high values actually require attention and precautionary actions.

Therefore, considering the approaches of forecasting mentioned above and the missing data issues, in chapter 5, an approach is proposed to tackle the missing data problem using the miss-forest

``

imputation technique, a multivariate ML-based imputation technique to impute missing observations in ambient air quality and meteorological datasets. The employed multivariate imputation's effectiveness was examined using the imputed data for training ML models to forecast the critical pollutants levels and AQI.

``

Chapter 3

Concepts of Machine Learning Algorithms

This chapter gives a conceptual mathematical overview of different machine learning algorithms. All methodologies utilized in the cases studies in chapters 4 and 5 are conceptually explained in this chapter. In addition, the evaluation metrics for comparing and evaluating the performance of the models are defined in section 3.8.

3.1 Artificial Neural Networks

ANNs are one of the most robust ML methods. They are known for their superior ability to capture and map complex nonlinear relations between data [92]. The learning approach in an ANN aims to simulate the actual biological neural networks in humans' brains. The ANN consists of connected units (artificial nodes) transmitting and processing information in specific functions between them to map complex relations. The simplest type of ANN is the feedforward neural network. In this type of network, weight parameters connect these networks' units. These weights regulate the learning process to map the input features into the targets with the lowest possible error between outputs and actual targets. Typically, the nodes are distributed into layers; each layer with its activation functions may perform different processes on their input before being fed into the following layers. The layer that takes the input features is the input layer; the layer that generates the final prediction of the target is the output layer, the layers between these two layers are the hidden layers. The number of nodes in the input layer is similar to the number of input features; the output layer units are equivalent to the size of the targets, while the number of nodes in a hidden layer is a hyperparameter that requires proper tuning and selection to avoid overfitting or underfitting the built models[93].

As shown in Figure 9, in this network, the input predictors(x) flow sequentially from the input layer to an intermediate layer (hidden layer) and finally to an output layer to estimate the final forecasted target ($\hat{y}$).

The feedforward equations for estimating the final target $\hat{y}$ are defined as follows:

$$H_1 = f_1(w_1 X + b_1) \quad (1)$$

$$\hat{y} = f_2(w_2 H_1 + b_2) \quad (2)$$

Where f_1 is the activation function in the hidden layer, it is generally selected to be the sigmoid

``

function. f_2 is the activation function in the output layer, a linear function in the case of regression problems. w_1 and w_2 are the weight matrices to connect the layers, and $b_1$and b_2 are the biases.

The weights and biases are initialized randomly at the first epoch (iteration) of training and adjusted according to the selected training algorithm. The backpropagation with gradient descent algorithm is the most common approach for adjusting the networks' parameters. The backpropagation equation for updating the network's parameters (w_1, b_1, w_2, b_2) at iteration k using gradient descent with momentum algorithm is defined as follows:

$$\varphi(k) = \varphi(k-1) - \alpha \frac{\partial E(k-1)}{\partial \varphi(k-1)} + \beta \Delta\varphi(k-1) \quad (3)$$

φ represent the parameters that are being trained, E is total sum square error (objective function) in iteration k, β is a momentum factor [0,1], and α is the learning rate between 0 and 1.

The gradient descent with momentum algorithm was mainly used in our applications because training a network with accounting the history of the parameter's updates eases the training process. Moreover, the momentum hyperparameter directly accelerates the training process and soothes out the noisy oscillations [94].

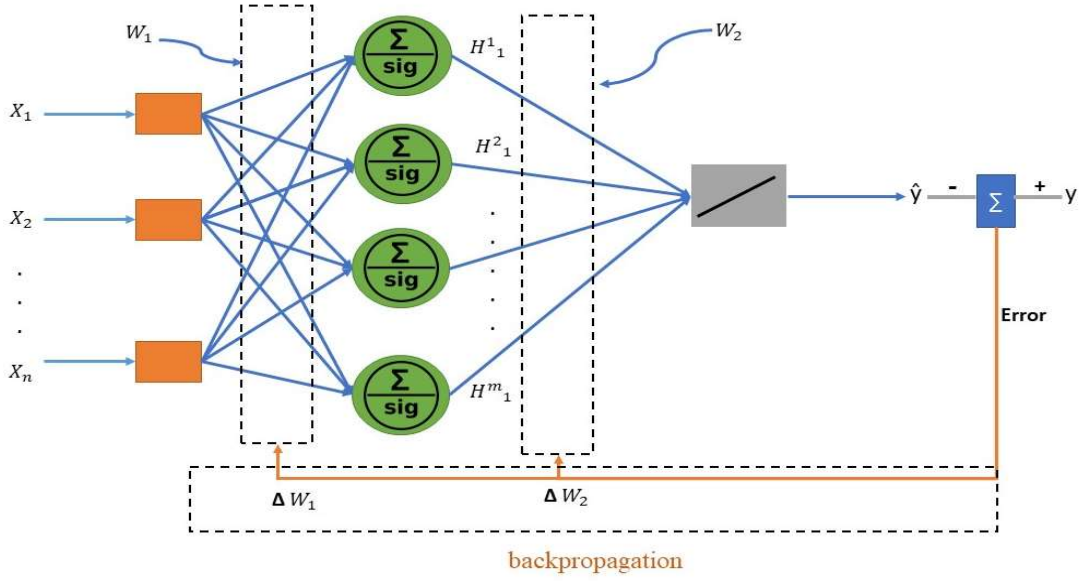

Figure 9: Structure of ANN

``

3.2 Deep Neural Network

The deep neural network (DNN) is simply a neural network with more than one hidden layer [95]. The additional layers in this network allow further data processing into the activation functions aiming to improve the forecasting accuracy. A DNN can replace the feature selection or pre-processing step before the ANN because, with training, the first layers can learn what attributes are significant to predict the target accurately and what attributes are negligible[96]. Nevertheless, the additional layers increase the number of hyperparameters requiring tunings, such as the number of nodes in these layers, the type of activation functions, and the number of the extra layers themselves. In addition, the multiple complex computing tasks that happen in these networks are expensive and require advanced computing machines; otherwise, they are very time-consuming [97]. The general training procedure of a DNN and weights and biases adjustments is similar to the one followed for training an ANN.

3.3 Long Short-Term Memory

As explained and showed in section 2.1.4, LSTM is a type of a RNN. This method is mainly introduced to overcome the short-term memory issue that appears in RNN. Moreover, it prevents the error gradient accumulation issue that causes the early truncation of the training process of the RNN.

As seen in Figure 10 below, inside the LSTM node, multiple parallels and sequential computation tasks happen in different functions to generate this unit's output. The mathematical processes that occur inside the LSTM node are as follows:

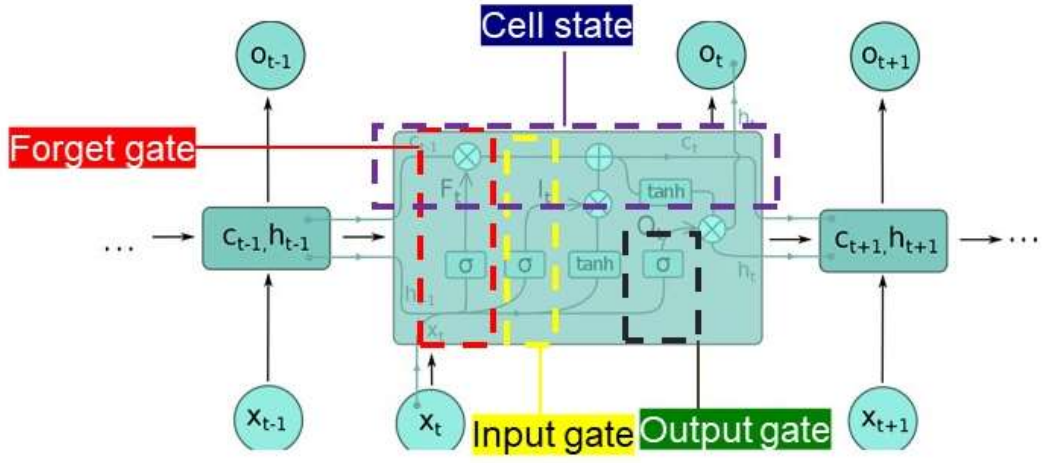

Figure 10: Structure of LSTM unit

``

Forget gate: $$f_t = sigmoid\left(W_f X_t + W_{fl} h_{t-1} + b_f\right) \quad (4)$$

Input gate: $$i_t = sigmoid(W_i X_t + W_{il} h_{t-1} + b_i) \quad (5)$$

Output gate: $$O_t = sigmoid(W_o X_t + W_{ol} h_{t-1} + b_o) \quad (6)$$

Candidate: $$\widetilde{C_t} = tanh(W X_t + W_l h_{t-1} + b) \quad (7)$$

Memory cell: $$C_t = f_t .* c_{t-1} + i_t .* \widetilde{C_t} \quad (8)$$

Hidden state: $$h_t = O_t .* \tanh(C_t) \quad (9)$$

The equations show that the LSTM has four different gates (forget, input gate, cell state, and output gate). The forget gate is the gate that decides what previously stored information to remove and what new information to add. The input gate takes the same inputs as the forget gate and processes them into sigmoid and tanh functions. The sigmoid function decides what information should be updated, and the tanh bounds the information between -1 and 1 to regulate the information's flow. Next, the outputs of the sigmoid and tanh functions are multiplied to generate the input gate's output. Afterward, the input gate outputs and the forget gate outputs are added to give the new cell state. Finally, the result passed to the output gate, which calculates the following hidden state.

3.4 Bagging Regression Tree

Bagging tree (BT) is a bootstrap ensemble model that improves the performance of a single decision tree model by combining prediction results from multiple decision trees. Each decision tree model is trained using a sub-set of the overall training set randomly drawn with replacement. The final prediction is estimated by taking the average value of the predictions of each tree model.

``

The induvial decision tree model is one of the simplest commonly used supervised models. As shown in Figure 11, this algorithm is nothing but a set of nested if statements or true/false questions with conditions sat based on the values of the attributes until the final node (leaf node) is reached. In the classification problems, the pure final leaf node will contain only one class. Therefore, when predicting a class using the single decision tree, the decision conditions that satisfy the features of the prediction target will lead to a final leaf node, having a specific class; this class will be assigned as the predicted class of that target from this single tree. In a similar approach, other trained trees will classify the target, and the final prediction of the bagged tress is estimated through majority voting of the classifications from all of these tree models [98].

On the other hand, for regression problems, decision conditions will not classify the data to a final value in the leaf node; the regression tree leaf node will contain a range of values; the regression tree classifies the data into regions. The final prediction value is the average of the values in that region. Finally, the bagged tree model will take the average of predicted values from the different trees.

Now, the question remains, what are the optimal splitting rules that result in the best model. Here is where the machine learning concepts appear. Multiple different decision rules and conditions are tested throughout the training process to compare and select the optimal conditions that maximize the information gain function for the classification problems and the variance reduction function for the regression problems [99].

Although this tree algorithm is easy to train and understand, it fails to generalize and usually falls in the overfitting issue when tested on unseen data. This is because their dependence on training datasets is high. This overfitting issue can be explained by the greedy approach used for training the decision trees, where the nodes are trained once to then train the following ones without the re-consideration of the previous ones. Thus, although the ensembles of tress might improve the globality of the model, in some cases, it is not enough. To overcome this issue, the RF models were introduced to alter the training data randomly to produce different trained models with higher flexibility. More details and explanations of the RF models are presented in the following section.

``

3.5 Random Forest

As mentioned before, the RF model is based on the ensemble trees concept. However, in RF, the trees are trained on a random subset of the full features, not all features, using randomly selected samples from the overall dataset. RF model follows the same training algorithm of the bagging tree described in the previous section (section 3.4).

RF model is used not only for building forecasting models but also for measuring the importance of features for feature selection and filtration. The application of RF for feature selection is tested in the AQI forecasting case study in chapter 5. Moreover, in this case study, the RF model is also used to impute missing data; all details will be further explained later in chapter 5.

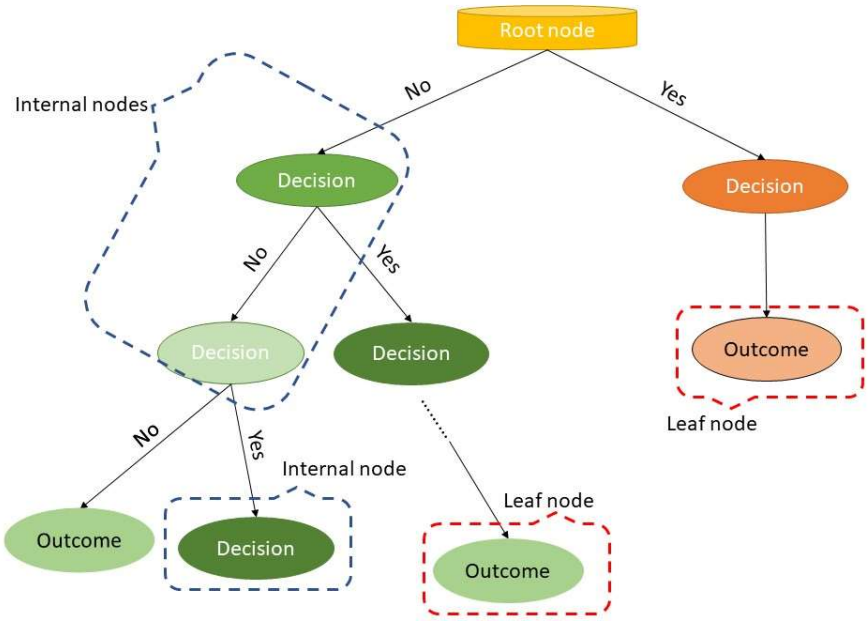

Figure 11: Structure of decision tree

3.6 Support Vector Machine

The literature review in section 2.1 concludes that the SVM / SVR is one of the most powerful ML algorithms that perform adequately for different regression and classification problems[100]. The objective of SVM is to find the hyperplane in N-dimensional space (N - number of features).

As shown in Figure 12, the data are scattered on the sides of the thresholds of that hyperplane. These thresholds represent the margins. Comparing SVR to linear regression, the SVR algorithms aim to find

``

the optimal line within these two margins[101]. In comparison, linear regression aims to minimize the distance between the data point and a straight line. The main objective of SVR is not to minimize the error to a certain degree; the aim is to ensure that the error lies within an acceptable range. Thus, the hyperplane that satisfies the SVR should satisfy the equation below[66].

$$-\varepsilon < Y - wx + b < +\varepsilon \tag{10}$$

What makes SVR unique is that its cost function minimizes the training error and the regularization term[101]. This specialty improves the model's overall generalization and results in reliable prediction results when tested on a new testing dataset.

The mathematical representation of general objective function for training the SVR model is as follows:

$$minimize\ \frac{1}{2}w^T w + C\sum_{i=1}^{m}(\xi^+{}_i + \xi^-{}_i)$$

$$subject\ to\ \begin{Bmatrix} y_i - w^T\varphi(x_i) - b \le \varepsilon + \xi^+{}_i \\ w^T\varphi(x_i) + b - y_i \le \varepsilon + \xi^-{}_i \\ \xi^+{}_i\xi^-{}_i \ge 0 \end{Bmatrix} \tag{11}$$

Where w is the weight vector, b is the bias, and φ is the high dimensional feature space, linearly mapped from the input space x (linear projection of the input space to the feature space). The objective is to find the optimal function with the slightest double-sided deviation ($\pm\varepsilon$) from the targets. Usually, the minimization of the langrage function is used for the target dual optimization problem. In SVR, the dual form of the constraint optimization problem (equation 12) is represented as follows:

$$minimize\ \frac{1}{2}\sum_{i,j=1}^{m} K(x_i, x_j)(\alpha_i^+ - \alpha_i^-)(\alpha_j^+ - \alpha_j^-) + \varepsilon\sum_{i=1}^{m}(\alpha_i^+ - \alpha_i^-) - \sum_{j=1}^{m}(\alpha_j^+ - \alpha_j^-)$$

$$subject\ to\ \begin{Bmatrix} \sum_{i=1}^{m}(\alpha_i^+ - \alpha_i^-) = 0 \\ \alpha_i^+, \alpha_i^- \epsilon [0, C], \end{Bmatrix} \tag{12}$$

K is a kernel function; by this objective function formulation, the SVR becomes employable for nonlinear relations while preserving the simplicity and practical computation of the linear SVR[101]. Thus K, C, and ε are the parameters of the SVR that we train the models for optimally estimating them.

``

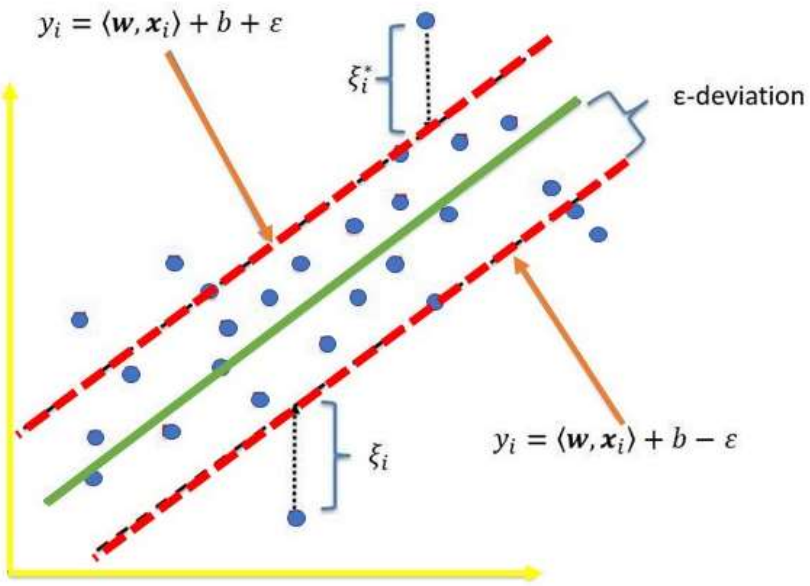

Figure 12: Illustration of support vector regression

3.7 Ensemble Models

Ensemble models are generally built based on the hypothesis "unity is strength. " The basic idea of ensemble learning is to train multiple models for the same problem to use them as input blocks to build another model [102]. The main advantage of ensemble modeling is that when combining models with high bias and low variance with other models with lower bias and higher variance, a balance between them is achieved by preserving the advantages of the combined models.

As mentioned, for training an ensemble model, a pre-training of base models is required. Some ensemble methods require homogenous base models trained in different ways. Other heterogeneous models use heterogenous base learners and combine them, creating a heterogeneous ensemble model. Combining these base models is also important and affects the overall performance of the final model [103]. Combining models can be achieved through the following three approaches:

1- Bagging: This method combines homogenous predicters after being independently trained in parallel using the deterministic average, aiming to produce a model with lower variance.

``

2- Boosting: In boosting, the base models are trained sequentially; each model depends on the previous one. Then, they are combined either using adaptive boosting or gradient boosting deterministic algorithms.

3- Stacking/Blending: In this approach, heterogenous methods are combined after being parallelly trained and connected through training a final model taking the predictions from the obtained base models. The base models are used to train the final meta-model in blending the hold-out validation predictions determined by the base models.

In chapter 4, after training different ML models, a blending ensemble model is constructed using these trained models.

3.8 Evaluation Metrics

After clearly explaining the different ML forecasting models, the objective of the following chapters is to implement these methods into real-life problems to propose approaches for forecasting time series data. Employing the methods for testing them necessitates a clear definition of metrics that can be used to evaluate these built models, conduct comparisons, investigate superiorities, and draw out conclusions and recommendations. These evaluation metrics are defined based on the forecasting error with respect to the actual values. Several metrics are discussed and explained in this section to illustrate the basis of selecting a superior model or the analysis of the results that are conducted in the case studies.

The following notation is followed in defining these metrics:

y_t: Actual observed value.

$\hat{y}_t$: Forecasted value.

$\bar{y} = \frac{1}{n}\sum_{t=1}^{t=n} y_t$: Mean value of the data set where n is the size of that set.

Equations (13-17) show the MAE, mean absolute percentage error (MAPE), MSE, RMSE, R^2 formulas, respectively.

$$MAE=\frac{1}{n}\sum_{i=1}^{i=n}|y_t-\hat{y}_t| \quad (13)$$

``

$$\text{MAPE} = \frac{1}{n}\sum_{i=1}^{i=n}\left|\frac{y_t - \hat{y}_t}{y_t}\right| \times 100 \quad (14)$$

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{i=n}(y_t - \hat{y}_t)^2 \quad (15)$$

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{i=n}(y_t - \hat{y}_t)^2} \quad (16)$$

$$R^2 = 1 - \frac{\sum_{i=1}^{i=n}(y_t - \hat{y}_t)^2}{\sum_{i=1}^{i=n}(y_t - \bar{y})^2} \quad (17)$$

``

Chapter 4

Case Study 1: Regional Wind Power Forecasting

4.1 Motivation and Contribution

Recalling the research gaps drawn out from the comprehensive literature review in section 2.1.10, the following points motivated us to conduct this case study:

1- Minimal studies have been conducted for regional wind power forecasting; most studies consider single locations or stations. Regional electrical grid optimal scheduling and managing would be achieved by constructing a model that forecasts the wind power for multiple locations in a specific region.

2- While the one-step-ahead forecasting has been extensively studied and tested, the multi-step ahead forecasting proposed models remains a complex task that is not considered adequately in literature and needs to be more encountered by researchers.

3- Currently, most of the published studies do not look at the problem of renewable power forecasting through the core structure of the ML model. The mathematical correlations between the input features and the renewable power prediction targets are not fully systematically disclosed and explained. Moreover, the input attributes that majorly affect the forecasting behaviors and precision are not entirely unambiguously indicated.

4- Lack of comprehensive and fair assessment of feature selection and ML forecasting methods for regional forecasting.

To sum up, in this chapter, one-step ahead and multi-step ahead regional wind power forecasting is studied by constructing different ML algorithms and an ensemble model combining them to conduct a fair assessment between these models, to build a reliable regional wind power forecasting model, and to comprehensively and carefully investigate the significant predictors required for this forecasting problem. The flow chart in Figure 13 summarizes the proposed problem formulation of this case study. In the following sections, each step is clearly explained and discussed.

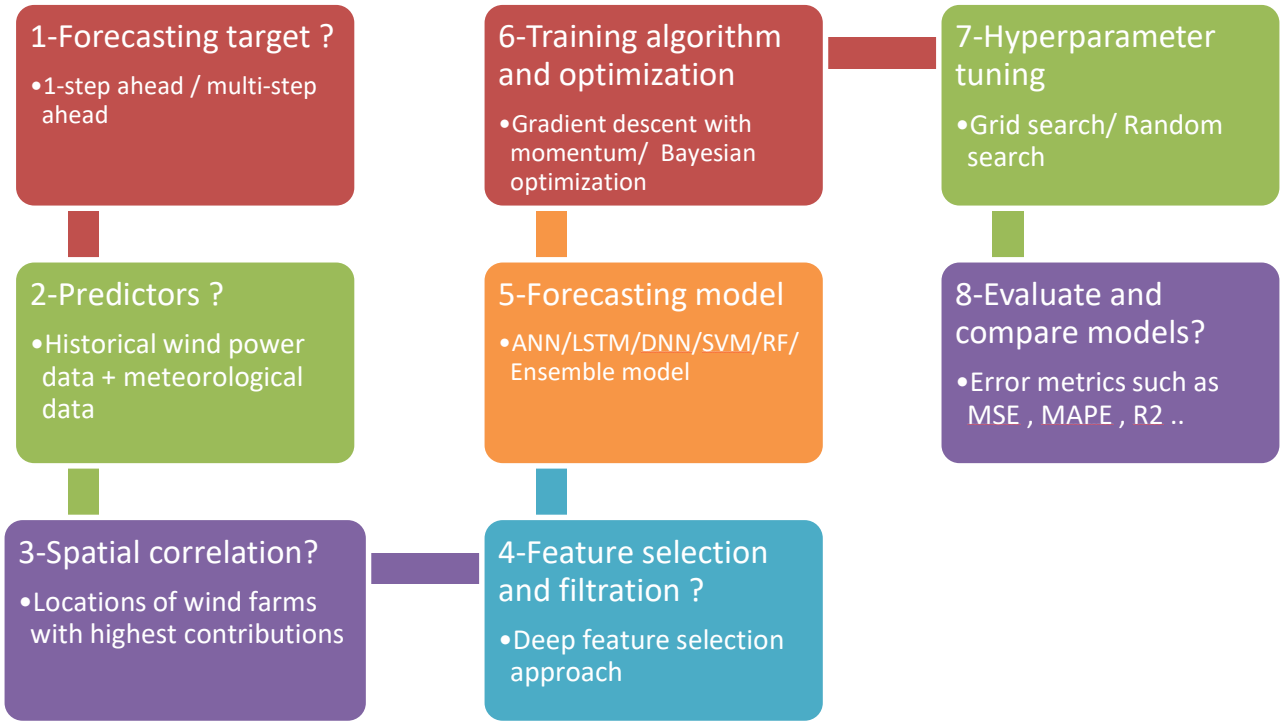

Figure 13: Flowchart of the proposed modeling approach

4.2 Raw Dataset Description and Pre-analysis

4.2.1 Power Data

This section aims to provide a brief insight into the wind power data used in this case study. This analysis is performed based on the available wind power data on Ontario Independent Electricity System Operator (IESO) website (public IESO data) during a period of eight years (2010-2018).

According to the Canadian Wind Energy Association, Ontario produces clean wind power with a leading rate across Canada as of December 2019[104]. Ontario's annual actual wind power production has increased gradually over the past ten years to reach 10.57 TWh in 2018. The increment trend is presented in Figure 14. Of course, the growth of wind power was associated with the growth of the

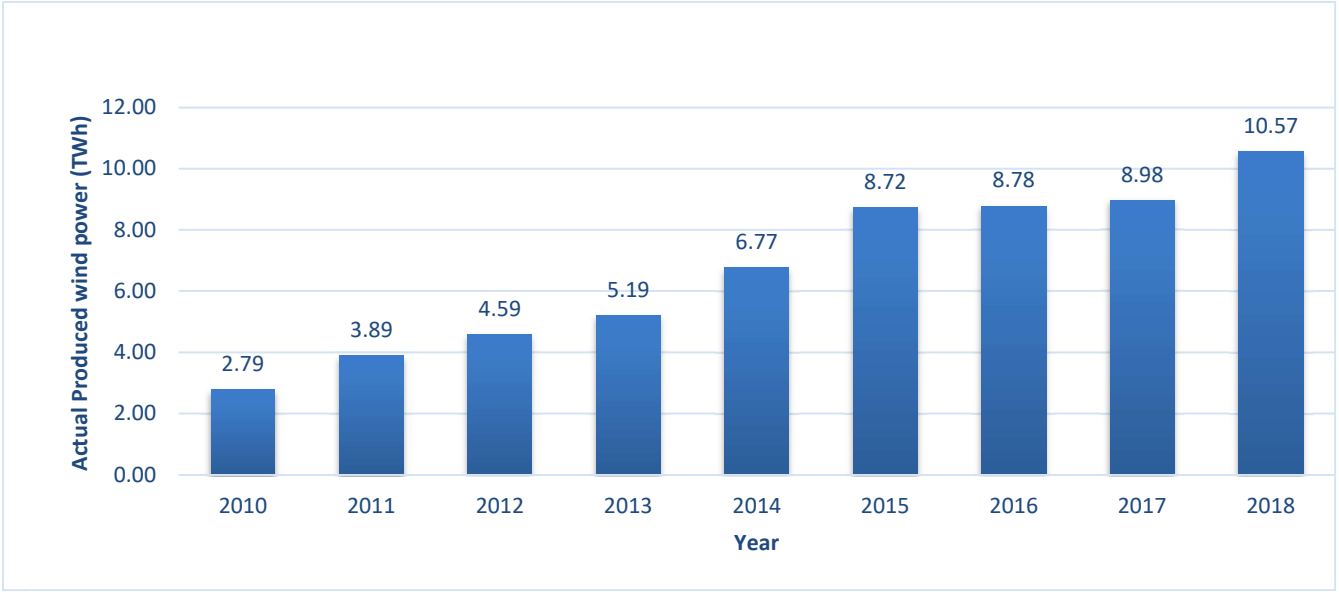

Figure 14: Ontario's annual wind energy output 2010-2018

number of functioning wind farms across the province. Table 2 below demonstrates the total number of operating farms through the study period. As shown in Figure 15, Ontario's wind farms are scattered along the region; however, it can be seen that the southern region is more dominant in the production where most of the farms are located in this area. Each wind farm has various numbers of wind turbines with different physical specifications and production capacities. The contribution of the wind farms to the total production was conducted to uncover the locations of the farms with the highest contribution to Ontario's overall wind power production. The location of these farms is essential due to the direct relations between the wind power and the metrological measurements at these locations where wind

``

speed is the primary key factor of estimating or studying wind power production. Moreover, other weather conditions could be necessary for building accurate, reliable wind power forecasting models, such as air temperature and relative humidity. Table 3 summarizes the individual production contribution of each wind farm to the overall production throughout the analysis period.

Table 2: Number of Ontario's operational wind farms

Year	Number of wind farms
2010	11
2011	15
2012	15
2013	18
2014	26
2015	33
2016	37
2017	38
2018	41

Independent Electricity System Operator |public IESO data

Figure 15: Location of Ontario's operational wind farms (2017)

``

Table 3: Annual production contribution percentage to the overall wind power production [104]

Wind Farm	Annual production contribution percentage to the overall wind power production								
	2010	2011	2012	2013	2014	2015	2016	2017	2018
ADELAIDE	-	-	-	-	1.0%	1.9%	1.5%	1.3%	1.4%
AMARANTH	15.5%	12.0%	10.1%	9.1%	7.1%	5.4%	4.8%	5.0%	4.0%
AMHERST ISLAND	-	-	-	-	-	-	-	-	1.0%
ARMOW	-	-	-	-	-	-	3.9%	4.2%	4.1%
ARNPRIOR	-	-	-	-	-	-	-	-	1.4%
BELLE RIVER	-	-	-	-	-	-	-	1.0%	2.3%
BLAKE	-	-	-	-	1.2%	2.1%	1.7%	1.5%	1.5%
BORNISH	-	-	-	-	1.3%	2.3%	1.8%	1.6%	1.6%
BOW LAKE	-	-	-	-	-	0.1%	0.6%	0.5%	0.5%
BOW LAKE 2	-	-	-	-	-	0.1%	1.2%	1.1%	1.0%
COMBER	-	2.5%	10.1%	9.4%	7.3%	5.4%	5.1%	4.7%	4.0%
DILLON	-	5.5%	5.1%	4.9%	3.6%	2.7%	2.6%	2.3%	1.8%
EAST LAKE	-	-	-	3.0%	4.3%	3.3%	3.2%	2.8%	2.6%
ERIEAU	-	-	-	2.8%	4.5%	3.3%	3.2%	2.9%	2.7%
GOSFIELDWGS	1.8%	3.7%	3.0%	2.8%	2.1%	1.6%	1.4%	1.5%	1.2%
GOSHEN	-	-	-	-	-	2.2%	2.6%	2.4%	2.4%
GOULAIS	-	-	-	-	-	0.5%	0.7%	0.8%	0.7%
GRAND VALLEY 3	-	-	-	-	-	-	1.2%	1.5%	1.2%
GRANDWF	-	-	-	-	0.4%	4.3%	3.4%	3.0%	3.0%
GREENWICH	-	2.0%	5.7%	4.8%	2.8%	2.6%	2.2%	2.2%	2.2%
JERICHO	-	-	-	-	1.0%	4.4%	3.7%	3.3%	3.3%
K2WIND	-	-	-	-	-	6.6%	7.6%	6.4%	6.3%
KINGSBRIDGE	3.8%	2.7%	2.4%	2.1%	1.6%	1.2%	1.1%	1.2%	1.0%
LANDON	-	-	-	-	-	1.2%	1.2%	0.9%	1.0%
MCLEANSMTNWF	-	-	-	-	1.4%	1.5%	1.6%	1.6%	1.4%
NORTH KENT	-	-	-	-	-	-	-	-	1.9%
PAROCHES	-	0.3%	3.1%	3.0%	2.3%	1.7%	1.6%	1.5%	1.3%
PORT BURWELL	8.1%	6.1%	5.0%	4.8%	3.4%	2.1%	1.8%	1.7%	1.7%
PORTALMA-T1	6.8%	-	-	-	-	2.8%	2.3%	2.1%	2.6%
PORTALMA-T3	6.6%	16.0%	13.1%	12.4%	9.5%	3.0%	2.5%	2.2%	2.8%
PRINCEFARM	16.3%	11.2%	10.3%	8.4%	5.9%	4.6%	4.4%	4.7%	3.5%
South Kent	-	-	-	-	8.1%	8.1%	6.8%	5.0%	6.0%
RIPLEY SOUTH	7.2%	5.5%	4.5%	4.4%	3.0%	1.8%	1.6%	1.6%	1.6%
SANDUSK-LT_AG_T1	-	-	-	1.0%	4.5%	3.4%	3.5%	3.2%	2.8%
SHANNON	-	-	-	-	0.4%	3.1%	3.0%	2.9%	2.5%
SPENCE	0.4%	7.2%	6.0%	5.4%	3.8%	2.6%	2.2%	1.8%	1.9%
SUMMERHAVEN	-	-	-	2.5%	4.9%	3.5%	2.5%	2.1%	2.1%
UNDERWOOD	16.8%	12.6%	10.7%	9.4%	6.5%	4.8%	4.0%	3.8%	3.6%
WEST LINCOLN NRWF	-	-	-	-	-	-	1.4%	5.0%	4.3%
WOLFE ISLAND	16.7%	12.8%	10.8%	9.6%	7.9%	5.7%	5.4%	5.3%	4.4%
ZURICH	-	-	-	-	-	-	1.0%	3.4%	3.0%

``

According to the determined percentages, it can be observed that the individual contribution percentage of each wind farm decreased over time because the total number of wind farms has increased; nevertheless, its average actual production rates remained the same or even increased with years. Currently, Ontario's biggest wind farm is Henvey inlet with a capacity of 300 MW; this farm was launched in 2019 [105] (not included in our study period).

Generally, as observed in Figure 14, wind power production started to evolve and increase a lot after 2014. The South Kent wind farm project that started operating in 2014 with a capacity of 270MW is a direct reason for this evolution and growth [106]. As seen in Table 3, this wind farm has contributed with high rates since the start-up until today. After this, K2-wind was launched a year later with a similar capacity and almost similar contribution percentages [107]. After a year, West Lincoln, with a capacity of 230 MW, started operating in 2016; this farm is located in the Niagara region in southern Ontario, where most of the other farms function with high production rates [108]. When observing its contribution during 2017 and 2018, West Lincoln farm contributed at higher rates than the other farms.

On the other hand, when looking throughout the entire study period, Amaranth farm also contributed with high rates since 2010, where it contributed by 15.5 % of the total annual outputs of 2010 and attained its moderately high percentage even with the increased number of total wind farms across the years [109]. Amaranth is one of the largest wind farms in Ontario, with a capacity of 199.5 MW. Similarly, wolfe island maintained a consistently high contribution from 2010 until 2018; wolf island was launched in 2009 [110].

As mentioned before, studying and observing the farms' contribution is vital to specify their locations and involve their meteorological conditions when building the wind power forecasting models. Based on the calculations and analysis above, five wind farms were selected to consider the weather conditions surrounding them to build the forecasting models. The selected wind farms and some related information about them are listed in Table 4.

From this section till the end of this case study, the locations of these wind farms are used to determine historical weather data that will be used for forecasting in the following sections.

``

Table 4: Summary of selected wind farms

Farm	Capacity (MW)	Lunching year	Location	Reference
South Kent	270	2014	42° 21' 1.3" N 82° 7' 11.8" W	[106]
K2-Wind	270	2015	43° 53' 44.6" N 81° 37' 21.1" W	[107]
West Lincoln	230	2016	42° 52′ 38″ N 79° 30′ 19″ W	[108]
Amaranth	199.5	2008	44° 06′ 00″ N 80° 16′ 15″ W	[109]
Wolfe Island	197	2009	44° 10′ 00″ N 76° 28′ 00″ W	[110]

4.2.2 Meteorological Data

As concluded in section 2.1, usually published papers in the scope of forecasting wind power apply their proposed forecasting models for predicting the output power of one specific site, where the weather data are primarily measured and recorded regularly with the power data. Nevertheless, in this case study, the target is the regional wind power from all the wind farms registered and owned by the IESO. Since weather conditions across Ontario differ, and no overall accurate representation of the entire province could guarantee precise weather features for the forecasting models, selecting a weather station measuring the essential parameters has become an issue that requires further analysis and search. Our proposed study addressed this matter by including the weather conditions recorded by weather stations near the wind farms with the highest contribution to Ontario's overall wind power output.

In the previous section (section 4.1.1), in Table 4, five wind farms were selected to be the most contributing wind farms to the overall regional wind power production. Using the longitude and latitude of the five selected wind farms, the nearest five weather stations were located to use their historical weather data as predictors to the ML models. The data of these weather stations were collected from the Government website, where all the historical weather data from various weather stations are available. The selected weather stations record more than 25 hourly weather parameters; six parameters are used in this study: temperature, relative humidity, wind speed, wind direction, atmospheric pressure, and dew point temperature. These factors were mainly used in all the research papers reviewed in section 2.1. The five stations' summary information is listed in Table 5. Six weather features were selected from each weather station, resulting in (6×5=30) weather parameters considered as input predictors to the ML forecasting models.

``

Table 5: Summary of selected weather stations

Station name	Longitude	Latitude	Climate ID	Elevation (m)
Chatham Kent (SK)	42°18'21.000" N	42°18'21.000" N	6131414	196.60
Goderich (KW)	81°43'00.000" W	43°46'00.000" N	6122847	213.70
Welland-Pelham (WL)	79°20'00.000" W	42°58'00.000" N	6139449	178.0
Mono Centre (AM)	80°01'28.010" W	44°01'56.100" N	6157000	436.0
Kingston A(WI)	76°35'48.000" W	44°13'33.000" N	6104149	92.40

4.3 Data Pre-processing and Feature Engineering

In the previous section, power data from 2010 – 2018 were used for analysis; however, three years of hourly data (2016- 2018) (3x8760) were used for training and building the forecasting model. Before filtering un-important features and selecting the appropriate lags, data were pre-processed and extracted. This section illustrates the steps taken for preparing the data for the deep feature selection performed in section 4.4.

4.3.1 Data Splitting for Training, Validation, and Testing

The overall dataset (3x8760 samples) was divided into three sets: training, validation, and testing datasets with 72 %, 13 %, and 15 %, of the overall size of data, respectively. The training set is used to train the forecasting model, the validation set is used to test the learned parameters at each iteration during the training process, and the testing set is used to test the fitted parameters after the learning stops. The model's training is truncated when the MSE of the validation set starts to increase even though the training MSE is decreasing. This data splitting step prevents overfitting of the model, and generalization is ensured [111].

4.3.2 Outliers and Missing Data Handling

Outliers in data: Data available on the IESO website is quite reliable, and no negative or significantly out of range high values were observed in the recorded wind power data.

Missing data: No missing data in the power data were observed, and for the weather data, missing entry was linearly interpolated using the previous and following observation. This estimation is considered reasonable since high temporal relations exist between the meteorological parameters. No

``

missing data were discarded because temporal relations between data are essential for the forecasting model.

4.3.3 Feature Engineering (Extraction)

Four new features were extracted to account for the temporal relations between weather and power data:

1- Year (2016-2018)
2- Month (1-12)
3- Day of the month (1-30/31)
4- Hour (0-23)

The presence of these features as predictors of the exogenous forecasting model would help the ML model capture the temporal relations and repeated trends in wind power.

An additional feature was added to relate the wind power at time=t to the wind power before 24 hours (time=t-24). This feature would also help the ML model in capturing the seasonality in data. It was calculated as illustrated in equation 18, where S is the seasonality feature, and WP is the wind power.

$$S(t)=WP(t)-WP(t-24) \quad (18)$$

Two additional features were added to account for the production of wind farms based on the number of the operating wind farms (from Table 2) and the summation of their designed capacities (Max output wind power).

4.3.4 Data Scaling

Scaling data to become within [0,1] is vital since predictors and also the added features (year, month, day, and hour, seasonality) differ in ranges. High variations between data could slow the training process of the ML engines and cause the issue of falling into the minimal local values, which results in unreliable, poor forecasting models.

In this case study, the predictors were scaled to become between 0 and 1 by applying equation 19

$$P_i^* = \frac{P_i(t)-\min(P_i)}{\max(P_i)-\min(P_i)} \quad (19)$$

``

P is a vector of predictors at time = t, i is the index of the predictor, and min is the minimum value of predictor i, and max is the maximum value of predictor i.

4.4 Deep Feature Selection

Filtering ineffective features is essential to improve the reliability of the forecasting model. Until this step, 37 exogenous predictors were selected (5 stations× 6 meteorological parameters + 4 temporal features + seasonality+ No. of operating farms+ Max wind power) at time t with additional endogenous feature (Wind power) at time = t to predict wind power at time = t+1.

This section proposed a deep feature selection approach to the three years data (2016-2018) to filter predictors, select appropriate lag features, and reduce data dimensionality.

Figure 16 illustrates the proposed deep feature selection approach. In the following sections (4.4.1-4.4. the results determined from applying these steps are discussed.

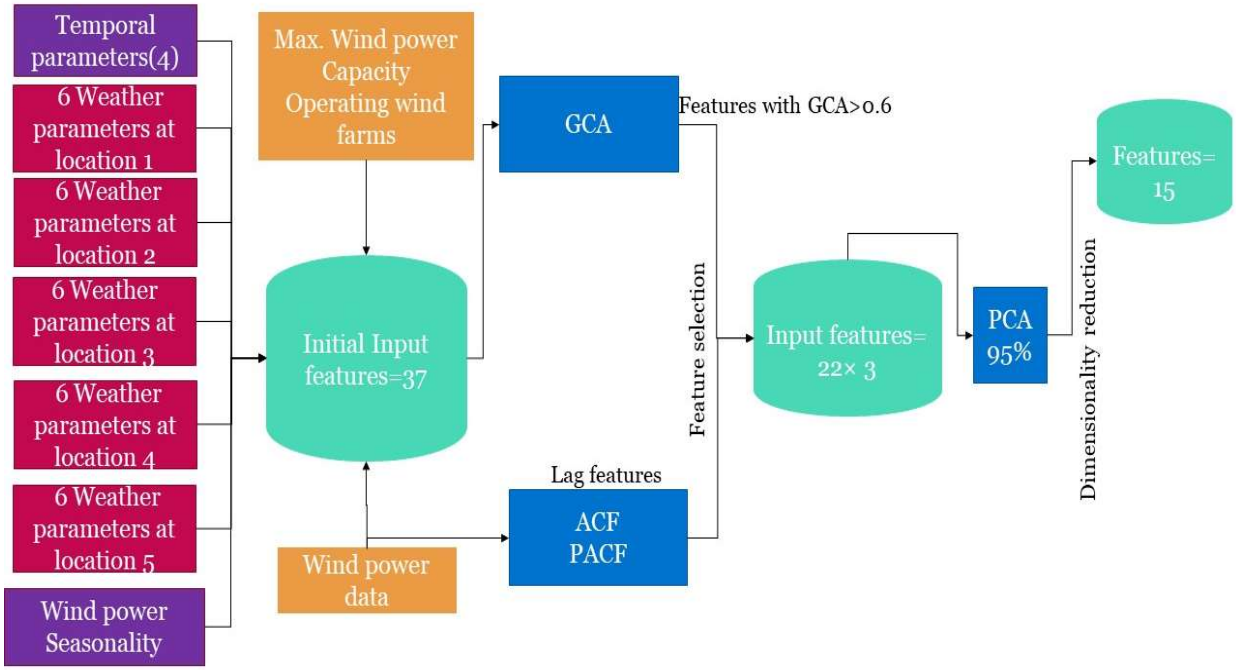

Figure 16: Proposed deep feature selection approach.

``

4.4.1 Grey Correlation Analysis Features Selection

This section will apply the grey correlation analysis (GCA) approach to select the most important features out of these 38 features. GCA approach is a good reliable approach for feature filtering and selection when dealing with big data (See Appendix C for details about GCA calculations). The determined correlation grades of the predictors to the target variable (Wind power) are presented in Figure 17. As expected, the highest correlation is observed for wind power with its previous reading. For the weather conditions, the results show that the wind speed from the different stations has the highest correlations compared to the other environmental data such as pressure and humidity. Similarly, the wind direction parameters have high grades of correlation. To select only the related and vital input features, predictors with GCA grades <0.6 were discarded to end up with 22 features (Previous wind power +Wind speed ×5 stations + Wind direction ×5 stations + Temperature ×5 stations + Year + month + day + hour + seasonality+ No. of operating plants).

``

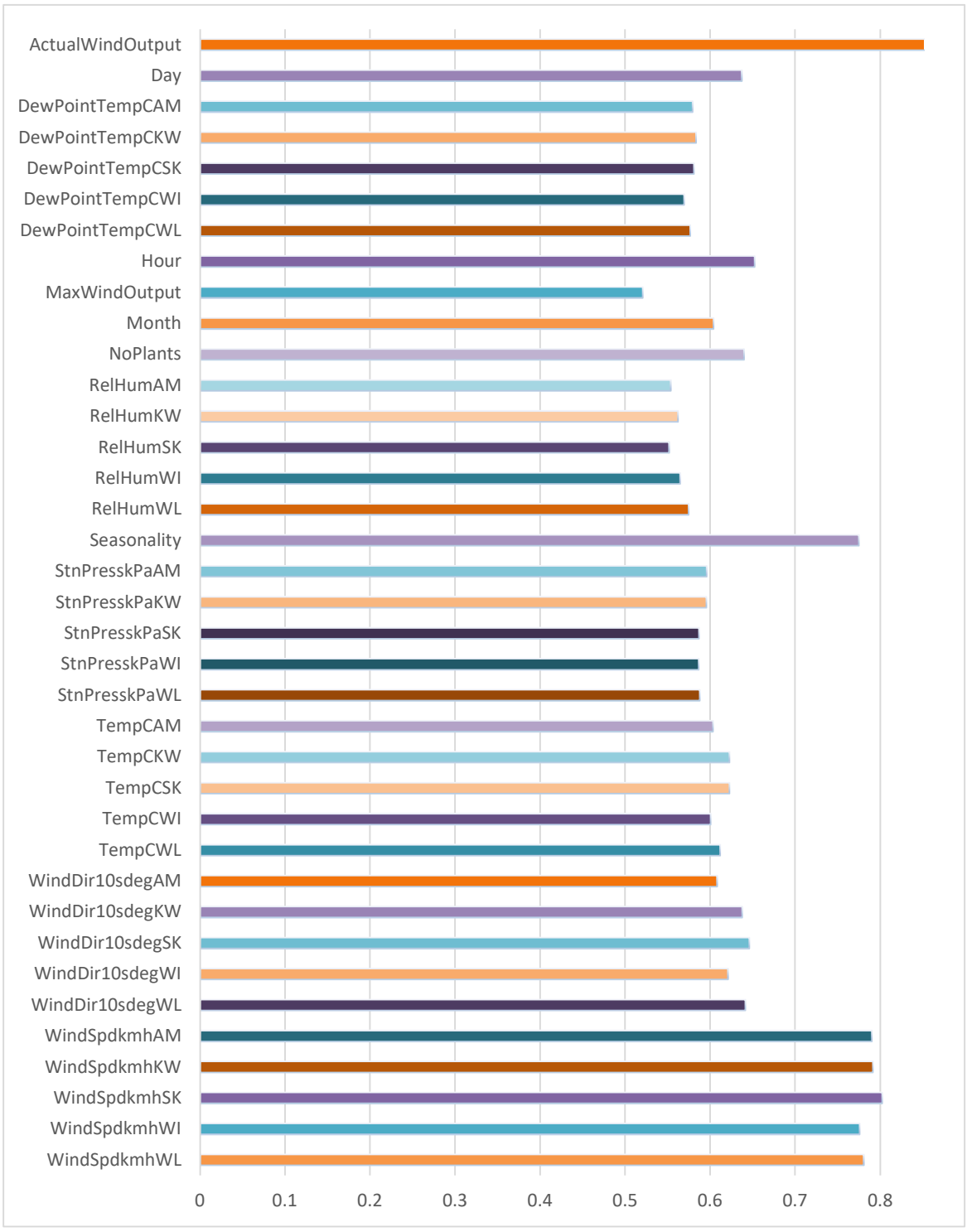

Figure 17: Grey correlation grades

``

4.4.2 ACF and PACF for Lag Feature Selection

Autocorrelation function (ACF) and partial autocorrelation function (PACF) are applied to determine the lag feature of wind power with respect to the previous hours (See Appendix C for a conceptual explanation of ACF and PACF). ACF and PACF are plotted in Figure 19 and Figure 18. As shown, wind power has a correlation =1 at lag =0 (itself) high correlation up until lag 24, where the correlation remains relatively high. This indicates that the seasonality feature created in the previous section was necessary, and that is why it showed a high grey correlation grade. Nevertheless, after removing the internal relation in PCF, it is observed that the lags 1, 2, and 3 are significant, and the correlation afterward starts to reach very low values. Based on that, a lag feature of 3 was used for all the 22 selected features to ensure the accounting of the previous temporal relations between all features.

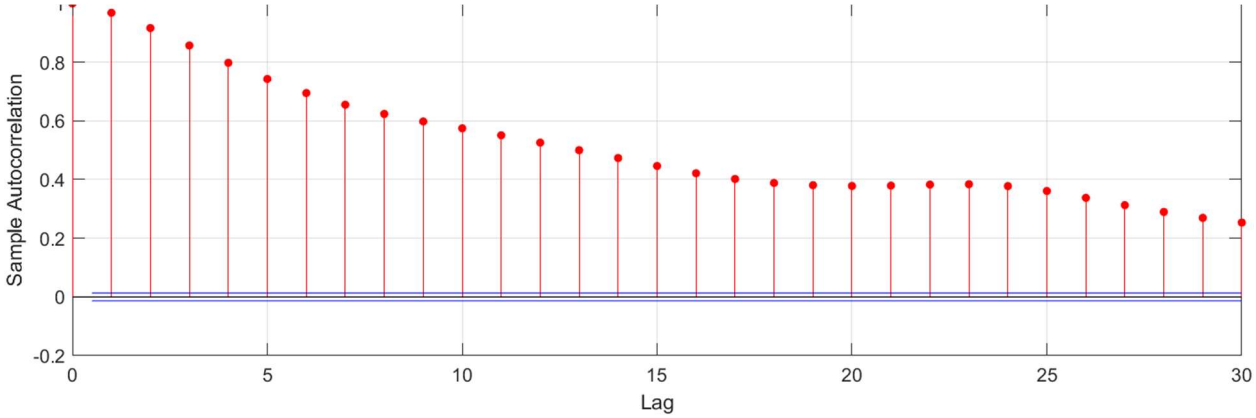

Figure 19: Correlogram of wind power lags

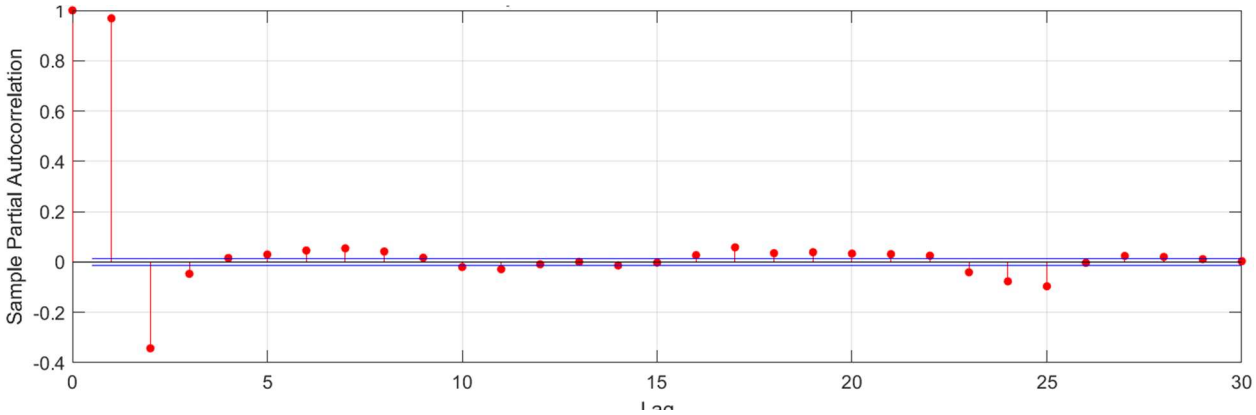

Figure 18: Partial correlogram of wind power lags

``

4.4.3 Principal Component Analysis for Dimensionality Reduction

After selecting the crucial predictors and the lag feature, the overall number of input features becomes (3×22=66) features. In our proposed model, PCA is employed to re-represent these predictors (See Appendix C for a conceptual explanation of PCA). This approach will result in lower dimension features approximately representing the original data in a lower dimension. As seen in Table 6, 90 % of the original information is cumulatively preserved 15 components, and the unnecessary, redundant information is scattered in the remaining components. Thus, 90 % of the data can be represented by 15 principal components. These 15 features are then used as input features to the ML prediction models for predicting 1-h ahead wind power.

Table 6: Principal component analysis results of the first 15 components

Principal component	Eigenvalue	Variation %	Cumulative variation %
1	3.508	24.324	24.324
2	2.292	15.893	40.216
3	1.544	10.706	50.922
4	1.015	7.035	57.957
5	0.936	6.491	64.448
6	0.882	6.116	70.563
7	0.764	5.299	75.862
8	0.519	3.601	79.463
9	0.403	2.798	82.261
10	0.325	2.253	84.514
11	0.244	1.693	86.206
12	0.226	1.566	87.772
13	0.162	1.122	88.895
14	0.130	0.904	89.799
15	0.116	0.802	90.601

4.5 Simulation Results 1 (1-step ahead Forecasting)

In this section, the final selected constructed input features (15 features) are used to build and train different ML algorithms to forecast Ontario's hourly wind power production. For each model, the hold-out validation error is tracked throughout the training process to truncate the training process whenever this validation error starts to increase while the training error is decreasing.

The training process aims to find the optimal model parameters that minimize the cost function (the error function). Some hyperparameters related to the construction of the model, such as the number of

``

nodes, hyperparameters of SVR, are tuned using grid search, random search, or Bayesian optimization. See Appendix D for a conceptual explanation of these hyperparameter tuning methods. The validation error is also used for this tuning step to avoid bias tuning and training towards the training dataset.

4.5.1 ANN Forecasting Model

In this step, in the beginning, multiple experiments have been conducted to determine the optimal number of nodes for the ANN for hourly wind power forecasting. As mentioned, the training processes for all of these experiments are truncated according to the validation error to avoid overfitting, and all the networks were trained using backpropagation with gradient descent algorithms with a hidden sigmoid function. As seen in Figure 20, the validation error metrics reach their lowest values with a network of 25 nodes, and then these metrics start to increase. This increment could reflect that the overfitting due to too many nodes or the network started to memorize the information in training data. Thus, the optimal number of nodes of the ANN was selected to be 25 nodes.

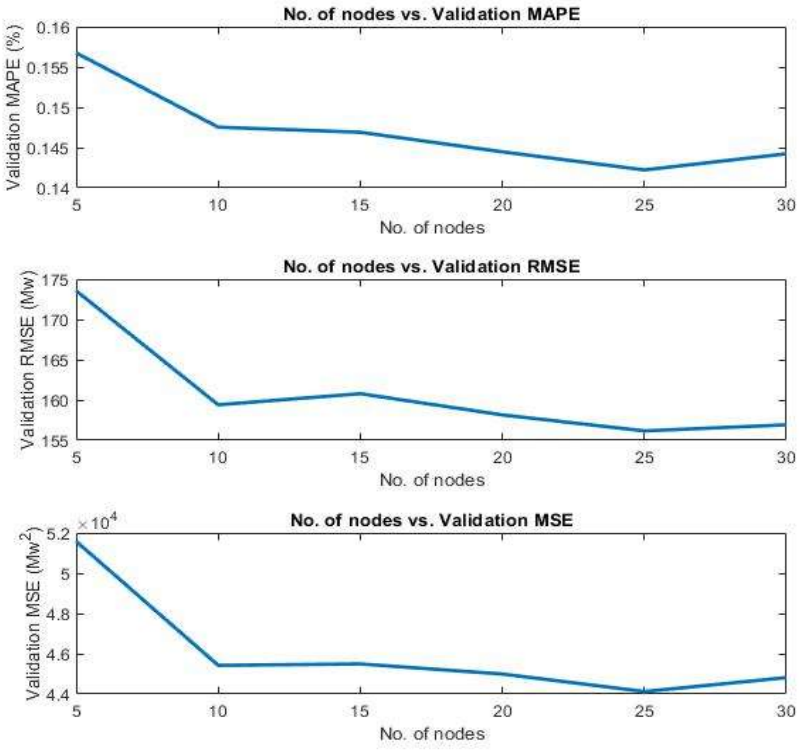

Figure 20: Different error criteria variations for various nodes of an ANN

``

Afterward, the network was re-trained with the 25 nodes, 0.001 learning rate, 0.9 momentum to determine the optimal weights and biases of the forecasting model. The learning curve of this ANN is presented in Figure 21.

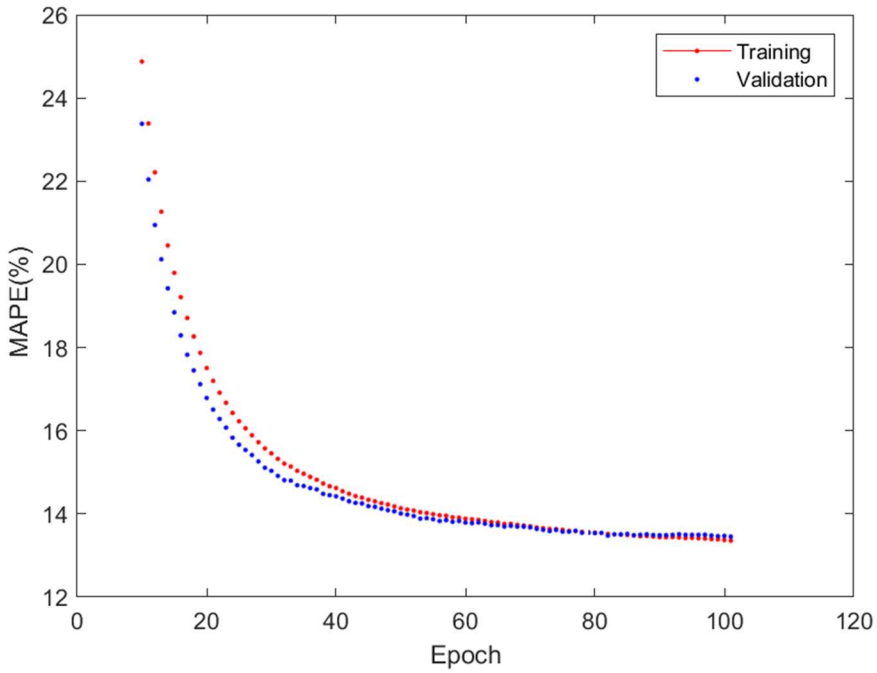

Figure 21: Learning curve of ANN (1-h ahead)

Finally, the evaluation metrics defined in section 3.8 are used to evaluate the results using the training dataset and another unseen dataset (Testing dataset). Table 7 lists these final determined results. As shown in this Table, this built ANN was able to generalize the forecasting and attain reliable forecasting results when tested on the new dataset where the error criteria for the testing set were moderately as good as the training ones. Figure 22 compares the actual to the predicted hourly power data for 15 days selected from the testing dataset. As seen, the model captured the overall trend; slight overestimations can be seen at peaks. However, the predictions remain reliable.

Table 7: Training and testing error criteria of ANN forecasting model

Error Criteria	Training	Testing
R^2	0.940	0.945
MSE [MW^2]	4.056 E+04	4.47E+04
RMSE [MW]	201.45	211.35
MAPE[%]	13.4%	13.5%
MAE [MW]	143.68	150.33

``

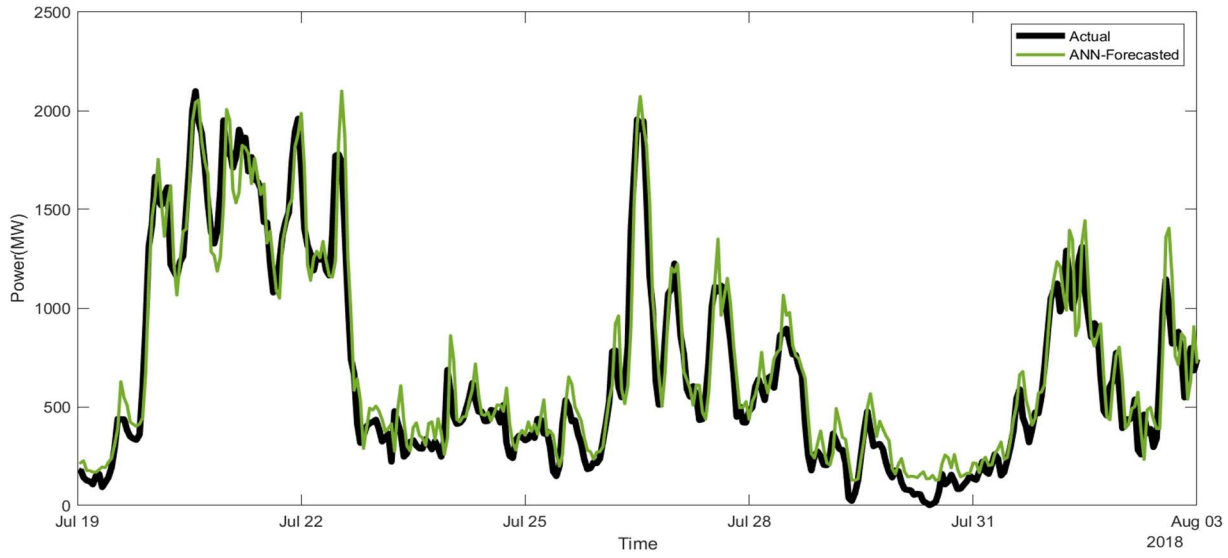

Figure 22: ANN wind power prediction results between 19/Jul-03/Aug

4.5.2 DNN Forecasting Model

Training a DNN with more than one hidden layer increases the number of hyperparameters the require tuning, including the number of hidden layers, the number of nodes in these layers, the activation functions of these layers, in addition to the hyperparameters that are already needed for tuning a one-layer neural network and related to the training algorithms such as the learning rate. Building a DNN considering all of these issues and hyperparameters would be very computationally expensive and require multiple optimization algorithms with different settings. Moreover, in too many problems sticking to a one-layer neural network, in fact, performed robustly and adequately. Therefore, in this case study, a DNN with only two layers was constructed; the number of hidden units in the first layer was selected to be 10, and the number of hidden units in the second layer was determined through grid search. As illustrated in Figure 23, the MAPE reduced almost more than 1 % when increasing the number of nodes in the second layer from 15 to 30, while the reduction in the MAPE and the other metrics was almost negligible for the set of nodes ranging from 30 to 60. Therefore, the optimal number of nodes in the second layer was selected to be 30. This selection was not only made by considering the validation error or the accuracy of the model only; it was made to ease the training and simulation of the model, where too many nodes and layers require high computational machines and could be time-consuming

``

.

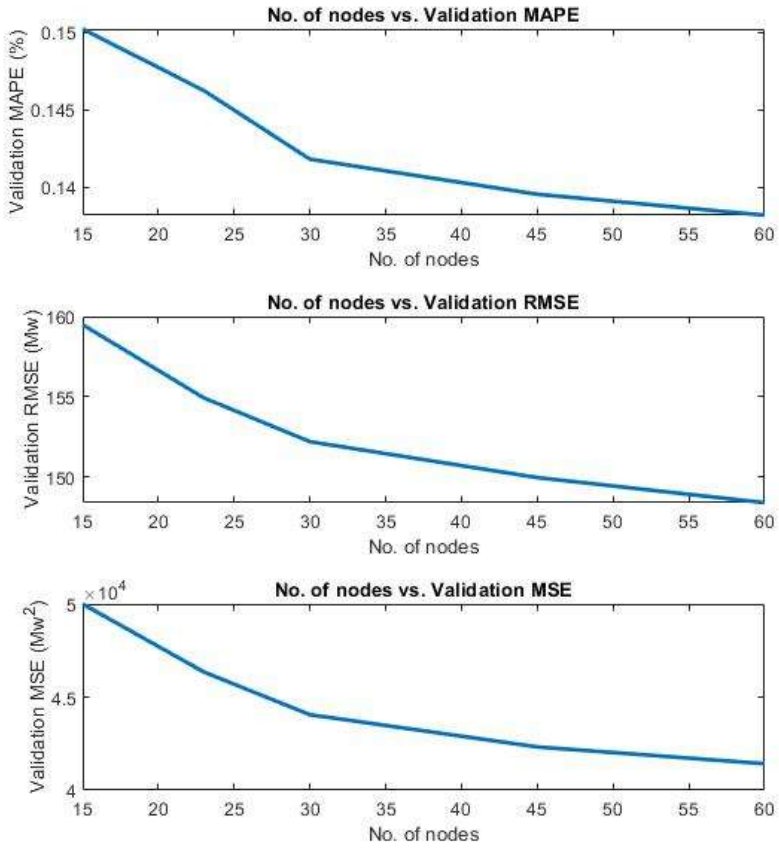

Figure 23: DNN different error criteria variation for various number of nodes in layer 2

When testing different combinations of activation functions in the two layers, it was investigated that using a sigmoid function and then a tanh function improved the forecasting slightly. It is essential to mention that training this network mini-batch gradient descent with a momentum algorithm was used. The mini-batch concept was employed because data are processed inside this network with different activation functions and layers; the mini-batch concept will accelerate the training process and reduce the training time. The mini-batch size was selected to be 10; this selection is made upon related applications in the literature [112]. As seen in Figure 24, the mini-batch technique results in a noisy cost function in contrast to the batch gradient descent, where the cost function decays smoothly.

Nevertheless, since our code is built to store the best parameters that determined the best validation error metrics and truncate the training process when overfitting starts the gradient descent with momentum performed well, and the final evaluation metrics of the build model are presented in Table

``

8. Overall, although the model performed well, compared to those error metrics obtained from the ANN, ANN is considered better. This indicates that the employment of the DNN in our case is not superior, mainly because it is time-consuming compared to the ANN. However, it should be mentioned that the DNN's hyperparameters are much more; this could also be a direct result of the superiority of the ANN in this case.

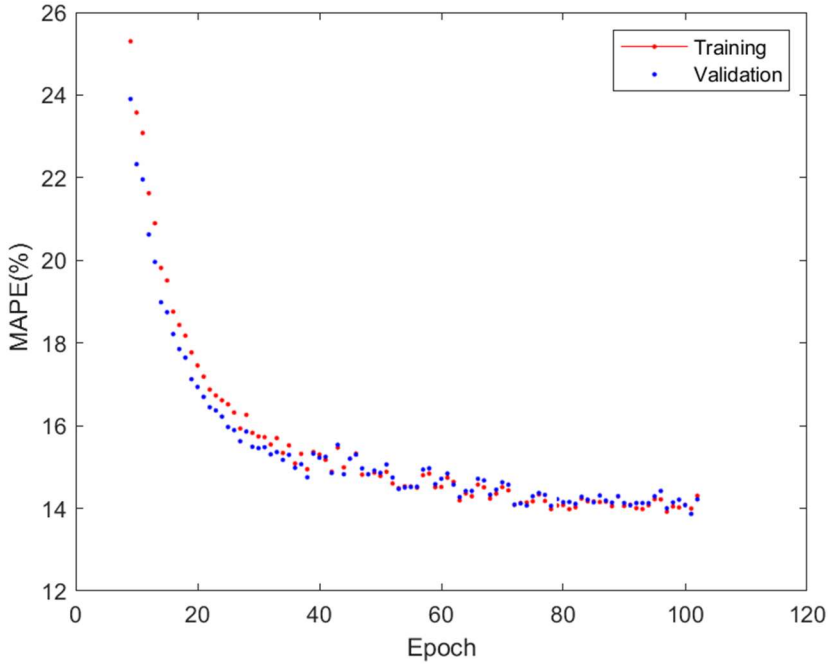

Figure 24: Learning curve of DNN (1-h ahead)

Table 8: Training and testing error criteria of DNN forecasting model

Error Criteria	Training	Testing
R^2	0.93	0.94
MSE [MW^2]	4.47 E+04	5.19+04
RMSE [MW]	211.41	227.71
MAPE[%]	14.0%	14.6%
MAE [MW]	152.38	164.69

The predictions of the same 15 days from the testing set are plotted in Figure 25. As illustrated, the predictions are noisy in some spots, and overestimations and underestimations at the peaks can be clearly observed; however, the general trend is mapped.

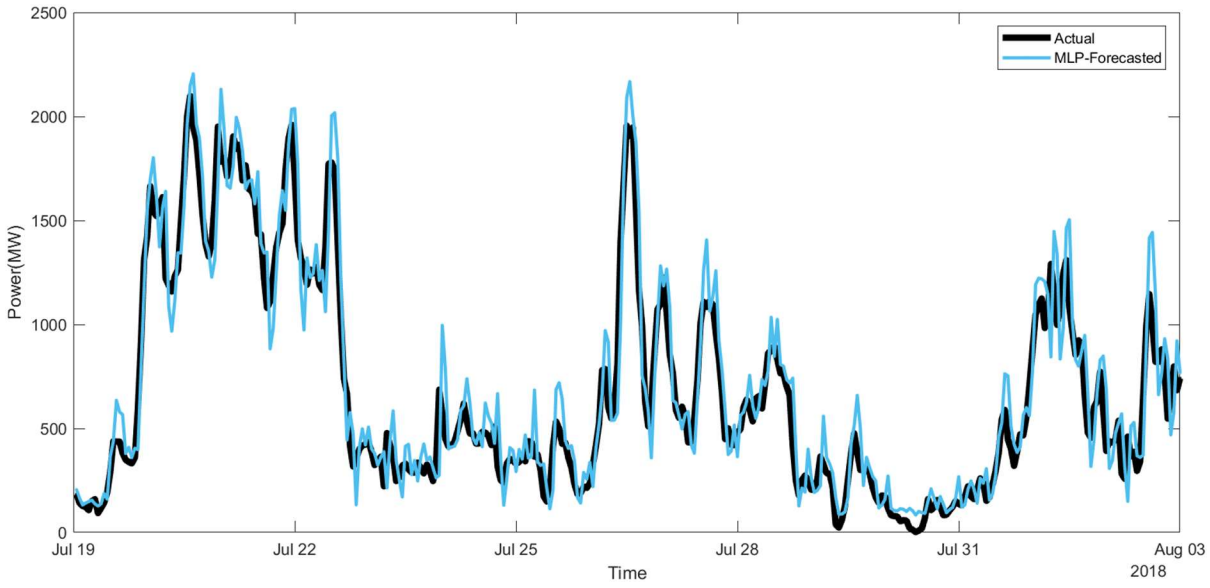

Figure 25: DNN wind power prediction results between 19/Jul-03/Aug

4.5.3 LSTM Forecasting Model

Being a type of RNN, LSTM proved its superiority to map time series data due to the recursive structure and the gates that preserve long previous sequences. The same 15 input features developed and used in previous sections are used now to train LSTM for wind power forecasting.

As previously shown in section 3.3, the units of LSTM consist of multiple gates where different mathematical calculations and operations happen; this series of operations slow down the training process. Therefore, similar to DNN in the previous section, the LSTM model was trained using mini-batch gradient descent with a momentum algorithm to help in accelerating the training process.

Similar to previous models, multiple experiments with a different number of nodes were conducted to determine the optimal number of nodes. As presented in Figure 26, the error metrics start to increase for nodes higher than 30 but then decrease for a higher number of nodes. However, training and simulating the network with more nodes will become much more challenging and time-consuming. Hence, by a trade-off decision between accuracy and ease of training and simulation, 30 LSTM nodes were selected to train the final LSTM model. Figure 27 represents the training curve of this LSTM network.

``

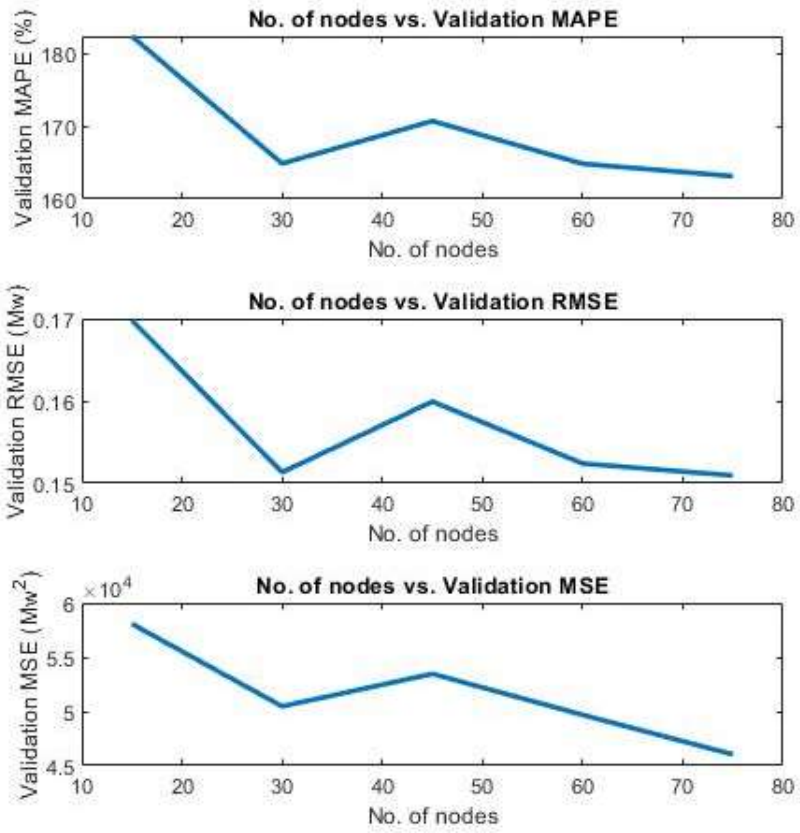

Figure 26: LSTM different error criteria variation for various number of nodes

According to the calculated error metrics (Table 9)from simulating the final trained model using the training and testing datasets, it can be seen that the LSTM converged at values of MSE higher than those reached by the DNN and ANN. This earlier convergence results in less accurate predictions and higher MAE. Nevertheless, the testing metrics results are highly close to the training ones, indicating a suitable model generalization. When observing the plotted prediction in Figure 28, it can be seen that this forecasting model is susceptible to variations and resulted in a noisy prediction that oscillate a lot and overestimate minimal variations across hours of predictions.

Table 9: Training and testing error criteria of LSTM forecasting model

Error Criteria	Training	Testing
R^2	0.93	0.93
MSE [MW^2]	5.09E+04	5.36 E+04
RMSE [MW]	225.69	231.57
MAPE[%]	15.0%	14.7%
MAE [MW]	161.60	162.84

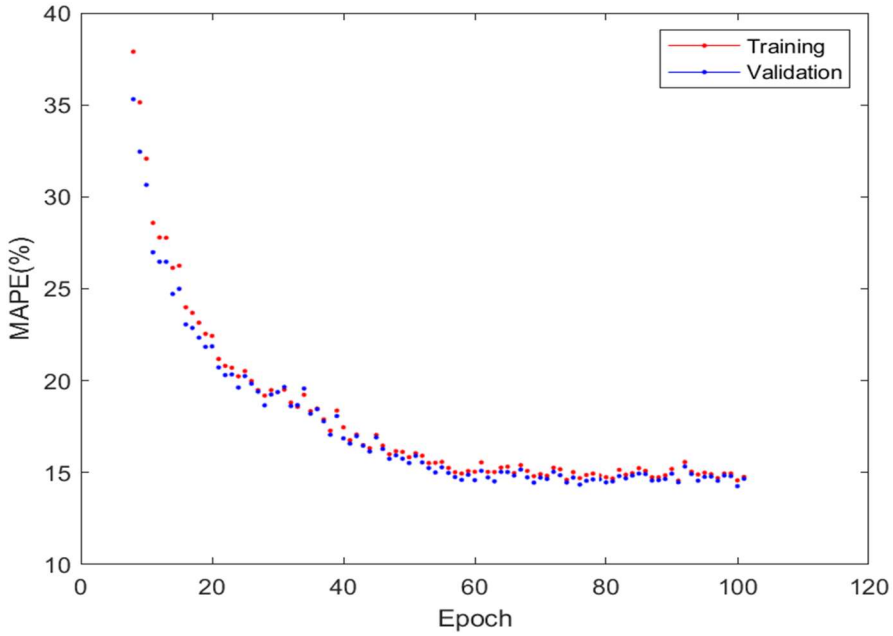

Figure 27: Learning curve of LSTM (1-h ahead)

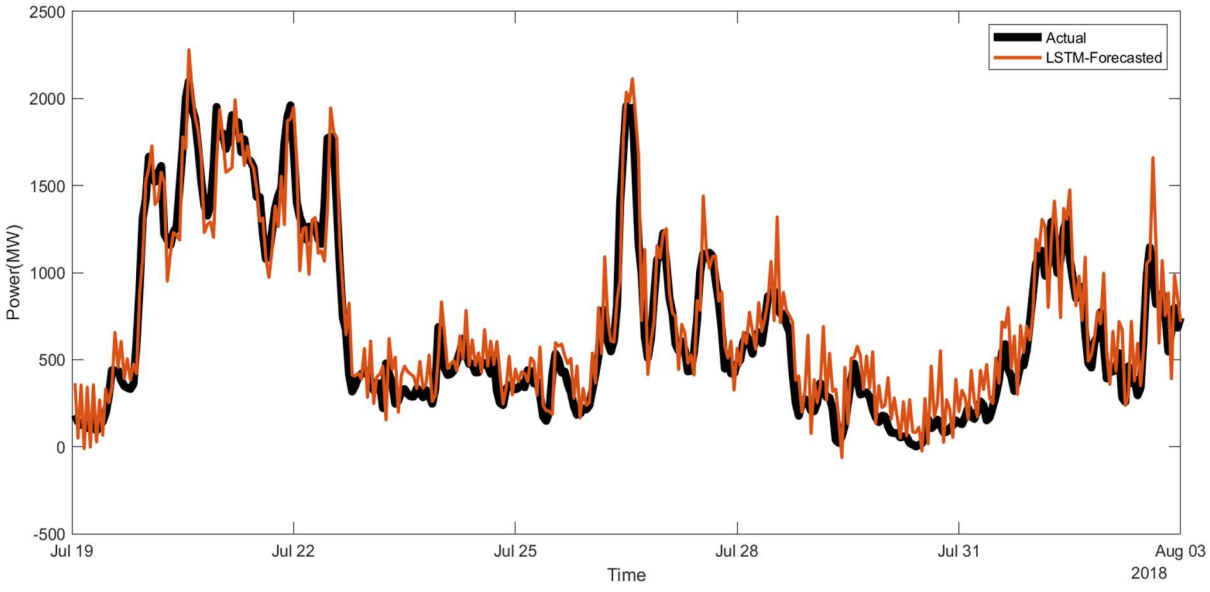

Figure 28: LSTM wind power prediction results between 19/Jul-03/Aug

``

4.5.4 BT Forecasting Model

As previously discussed, ensembling tree-based models together can reduce the variance of the tree models. In this section, the bagging tree method is trained to forecast the hourly wind power targets.

The most crucial hyperparameter for the BT model is the number of weak individual trees we aim to combine to create the final ensemble model. In this case study with trial and error, 30 individual trees were trained. Each tree is trained using a subset of training data (65 % of the training data) randomly drawn with replacement from the shuffled training dataset. The training process of each tree is truncated based on the validation error. It is essential to mention that all the 15 features in the training subsets were used for building the individual trees (Bagging approach is used, this approach is explained previously in section 3.4)

Table 10 lists the error metrics determined after simulating the final ensemble tree trained model using the complete training dataset and the unseen testing dataset. As expected for the training dataset, combined tress performed adequately and, in fact, surpassed all the previous tested models; However, this forecasting accuracy considerably dropped when the model was tested using the unseen data. For instance, MAPE increased from 11.1 % for the training set to 22.8 % for the testing set. This gap between the metrics for the two sets proves the high variance of the tree models and their high dependence on the training dataset. Even though for building the trees, the training sets were constantly altered and differed from tree to tree.

Figure 29 compares the actual and the predicted wind power values during the 15 days selected period. As illustrated, the high MAE is quite evident in that plot where gaps between the forecasted power and the actual one are considerable. Nevertheless, it is vital to mention that the tree models are easy to train and simulate; therefore, the selection of this model remains a trade-off decision based on the importance of the level of accuracy required for the different applications and purposes.

Table 10: Training and testing error criteria of BT forecasting model

Error Criteria	Training	Testing
R^2	0.96	0.86
MSE [MW^2]	2.69E+04	11.75 E+04
RMSE [MW]	163.94	342.74
MAPE[%]	11.1%	22.8%
MAE [MW]	118.28	249.17

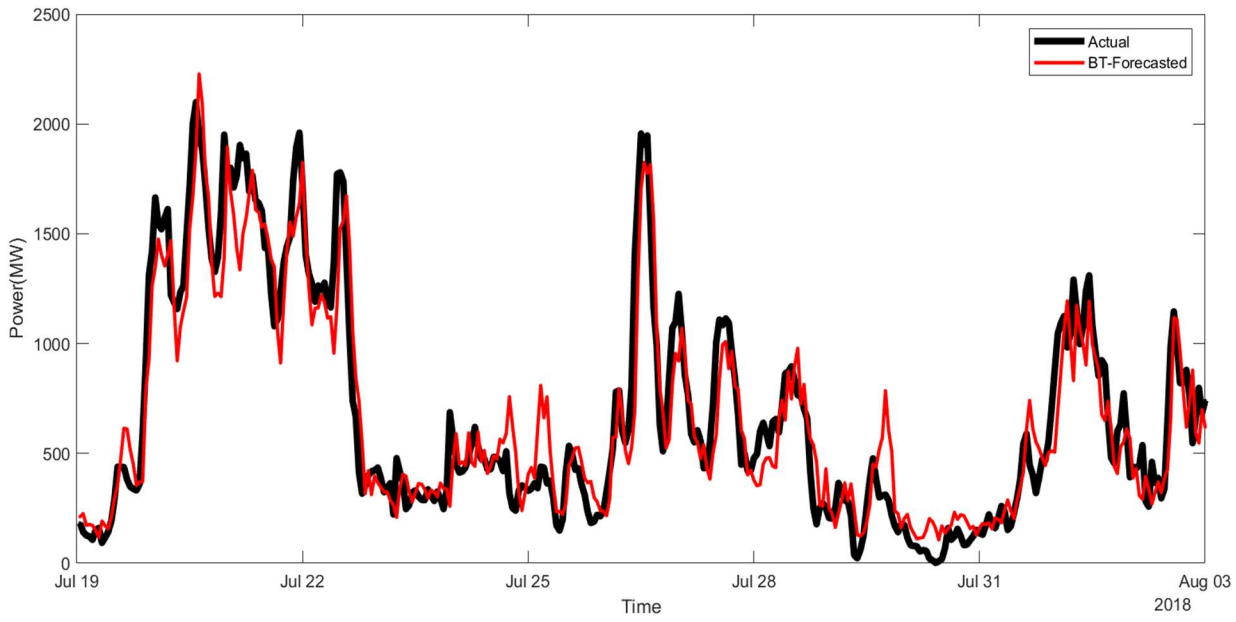

Figure 29: BT wind power prediction results between 19/Jul-03/Aug

4.5.5 SVM Forecasting Model

In this section, a SVR model is built to predict the wind power targets with the same 15 features used for training the other models.

Although SVR is a robust model that has been proven to be able to capture various nonlinear relationships for different applications, the training process of and determining its optimal parameters is a challenging process that requires advanced optimization packages. In this case study, a Bayesian optimization algorithm (See Appendix D for a conceptual explanation of Bayesian optimization) was employed to estimate the SVR model's optimal hyperparameters estimation.

The three hyperparameters that the SVR model is being trained for are:

1- The kernel functions

2- C (Regularization parameter)

3- ε (The intensive zone)

Table 11 shows the chosen search space for each hyperparameter and their obtained optimal values using Bayesian optimization.

``

Table 11: Determined Bayesian search optimal parameters

Hyperparameter	Search space	Optimal value
Kernel function	[Linear, quadratic, cubic]	Linear
C	[0.001,1000]	0.9709
ε	[0.854,8.54E+04]	121.6884

Finally, Table 12 represents the evaluation metrics determined using the optimized SVR model with the training and testing. The results show that the model is generic and very reliable. The MAPE and R^2 of the testing set are even slightly better than those of the training set. Moreover, the determined MAE is the lowest determined among the different tested methods in this study. The model's low variance results from the built-in regularization property in the SVR and SVM, which in most cases makes the SVR models superior compared to other models. The predictions of the same 15 days used in the previous section are shown in Figure 30; as seen, the SVR mode reliably mapped the relations and predicted the power with high accuracy.

Table 12: Training and testing error criteria of SVR forecasting model

Error Criteria	Training	Testing
R^2	0.94	0.95
MSE [MW^2]	38783.00	41425.00
RMSE [MW]	196.93	203.53
MAPE[%]	13.2%	13.0%
MAE [MW]	140.74	144.80

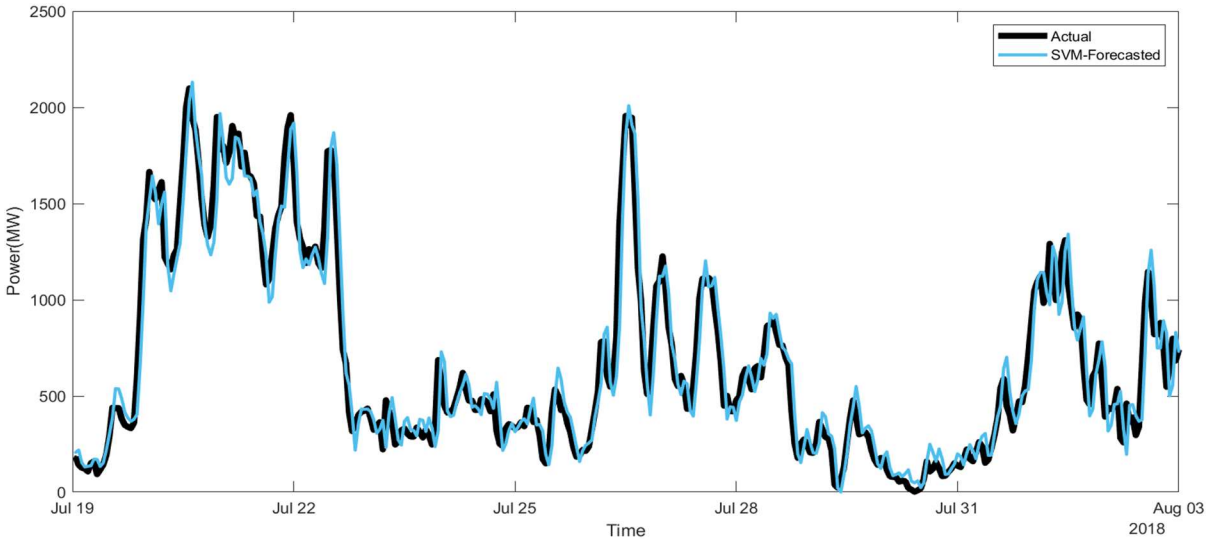

Figure 30: SVR wind power prediction results between 19/Jul-03/Aug

``

4.5.6 Ensemble Forecasting Model

This section tests ensemble learning by constructing an ensemble model combining the predictions from the models trained in the previous sections.

As illustrated in Figure 31, a rotational quadratic Gaussian regression model is fitted using the predictions from trained models (ANN, DNN, LSTM, and SVM) to improve the final predictions of the hourly wind power. Table 13 below lists the calculated error metrics of this final ensemble model.

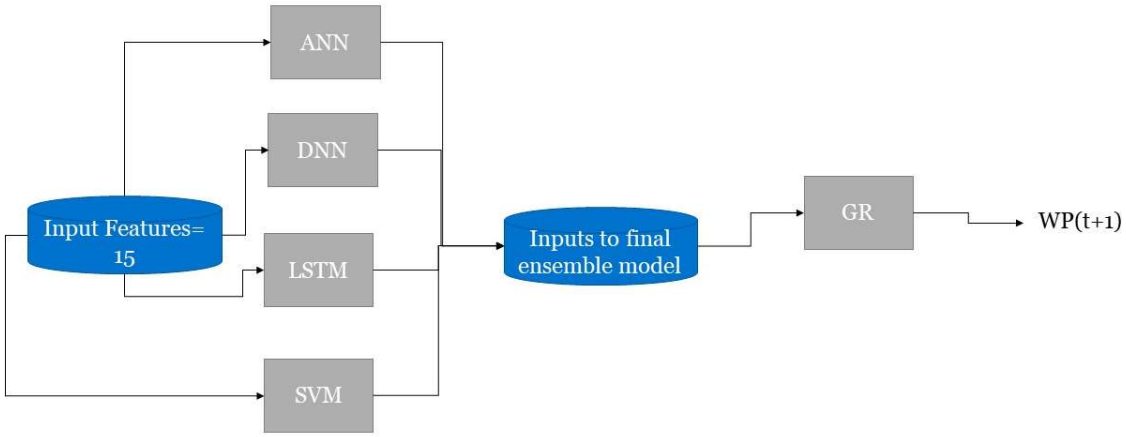

Figure 31: Proposed ensemble forecasting model

Table 13: Training and testing error criteria of an ensemble forecasting model.

Error Criteria	Training	Testing
R^2	0.94	0.95
MSE [MW^2]	3.76 E+04	4.11 E+04
RMSE [MW]	193.96	202.71
MAPE[%]	12.9%	12.8%
MAE [MW]	138.07	144.38

The results above in Table 13 show that the testing MSE (objective function) reached the lowest value for the ensemble model compared to the trained models in previous sections. Similarly, the MAPE % reached a value of 12.8% for the testing dataset, indicating the model's reliable generalization and accuracy. Nevertheless, when comparing it specifically to the SVM model, the obtained results are close to each other. This shows that the superiority of the ensemble model over the SVR could be

``

negligible and confirm the robustness of the individual SVR models and their ability to map and capture different relationships between data.

Finally, the forecasted wind power by the ensemble model during the same period studied in previous sections is plotted in Figure 32. As illustrated, the model predicted the power values smoothly with high accuracy at peaks and adapted the minor oscillations in values.

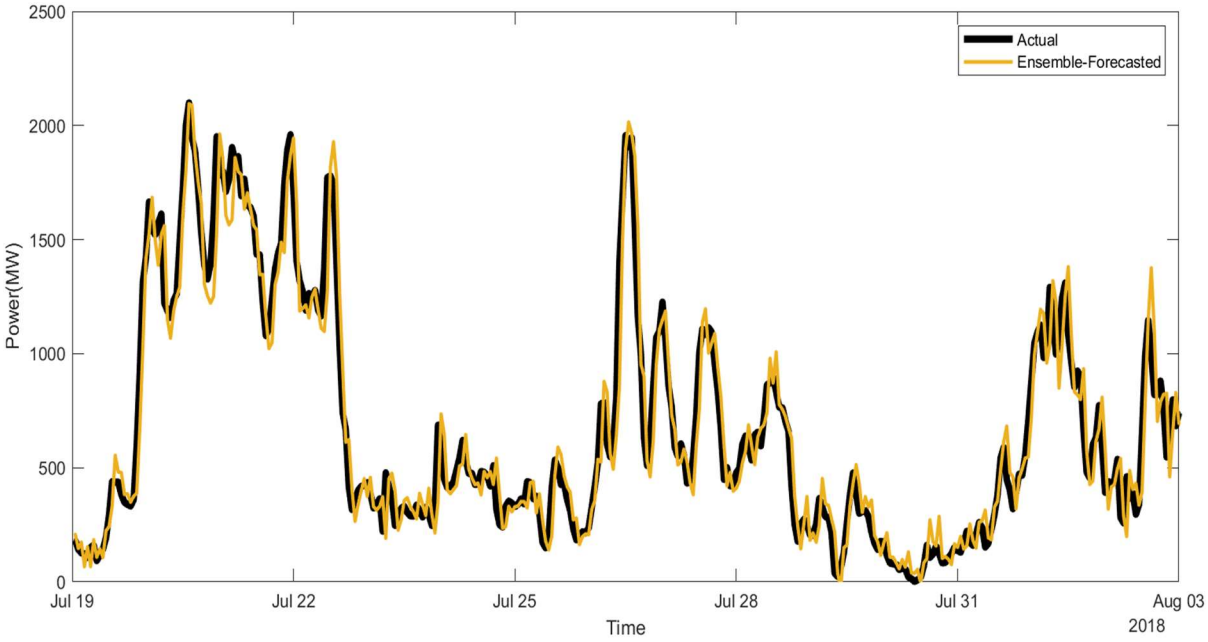

Figure 32: Ensemble model wind power prediction results between 19/Jul-03/Aug

4.5.7 Comparative Discussion

Table 14 summarizes all tested models' evaluation metrics to compare them and evaluate their performances and conduct the superiority of one model over the others.

By evaluating and analyzing the performance of the models when tested on unseen testing data, it can be concluded that SVR/SVM is one of the most promising robust ML-based forecasting models. This algorithm can build reliable generic models that can perform well with new data where the testing MAPE % reached a value of 13 % for the testing predictions. Although almost a similar MAPE was calculated from the ensemble model, the ensemble model results indicate that the utilized approach for combining the models was insufficient to improve the predictions. Hence, using a different ensembling approach such as boosting in future work could increase the accuracy of forecasting results.

``

When comparing the ANN and DNN, it can be seen that the additional layers did not improve the forecasting precision. Although a hyperparameter tuning was employed to tune the number of nodes in the second and first layers and for activation function selection, this tuning was not enough. The assumption of inadequate tuning of this network is based on the fact that the random search was employed and the tuning process itself needed a long computational time. Moreover, other learning-related hyperparameters such as batch size and learning rate were selected based on experience and other scholars' applications without turning them. Therefore, employing more powerful optimization algorithms is expected to improve the overall performance of a DNN. Eventually, a better performance by an ANN is considered favorable since training and simulating this network is easier and faster.

When comparing the DDN with the LSTM, the DNN reached a lower MSE; however, the LSTM's MAE is lower. Generally, the LSTM performance was expected to be better due to the recursive nature of the units in this network. Nonetheless, as illustrated and observed from the obtained results, no clear superiority of the LSTM model over the non-recursive network. Where SVR and ANN mapped and presented the data with overall higher accuracies.

Table 14: Training and testing error criteria of different forecasting models.

	Training					Testing				
Model	**R^2**	**MSE [MW]**	**RMSE [MW]**	**MAPE**	**MAE [MW]**	**R^2**	**MSE [MW]**	**RMSE [MW]**	**MAPE**	**MAE [MW]**
ANN	0.94	4.06E+04	201.45	13.4%	143.68	0.945	4.47E+04	211.35	13.5%	150.33
MLP	0.93	4.47E+04	211.41	14.0%	152.38	0.94	5.19E+04	227.71	14.6%	164.69
LSTM	0.93	5.09E+04	225.69	15.0%	161.60	0.93	5.36E+04	231.57	14.7%	162.84
SVM	0.94	3.88E+04	196.93	13.2%	140.74	0.95	4.14E+04	203.53	13.0%	144.80
BT	0.96	2.69E+04	163.94	11.1%	118.28	0.86	1.17E+05	342.74	22.8%	249.17
Ensemble	0.94	3.76E+04	193.96	12.9%	138.07	0.95	4.11E+04	202.71	12.8%	144.38

For bagging tree ensemble, although the training process of this model is straightforward and time-efficient, as the results confirmed and illustrated, this kind of algorithm builds data-dependent models. Similar to the ensemble model concluded idea, using different combining approaches for optimally

``

enesambling the trees could improve the overall generalization of the trees. The implication of RF is also expected to be practical and reliable. Further consideration of the abilities of the RF algorithm is conducted in case study 2 (chapter 5).

To sum up, although better precisions were reported in the literature for wind power forecasting, our obtained accuracies are considered adequate for regional forecasting. This notation is based on the fact that all the published and conducted studies focus on specific sites or even one wind turbine with a known hub height, physical characteristics, and efficiency coefficients. Furthermore, unlike the carefully measured weather parameters affecting a single turbine at specified heights, the meteorological data used in this study were from different locations across Ontario, measured with different apparatuses with different settings. Hence, these factors would definitely affect the overall forecasting results, even though the ML models will try to capture and adapt them and train the models for reliable predictions. Hence, with further spatial and weather data availability and more comprehensive optimization and tuning, a general regional model representing Ontario's wind power could be constructed to be used for other purposes such as electricity pre-scheduling to avoid surplus production of other resources.

4.6 Multi-Step Ahead Forecasting

As mentioned before, multi-step forecasting is not an easy task that requires further attention from scholars. This type of forecasting objective requires compelling models to handle error accumulation resulting from the forecasting targets' temporal dependencies between each other. Before discussing the approaches that will be considered in this section for multi-step wind power forecasting, it is essential to mention that multi-step forecasting can be achieved using different approaches, namely:

1- Direct multi-step forecasting: In this approach, N models are individually built for N forecasting steps without considering the dependencies of these steps on each other. This methodology requires multiple models training and tuning, which is eventually very expensive and time-consuming [113].

2- Recursive multi-step forecasting: In this method, a one-step model is trained for multiple steps forecasting using the predictions of previous steps to predict the following ones. Although this approach accounts for the dependencies of steps on each other, it can cause degradation of the training process, where error accumulates from step to step with the increasing length of the horizon (number of steps)[114].

``

3- Direct-Recursive multi-step forecasting: This methodology combines methods 1 and 2 together, where a model for each step forecasting is built, but the predictions from the model of the previous steps are used to build the model for the following step. In other words, prediction is performed sequentially to use the predictions from the previous model as inputs to the following model. Although hybrid strategy requires multiple model training, it is expected to overcome the drawbacks of the two individual models [115].

4- Multiple output multi-step forecasting: This methodology aims to build one model capable of forecasting the entire target sequence in one simulation step without the recursive processing of data. These multiple outputs models are complex since they focus not only on capturing and mapping the relationship between inputs and outputs but also on learning the relations between outputs to capture temporal dependencies between them. This complexity slows down the training process and requires more training data to avoid over or underfitting[116].

The direct multi-step forecasting and the (MIMO) methods are tested and compared in the following subsections of this section. Similar data and pre-processing approaches followed for one-step forecasting will be followed for multi-step forecasting. Some minor modifications of the selected lag features will be applied to handle the longer horizon of forecasting. These modifications would, of course, affect the dimensionality of data; therefore, dimensionality reduction of the new features will be considered to ease the training processes. The following section will discuss these modifications to features and dimensionality.

4.6.1 Feature Selection and Dimensionality Reduction

The same 21 features selected in section 4.4.1 are the first step for feature selection for multi-step forecasting: (Previous wind power +Wind speed ×5 stations + Wind direction ×5 stations + Temperature ×5 stations + Year + month + day + hour + seasonality+ No. of operating plants).

For the lag feature, the lag feature selected for one-step ahead was 3; the last 3 hours features were considered to forecast the following one. Now, For the case of multiple steps, with trial and error, a lag of 6 was chosen to be enough for this case study to forecast wind power three hours ahead. Therefore, the new input features become (6×22=132 features). The dimensionality of these features is considerably high; therefore, using the PCA method for dimensionality reduction would ease and accelerate the training process. As seen in Table 15, 93 % of the original information is cumulatively

``

stored in the first 29 principal components. Hence, the final input vector to the MIMO forecasting model was chosen to be these 29 components.

Table 15: Principal component analysis results of the first 29 components

Principal component	Eigenvalue	Variation %	Cumulative variation %
1	7.014	24.317	24.317
2	4.412	15.295	39.612
3	2.983	10.342	49.954
4	2.009	6.964	56.918
5	1.854	6.427	63.345
6	1.536	5.326	68.671
7	1.313	4.552	73.223
8	0.893	3.095	76.318
9	0.659	2.285	78.603
10	0.523	1.813	80.416
11	0.476	1.651	82.067
12	0.392	1.358	83.425
13	0.336	1.163	84.588
14	0.250	0.867	85.455
15	0.224	0.778	86.233
16	0.218	0.756	86.989
17	0.193	0.670	87.659
18	0.185	0.641	88.300
19	0.174	0.603	88.904
20	0.166	0.576	89.480
21	0.154	0.533	90.012
22	0.152	0.528	90.540
23	0.123	0.428	90.968
24	0.115	0.400	91.368
25	0.112	0.387	91.755
26	0.109	0.378	92.133
27	0.109	0.376	92.509
28	0.107	0.372	92.881
29	0.098	0.341	93.222

``

4.6.2 Simulation Results 2 (3-steps ahead Forecasting –Direct Strategy)

In this section, three independent models using the same 29 input features are trained to predict one-step wind power. The models are built parallelly using the same features; the first model's target is forecasting the wind power 1-h ahead, while the second is trained to forecast 2-h ahead, and finally, the third is constructed for 3-h ahead forecasting. Figure 33 below illustrates this proposed forecasting process. As shown, the three models were selected to be SVR; this selection was made based on the obtained results in the 1-step ahead forecasting, where SVR showed a robust performance and generalization when compared to other models. The hyperparameters of these SVR models are optimally obtained using random search. Table 16 lists the obtained parameters for the three models.

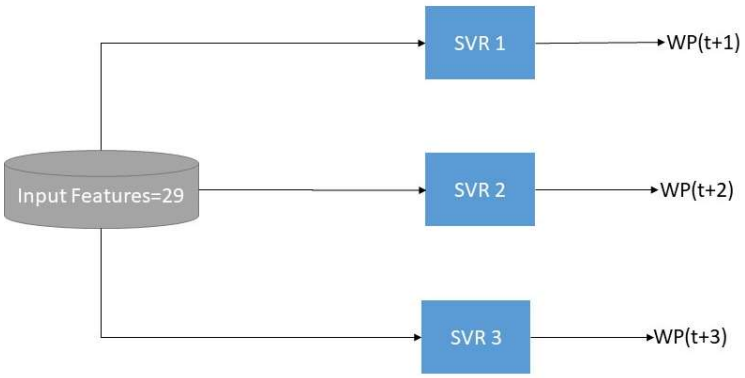

Figure 33: Proposed direct multi-step forecasting model.

Table 16: Random search optimal parameter results

Hyperparameter	Search space	Optimal value
Model 1		
Kernel function	[Linea,quadratic, qubic]	Cubic
C	[0.001,1000]	1.72
ε	[0.845,8.54E+04]	26.21
Model 2		
Kernel function	[Linea,quadratic, qubic]	Linear
C	[0.001,1000]	9.9308
ε	[0.845,8.54E+04]	2.7887
Model 3		
Kernel function	[Linea,quadratic, qubic]	Linear
C	[0.001,1000]	56.89
ε	[0.845,8.54E+04]	1.1665

``

According to the error criteria of each model in Table 17, the accuracy of the predictions reduces with longer horizons. For instance, the MAPE of the third step model for testing the data set is considered high, and according to [117], A MAPE > 30 % is considered reasonable forecasting. Overall, this decline of accuracy is, in fact, expected because the temporal relation between this third step and the previous 2 steps was not considered; this is one of the drawbacks of the direct multi-step forecasting that was previously mentioned. Moreover, as seen, the gap between the training and testing sets metrics is considerable, although the SVR models are known for their generalization ability.

Table 17: Training and testing error criteria of direct multi-step forecasting models.

Error Criteria	Training			Testing		
	Step 1	Step 2	Step 3	Step 1	Step 2	Step 3
R^2	0.91	0.85	0.79	0.88	0.76	0.67
MSE [MW^2]	5.84E+04	1.05E+05	1.46E+05	1.01E+5	1.91E+05	2.67E+05
RMSE [MW]	241.66	324.04	382.1	317.80	437.03	516.72
MAPE[%]	16.2%	22.0%	26.2%	20.0%	28.11%	33.67%
MAE [MW]	171.82	231.33	274.0	219.90	304.37	362.17

4.6.3 Simulating Results 3 (3-steps ahead Forecasting – MIMO Strategy)

This section evaluates the MIMO models for 3-steps ahead forecasting based on different error metrics by simulating them using the training and testing datasets.

Training a network for multiple outputs requires some modifications to the cost function. For this forecasting case study, for the sake of simplifying the application, the average of MSE for all the horizons is selected to be the cost function, that we aim to find the optimal parameters that minimize it. Similar to the previous training processes, the cross-validation concept is employed to avoid overfitting and truncate the training process.

4.6.3.1 MIMO ANN Forecasting Model

MIMO ANN with 60 nodes and sigmoid hidden activation function in hidden nodes was trained to forecast the 3-h ahead wind power. Table 18 lists the error metrics of each forecasting step. According to these results, the ANN model has not been robust enough for this multioutput problem, where the accuracy for the second and third steps is not as high as the one for the first step. Nevertheless, comparing these results to those obtained from the direct approach, it can be seen that, unlike the direct approach where the accuracy of the first step is considerably higher than the one for the second and

``

third step, the MIMO approach forecasted the three steps with a balanced accuracy, where on the expense of the first step precision the accuracies of the second and third steps increased. Furthermore, better reliable performance can be observed when the model is tested on the unseen data.

Table 18: Training and testing error criteria of MIMO ANN forecasting model

Error Criteria	Training			Testing		
	Step 1	Step 2	Step 3	Step 1	Step 2	Step 3
R^2	0.89	0.83	0.76	0.88	0.84	0.77
MSE [MW^2]	7.46E+04	1.15E+05	1.61E+05	9.32E+4	1.33E+05	1.83E+05
RMSE [MW]	273.13	339.12	401.25	305.29	364.69	427.78
MAPE[%]	18.6%	22.8%	26.7%	20.1%	23.7%	27.5%
MAE [MW]	211.28	259.45	304.71	234.90	277.84	321.74

4.6.3.2 MIMO LSTM Forecasting Model

As seen in Table 19, MIMO LSTM performed better than MIMO ANN; error metrics such as MSE, MAE, and MAPE for all the steps are considerably better than ANN. This superior performance could be explained by the recursive structure of the LSTM, where the information from previous steps is used in the following steps. The MIMO LSTM was trained with 45 hidden units, this size of these nodes was determined with trial and error.

Table 19: Training and testing error criteria of MIMO LSTM forecasting model.

Error Criteria	Training			Testing		
	Step 1	Step 2	Step 3	Step 1	Step 2	Step 3
R^2	0.93	0.87	0.80	0.88	0.84	0.77
MSE [MW^2]	4.94E+04	9.09E+04	1.36E+05	6.58E+4	1.15E+05	1.68E+05
RMSE [MW]	222.26	301.50	368.78	256.52	339.12	409.88
MAPE[%]	15.2%	20.5%	25.0%	16.9%	22.0%	26.6%
MAE [MW]	163.3	220.65	269.87	190.02	284.06	299.63

4.6.4 Comparative Discussion

Multi-step step ahead forecasting is essential for some scheduling and managing objectives; however, this task is a complex task that requires special considerations of the dependence of the targets on each other. When comparing the direct and MIMO multi-step forecasting methodologies, it can be seen that although the MIMO approach is harder to train and require advanced machines, it can perform better, especially if the trained model has the built-in recursive property. Thus, the MIMO method with a type of recursive properties can capture the dependencies between steps. Moreover, it avoids the time-

``

consuming multi-model training for each step. From that perspective, in future studies, an attempt to apply different ML for MIMO forecasting could provide better insight into the ability of this approach.

4.7 Conclusion (Case Study 1)

This case study addressed Ontario's wind power forecasting comprehensively from different perspectives. Besides the proposed deep feature selection approach, a comparative analysis was conducted in this case study to compare the performances for different ML algorithms for one-step and multi-step ahead forecasting.

- For one-step ahead forecasting, by evaluating and analyzing the performance of the models when tested on unseen testing data, it can be concluded that SVR/SVM is one of the most promising robust ML-based forecasting models. This algorithm can build reliable generalized models that can perform well with new data where the testing MAPE % reached a value of 13 % for the testing predictions. Although almost a similar MAPE was calculated from the ensemble model, the ensemble model results indicate that the utilized approach for combining the models was insufficient to improve the predictions. Hence, using a different ensembling approach such as boosting in future work could increase the accuracy of forecasting results.

- Multi-step step ahead forecasting is essential for some scheduling and managing objectives; however, this task is a complex task that requires special considerations of the dependence of the targets on each other. When comparing the direct and MIMO multi-step forecasting methodologies, it can be seen that although the MIMO approach is harder to train and require advanced machines, it can perform better, especially if the trained model has the built-in recursive property. Thus, the MIMO method with a type of recursive properties can capture the dependencies between steps. Moreover, it avoids the time-consuming multi-model training for each step. From that perspective, in future studies, an attempt to apply different ML for MIMO forecasting could provide better insight into the ability of this approach.

To sum up, although better precisions were reported in the literature for wind power forecasting, our obtained accuracies are considered adequate for regional forecasting. This notation is based on the fact that all the published and conducted studies focus on specific sites or even one wind turbine with a known hub height, physical characteristics, and loss coefficients. Furthermore, unlike the carefully measured weather parameters affecting a single turbine at specified heights, the meteorological data used in this study were from different locations across Ontario, measured with different apparatuses with different settings. Hence, these factors would definitely affect the overall forecasting results, even

``

though the ML models will try to capture and adapt them and train the models for reliable predictions. Hence, with further spatial and weather data availability and more compressive optimization and tuning, a general regional model representing Ontario's wind power could be constructed to be used for other purposes such as electricity pre-scheduling to avoid surplus production of other sources.

``

Chapter 5
Case study 2: Air Quality Index Forecasting

5.1 Motivation and Contribution

As discussed and shown in the brief background and literature review in section 2.2, the AQI accurate forecasting relies on the forecasted pollutants levels in the ambient air. One significant issue associated with predicting these levels is the missing air monitoring data and the missing metrological data. As a result of temporal dependencies between these data, discarding observation with missing variables for training or building prediction models is generally impractical and affects the model's ability to capture the time relations between data. Furthermore, imputing missing observations with mean or median values or any other single imputation approaches could fail to map extreme or abnormal behaviors in the data. Therefore, assigning values to these missing incidents with the consideration of other factors is essential, especially for the case of AQI predictions where the extreme and high values actually require attention and precautionary actions.

This chapter tackles the missing data problem using the miss-forest imputation technique, a multivariate random forest-based imputation technique to impute missing observations in meteorological and pollutant levels data. Afterward, the effectiveness of the employed multivariate imputation is examined by using the imputed data for training ANN models to forecast the criteria pollutants levels and AQI. Pre-processing of data and feature selection is comprehensively conducted before building models using different methodologies; as part of investigating features' importance, random forest modeling was also employed to detect feature's importance.

Then, further analysis is performed to compare models built using the proposed imputation technique with models trained using linear imputation. In order to conduct a fair comparison between the two imputation approaches and test their performance, the testing set of data (unseen by the fitted models) was selected to be complete with no missing data. This selection makes the actual data of the testing set similar for both built models. By this, the generalization of the constructed models by differently imputed datasets could be fairly compared and analyzed.

To sum up, the following points mainly motivated this case study:

1- Testing multivariate imputation technique for imputing missing environmental and meteorological parameters in datasets.

``

2- Forecasting the criteria for pollutants levels in the air to report the predicted AQI of the following hours. Accurate predictions would help the inhabitants prevent hazardous and low-quality air exposure. Moreover, it would help decision-makers plan future operating conditions and/or cut off certain polluting activities at predicted peak pollution hours.

3- Evaluating the robust performance of random forest models, not only as a forecasting model but also for missing data imputation and features selection.

5.2 Data Description and Feature Engineering

5.2.1 Raw Data Sources and Pre-analysis

The air quality monitoring data for training and testing the proposed model was collected by Kuwait Environmental Public Authority (KEPA) from a station located in Al-Jahra, a city in Kuwait. This data includes three types of hourly observations:

1- Concentrations of different gaseous and particulate pollutants.
2- Meteorological condition measurements such as ambient temperature, wind speed, wind direction, and pressure, etc.
3- Temporal data; the time of recorded observations, including the year, month, and day in addition to the hour of the day.

Three years of hourly data (24-2-2013- 23-2-2015) were gathered for this study.

Out of 38 different recorded parameters, the listed parameters in Table 20 are selected for this case study.

``

Table 20: Summary of selected parameters

Type	Variable	Measurement unit
Meteorological	Temperature	Celsius
	Wind speed	m/s
	Wind direction	deg
	Relative humidity	%
Criteria pollutants levels	CO	mg/m^3
	NO_2	$\mu g/m^3$
	O_3	$\mu g/m^3$
	PM10	$\mu g/m^3$
	PM2.5	$\mu g/m^3$
	SO_2	$\mu g/m^3$

5.2.2 Data Splitting

The overall dataset was divided into three sets: training, validation, and testing datasets with 80 %, 10 %, and 10 %, of the overall data size, respectively. The training set is used to train the forecasting model, the validation set is used to test the learned parameters at each iteration during the training process, and the testing set is used to test the fitted parameters after the learning stops. The model's training is truncated when the MSE of the validation set starts to increase even though the training MSE is decreasing. This data splitting step prevents overfitting of the model, and generalization is ensured[111].

5.2.3 Missing Data Imputation

5.2.3.1 Conceptual Explanation Missing Data Imputation Method by Random Forest

As mentioned before, missing recorded measurements is a pretty common issue when dealing with real-life data. In research, two schemes of imputing these missing data are regularly followed, namely 1- Single imputation and 2- Multiple imputations.

The single imputation approach is a faster approach. The missing entry for a specific variable is simply assigned to that variable's mean or median value without considering other variables or even other related non-missing observations of the same variable. Conversely, missing values in the multiple

``

imputation techniques are estimated with lower biases and uncertainties using data analysis and regression tools[118]. In these approaches, models are built on the non-missing data to estimate the missing ones. In our study, the miss-forest imputation technique is employed and examined.

Miss-forest imputation methodology employs the RF algorithm for estimating missing data. This method can be summarized into four steps as follows:

1- **Initialization:** In this step, all missing observations of a specific variable are substituted by the mean value of this variable; a mean single imputation is performed as an initial step.
2- **Imputation**: The imputation of missing data is performed in sequential order of missing entries for each variable. The variable with the missing entries being imputed is treated as a target variable (dependent variable) for training the RF model [119]. Other variables are used as predictors for this target variable. The complete non-missing entries of the target variable are used for training the RF model, whereas the missing ones are replaced by the estimated values using the trained model[118].
3- **Repetition**: Step 2 is repeated for all variables with missing entries by assigning other variables to be the predictors to build the RF model.
4- **End**: Once RF models for all the variables with missing entries are trained, the first imputation iteration is achieved. Then, steps 2 and 3 will be repeated until the squared difference between the new and the previous imputation results increases. When detecting this increment, the imputation process will stop, and the final results will be selected to be the results determined from the previous iteration [118],[120].

5.2.3.2 Application of Proposed Imputation Technique

The missing observations in our dataset are scattered randomly in the training and validation sets. From the analysis performed to conduct the percentage of missing entries in each variable, high missingness rates (>6 %) are observed in pollutants concentrations. On the other hand, moderately lower missingness percentages are for the meteorological data. The missingness percentages in the data are summarized in Table 21.

As mentioned before, the missing entries are imputed using two different imputation approaches, namely, miss-forest imputation and linear interpolation; these two approaches imputed the missing data in the training and validation sets. Using different imputation techniques assign missing data differently; both data sets are then used to train ANN models to forecast the pollutants level and the AQI.

``

Nevertheless, because different methods imputed the missing entries, the imputed forecasting targets for the two datasets are not similar

Therefore, comparing the obtained results to conduct the superiority of one approach over the other will not be fair in this case. Thus, to achieve a fair comparison between the models, the testing set was selected to be complete with no missing data. In this case, the targets of both models are the same; therefore, the model that results in better testing predictions can be considered superior with a better generalization when tested on the unseen data, which reflects an overall better estimation of missing data.

Table 21: Percentages of missing observations

Variable	Missingness %
NO_2 Conc.	10.96 %
PM2.5 Conc.	10.36%
O_3 Conc.	10.30%
SO_2 Conc.	8.01%
PM10 Con.	7.89%
CO Conc.	6.70%
Temperature	4.59%
Relative humidity	1.89%
Wind speed	1.16%
Wind direction	1.16%
Time (year, month, day, hour)	0%

5.2.4 Feature Engineering (Extraction)

In this section, the time features, including month, day, and hour, are encoded to become cyclic using sin and cos functions. This encoding will improve forecasting models' ability to capture the cyclic temporal and seasonal relations between predictors and targets, ultimately increasing the model accuracy[121]. The following equations (20 and 21) were used to transform the temporal feature X into a cyclic feature.

Where $max.(X)$ = 12, 31, and 24 for the month, day, and hour feature, respectively.

``

$$X_{sin} = \text{Sin}\left(\frac{2\pi X}{max.(X)}\right) \quad (20)$$

$$X_{cos} = Cos\left(\frac{2\pi X}{max.(X)}\right) \quad (21)$$

After this feature engineering step, a total number of 17 features (Concentrations of 6 pollutants + Temperature + Wind speed +Wind direction + Relative humidity + Sin (month, day, hour) + Cos (month, day, hour) +Year) are selected to be the inputs of the ANN.

In section 5.3, unnecessary features, if found, will be filtered, and the appropriate lag features of the target pollutants levels are selected.

5.2.5 Data Scaling

Scaling data to become within [0,1] is vital since predictors differ in ranges. High variations between data could slow the training process of the ML engines and cause the issue of falling into the minimal local values, which results in unreliable, poor forecasting models [122].

In this case study, the predictors were scaled to become between 0 and 1 as follows:

$$P_i^* = \frac{P_i(t) - \min(P_i)}{\max(P_i) - \min(P_i)} \quad (22)$$

P is a vector of predictors at time = t, i is the index of the predictor, and min is the minimum value of predictor i, and max is the maximum value of predictor i, and P* is the scaled value of predictor i.

5.3 Feature Selection

5.3.1 Feature Filtering and Selection

Proper selection of predictors is essential because high dimensions of irrelevant features could delay the training process and necessitate the need for expensive, time-consuming computation machines[123]. Although previous research and experience were considered for carefully selecting attributes for forecasting the pollutants level, this step before predicting is vital. It could highly affect the forecasting results and give an insight for feature selection for related applications and studies. In our study, the famous Boruta algorithm analyzes the importance of the selected predictors and filters the irrelevant ones if found.

``

Boruta's main objective is to conduct the significance of the feature by testing its effect on a random forest model through multiple iterations [124]. First, a copy of the predictors is created and randomly shuffled across the observations in each iteration to create shadow variables. The shadow variable will erase out the actual relation between the predictors and the target. Next, the RF model is trained using the doubled dataset (The real predictors + their shadows). After training, a statistical Z-test is conducted to disclose the significance of the predictor and compare it to the importance of the shadow variable. To declare the feature as an important feature, its relevance must be higher than the maximum significance of all shadow variables ($Z\text{-score}_{actual} > \max. Z\text{-score}_{shadow}$). After filtering out insignificant variables and their shadows, the previous steps are repeated until a filter/keep decision for all features is made[124]. R studio package for Boruta was employed in our study to determine the important features for forecasting each pollutant.

Table 22 lists the final selected predictors for each forecasting target. As illustrated, almost all the features nominated in the previous section were declared to be necessary. Although sin hour was

Table 22: Summary of selected features per target

Variable \ Target	O_3 Conc.	SO_2 Conc.	NO_2 Conc.	CO Conc.	PM10 Conc	PM2.5 Conc
Year	*	*	*	*	*	*
sine month	*	*	*	*	*	*
cosine month	*	*	*	*	*	*
sine day	*	*	*	*	*	*
cosine day	*	*	*	*	*	*
sine hour	*	*	*	*	Rejected	Rejected
cosine hour	*	*	*	*	*	*
O_3 Conc.		*	*	*	*	*
SO_2 Conc.	*		*	*	*	*
NO_2 Conc.	*	*		*	*	
CO Conc.	*	*	*		*	*
PM10 Conc	*	*	*	*		*
PM2.5 Conc	*	*	*	*	*	
Wind Speed	*	*	*	*	*	*
Wind direction	*	*	*	Tentative	*	*
Temperature	*	*	*	*	*	*
RelativeHumidity	*	*	*	*	*	*

``

rejected for PM10 and PM2.5, a decision to keep it was made since it was accepted for all the other targets. A similar conclusion was made for the wind direction for predicting the level of CO.

5.3.2 Lag Feature Selection

The final feature selection step is critical. This step is responsible for selecting the lag feature of the dependent variable, i.e., the pollutant levels. Like with any other time-series data, the pollutant levels in ambient air have significant dependencies on their observations at previous time steps. Therefore, ACF and PACF were used to select the appropriate lag features in our study. ACF and PACF are plotted in Figure 35-Figure 36 for (O_3, SO_2, NO_2, CO, and PM10, PM2.5) respectively. As one can notice from the ACF plots, level patterns of O_3, NO_2, and CO are repeated every 24 hours. This repeated pattern cannot be seen in the ACF plots of the other pollutants. Nevertheless, when removing internal relations between lags and plotting the PCAF, lags ranging between 1 and 4 are considered significant for the different pollutants. Although in the case of CO, only the zero-lag (the observation with itself) and 1 lag are considerable, in this study, the lag feature of all targets (pollutants levels) was selected to be 4 because the lags<4 observed in ACF plots were high and neglecting them will not be sensible. By this, the final size of the input feature becomes 20.

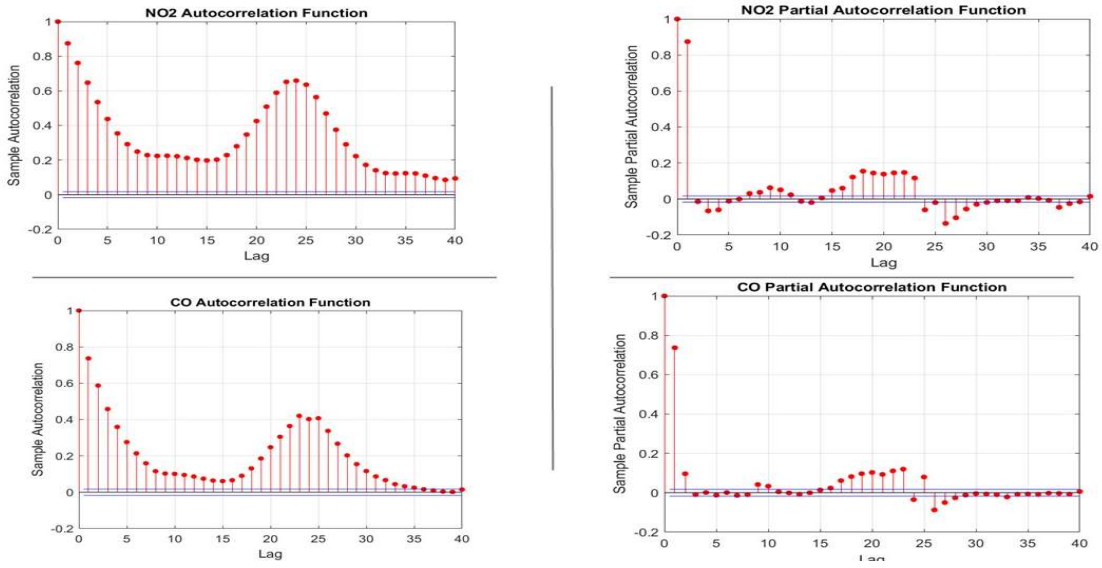

Figure 34: ACF and PACF plots for NO_2 and CO

``

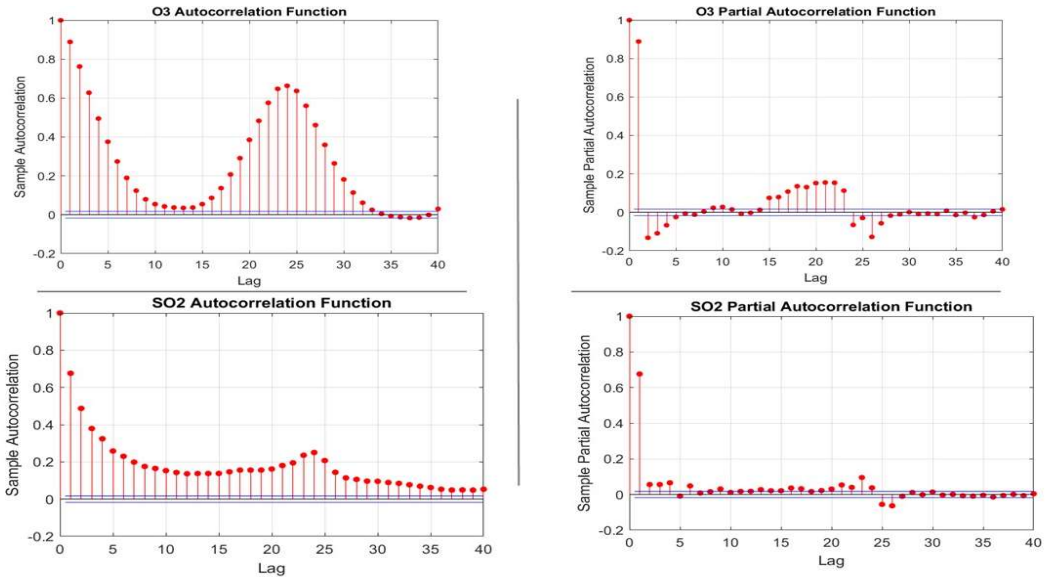

Figure 35: ACF and PACF plots for O_3 and SO_2

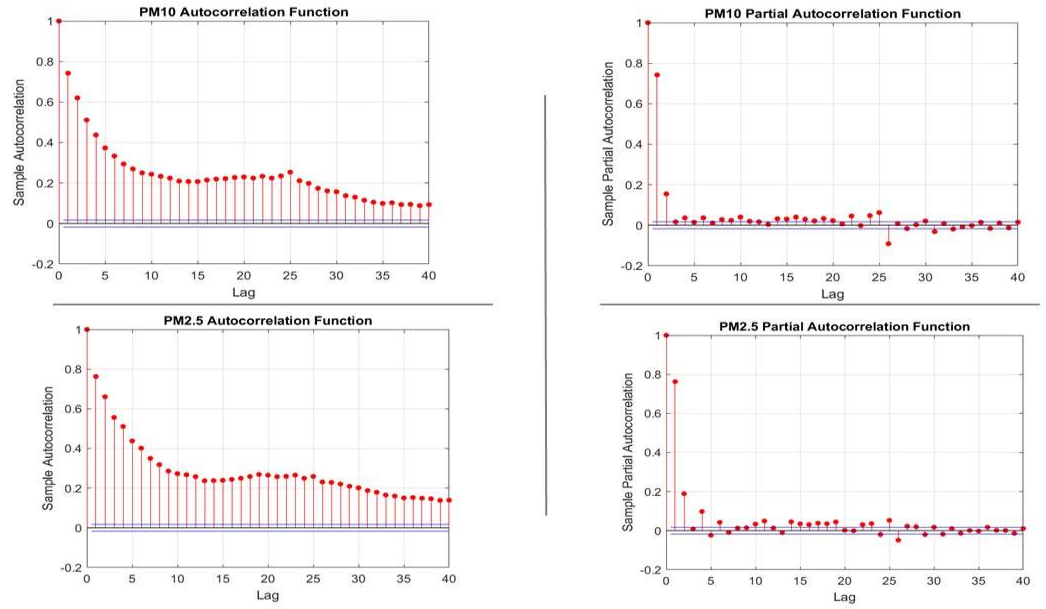

Figure 36: ACF and PACF plots for PM10 and PM2.5

``

5.4 Numerical Study

5.4.1 Forecasting Targets

Reporting the AQI of O_3 at a specific hour requires the midpoint 8-hour average ozone concentration[83]. Therefore, forecasting the AQI at time=t+1, concentrations at time = [t-3, t+4] are needed. Based on that, at time =t, our built forecasting model predicts the average O_3 concentrations of the next 4 hours [t+1, t+4]. Then the midpoint 8-h average concentration is calculated using the following equation:

$$\frac{4 \quad \text{h average of actual previous hours} + \text{forecasted } 4 \quad \text{h average of the following hours}}{2} \tag{22}$$

A similar approach is considered to forecast the 4-h average of CO concentration and calculate its midpoint 8-h average.

For PM10 and PM2.5, the 24-hour average concentration is required; thus, the prediction model is designed to predict the 12-h average concentration of the following hours. Similarly, then the midpoint 12-h concentration is calculated using the following equation:

$$\frac{12 \quad \text{h average of actual previous hours} + \text{forecasted } 12 \quad \text{h average of the following hours}}{2} \tag{23}$$

For NO_2 and SO_2, the 1-h concentrations of the next hour are forecasted.

5.4.2 Settings of the Pollutants Forecasting Models

In this study, ANN is applied to build the forecasting models of the six pollutants. As mentioned before, training an ANN with insufficient nodes generates unreliable under-fitted models. Whereas selecting a large number of hidden neurons when training an ANN provides an overfitted model lacking generalization performing poorly with another unseen dataset[125]. Therefore, finding the appropriate number of nodes when training an ANN is crucial and highly affects forecasting performance [126]. Besides that, excessive training by setting the iterations of training to a large number would result in a biased fit towards the training set and also cause the overfitting issue.

Therefore, the data division accompanied the grid search optimization to address the overfitting issue and find the optimal number of nodes. The network will be trained multiple times with different sizes

``

of neurons between 2 and the input layer nodes (20); the number of neurons resulting in the lowest validation error is selected to be the optimal size of hidden nodes.

The training procedure of the ANN is presented by pseudo-code in Table 23. The ANN for all the pollutants was trained using mini-batch gradient descent with a momentum algorithm with a batch size of 32, a learning rate=0.0001, and a momentum =0.9. The determined optimal number of nodes for the models of the six pollutants is presented below in Table 24.

Table 23: The training procedure of the ANN is presented by pseudo-code.

```
For neuros 1: neurons max do
    While epoch< epoch max do
        For each input of the training dataset Do
            Drop out nodes.
            Forward propagation.
            Evaluate loss function of each input.
            Back-propagate error.
            Update weights and biases of the hidden and output layer.
        End for
        For each input in the Validation data set do
            Calculate Validation cost function and error metrics.
        End for
        If Validation cost < best determined validation cost function (BestV)
            BestV=Validation cost
            epoch=epoch+1
        Else
            epoch=epoch max
        End if
    End while
End for
```

Table 24: Optimal number of nodes

Pollutant	Optimal no. of nodes (Miss-forest imputed dataset)	Optimal no. of nodes (linear imputed dataset)
O_3	12	12
NO_2	10	12
SO_2	16	18
CO	14	16
PM10	20	12
PM2.5	18	16

``

5.5 Criteria Pollutants Forecasting Results

In this section, the determined and selected features from the previous section will be utilized to build forecasting models of criteria pollutants to predict the AQI and its corresponding criteria pollutant. As previously stated, the prediction models are built using two different datasets. In this section, the forecasting results obtained from the two imputed datasets are reported and compared for the six pollutants (O_3, NO_2, SO_2, CO, PM10, and PM2.5, respectively).

5.5.1 Ozone (O_3)

Error criteria for the two forecasting models of ozone are shown in Table 25. As shown, the Miss - forest trained imputed dataset generated a model with lower error metrics for both the training and testing datasets. Since the actual testing data set is complete with no missing data, it can be seen that MSE, RMSE, and MAE are lower for the model of the miss-forest imputed data, indicating a better generalization of the built model when tested on the new unseen data. Figure 38 and Figure 39 show the testing prediction results of the O_3 level for both models. Figure 37 represents more detailed forecasted levels in fifteen days selected from the testing dataset.

Table 25: Forecasting models' error metrics for O_3 measurement in the training and testing sets.

	Miss-forest Imputed dataset		**Linear Imputed dataset**	
	Training	**Testing**	**Training**	**Testing**
R^2	0.97	0.93	0.97	0.87
MSE [µg/m³]²	26.80	31.11	28.45	55.23
RMSE [µg/m³]	5.18	5.58	5.33	7.43
MAE [µg/m³]	4.03	4.55	4.22	6.56

``

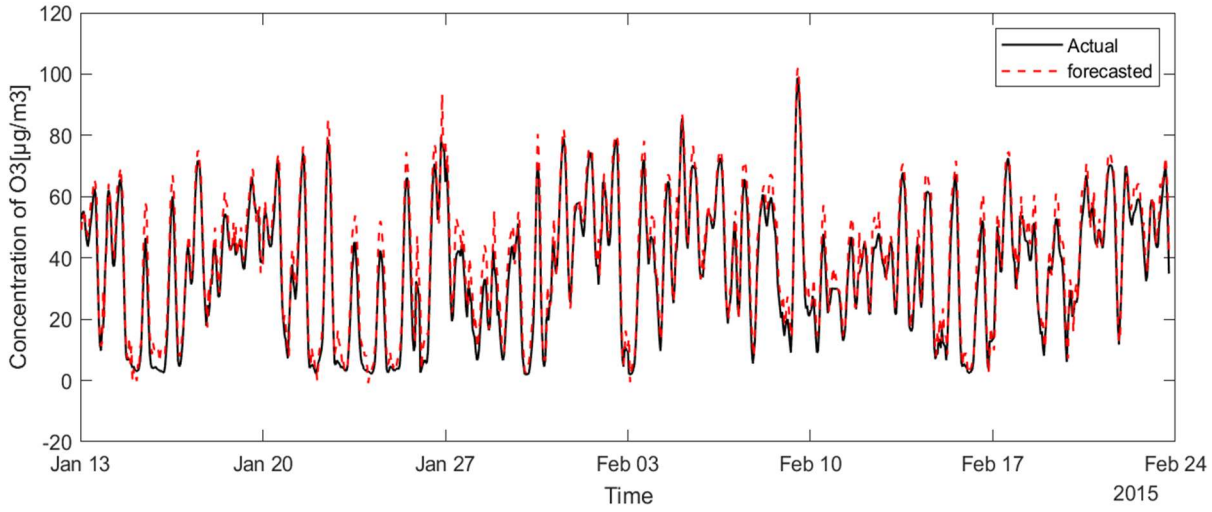

Figure 37: O_3 testing forecasts using the model training by the **miss-forest** imputed dataset

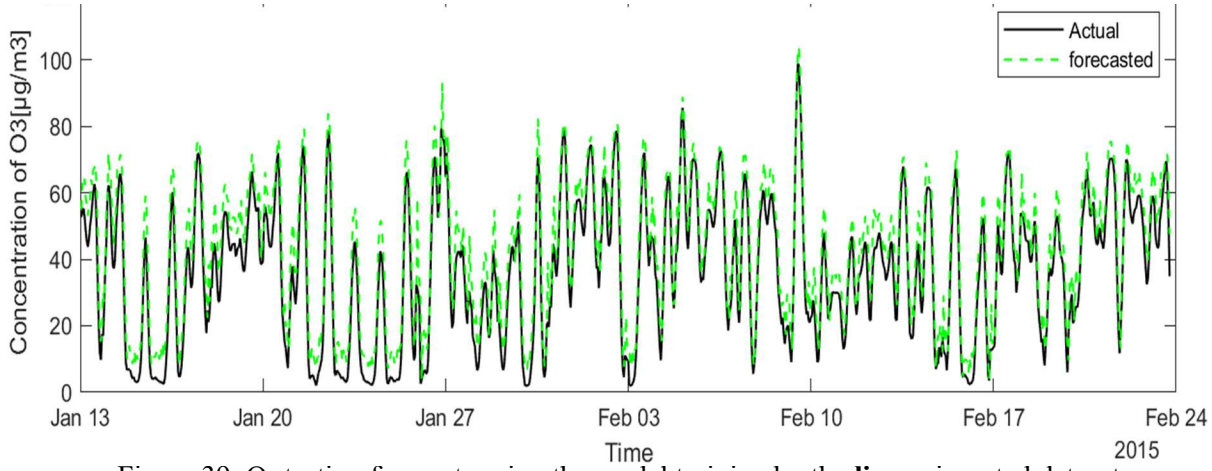

Figure 39: O_3 testing forecasts using the model training by the **linear** imputed dataset.

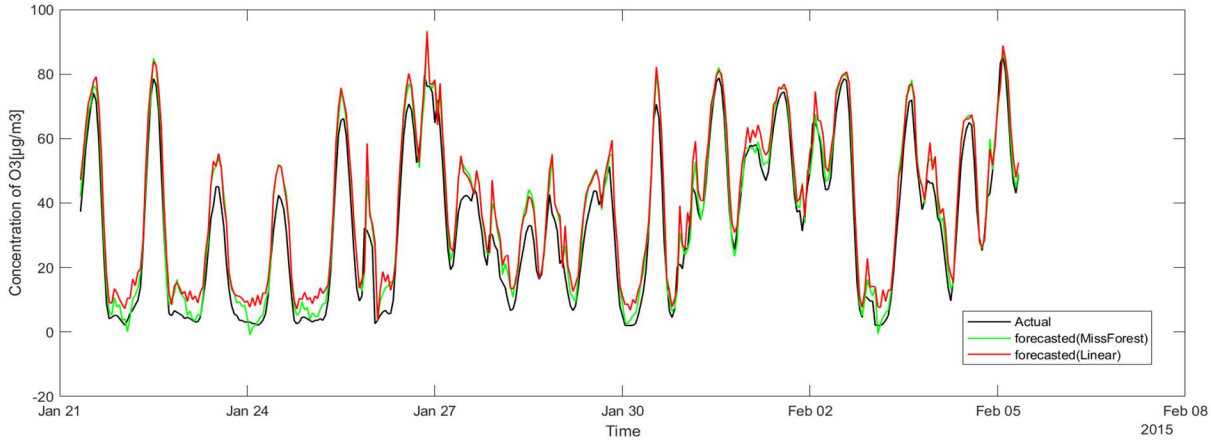

Figure 38: prediction results on Jan 21-Feb 08 -2015

``

5.5.2 Nitrogen Dioxide (NO_2)

The evaluation metrics of the two trained models for forecasting the hourly Nitrogen dioxide levels are presented in Table 26. When comparing the two models, the model trained by the linear interpolation imputed dataset performed slightly better with respect to the error criteria for both the training and testing datasets. When comparing the MAE, the values are almost similar, with a difference of decimal points. The forecasts of the testing set are illustrated in Figure 40 and Figure 41, and a comparison between the two forecasts over 15 days is also presented in Figure 42.

Table 26: Forecasting models' error metrics for NO_2 measurement in the training and testing sets.

	Miss-forest Imputed dataset		**Linear Imputed dataset**	
	Training	**Testing**	**Training**	**Testing**
R^2	0.84	0.78	0.87	0.76
MSE [µg/m³]²	166.62	160.37	142.80	144.798
RMSE [µg/m³]	12.91	12.66	11.95	12.03
MAE [µg/m³]	9.16	9.99	7.86	9.95

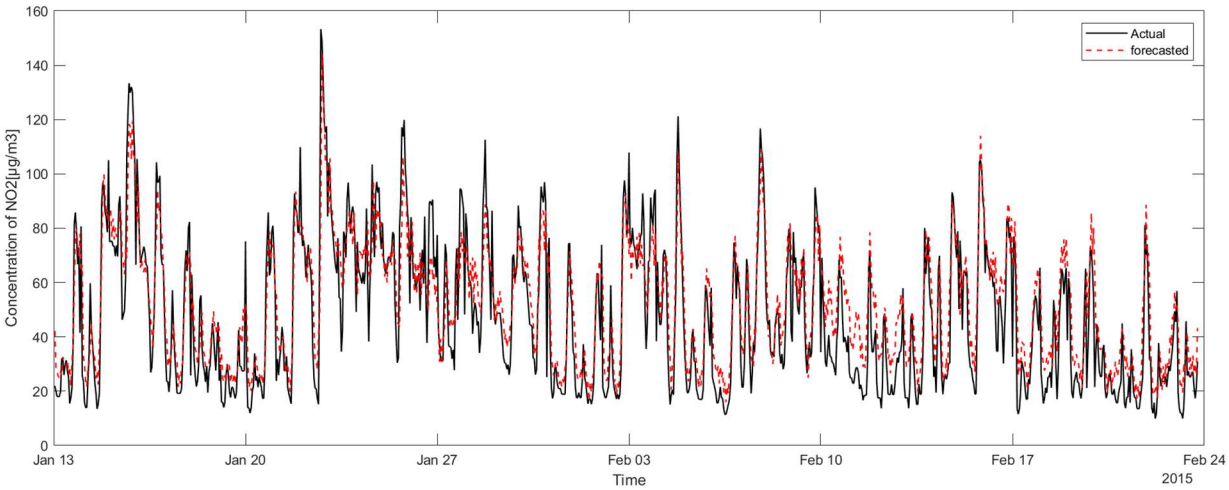

Figure 40: NO_2 testing forecasts using the model training by the **miss-forest** imputed dataset.

``

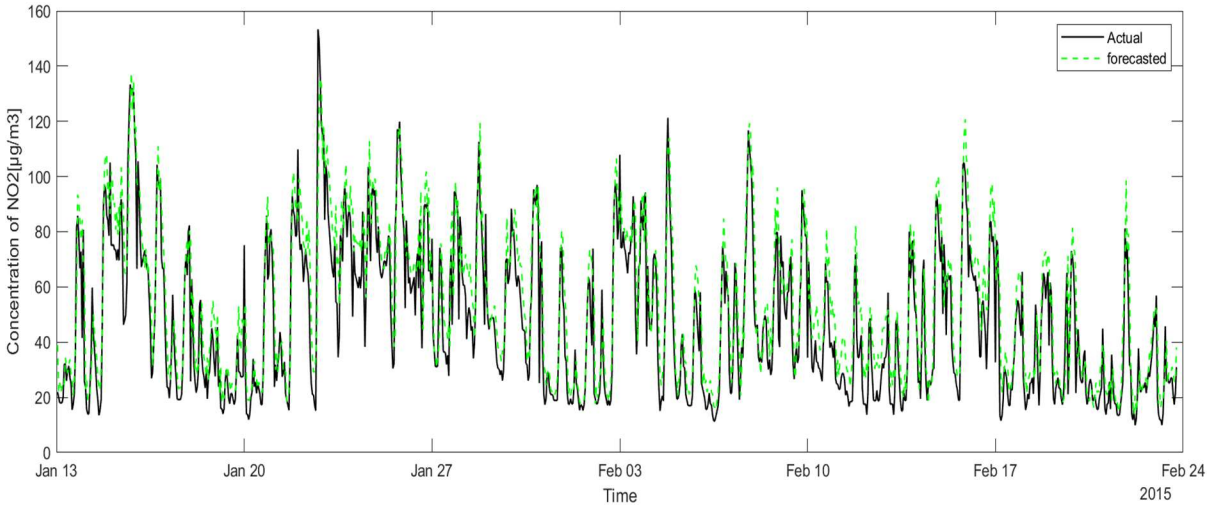

Figure 41: NO_2 testing forecasts using the model training by the **linear** imputed dataset.

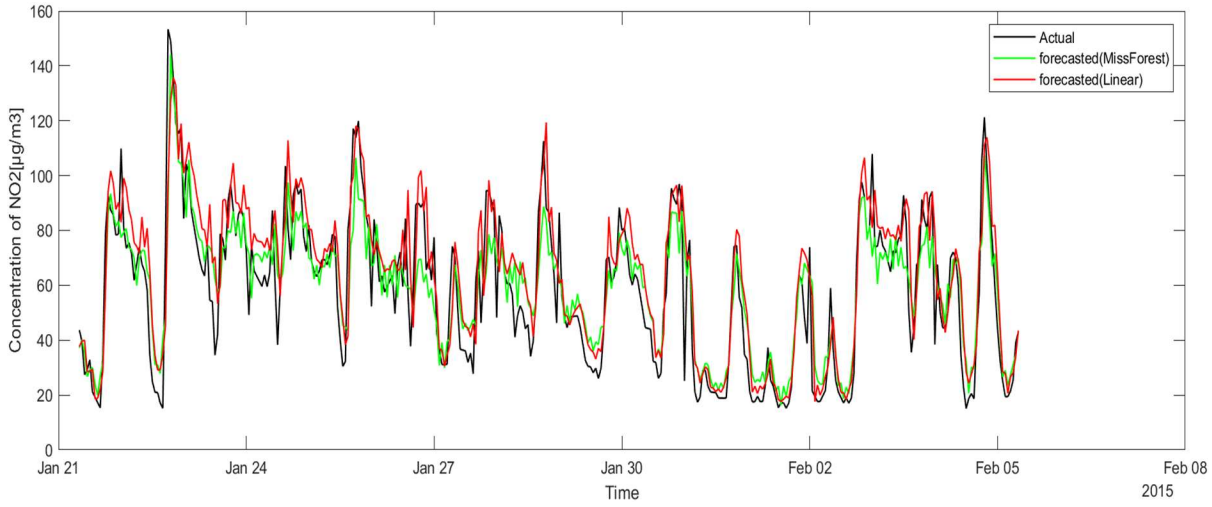

Figure 42: NO_2 prediction results on Jan 21-Feb 08 -2015

5.5.3 Sulfur Dioxide (SO_2)

Based on the results reported in Table 27, both studied models did not predict the SO_2 levels with high accuracies. Comparing the two methods, the miss-forest provided outputs with higher accuracy and a better generalization where the metrics of the testing set are close to the training set.

Figure 43 and Figure 44 show the prediction of the two models. Figure 45 compares the results of the two examined models, where it is clear that both models perform not very accurately at high levels of SO_2 and underestimate these peak concentrations.

``

Table 27: Forecasting models' error metrics for SO_2 measurement in the training and testing sets.

	Miss-forest Imputed dataset		Linear Imputed dataset	
	Training	**Testing**	**Training**	**Testing**
R^2	0.53	0.51	0.49	0.33
MSE [µg/m³]²	297.40	191.02	455.13	261.05
RMSE [µg/m³]	17.25	13.82	21.33	16.16
MAE [µg/m³]	6.61	6.01	7.20	9.75

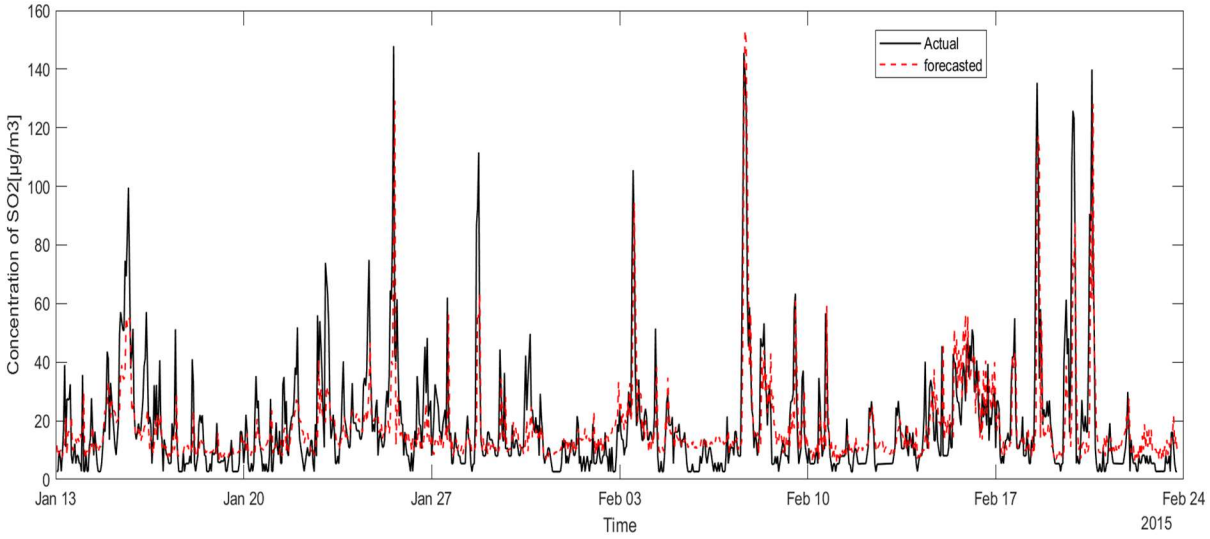

Figure 43: SO_2 testing forecasts using the model training by the **miss-forest** imputed dataset.

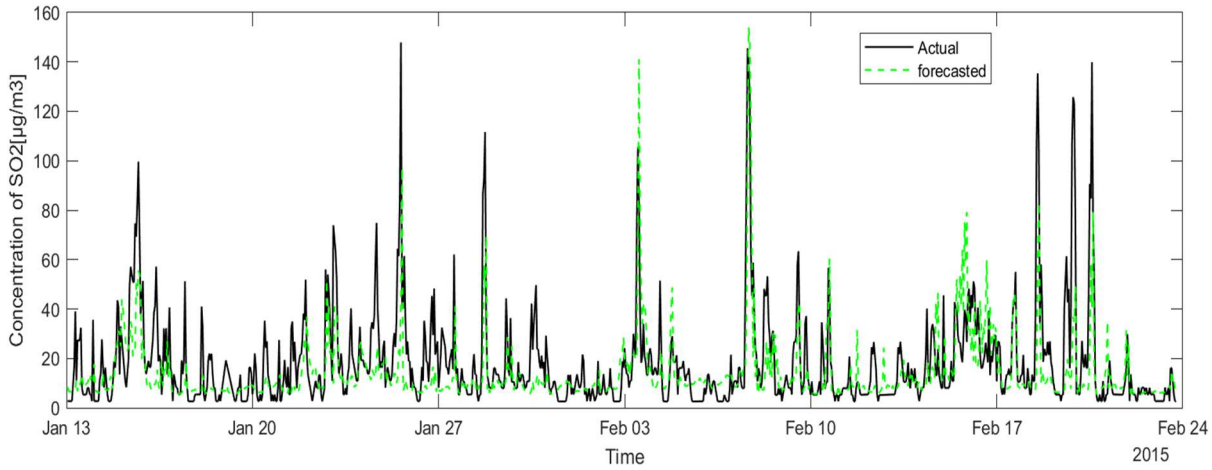

Figure 44: SO_2 testing forecasts using the model training by the **linear** imputed dataset.

``

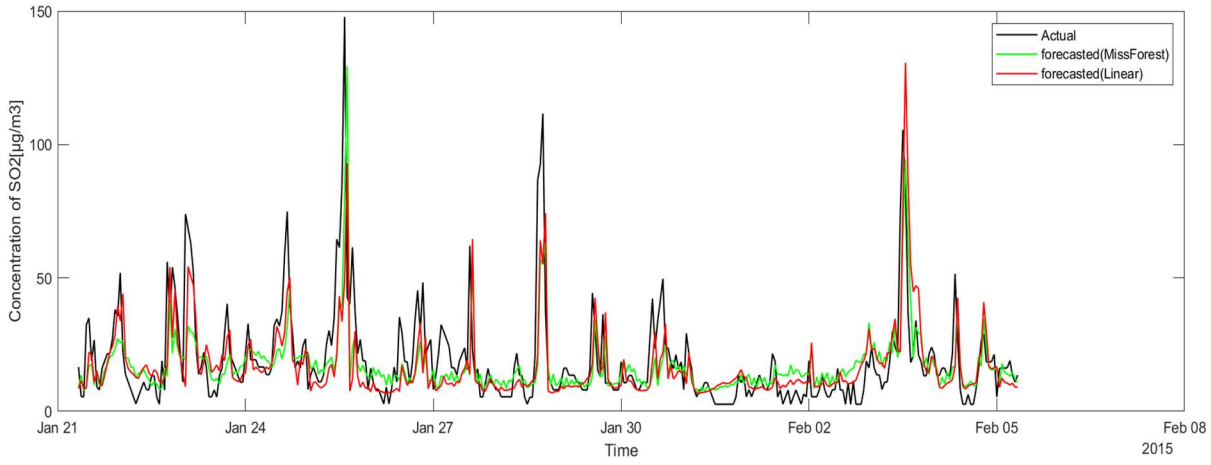

Figure 45: SO_2 prediction results on Jan 21-Feb 08 -2015

5.5.4 Carbon Monoxide (CO)

As demonstrated in Table 28, CO forecasted levels are more accurate for the linear imputed model; nevertheless, both models could be considered accurate and reliable. Figure 46 and Figure 47 illustrate the forecasts of the testing sets, and Figure 48 compares the predictions of the two considered models in a period of 15 days.

Table 28: Forecasting models' error metrics for CO measurement in the training and testing sets.

	Miss-forest Imputed dataset		**Linear Imputed dataset**	
	Training	**Testing**	**Training**	**Testing**
R^2	0.92	0.92	0.93	0.92
MSE [µg/m³]²	0.02	0.05	0.023	0.04
RMSE [µg/m³]	0.16	0.21	0.15	0.20
MAE [µg/m³]	0.10	0.17	0.09	0.16

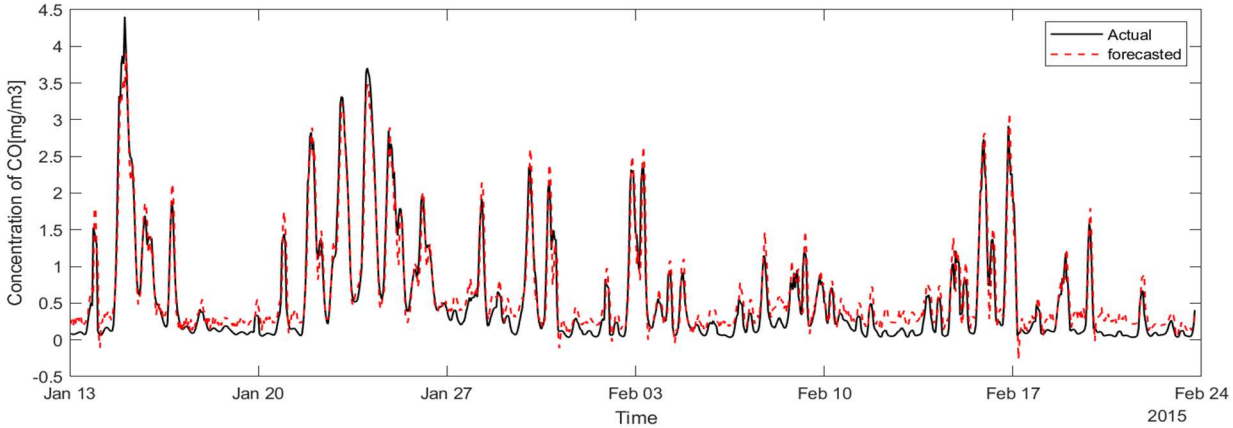

Figure 46: CO testing forecasts using the model training by the **miss-forest** imputed dataset

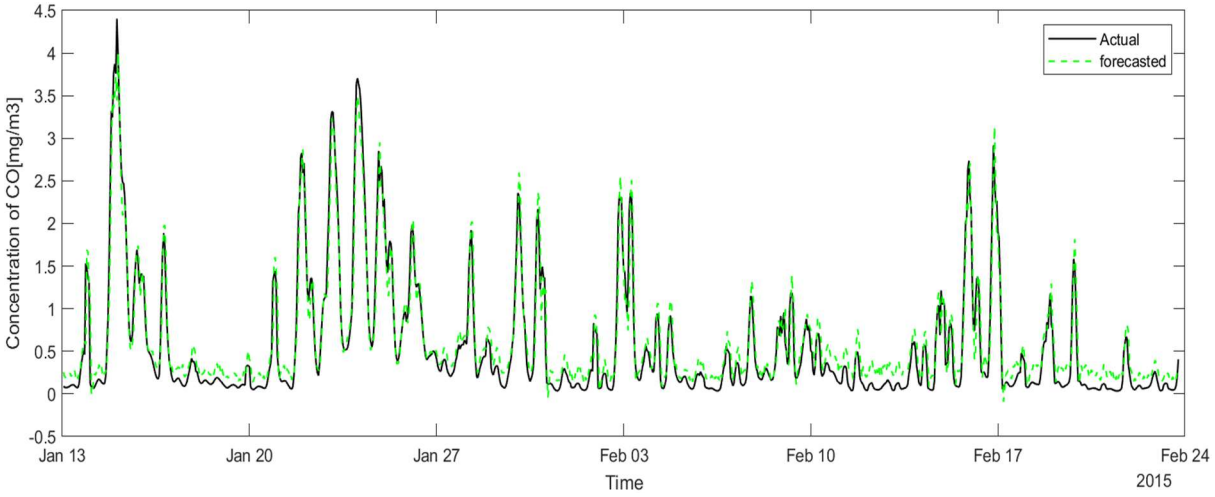

Figure 48: CO testing forecasts using the model training by **linear** imputed dataset

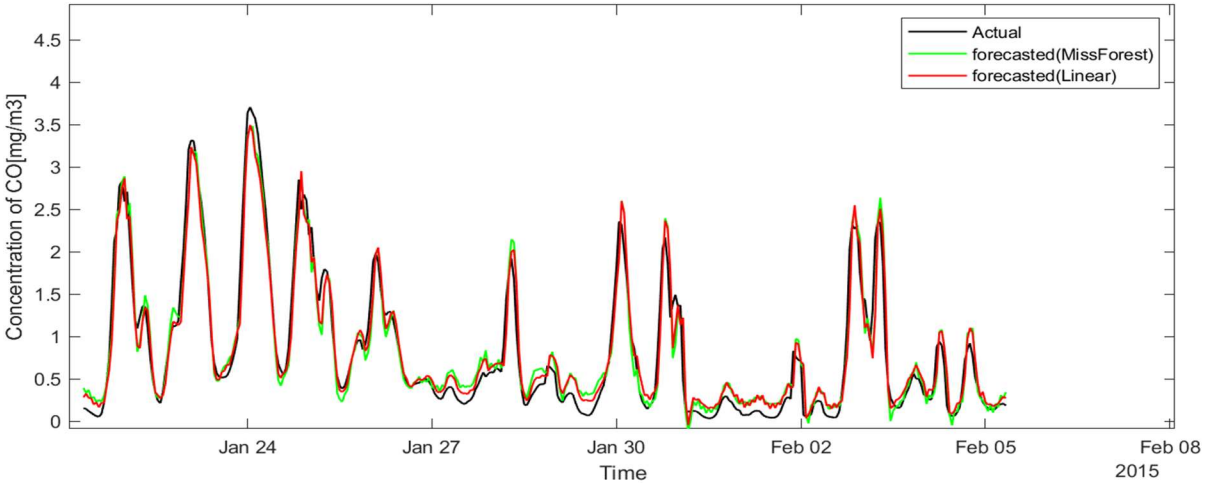

Figure 47: CO prediction results on Jan 21-Feb 08 -2015

5.5.5 Particulate Matter 10 (PM10)

When comparing the results reported in Table 29, the MSE of the linear imputed dataset is lower than the one achieved by the miss-forest imputed model. On the other hand, lower MAE is achieved by the miss-forest imputed model. Generally, all error metrics of both models are considered reliable. Figure 49 and Figure 50 show the estimated concentrations of both models, and Figure 51 compares these predicted concentrations on Jan 21st to Feb 8^{th,} 2015.

``

Table 29: Forecasting models' error metrics for PM10 measurement in the training and testing sets.

	Miss-forest Imputed dataset		Linear Imputed dataset	
	Training	**Testing**	**Training**	**Testing**
R^2	0.978	0.97	0.97	0.98
MSE [µg/m³]²	252.39	482.35	371.86	347.36
RMSE [µg/m³]	15.89	21.96	19.28	18.64
MAE [µg/m³]	6.14	7.98	7.58	9.40

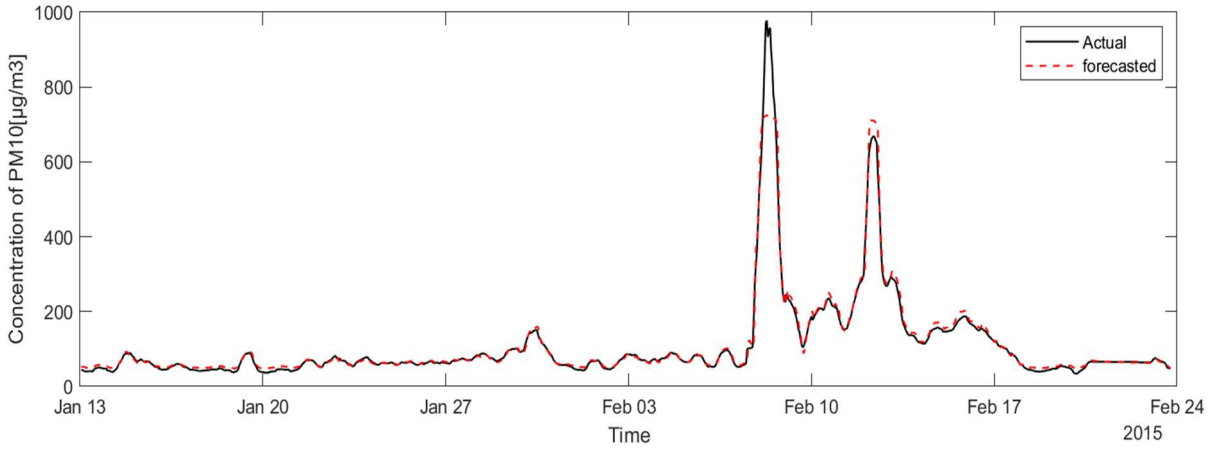

Figure 49: PM10 testing forecasts using the model training by **miss-forest** imputed dataset

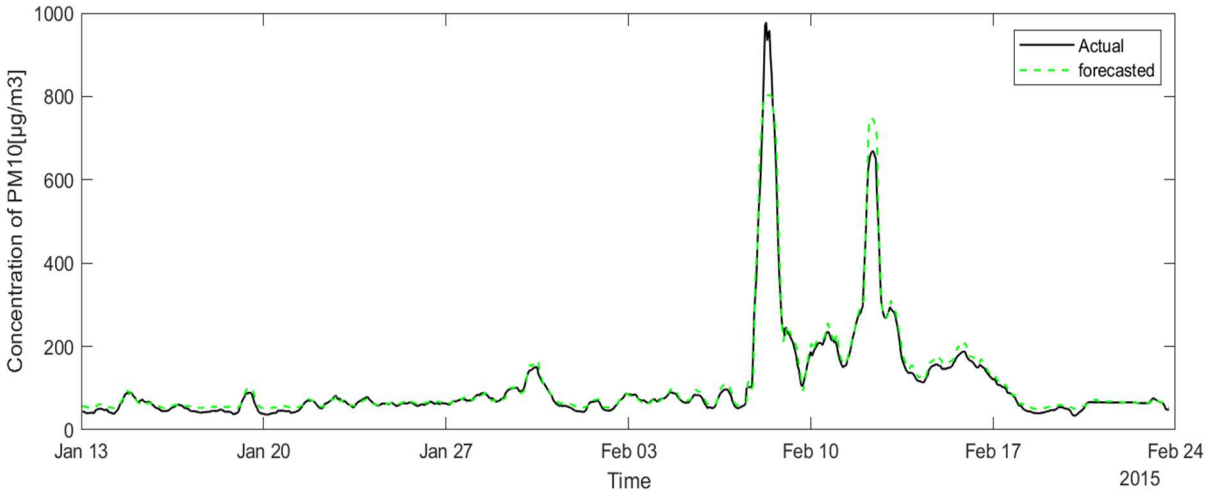

Figure 50: PM10 testing forecasts using the model training by **linear** imputed dataset

``

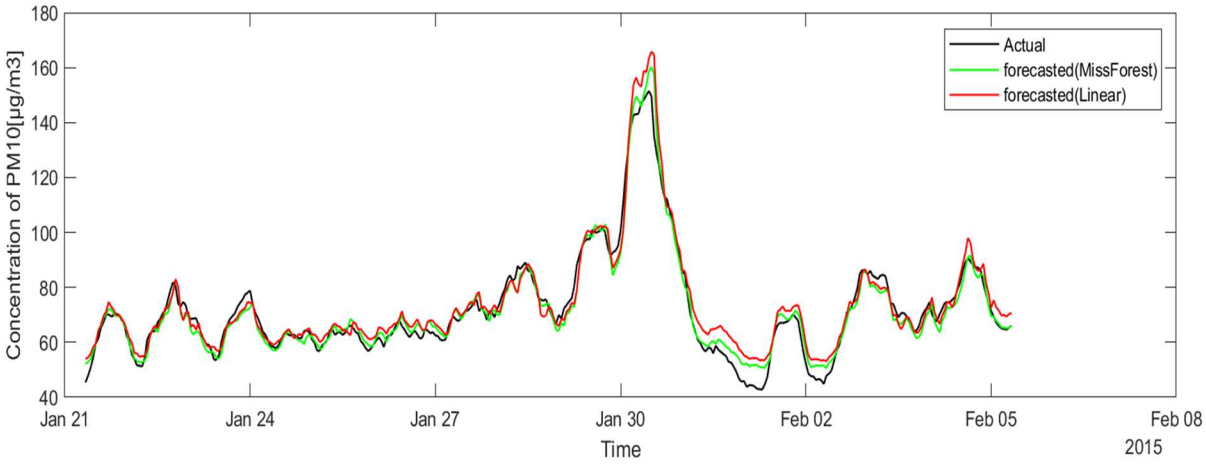

Figure 51: PM10 prediction results on Jan 21-Feb 08 -2015

5.5.6 Particulate Matter 2.5 (PM2.5)

According to the reported estimates in Table 30, similar to PM10, the two models for forecasting PM2.5 are considered reliable with a slight superiority of miss-forest when comparing the MAE. Results of both built models are presented and compared in Figure 53-Figure 52. From Figure 52, it can be observed that both the minimum and maximum values of concentration are overestimated in some cases by both models.

Table 30: Forecasting models' error metrics for PM2.5 measurement in the training and testing sets.

	Miss-forest Imputed dataset		Linear Imputed dataset	
	Training	**Testing**	**Training**	**Testing**
R^2	0.98	0.97	0.98	0.97
MSE [µg/m³]	27.96	14.11	26.21	12.75
RMSE [µg/m³]	5.29	3.76	5.12	3.57
MAE [µg/m³]	2.46	2.78	2.36	2.89

``

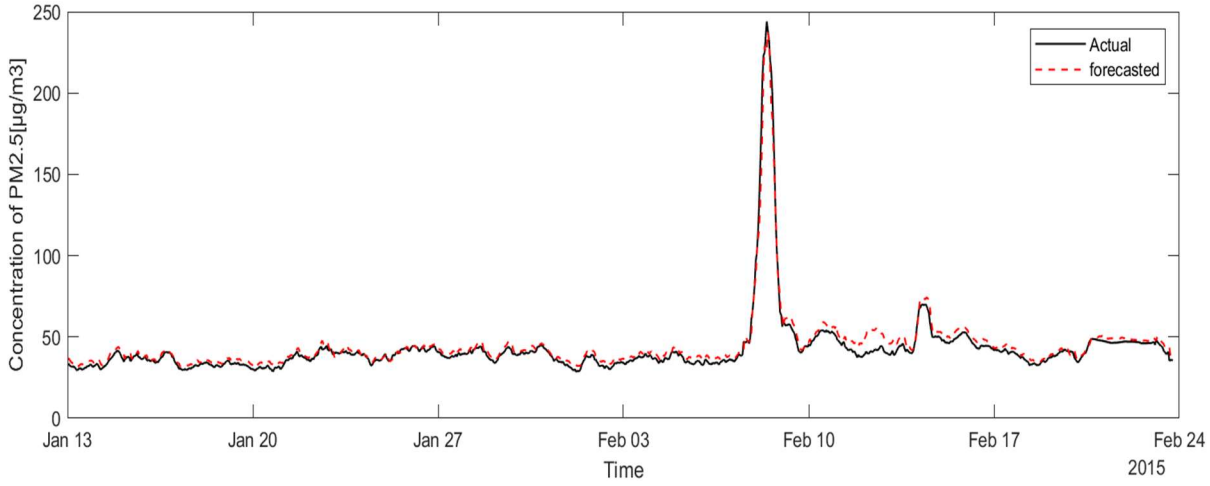

Figure 52: testing forecasts using the model training by **miss-forest** imputed dataset

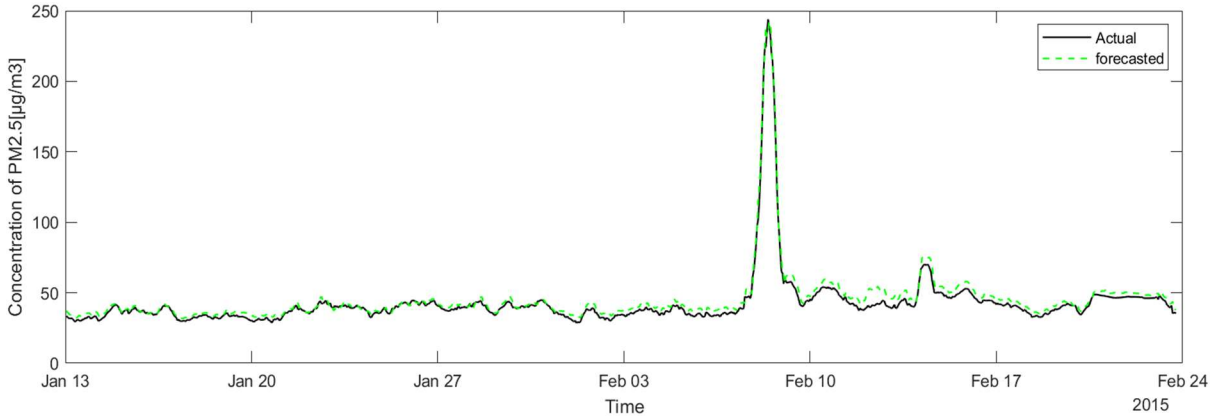

Figure 54: PM2.5 testing forecasts using the model training by **linear** imputed dataset.

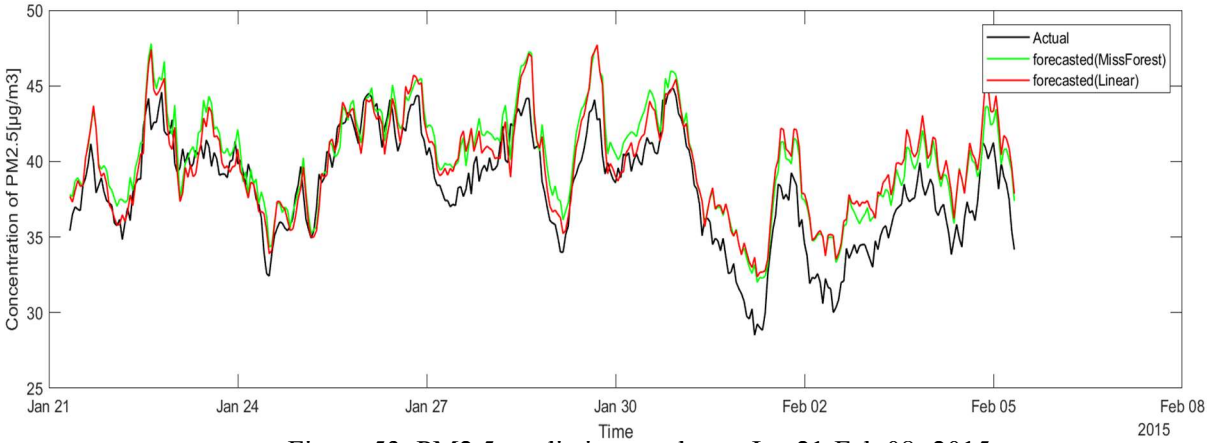

Figure 53: PM2.5 prediction results on Jan 21-Feb 08 -2015.

``

5.6 Hourly Forecast of Air Quality Index (AQI)

The AQI calculated from the actual and forecasted data is analyzed and compared in this section. Figs. 55-58 represent a detailed confusion matrix chart comparing the actual and forecasted results of both models' training and testing sets. Table 31 summarizes the error metrics of forecasting the AQI values. Upon inspection, one observes that the miss-forest imputation technique outperforms the linear one in forecasting generalization, e.g., the miss-forest MAE value was 3.27 compared to 4.69 for the linear imputed data. The linear imputed datasets achieved an overall classification accuracy of 95.75% on the training set and 90.31% on the testing data set, whereas the miss-forest imputed dataset performed better and achieved a classification accuracy of 95.65% and 92.48% for training and testing sets, respectively. These results confirm that coupling ANN with miss-forest imputed data leads to higher accuracy forecasting and better generalization when tested on unseen data. Conversely, the gap between the testing and training accuracies of the linear imputed model is considerable and reflects the weakness of this model to perform with moderately consistent accuracy when validated with unseen data.

In addition to the importance of forecasting the AQI and accurately classifying its category, it is also essential to report the corresponding critical pollutant with the highest AQI value. Reliable reporting of this pollutant can contribute to investigating the sources of pollution to take prior precautionary actions. Table 32 shows a precise count of the true and false defined categories and critical pollutants for both training and testing sets of the two models. From that table, similar superiority of the miss-forest model can be observed where both the air quality category and the corresponding pollutant were correctly classified 95.64 % of overall cases of the training set and 92.41% for the testing set.

``

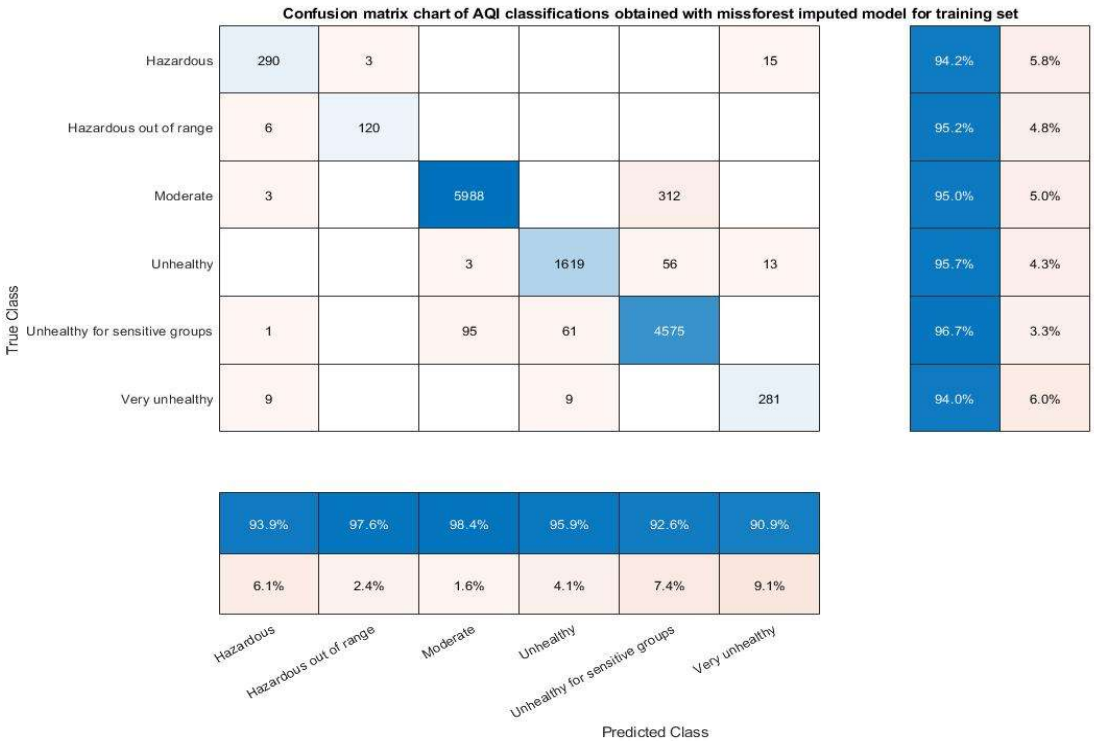

Figure 55: Training set confusion matrix for the AQI categories from the **miss-forest** model.

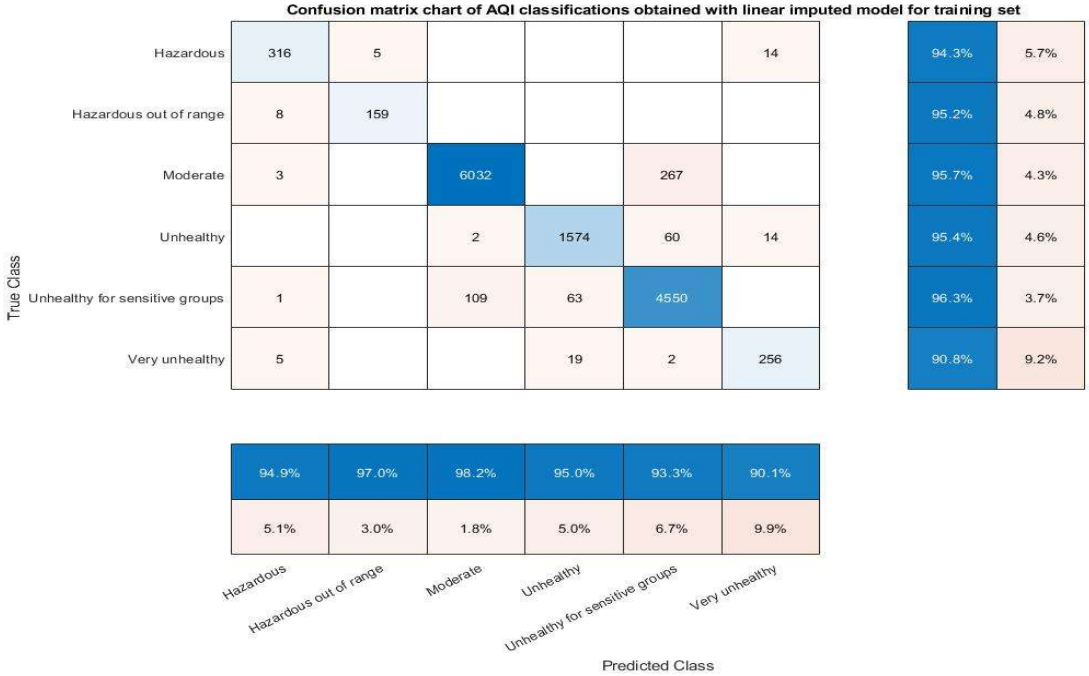

Figure 56: Training set confusion matrix for the AQI categories from the **linear** model.

``

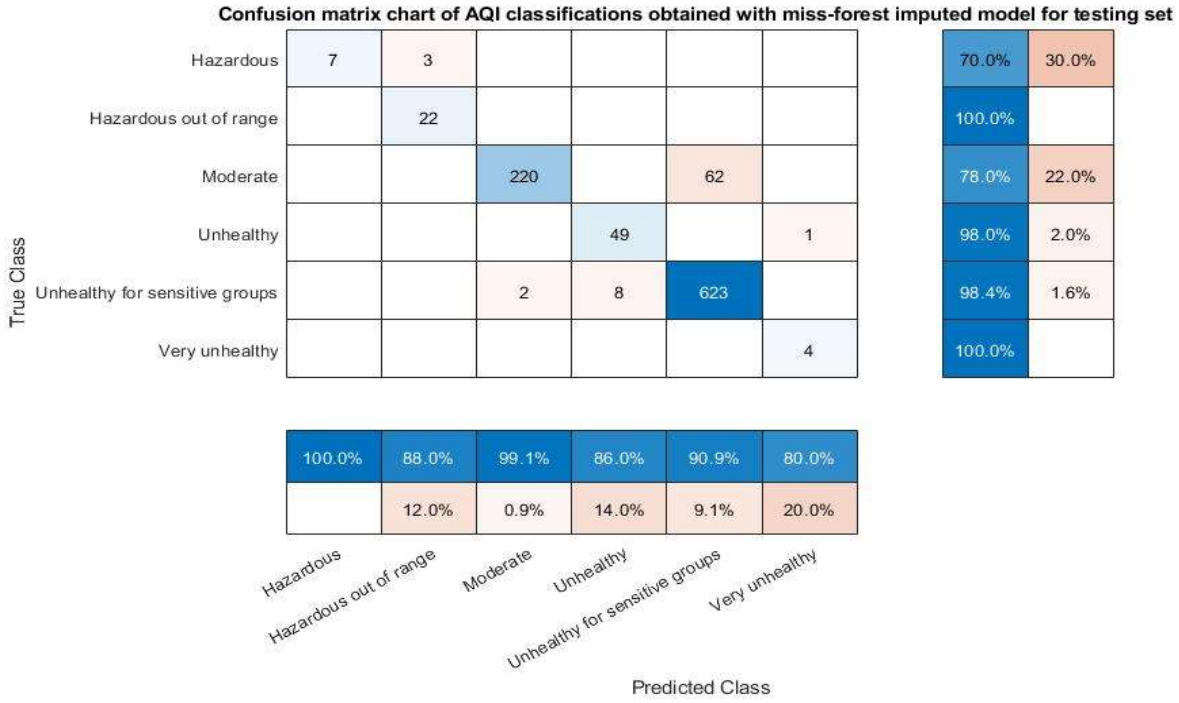

Figure 57: Testing set confusion matrix for the AQI categories from the **miss-forest** model

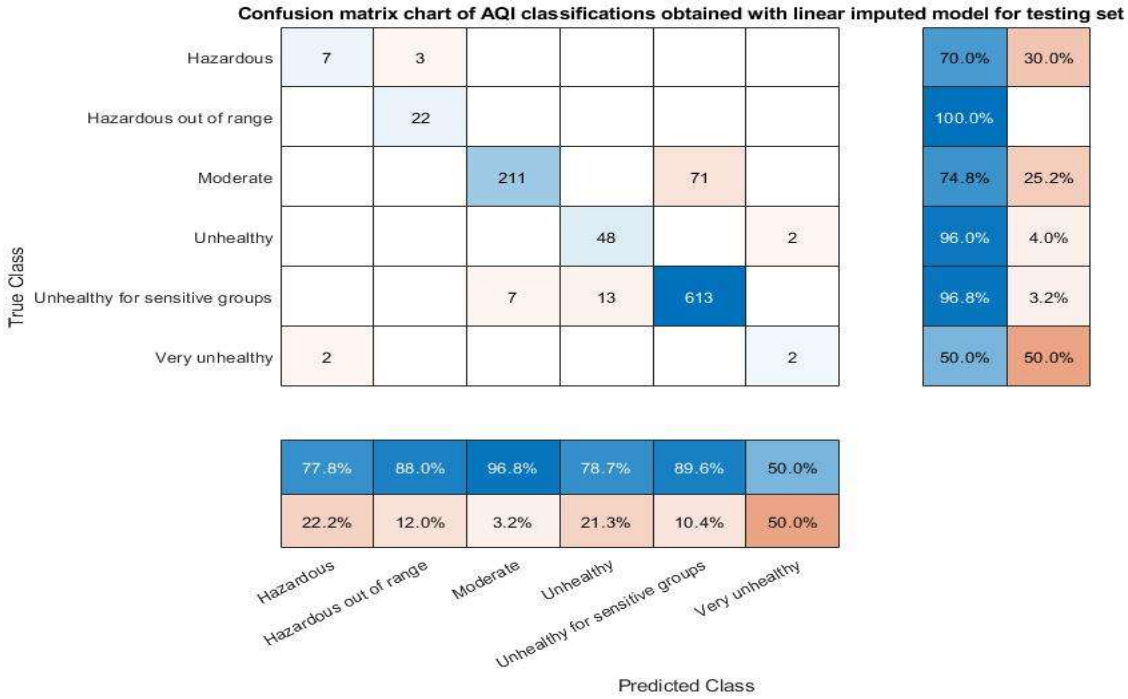

Figure 58: Testing set confusion matrix for the AQI categories from the **Linear** model

``

Table 31: Forecasting models' error metrics for AQI in the training and testing sets.

	Miss-forest Imputed dataset		Linear Imputed dataset	
	Training	**Testing**	**Training**	**Testing**
R^2	0.81	0.93	0.78	0.98
MSE	131.25	95.18	178.56	297.05
RMSE	11.46	9.76	13.36	17.24
MAE	3.00	3.27	3.34	4.69

Table 32:Count of true and false forecasted

Condition	Training miss-forest imputed	Testing miss-forest imputed
Category=True & Critical pollutant=True	12597	924
Category=False & Critical pollutant=True	551	76
Category=True & Critical pollutant=False	267	1
Category=False & Critical pollutant=False	35	0
Condition	**Training linear imputed**	**Testing linear imputed**
Category=True & Critical pollutant=True	12646	901
Category=False & Critical pollutant=True	536	98
Category=True & Critical pollutant=False	251	2
Category=False & Critical pollutant=False	36	0

5.7 Conclusion (Case study 2)

Forecasting AQI is a task that requires attention to multiple factors, including the missing observations raw data, the high inconsistency in data, the proper selection of predictors and lags, and the high temporal correlations between the concentrations of pollutants. Moreover, this task requires the appropriate choice of a robust, reliable methodology for training and building the forecasting

``

models and tuning their hyperparameters. This chapter proposed an approach considering and tackling all of these mentioned important accuracy affecting factors.

For missing data imputation, two different imputation methodologies were tested by training an optimizable ANN for forecasting the six pollutants levels and classify the hourly AQI and identify the pollutant with the highest AQI. Although both trained models performed adequately, more generalized forecasting was observed by the models trained using the miss-forest imputed dataset.

All the pre-processing and data preparation steps were comprehensively considered after imputing the missing observations before building the forecasting models. Although modeling the pollutants' levels with the selected features was adequate for almost all the pollutants, further analysis and testing should be taken for forecasting SO_2 levels since its predictions were the least accurate compared to other pollutants.

``

Chapter 6
Thesis Conclusions and Future Work

Time series forecasting with machine learning algorithms is a complex task requiring the consideration of different factors, such as feature selection, feature scaling, feature extraction, and dimensionality reduction. Besides that, it requires a decent size of training data and powerful software. Moreover, the construction of the forecasting model itself needs the implications of different optimization algorithms for both the models' parameters optimization and tuning their hyperparameters.

The main objective of this research was to test different ML algorithm for time series forecasting, evaluate them, and employ them for different purposes in different applications. This research considered two different case studies of time-series forecasting:

1- Regional wind power forecasting.
2- Air quality index (AQI) forecasting.

The first case study addressed Ontario's wind power forecasting comprehensively from different perspectives. Besides the proposed deep feature selection approach, a comparative analysis was conducted in this case study to compare the performances for different ML algorithms for one-step and multi-step ahead forecasting.

- For one-step ahead forecasting, by evaluating and analyzing the performance of the models when tested on unseen testing data, It can be concluded that SVR/SVM is one of the most promising robust ML-based forecasting models. This algorithm can build reliable generalized models that can perform well with new data where the testing MAPE % reached a value of 13 % for the testing predictions. Although almost a similar MAPE was calculated from the ensemble model, the ensemble model results indicate that the utilized approach for combining the models was not adequate to improve the predictions. Hence, using a different ensembling approach such as boosting in future work could increase the accuracy of forecasting results.
- Multi-step step ahead forecasting is essential for some scheduling and managing objectives; however, this task is a complex task that requires special considerations of the dependence of the targets on each other. When comparing the direct and MIMO multi-step forecasting methodologies,

``

it can be seen that although the MIMO approach is harder to train and require advanced machines, it can perform better, especially if the trained model has the built-in recursive property. Thus, the MIMO method with a type of recursive properties can capture the dependencies between steps. Moreover, it avoids the time-consuming multi-model training for each step. From that perspective, in future studies, an attempt to apply different ML for MIMO forecasting could provide better insight into the ability of this approach.

To sum up, although better precisions were reported in the literature for wind power forecasting, our obtained accuracies are considered adequate for regional forecasting. This notation is based on the fact that all the published and conducted studies focus on specific sites or even one wind turbine with a known hub height, physical characteristics, and loss coefficients. Furthermore, unlike the carefully measured weather parameters affecting a single turbine at specified heights, the meteorological data used in this study were from different locations across Ontario, measured with different apparatuses with different settings. Hence, these factors would definitely affect the overall forecasting results, even though the ML models will try to capture and adapt them and train the models for reliable predictions. Hence, with further spatial and weather data availability and more compressive optimization and tuning, a general regional model representing Ontario's wind power could be constructed to be used for other purposes such as electricity pre-scheduling to avoid surplus production of other sources.

From the second case study, it can be concluded that forecasting AQI is a task that requires attention to multiple factors, including the missing observations raw data, the high inconsistency in data, the proper selection of predictors and lags, and the high temporal correlations between the concentrations of pollutants. Moreover, this task requires the appropriate choice of a robust, reliable methodology for training and building the forecasting models and tuning their hyperparameters. Therefore, this research proposed an approach considering and tackling all of these mentioned important accuracy affecting factors.

For missing data imputation, two different imputation methodologies were tested by training an optimizable ANN for forecasting the six pollutants levels and classify the hourly AQI and identify the pollutant with the highest AQI. Although both trained models performed adequately, more generalized forecasting was observed by the models trained using the miss-forest imputed dataset. All the pre-processing and data preparation steps were comprehensively considered after imputing the missing observations before building the forecasting models. Although modeling the pollutants' levels

``

with the selected features was adequate for almost all the pollutants, further analysis and testing should be taken for forecasting NO_2 levels since its predictions were the least accurate compared to other pollutants.

In future work, an investigation of the additional features that can improve the accuracy of the forecasted SO_2 is intended because it would enhance the overall forecasted AQI. In addition, the employment of other robust ML forecasting methods such as support vector machines could be expected to improve the predictions' accuracy and increase reliability.

``

Appendix A

Summary of Reviewed Recent Papers Utilizing Machine Learning Algorithms for Renewable Power Forecasting

Table A.1: Summary of ANN methods

References	Forecasting target and horizon	ANN method	Important conclusions	Accuracy of the proposed model
Nielson et al. (2020)	Monthly wind power	FFBP-ANN	- The atmospheric effects on wind power curved could be explained by incorporating wind density into the input's features.	Improved MAE% by 59% compared to IEC*
Chen et al. (2019)	Hourly wind speed	FFBP-ANN ADALINE -NN RBF-ANN	- The time-series input size dramatically affects the performance of the ANN approaches. - Different learning rates resulted in considerably different interpretations.	RBF-ANN: MAE = 1.112 m/s
Grassi and Vecchio (2010)	Monthly generate wind power.	Two-layer FFNN with sigmoid and Tanh activation functions	- Considering maintenance hours as an input has a significant impact on increasing the forecasting reliability.	Training MAE = 0.0109 MWh
Liu et al. (2017)	48 h-ahead of wind power	ANFIS combining BPNN, LSSVM, RBANN	- Utilization of PCC for input selection contributes to improving accuracy. - Combining three ANN approaches as inputs to ANFIS ensures accurate results throughout the four seasons.	Spring: MAPE = 11.76% Fall: MAPE = 6.7%
Abuella and Chowdhury (2015a)	Hourly solar power	FF-ANN	- Input data pre-processing and clustering improve results. - Elimination of night hours results in better predictions.	Testing $R^2 = 0.9665$
O'Leary and Kubby (2017)	Hourly solar power	ANN	- Masking the ANN input data into specific categories according to the error rate was why the highly reliable drawn-out forecasting results.	MAE = 5.99 W
Ozoegwu (2019)	Monthly mean daily global solar radiation	Hybridization of non-linear autoregressive + ANN	- Forecasting results by the proposed model are satisfactory for longer-term forecasting horizons (up to 2 years-ahead) under different climate conditions.	$R^2 = 0.92$
Yagli et al. (2019)	Hourly global horizontal irradiance (GHI)	68 ML and statistical approaches	- This study can offer researchers' guidance in using the appropriate forecasting approach based on the climate conditions for GHI furcating.	N/A
Ghimire et al. (2019)	Daily global horizontal irradiance	FFBP-ANN	- Nearest component analysis incorporates in choosing the appropriate features that result in better predictions. - Integrating optimization techniques is recommended and would increase the reliability of ANN and ELM forecasting approaches.	$R = 0.967$

IEC, The International Electrotechnical Commission.

``

Table A.2: Summary of RNN/ELM/SVM methods

References	Forecasting targe and horizon	RNN method	Important conclusions	Accuracy of the proposed model
Syu et al. (2020)	Ultra-short-term wind speed Multi-step	GRU	- Number of previous steps as inputs to GRU is essential and affects forecasting results. - GRU requires less parameter tuning and shorter training time.	Step 1: MAE = 0.130 m/s Step 2: MAE = 0.222 m/s Step 3: MAE = 0.302 m/s
Yu et al. (2019)	Short-term wind power	LSTM- enhancement forget gate (LSTM-EFG)	- Proposed approach accelerates convergence. - Pre-processing of data represented in clustering and filtering noticeably improves prediction results.	Clustering increases the accuracy by 18.3%
Niu et al. (2020)	Wind power at different time horizons (MIMO) forecasting	Attention mechanism GRU	- The proposed model can continuously extract the spatial-temporal features in data, which boosts the forecasted power accuracy. - MIMO model offers stable forecasting results at different time horizons.	Step 1: MAPE = 4.35% Step 2: MAPE = 7.97% Step 3: MAPE = 11.54%
Aslam et al. (2019)	Long-term solar radiation forecasting (1-year interval)	Comparative study between RNN, LSTM, and GRU and other ML approaches.	- Reinforce the effectiveness of different RNN models in forecasting.	N/A
Rana et al. (2016)	Half-hourly PV power output	Elman RNN.	- Multivariate input model to Elman RRN is more reliable. - Elman RNN-based model can alternate the conventional persistence forecasting methodologies.	Univariate model: MAE = 90.95 kW
Hosseini et al. (2020)	Direct normal irradiance. At different horizons	GRU and LSTM (multivariate)	- Highlight the effectiveness of incorporating wind speed and cloud coverage as inputs to the forecasting networks. - GRU is adequate, with no superiority of LSTM.	The multivariate model outperforms the univariate model by 34.42%
Hossain and Mahmood (2020)	Short-term PV power. (6-12-24 h ahead)	LSTM	- Using k-means clustering to create numerical synthetic approximations of the sky type and incorporating them as inputs to the LSTM dramatically improved the forecasting results.	MAE Fall for 24 h: 0.36 MW
Ref				
He and Xu (2019)	Ultra-short term wind speed	SVM with wavelet + polynomial hybrid kernel function	- The SVM model's combined kernel function can outperform the single regular functions and improve the interpolation of the model's extrapolation ability. - Pre-processing of historical data and clustering also contribute to increased accuracy.	The hybrid function reduced the mean error by 3.94 %.
Tabari et al. (2012)	Short-term wind power	SVM with a linear kernel function.	- SVM models tackle the issue of local optimality that appears in other ML networks.	DBSCAN clustering reduced MAE by 54%.
Quej et al. (2017)	Daily global solar irradiance	SVM RBF-kernel function	- SVM with kernel function can tolerate noisy input data in humid areas and result in an accurate model with reasonable computational time. It outperforms the ANFIS and ANN models.	RMSE = 3.089 $MJm^{-2}d^{-1}$
Ahmad et al. (2020)	PV module and power generation parameters	Polynomial kernel SVM RBF kernel SVM	- Detailed SVM-based modeling study can offer helpful guidance for solar power-related problems.	Poly kernel: total RMSE = 5.1674 W BBFkernel: total RMSE = 2.342W
Hamamy and Omar (2019)	Solar irradiance at different time horizons.	LSSVM with RBF kernel	- LSSVM models perform better for short-term forecasting; the longer the forecasting term, the lower the predictions' accuracy.	Not given
Li et al. (2019)	Wind power	ELM with kernel mean P-power error loss function	- The ELM model with the special loss function can outperform the classical BP-ANN; nevertheless, the fast convergence of ELM is lost when the proposed loss function is introduced to ELM. Hybridizing ELM with optimization algorithms to estimate optimal weights and biases is effective and recommended.	MAE = 255.4860kW
Hossain et al. (2017)	Hourly and daily PV solar power	Classical ELM	- ELM can perform robustly for long-term forecasting with short training time.	Testing $R^2 = 0.8936$

``

Table A.3: Summary of ML-Metaheuristics methods

References	Forecasting target and horizon	ML method-metaheuristic method	Important conclusions	Accuracy of the proposed model
Jafarian-Namin et al. (2019)	Monthly wind power generation	ANN-GA ANN-PSO	- Hybridization with optimization algorithms increases accuracy - GA is superior to PSO in this case.	ANN-GA: R^2 = of 0.9212 GA reduced RMSE by 3.9% PSO reduced RMSE by 3.07%
Pedro and Coimbra (2012)	1 and 2 h-ahead solar power output	ANN-GA	- GA-ANN surpassed ARIMA and KNN approaches in improving the accuracy of forecasting	1 h ahead R^2 = 0.96 2 h ahead R^2 = 0.93 RMSE improved by; 32.3% for 1 h 35.1% for 2 h
Flores et al. (2019)	Day-ahead wind speed	NN-DE	- NN-DE approaches resulted in the highest accuracy predictions compared to MLP-ANN	SMAPE = 33.552 m/s
Salcedo-Sanz et al. (2011)	Short-term wind speed	SVM-EP SVM-PSO	- Proposed models surpassed the multi-layer perceptron forecasting approach	EP-SVMr: MAE = 1.7921 m/s PSO-SVMr:MAE = 1.7823 m/s
Tian et al. (2020)	Short-term wind speed	LSSVM-BSOA	- LSSVM-BSOA is superior to LSSVM-GA, LSSVM-PSO, and other standalone approaches.	MAE = 0.1113 m/s
Wang et al. (2015)	Short-term wind speed and multi-step ahead	SVM-COA SVM-PSO	- Implementation of the COA-SVR model is advanced to that of the PSO-SVR and GA-SVR methods,	1 step ahead: MAE = 0.6836 m/s 2 steps ahead: MAE = 1.0051 m/s
Yin et al. (2017)	Wind power forecasting and multi-step ahead	ELM-CSO	- CSO can address the premature convergence of ELM and surpass other algorithms.	1 step ahead: MAE=0.104 m/s 2 steps ahead: MAE = 0.157 m/s 3 steps ahead: MAE = 0.186 m/s
Zhang et al. (2017)	Wind speed a mean half-hour	HBSA-ELM	- Compared to other approaches, the proposed model has robust performance in capturing the non-linear attributes of wind speed	MAE = 0.372 m/s
Du et al. (2019)	Short term wind power multi-step ahead	WNN-MOMFO WNN-MOWCA WNN-MOMVO WNN- MOWOA	- Integration of the multi-objective optimizers improves the predictions compared to GNN and WNN	1 step ahead: MAPE = 5.0661% 2 steps ahead: MAPE = 7.7877% 3 steps ahead: MAPE = 10.6968%
Li et al. (2020)	Short-term wind power	SVM-DA-DEA	- The model showed a better performance compared to other models such as ANN	R^2 = 0.9791 for winter dataset R^2 = 0.9544 for autumn dataset.
Vinothkumar and Deeba (2020)	Short-term wind speed	SVM-ALO-PSO LSTM-ALO-PSO	- The effectiveness of the proposed optimizer integrated into the ML forecasters is proven comparing to other benchmarking forecasting approaches	ALO-PSO MAE = 0.0027 m/s ALO-LSTM = 0.0126 m/s
Kumar et al. (2018)	PV power generation	ELM-PSO ELM-APSO ELM-CRPSO	- ELM model surpassed BP-ANN - The hybridized ELM with different PSO approaches reduced the prediction error.	ELM: MAPE = 2.9417% PSO-ELM: MAPE = 2.7736% CRPSO-ELM: MAPE = 2.2207% APSO-ELM: MAPE = 1.440%
Liu et al. (2019)	Short-term PV power outputs	ICSO-ELM	- Improving the CSO considerably enhanced the forecasting ability of the ELM	MAPE = 5.54%

``

Appendix B

Air Quality Index Calculation Procedure

The presented procedure for calculating air quality index is from the US Environmental Protection Agency (US EPA), "Criteria air pollutants," America's Children and the Environment, USEPA, Washington, DC, USA, 2015."

The AQI is the highest value calculated for each pollutant as follows:

a. Identify the highest concentration among all of the monitors within each reporting area and truncate as follows:

Ozone (ppm) – truncate to 3 decimal places

$PM_{2.5}$ ($\mu g/m^3$) – truncate to 1 decimal place

PM_{10} ($\mu g/m^3$) – truncate to integer

CO (ppm) – truncate to 1 decimal place

SO2 (ppb) – truncate to integer

NO2 (ppb) – truncate to integer

b. Using Table 6, find the two breakpoints that contain the concentration.

c. Using Equation 1, calculate the index.

d. Round the index to the nearest integer.

Equation 1:

$$I_p = \frac{I_{Hi} - I_{Lo}}{BP_{HI} - BP_{Lo}}(C_p - BP_{Lo}) + I_{Lo}.$$

Where I_p = the index for pollutant p

C_p = the truncated concentration of pollutant p

BP_{Hi} = the concentration breakpoint that is greater than or equal to C_p

BP_{Lo} = the concentration breakpoint that is less than or equal to C_p

I_{Hi} = the AQI value corresponding to BP_{Hi}

I_{Lo} = the AQI value corresponding to BP_{Lo}

``

Table 6: Breakpoints for the AQI

These Breakpoints...							...equal this AQI	...and this category
O_3 (ppm) 8-hour	O_3 (ppm) 1-hour[1]	$PM_{2.5}$ ($\mu g/m^3$) 24-hour	PM_{10} ($\mu g/m^3$) 24-hour	CO (ppm) 8-hour	SO_2 (ppb) 1-hour	NO_2 (ppb) 1-hour	AQI	
0.000 - 0.054	-	0.0 – 12.0	0 - 54	0.0 - 4.4	0 - 35	0 - 53	0 - 50	Good
0.055 - 0.070	-	12.1 – 35.4	55 - 154	4.5 - 9.4	36 - 75	54 - 100	51 - 100	Moderate
0.071 - 0.085	0.125 - 0.164	35.5 – 55.4	155 - 254	9.5 - 12.4	76 - 185	101 - 360	101 - 150	Unhealthy for Sensitive Groups
0.086 - 0.105	0.165 - 0.204	(55.5 - 150.4)[3]	255 - 354	12.5 - 15.4	(186 - 304)[4]	361 - 649	151 - 200	Unhealthy
0.106 - 0.200	0.205 - 0.404	(150.5 - (250.4)[3]	355 - 424	15.5 - 30.4	(305 - 604)[4]	650 - 1249	201 - 300	Very unhealthy
([2])	0.405 - 0.504	(250.5 - (350.4)[3]	425 - 504	30.5 - 40.4	(605 - 804)[4]	1250 - 1649	301 - 400	Hazardous
([2])	0.505 - 0.604	(350.5 - 500.4)[3]	505 - 604	40.5 - 50.4	(805 - 1004)[4]	1650 - 2049	401 - 500	Hazardous

[1] Areas are generally required to report the AQI based on 8-hour ozone values. However, there are a small number of areas where an AQI based on 1-hour ozone values would be more precautionary. In these cases, in addition to calculating the 8-hour ozone index value, the 1-hour ozone value may be calculated, and the maximum of the two values reported.

[2] 8-hour O_3 values do not define higher AQI values (≥ 301). AQI values of 301 or higher are calculated with 1-hour O_3 concentrations.

[3] If a different SHL for PM2.5 is promulgated, these numbers will change accordingly.

[4] 1-hour SO_2 values do not define higher AQI values (≥ 200). AQI values of 200 or greater are calculated with 24-hour SO_2 concentrations.

``

Appendix C

Feature Selection and Dimensionality Reduction Methods

- **Grey correlation analysis:**

GCA is a powerful benchmarking tool for determining the correlation grades between predictors and prediction targets. The higher the correlation grade of a feature, the higher its influence on the dependent target. This method was used in case study 1 in chapter 4 to select features with high correlations with the target wind power and filter out features with lower influences.

The GCA grades are calculated as follows: Assuming D is a matrix containing the m samples of n features, the number of rows is the number of time samples, and the number of columns represents the number of features (predictors).

$$D = \begin{pmatrix} \lambda_1(1) & \cdots \lambda_z(1) \cdots & \lambda_n(1) \\ \vdots & \vdots & \vdots \\ \lambda_1(m) & \cdots \lambda_z(m) \cdots & \lambda_n(m) \end{pmatrix}$$

To adopt the differences between the magnitudes of the different features, first of all, all the data are normalized as follows :

$$\lambda_z^*(t) = \frac{\lambda_z(t) - \min \lambda_z(t)}{\max \lambda_z(t) - \min \lambda_z(t)}$$

Where z and t are the and t are feature index and time index, respectively.

Using these normalized features, the grey coefficients are calculated using the four equations below :

$$\Omega(\lambda_O^*(t), \lambda_z^*(t)) = \frac{\Delta_{\min} + \xi \Delta_{\max}}{\Delta_{oz}(t) + \xi \Delta_{\max}} \quad \xi \in (0, 1)$$

$$\Delta_{oz}(t) = |\lambda_O^*(t) - \lambda_z^*(t)|$$

$$\Delta_{\max} = \max |\lambda_O^*(t) - \lambda_z^*(t)| \quad z = 1, ...,n$$

$$\Delta_{\min} = \min |\lambda_O^*(t) - \lambda_z^*(t)| \quad z = 1, ...,n$$

Where ξ is a distinguishing factor and has a value of [0.5,1], a value of 0.6 was used in our study. $\lambda_O^*(t)$ is the target feature at time =t. Finally, the grey correlation grade for different predictors is calculated by

$$\Gamma_Z(\lambda_O^*(t), \lambda_z^*(t)) = \frac{\sum_{t=1}^{m} \Omega(\lambda_O^*(t), \lambda_z^*(t))}{m}$$

``

- **Autocorrelation and partial autocorrelation functions:**

The autocorrelation function (ACF) and partial autocorrelation functions (PACF) are well-known functions to investigate how a variable series is correlated with itself at different time lags. The ACF and PACF function were used in the two proposed case studies to select the appreciate time-lag features.

ACF estimated the correlation of a variable between two lags ρ_h defined as:

$$\rho_h = Corr(x_{1t}, x_{1(t-h)}) = \frac{\gamma_h}{\gamma_0}$$

where $x_{1t}$is the target variable at time t, $x_{1(t-h)}$ is target at time t-1, $\gamma_h$is the covariance of the variable at lag h, and γ_o is the current covariance of the variable.

On the other hand, PACF represents the correlation between variable at lad h and lag t-h, after removing all the dependence on other variables between the two lags, which is defined as:

$$\phi_h = Corr\{[x_t - P(x_t|x_{t-h+1}, \ldots, x_{t-1})], [x_{t-h} - P(x_t|x_{t-h+1}, \ldots, x_{t-1})]\}$$

where P(A|B) is the correlation between A and B

- **Principal component analysis:**

The high dimensional features affect the prediction accuracy and require high computation costs. The principal component analysis is employed for dimensionality reduction and redundant and related or necessary information removal. Thus, avoiding by PCA the risk of overfitting and maintaining reliable prediction models. PCA transforms the data linearly by creating the linear combinations between them. By employing PCA, X matrix, the observations matrix is into a covariance matrix $\sum$, then the contribution rate (CR) and the cumulative contribution (CC) of i th principal component are, respectively, computed as follows:

$$CR_i = \frac{\lambda_i}{\sum_{j=1}^{p} \lambda_j}$$

$$CC_i = \frac{\sum_{j=1}^{i} \lambda_i}{\sum_{j=1}^{p} \lambda_j}$$

where w λ_i Is the eigenvalue corresponding to the ith principal component, and p is the number of parameters.

``

Appendix D

Hyperparameter Tuning Methods

- **Grid search**

. This method tries every possible combination of each set of hyper-parameters. Using this method, we can find the best set of values in the parameter search space. This usually uses more computational power and takes a long time to run since this method needs to try every combination in the grid size.

- **Random search**

The random search method randomly chooses the hyperparameter sample combinations from grid space instead of trying every possible combination. Thus, there is no assurance that the best parameter will be found. Nevertheless, this search can be highly effective in practice as computational time is significantly less.

- **Bayesian optimization for hyperparameter tuning**

similar to grid search, a parameter space with the range of input values is created for evaluation as a first step. However, in contrast to random or grid search, Bayesian approaches keep track of past evaluation results, which they use to form a probabilistic model mapping hyperparameters to a probability of a score on the objective function. Bayesian tuning aims to become "less wrong" with more data which these approaches do by continually updating the surrogate probability model after each evaluation of the objective function.

The steps of Bayesian optimization are summarized into four steps :

1- Create a surrogate probability model of the objective function.

2- Find the hyperparameters that perform best on the surrogate model.

3- Use these values on the true model to return the objective function and update the surrogate model.

4- Repeat steps 2 and 3 until maximum evaluations are reached.

``

References

[1] "Global energy transformation: A roadmap to 2050 (2019 edition)." https://www.irena.org/publications/2019/Apr/Global-energy-transformation-A-roadmap-to-2050-2019Edition (accessed Aug. 17, 2020).

[2] A. Zerrahn, W. P. Schill, and C. Kemfert, "On the economics of electrical storage for variable renewable energy sources," *European Economic Review*, vol. 108, pp. 259–279, 2018, doi: 10.1016/j.euroecorev.2018.07.004.

[3] D. Anderson and M. Leach, "Harvesting and redistributing renewable energy : on the role of gas and electricity grids to overcome intermittency through the generation and storage of hydrogen," vol. 32, pp. 1603–1614, 2004, doi: 10.1016/S0301-4215(03)00131-9.

[4] J. V. Lara-Fanego, J.A. Ruiz-Arias, D. Pozo-Va´zquez ⇑, F.J. Santos-Alamillos, "Evaluation of the WRF model solar irradiance forecasts in Andalusia," vol. 86, pp. 2200–2217, 2012, doi: 10.1016/j.solener.2011.02.014.

[5] P. Applications, S. S. Al-zakwani, A. Maroufmashat, A. Mazouz, M. Fowler, and A. Elkamel, "energies Allocation of Ontario ' s Surplus Electricity to Di ff erent," no. 2017, 2019.

[6] S. Chakraborty, T. Senjyu, A. Y. Saber, A. Yona, and T. Funabashi, "A fuzzy binary clustered particle swarm optimization strategy for thermal unit commitment problem with wind power integration," *IEEJ Transactions on Electrical and Electronic Engineering*, vol. 7, no. 5, pp. 478–486, 2012, doi: 10.1002/tee.21761.

[7] M. Santhosh and C. Venkaiah, "Sustainable Energy , Grids and Networks Short-term wind speed forecasting approach using Ensemble Empirical Mode Decomposition and Deep Boltzmann Machine," *Sustainable Energy, Grids and Networks*, vol. 19, p. 100242, 2019, doi: 10.1016/j.segan.2019.100242.

[8] T. S. Nielsen, A. Joensen, H. Madsen, L. Landberg, and G. Giebel, "A new reference for wind power forecasting," *Wind Energy*, vol. 1, no. 1, pp. 29–34, Sep. 1998, doi: 10.1002/(SICI)1099-1824(199809)1:1<29::AID-WE10>3.0.CO;2-B.

[9] M. Lei, L. Shiyan, J. Chuanwen, L. Hongling, and Z. Yan, "A review on the forecasting of wind speed and generated power," *Renewable and Sustainable Energy Reviews*, vol. 13, no. 4, pp.

915–920, 2009, doi: 10.1016/j.rser.2008.02.002.

[10] G. Giebel, R. Brownsword, G. Kariniotakis, and C. Denhard, MichaelDraxl, *The State-Of-The-Art in Short-Term Prediction of Wind Power*. 2011.

[11] A. Murata, H. Ohtake, and T. Oozeki, "Modeling of uncertainty of solar irradiance forecasts on numerical weather predictions with the estimation of multiple confidence intervals," *Renewable Energy*, vol. 117, pp. 193–201, 2018, doi: 10.1016/j.renene.2017.10.043.

[12] R. Giebel, Gregor Kariniotakis, Georges Brownsword, "The state-of-the-art in short term prediction of wind power from a danish perspective To cite this version : HAL Id : hal-00529986 The State-of-the-Art in Short-Term Prediction of Wind Power From a Danish Perspective," 2018, [Online]. Available: https://hal-mines-paristech.archives-ouvertes.fr/hal-00529986.

[13] A. Ahmed and M. Khalid, "A review on the selected applications of forecasting models in renewable power systems," *Renewable and Sustainable Energy Reviews*, vol. 100, no. September 2018, pp. 9–21, 2019, doi: 10.1016/j.rser.2018.09.046.

[14] O. Karakuş, E. E. Kuruoğlu, and M. A. Altınkaya, "One-day ahead wind speed/power prediction based on polynomial autoregressive model," *IET Renewable Power Generation*, vol. 11, no. 11, pp. 1430–1439, Sep. 2017, doi: 10.1049/iet-rpg.2016.0972.

[15] Y. Jiang, G. Huang, X. Peng, Y. Li, and Q. Yang, "Journal of Wind Engineering & Industrial Aerodynamics A novel wind speed prediction method : Hybrid of correlation-aided DWT , LSSVM and GARCH," *Journal of Wind Engineering & Industrial Aerodynamics*, vol. 174, no. December 2017, pp. 28–38, 2018, doi: 10.1016/j.jweia.2017.12.019.

[16] A. A. Ezzat, M. Jun, Y. Ding, and S. Member, "Spatio-temporal asymmetry of local wind fields and its impact on short-term wind forecasting .," *Transactions on Sustainable Energy*, vol. X, no. X, pp. 1–11, 2018, doi: 10.1109/TSTE.2018.2789685.

[17] M. Ghofrani and M. Alolayan, "Time Series and Renewable Energy Forecasting," *Time Series Analysis and Applications*, WA, USA: InTech, 2018, pp. 78–92.

[18] E. Erdem and J. Shi, "ARMA based approaches for forecasting the tuple of wind speed and direction," *Applied Energy*, vol. 88, no. 4, pp. 1405–1414, 2011, doi: 10.1016/j.apenergy.2010.10.031.

``

[19] P. Gomes and R. Castro, “Wind Speed and Wind Power Forecasting using Statistical Models: AutoRegressive Moving Average (ARMA) and Artificial Neural Networks (ANN),” *International Journal of Sustainable Energy Development*, vol. 1, no. 2, pp. 41–50, 2012, doi: 10.20533/ijsed.2046.3707.2012.0007.

[20] A. Fentis, L. Bahatti, M. Tabaa, and M. Mestari, “Short-term nonlinear autoregressive photovoltaic power forecasting using statistical learning approaches and in-situ observations,” *International Journal of Energy and Environmental Engineering*, vol. 10, no. 2, pp. 189–206, 2019, doi: 10.1007/s40095-018-0293-5.

[21] P. Bacher, H. Madsen, and H. A. Nielsen, “Online short-term solar power forecasting,” *Solar Energy*, vol. 83, no. 10, pp. 1772–1783, 2009, doi: 10.1016/j.solener.2009.05.016.

[22] S. Atique, S. Noureen, V. Roy, V. Subburaj, S. Bayne, and J. Macfie, “Forecasting of total daily solar energy generation using ARIMA: A case study,” in *2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC)*, Jan. 2019, no. February, pp. 0114–0119, doi: 10.1109/CCWC.2019.8666481.

[23] S. Pasari and A. Shah, “Time Series Auto-Regressive Integrated Moving Average Model for Renewable Energy Forecasting,” in *Sustainable Production, Life Cycle Engineering and Management*, Pilani, India: Springer International Publishing, 2020, pp. 71–77.

[24] R. G. Kavasseri and K. Seetharaman, “Day-ahead wind speed forecasting using f -ARIMA models,” *Renewable Energy*, vol. 34, no. 5, pp. 1388–1393, 2009, doi: 10.1016/j.renene.2008.09.006.

[25] M. Abuella and B. Chowdhury, “Solar power probabilistic forecasting by using multiple linear regression analysis,” in *Conference Proceedings - IEEE SOUTHEASTCON*, 2015, vol. 2015-June, no. June, pp. 5–9, doi: 10.1109/SECON.2015.7132869.

[26] P. Lauret, M. David, and H. T. C. Pedro, “Probabilistic solar forecasting using quantile regression models,” *Energies*, vol. 10, no. 10, pp. 1–17, 2017, doi: 10.3390/en10101591.

[27] L. Massidda and M. Marrocu, “Use of Multilinear Adaptive Regression Splines and numerical weather prediction to forecast the power output of a PV plant in Borkum, Germany,” *Solar Energy*, vol. 146, pp. 141–149, 2017, doi: 10.1016/j.solener.2017.02.007.

[28] G. Wang, Y. Su, and L. Shu, “One-day-ahead daily power forecasting of photovoltaic systems

``

based on partial functional linear regression models," *Renewable Energy*, vol. 96, pp. 469–478, 2016, doi: 10.1016/j.renene.2016.04.089.

[29] H. Cai, X. Jia, J. Feng, W. Li, Y. M. Hsu, and J. Lee, "Gaussian Process Regression for numerical wind speed prediction enhancement," *Renewable Energy*, vol. 146, pp. 2112–2123, 2020, doi: 10.1016/j.renene.2019.08.018.

[30] B. Keshtegar, C. Mert, and O. Kisi, "Comparison of four heuristic regression techniques in solar radiation modeling: Kriging method vs RSM, MARS and M5 model tree," *Renewable and Sustainable Energy Reviews*, vol. 81, no. February 2017, pp. 330–341, 2018, doi: 10.1016/j.rser.2017.07.054.

[31] M. N. Akhter, S. Mekhilef, H. Mokhlis, and N. M. Shah, "Review on forecasting of photovoltaic power generation based on machine learning and metaheuristic techniques," *IET Renewable Power Generation*, vol. 13, no. 7, pp. 1009–1023, 2019, doi: 10.1049/iet-rpg.2018.5649.

[32] A. Mellit and S. A. Kalogirou, "Artificial intelligence techniques for photovoltaic applications: A review," *Progress in Energy and Combustion Science*, vol. 34, no. 5, pp. 574–632, 2008, doi: 10.1016/j.pecs.2008.01.001.

[33] A. Mellit, S. A. Kalogirou, L. Hontoria, and S. Shaari, "Artificial intelligence techniques for sizing photovoltaic systems: A review," *Renewable and Sustainable Energy Reviews*, vol. 13, no. 2, pp. 406–419, 2009, doi: 10.1016/j.rser.2008.01.006.

[34] P. M. V, Sindhu, Nivedha S, "an Empirical Science Research on Bioinformatics in Machine Learning," *Journal of Mechanics of Continua and Mathematical Sciences*, vol. spl7, no. 1, pp. 86–94, 2020, doi: 10.26782/jmcms.spl.7/2020.02.00006.

[35] J. Maldonado-Correa, J. Solano, and M. Rojas-Moncayo, "Wind power forecasting: A systematic literature review," *Wind Engineering*, vol. 2019, pp. 1–14, Dec. 2019, doi: 10.1177/0309524X19891672.

[36] P. Du, J. Wang, W. Yang, and T. Niu, "A novel hybrid model for short-term wind power forecasting," *Applied Soft Computing Journal*, vol. 80, pp. 93–106, 2019, doi: 10.1016/j.asoc.2019.03.035.

[37] J. Nielson, K. Bhaganagar, R. Meka, and A. Alaeddini, "Using atmospheric inputs for Arti ficial Neural Networks to improve wind turbine power prediction," *Energy*, vol. 190, p. 116273, 2020,

``

doi: 10.1016/j.energy.2019.116273.

[38] G. Li and J. Shi, "On comparing three artificial neural networks for wind speed forecasting," *Applied Energy*, vol. 87, no. 7, pp. 2313–2320, 2010, doi: 10.1016/j.apenergy.2009.12.013.

[39] Y. Hong, C. Lian, and P. P. Rioflorido, "A hybrid deep learning-based neural network for 24-h ahead wind power forecasting," *Applied Energy*, vol. 250, no. January, pp. 530–539, 2019, doi: 10.1016/j.apenergy.2019.05.044.

[40] G. Grassi and P. Vecchio, "Wind energy prediction using a two-hidden layer neural network," *Communications in Nonlinear Science and Numerical Simulation*, vol. 15, no. 9, pp. 2262–2266, 2010, doi: 10.1016/j.cnsns.2009.10.005.

[41] Z. Bashir and M. E. El-Hawary, "Short term load forecasting by using wavelet neural networks," in *2000 Canadian Conference on Electrical and Computer Engineering. Conference Proceedings. Navigating to a New Era (Cat. No.00TH8492)*, 2000, vol. 1, pp. 163–166, doi: 10.1109/CCECE.2000.849691.

[42] H. Jahangir, M. Aliakbar, F. Alhameli, A. Mazouz, A. Ahmadian, and A. Elkamel, "Short-term wind speed forecasting framework based on stacked denoising auto-encoders with rough ANN," *Sustainable Energy Technologies and Assessments*, vol. 38, no. June 2019, p. 100601, 2020, doi: 10.1016/j.seta.2019.100601.

[43] A. P. Marugán, F. P. G. Márquez, J. M. P. Perez, and D. Ruiz-Hernández, "A survey of artificial neural network in wind energy systems," *Applied Energy*, vol. 228, no. July, pp. 1822–1836, 2018, doi: 10.1016/j.apenergy.2018.07.084.

[44] J. Liu, X. Wang, and Y. Lu, "A novel hybrid methodology for short-term wind power forecasting based on adaptive neuro-fuzzy inference system," *Renewable Energy*, vol. 103, pp. 620–629, 2017, doi: 10.1016/j.renene.2016.10.074.

[45] M. Abuella and B. Chowdhury, "Solar power forecasting using artificial neural networks," in *2015 North American Power Symposium (NAPS)*, Oct. 2015, no. November, pp. 1–5, doi: 10.1109/NAPS.2015.7335176.

[46] D. O'Leary and J. Kubby, "Feature Selection and ANN Solar Power Prediction," *Journal of Renewable Energy*, vol. 2017, pp. 1–7, Nov. 2017, doi: 10.1155/2017/2437387.

``

[47] C. G. Ozoegwu, "Artificial neural network forecast of monthly mean daily global solar radiation of selected locations based on time series and month number," *Journal of Cleaner Production*, vol. 216, pp. 1–13, 2019, doi: 10.1016/j.jclepro.2019.01.096.

[48] G. M. Yagli, D. Yang, and D. Srinivasan, "Automatic hourly solar forecasting using machine learning models," *Renewable and Sustainable Energy Reviews*, vol. 105, no. February, pp. 487–498, 2019, doi: 10.1016/j.rser.2019.02.006.

[49] S. Ghimire, R. C. Deo, N. J. Downs, and N. Raj, "Global solar radiation prediction by ANN integrated with European Centre for medium range weather forecast fields in solar rich cities of Queensland Australia," *Journal of Cleaner Production*, vol. 216, pp. 288–310, 2019, doi: 10.1016/j.jclepro.2019.01.158.

[50] H. Wang *et al.*, "Taxonomy research of artificial intelligence for deterministic solar power forecasting," *Energy Conversion and Management*, vol. 214, no. January, p. 112909, 2020, doi: 10.1016/j.enconman.2020.112909.

[51] H. Su, E. Zio, J. Zhang, M. Xu, X. Li, and Z. Zhang, "A hybrid hourly natural gas demand forecasting method based on the integration of wavelet transform and enhanced Deep-RNN model," *Energy*, vol. 178, pp. 585–597, 2019, doi: 10.1016/j.energy.2019.04.167.

[52] M. Bianchini, M. Maggini, and L. C. Jain, "Handbook on Neural Information Processing," in *Intelligent Systems Reference Library*, vol. 49, M. Bianchini, M. Maggini, and L. C. Jain, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 29–65.

[53] A. Kisvari, Z. Lin, and X. Liu, "Wind power forecasting – A data-driven method along with gated recurrent neural network," *Renewable Energy*, vol. 163, pp. 1895–1909, 2021, doi: 10.1016/j.renene.2020.10.119.

[54] Y.-D. Syu *et al.*, "Ultra-Short-Term Wind Speed Forecasting for Wind Power Based on Gated Recurrent Unit," in *2020 8th International Electrical Engineering Congress (iEECON)*, Mar. 2020, pp. 1–4, doi: 10.1109/iEECON48109.2020.229518.

[55] R. Yu *et al.*, "LSTM-EFG for wind power forecasting based on sequential correlation features," *Future Generation Computer Systems*, vol. 93, pp. 33–42, 2019, doi: 10.1016/j.future.2018.09.054.

[56] Z. Niu, Z. Yu, W. Tang, Q. Wu, and M. Reformat, "Wind power forecasting using attention-

based gated recurrent unit network," *Energy*, vol. 196, p. 117081, 2020, doi: 10.1016/j.energy.2020.117081.

[57] A. Yona, T. Senjyu, T. Funabashi, and C. H. Kim, "Determination method of insolation prediction with fuzzy and applying neural network for long-term ahead PV power output correction," *IEEE Transactions on Sustainable Energy*, vol. 4, no. 2, pp. 527–533, 2013, doi: 10.1109/TSTE.2013.2246591.

[58] M. Aslam, J. M. Lee, H. S. Kim, S. J. Lee, and S. Hong, "Deep learning models for long-term solar radiation forecasting considering microgrid installation: A comparative study," *Energies*, vol. 13, no. 1, 2019, doi: 10.3390/en13010147.

[59] M. Rana, R. Chandra, and V. G. Agelidis, "Cooperative neuro-evolutionary recurrent neural networks for solar power prediction," in *2016 IEEE Congress on Evolutionary Computation (CEC)*, Jul. 2016, pp. 4691–4698, doi: 10.1109/CEC.2016.7744389.

[60] M. Hosseini, S. Katragadda, J. Wojtkiewicz, R. Gottumukkala, A. Maida, and T. L. Chambers, "Direct normal irradiance forecasting using multivariate gated recurrent units," *Energies*, vol. 13, no. 15, pp. 1–15, 2020, doi: 10.3390/en13153914.

[61] M. S. Hossain and H. Mahmood, "Short-Term Photovoltaic Power Forecasting Using an LSTM Neural Network and Synthetic Weather Forecast," *IEEE Access*, vol. 8, pp. 172524–172533, 2020, doi: 10.1109/access.2020.3024901.

[62] J. He and J. Xu, "Ultra-short-term wind speed forecasting based on support vector machine with combined kernel function and similar data," *EURASIP Journal on Wireless Communications and Networking*, vol. 2019, no. 1, p. 248, Dec. 2019, doi: 10.1186/s13638-019-1559-1.

[63] X. Huang, A. Maier, J. Hornegger, and J. A. K. Suykens, "Indefinite kernels in least squares support vector machines and principal component analysis," *Applied and Computational Harmonic Analysis*, vol. 43, no. 1, pp. 162–172, 1999, doi: 10.1016/j.acha.2016.09.001.

[64] S. Wang and C. Chen, "Short-Term Wind Power Prediction Based on DBSCAN Clustering and Support Vector Machine Regression," in *2020 5th International Conference on Computer and Communication Systems (ICCCS)*, May 2020, pp. 941–945, doi: 10.1109/ICCCS49078.2020.9118606.

[65] H. Tabari, O. Kisi, A. Ezani, and P. Hosseinzadeh Talaee, "SVM, ANFIS, regression and

``

climate based models for reference evapotranspiration modeling using limited climatic data in a semi-arid highland environment," *Journal of Hydrology*, vol. 444–445, pp. 78–89, 2012, doi: 10.1016/j.jhydrol.2012.04.007.

[66] V. H. Quej, J. Almorox, J. A. Arnaldo, and L. Saito, "ANFIS, SVM and ANN soft-computing techniques to estimate daily global solar radiation in a warm sub-humid environment," *Journal of Atmospheric and Solar-Terrestrial Physics*, vol. 155, no. September 2016, pp. 62–70, 2017, doi: 10.1016/j.jastp.2017.02.002.

[67] A. Ahmad, Y. Jin, C. Zhu, I. Javed, and M. W. Akram, "Support vector machine based prediction of photovoltaic module and power station parameters," *International Journal of Green Energy*, vol. 00, no. 00, pp. 1–14, 2020, doi: 10.1080/15435075.2020.1722131.

[68] F. Hamamy and A. M. Omar, "Least square support vector machine technique for short term solar irradiance forecasting," *AIP Conference Proceedings*, vol. 2129, no. July, 2019, doi: 10.1063/1.5118141.

[69] M. Malvoni and N. Hatziargyriou, "One-day ahead PV power forecasts using 3D Wavelet Decomposition," in *2019 International Conference on Smart Energy Systems and Technologies (SEST)*, Sep. 2019, pp. 1–6, doi: 10.1109/SEST.2019.8849007.

[70] N. Li, F. He, and W. Ma, "Wind power prediction based on extreme learning machine with kernel mean p-power error loss," *Energies*, vol. 12, no. 4, pp. 1–19, 2019, doi: 10.3390/en12040673.

[71] M. Hossain, S. Mekhilef, M. Danesh, L. Olatomiwa, and S. Shamshirband, "Application of extreme learning machine for short term output power forecasting of three grid-connected PV systems," *Journal of Cleaner Production*, vol. 167, pp. 395–405, 2017, doi: 10.1016/j.jclepro.2017.08.081.

[72] L. Yang and A. Shami, "On hyperparameter optimization of machine learning algorithms: Theory and practice," *Neurocomputing*, vol. 415, pp. 295–316, 2020, doi: 10.1016/j.neucom.2020.07.061.

[73] F. Hutter, J. Lücke, and L. Schmidt-Thieme, "Beyond Manual Tuning of Hyperparameters," *KI - Kunstliche Intelligenz*, vol. 29, no. 4, pp. 329–337, 2015, doi: 10.1007/s13218-015-0381-0.

[74] R. Jursa and K. Rohrig, "Short-term wind power forecasting using evolutionary algorithms for

the automated specification of artificial intelligence models," *International Journal of Forecasting*, vol. 24, no. 4, pp. 694–709, 2008, doi: 10.1016/j.ijforecast.2008.08.007.

[75] A. Sözen, E. Arcaklioğlu, M. Özalp, and E. G. Kanit, "Use of artificial neural networks for mapping of solar potential in Turkey," *Applied Energy*, vol. 77, no. 3, pp. 273–286, 2004, doi: 10.1016/S0306-2619(03)00137-5.

[76] M. Ding, L. Wang, and R. Bi, "An ANN-based approach for forecasting the power output of photovoltaic system," *Procedia Environmental Sciences*, vol. 11, no. PART C, pp. 1308–1315, 2011, doi: 10.1016/j.proenv.2011.12.196.

[77] T. Vinothkumar and K. Deeba, "Hybrid wind speed prediction model based on recurrent long short-term memory neural network and support vector machine models," *Soft Computing*, vol. 24, no. 7, pp. 5345–5355, 2020, doi: 10.1007/s00500-019-04292-w.

[78] C. Voyant *et al.*, "Machine learning methods for solar radiation forecasting : A review," *Renewable Energy*, vol. 105, pp. 569–582, 2017, doi: 10.1016/j.renene.2016.12.095.

[79] Z. Liu, L. Li, M. Tseng, and M. K. Lim, "Prediction short-term photovoltaic power using improved chicken swarm optimizer - Extreme learning machine model," *Journal of Cleaner Production*, no. xxxx, p. 119272, 2019, doi: 10.1016/j.jclepro.2019.119272.

[80] I. Manisalidis, E. Stavropoulou, A. Stavropoulos, and E. Bezirtzoglou, "Environmental and Health Impacts of Air Pollution: A Review," *Frontiers in Public Health*, vol. 8. 2020, doi: 10.3389/fpubh.2020.00014.

[81] V. Ramanathan, "Climate Change, Air Pollution, and Health: Common Sources, Similar Impacts, and Common Solutions," in *Health of People, Health of Planet and Our Responsibility*, Cham: Springer International Publishing, 2020, pp. 49–59.

[82] D. W. Connell, *Basic Concepts of Environmental Chemistry*. 2005.

[83] USEPA, "Technical Assistance Document for the Reporting of Daily Air Quality – the Air Quality Index (AQI)," *Environmental Protection*, no. May, pp. 1–28, 2013.

[84] S. Y. Lim, L. Y. Chin, P. Mah, and J. Wee, "Arima and Integrated Arfima Models for Forecasting Air Pollution Index in Shah Alam , Selangor," *The Malaysian Journal of Analytical Science*, vol. 12, no. 1, pp. 257–263, 2008.

``

[85] J. Zhu, “Comparison of ARIMA Model and Exponential Smoothing Model on 2014 Air Quality Index in Yanqing County, Beijing, China,” *Applied and Computational Mathematics*, vol. 4, no. 6, p. 456, 2015, doi: 10.11648/j.acm.20150406.19.

[86] S. Karthikeyani and S. Rathi, “A Survey On Air Quality Prediction Using Traditional Statistics Method,” *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, vol. 6, no. 3, pp. 942–946, 2020, doi: 10.32628/cseit2063197.

[87] P. G. Zhang, “Time series forecasting using a hybrid ARIMA and neural network model,” *Neurocomputing*, vol. 50, pp. 159–175, 2003, doi: 10.1016/S0925-2312(01)00702-0.

[88] C. Y. Wang, W. Y. Zhang, J. J. Wang, and W. F. Zhao, “The prediction of SO 2 pollutant concentration using a RBF neural network,” *Applied Mechanics and Materials*, vol. 55–57, pp. 1392–1396, 2011, doi: 10.4028/www.scientific.net/AMM.55-57.1392.

[89] M. Cai, Y. Yin, and M. Xie, “Prediction of hourly air pollutant concentrations near urban arterials using artificial neural network approach,” *Transportation Research Part D: Transport and Environment*, vol. 14, no. 1, pp. 32–41, Jan. 2009, doi: 10.1016/j.trd.2008.10.004.

[90] M. Castelli, F. M. Clemente, A. Popovič, S. Silva, and L. Vanneschi, “A Machine Learning Approach to Predict Air Quality in California,” *Complexity*, vol. 2020, no. Ml, 2020, doi: 10.1155/2020/8049504.

[91] S. Sankar Ganesh, P. Arulmozhivarman, and R. Tatavarti, “Forecasting Air Quality Index Using an Ensemble of Artificial Neural Networks and Regression Models,” *Journal of Intelligent Systems*, vol. 28, no. 5, pp. 893–903, Sep. 2017, doi: 10.1515/jisys-2017-0277.

[92] A. Tealab, “Time series forecasting using artificial neural networks methodologies: A systematic review,” *Future Computing and Informatics Journal*, vol. 3, no. 2, pp. 334–340, 2018, doi: 10.1016/j.fcij.2018.10.003.

[93] B. Wu, “An introduction to neural networks and their applications in manufacturing,” *Journal of Intelligent Manufacturing*, vol. 3, no. 6, pp. 391–403, 1992, doi: 10.1007/BF01473534.

[94] M. Sarigül and M. Avci, “Performance comparison of different momentum techniques on deep reinforcement learning,” *Journal of Information and Telecommunication*, vol. 1839, pp. 1–12, 2018, doi: 10.1080/24751839.2018.1440453.

``

[95] A. I. Georgevici and M. Terblanche, “Neural networks and deep learning: a brief introduction,” *Intensive Care Medicine*, vol. 45, no. 5, pp. 712–714, May 2019, doi: 10.1007/s00134-019-05537-w.

[96] D. Roy, K. S. R. Murty, and C. K. Mohan, “Feature selection using Deep Neural Networks,” in *2015 International Joint Conference on Neural Networks (IJCNN)*, Jul. 2015, vol. 2015-Septe, pp. 1–6, doi: 10.1109/IJCNN.2015.7280626.

[97] Y. Jiang *et al.*, “Expert Feature-Engineering vs. Deep Neural Networks: Which Is Better for Sensor-Free Affect Detection?,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 10947 LNAI, Springer International Publishing, 2018, pp. 198–211.

[98] L. Breiman, “Using iterated bagging to debias regressions,” *Machine Learning*, vol. 45, no. 3, pp. 261–277, 2001, doi: 10.1023/A:1017934522171.

[99] C. Kingsford and S. L. Salzberg, “What are decision trees?,” *Nature Biotechnology*, vol. 26, no. 9, pp. 1011–1012, 2008, doi: 10.1038/nbt0908-1011.

[100] K. Kim, “Financial time series forecasting using support vector machines,” *Neurocomputing*, vol. 55, no. 1–2, pp. 307–319, Sep. 2003, doi: 10.1016/S0925-2312(03)00372-2.

[101] D. Jap, M. Stöttinger, and S. Bhasin, “Support vector regression,” no. November 2007, pp. 1–8, 2015, doi: 10.1145/2768566.2768568.

[102] E. Yaman and A. Subasi, “Comparison of Bagging and Boosting Ensemble Machine Learning Methods for Automated EMG Signal Classification,” *BioMed Research International*, vol. 2019, 2019, doi: 10.1155/2019/9152506.

[103] L. Rokach, “Chapter 45 ENSEMBLE METHODS FOR CLASSIFIERS,” pp. 1–24, 2010, [Online]. Available: http://www.ise.bgu.ac.il/faculty/liorr/hbchap45.pdf.

[104] “Onario, Canadian Wind Energy Association. 2019.” https://canwea.ca/wind-energy/ontario-market-profile/#:~:text=Ontario remains Canada’s leader in,360 MW of generation capacity.

[105] “Henvey Inlet Wind Power Project, Ontario.” https://www.power-technology.com/projects/henvey-inlet-wind-power-project-ontario/.

[106] “South Kent Wind Farm,” [Online]. Available: https://www.renewable-

``

technology.com/projects/south-kent-wind-farm/.

[107] "K2 Wind." https://patternenergy.com/learn/portfolio/k2-wind.

[108] "Ontario's Niagara Region Wind Farm inaugurated as part of Global Wind Day." https://www.windpowerengineering.com/ontarios-niagara-region-wind-farm-inaugurated-part-global-wind-day/.

[109] "Melancthon." https://www.transalta.com/plants-operation/melancthon/.

[110] "Wolfe Island." https://www.transalta.com/plants-operation/wolfe-island/.

[111] J. Lever, M. Krzywinski, and N. Altman, "Points of Significance: Model selection and overfitting," *Nature Methods*, vol. 13, no. 9, pp. 703–704, 2016, doi: 10.1038/nmeth.3968.

[112] P. Yin, P. Luo, and T. Nakamura, "Small batch or large batch? Gaussian walk with rebound can teach," *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, vol. Part F1296, pp. 1275–1284, 2017, doi: 10.1145/3097983.3098147.

[113] S. Ben Taieb, A. Sorjamaa, and G. Bontempi, "Multiple-output modeling for multi-step-ahead time series forecasting," *Neurocomputing*, vol. 73, no. 10–12, pp. 1950–1957, 2010, doi: 10.1016/j.neucom.2009.11.030.

[114] J. Wang, Y. Song, F. Liu, and R. Hou, "Analysis and application of forecasting models in wind power integration: A review of multi-step-ahead wind speed forecasting models," *Renewable and Sustainable Energy Reviews*, vol. 60, pp. 960–981, 2016, doi: 10.1016/j.rser.2016.01.114.

[115] S. Ben Taieb and R. J. Hyndman, "Recursive and direct multi-step forecasting: the best of both worlds," *International Journal of Forecasting*, no. September, 2014.

[116] C. Fan, J. Wang, W. Gang, and S. Li, "Assessment of deep recurrent neural network-based strategies for short-term building energy predictions," *Applied Energy*, vol. 236, no. November 2018, pp. 700–710, 2019, doi: 10.1016/j.apenergy.2018.12.004.

[117] J. J. Montaño Moreno, A. Palmer Pol, A. Sesé Abad, and B. Cajal Blasco, "El índice R-MAPE como medida resistente del ajuste en la previsioṅn," *Psicothema*, vol. 25, no. 4, pp. 500–506, 2013, doi: 10.7334/psicothema2013.23.

[118] A. Liaw and M. Wiener, "Classification and Regression by randomForest," *R News*, vol. 2, no. 3, pp. 18–22, 2002.

``

[119] S. Sun, Z. Cao, H. Zhu, and J. Zhao, “A survey of optimization methods from a machine learning perspective,” *arXiv*, no. March, 2019.

[120] S. Hong and H. S. Lynn, “Accuracy of random-forest-based imputation of missing data in the presence of non-normality, non-linearity, and interaction,” *BMC Medical Research Methodology*, vol. 20, no. 1, pp. 1–12, 2020, doi: 10.1186/s12874-020-01080-1.

[121] M. Arhami, N. Kamali, and M. M. Rajabi, “Predicting hourly air pollutant levels using artificial neural networks coupled with uncertainty analysis by Monte Carlo simulations,” *Environmental Science and Pollution Research*, vol. 20, no. 7, pp. 4777–4789, Jul. 2013, doi: 10.1007/s11356-012-1451-6.

[122] N. M. Nawi, W. H. Atomi, and M. Z. Rehman, “The Effect of Data Pre-processing on Optimized Training of Artificial Neural Networks,” *Procedia Technology*, vol. 11, no. Iceei, pp. 32–39, 2013, doi: 10.1016/j.protcy.2013.12.159.

[123] T. R. Brick, R. E. Koffer, D. Gerstorf, and N. Ram, “Feature Selection Methods for Optimal Design of Studies for Developmental Inquiry,” *The Journals of Gerontology: Series B*, vol. 73, no. 1, pp. 113–123, Jan. 2018, doi: 10.1093/geronb/gbx008.

[124] F. Degenhardt, S. Seifert, and S. Szymczak, “Evaluation of variable selection methods for random forests and omics data sets,” *Briefings in Bioinformatics*, vol. 20, no. 2, pp. 492–503, 2019, doi: 10.1093/bib/bbx124.

[125] K. Gnana Sheela and S. N. Deepa, “An intelligent computing model for wind speed prediction in renewable energy systems,” *Procedia Engineering*, vol. 30, no. 2011, pp. 380–385, 2012, doi: 10.1016/j.proeng.2012.01.875.