Estimation of Wake Effect in Wind Farms, using Machine Learning Algorithms

Aamer Bilal Asghar[1,*], Khazina Naveed[2], Ayman A. Aly[3], Basem Alamri[4] and Róbert Štefko[5,*]

[1] Department of Electrical Engineering, COMSATS University Islamabad, Lahore 54000, Pakistan, e-mail: aamerbilal@cuilahore.edu.pk

[2] Department of Computer Science, Bahria University, Lahore 54000, Pakistan, e-mail: khazina.bulc@bahria.edu.pk

[3] Department of Mechanical Engineering, College of Engineering, Taif University, Taif 21944, Saudi Arabia, e-mail: aymanaly@tu.edu.sa

[4] Department of Electrical Engineering, College of Engineering, Taif University, P.O. Box 11099, Taif 21944, Saudi Arabia, e-mail: b.alamri@tu.edu.sa

[5] Faculty of Electrical Engineering and Informatics, Department of Electric Power Engineering, Technical University of Kosice, Kosice, Slovakia, e-mail: robert.stefko@tuke.sk

[*]Corresponding author

Abstract: Wind energy is one of the leading renewable sources. Multiple wind turbines are installed at a given site to produce more electrical energy, making wind farm efficiency optimization a vital area of study. The flow field in the wake of the first row of turbines is characterized by wind velocity deficit and high turbulent intensity. For this reason, a downstream turbines in a wind farm can capture less wind energy than the first-row turbine. The present research specifically focuses on intelligently estimating the wake speed in wind farm using well-known five machine learning algorithms. The data used is computationally synthesized from the Jensen wake model. Machine learning models such as Artificial Neural Networks (ANN), Random Forest Regression (RFR), Decision Tree Regression (DTR), Support Vector Machines (SVM), and the Adaptive Neuro-Fuzzy Inference System (ANFIS) are implemented to adopt the complex nonlinear relationship for accurately estimating the wake speed. Among the tested models, the Random Forest Model performed the best with an R2 score of 0.9905, Mean Square Error (MSE) of 1.26E-06, and Root Mean Square Error (RMSE) of 0.0011, demonstrating its effectiveness in accurately estimating wake effects. In comparison, the Decision Tree Regression also showed promising results with an R2 score of 0.9646, although it exhibited a slightly higher MSE of 4.75E-06, and RMSE of 0.0021. The outcomes of this research have the potential to revolutionize wind farm optimization by providing more adaptive and faster wake estimation.

Keywords: Wind Farm; wake effect; ANN; SVM; ANFIS; Decision tree; Random Forest

1 Introduction

Wind energy is captured using wind turbines to generate electricity. A wind farm consists of many wind turbines connected in parallel. Each wind rotor produces a wake, which is a turbulent region in the air. An optimal localization of wind turbines in wind farm, is essential, since it can lead to a significant increase in a fields power output. Dense configurations may seem like a good idea, but the wake effect, a known consequence of the close spacing between the turbines, is a problem. Each wind turbine generates a cone of slower, more turbulent air behind it, which causes downstream wind turbines to experience a drop in wind speed, which lowers energy production [1]. The abrupt fall in velocity caused by this wake results in a decrease in the amount of air and wind speed entering the downstream turbine, which reduces the amount of energy the downstream turbine can produce. The air exiting the wind turbine rotor initially has a diameter that nearly matches the rotor's diameter before spreading out conically. Various factors, including wind speed, turbine design, and rotor radius, influence the characteristics and development of turbine wakes [2]. An essential part of estimating energy production for large wind farm planning is modelling the wake effects [3]. Reducing power losses and prolonging blade life in a wind farm requires a complete understanding of how wind turbine waves behave. Numerical simulation of the wake effects in wind farms can provide such knowledge. The wake plays a vital role in wind energy capturing capacity. It refers to the trail of disturbed wind flow left behind a turbine, as illustrated in Figure 1. The wake, which is an invisible region behind the wind turbine where the wind is weaker and slower, is created when the blades rotate. A wind turbine's performance is significantly impacted by wake; if upstream wind turbine generates a huge wake, the subsequent downstream turbine will receive less wind and generate less energy, which will impact all the wind turbines in the row ahead, resulting in all the subsequent turbines having less energy as well [4].

The Jensen's wake model views momentum as being conserved inside the wake. Therefore, it is crucial to estimate the wake effect before installing the wind turbines, as well as how and when to install them to minimize the wake and maximize energy production [5].

According to Jensen's model, as illustrated in Figure 2, the wake grows linearly with downstream distance. The mean wind speed (u_0), also referred as free stream wind speed is taken as 12 m/s; u_1 is the decreased wake wind speed; R_1 is the downstream wake radius in meters (m). The R_1 and X (downstream distance of the turbine in meters) have a linear relationship as the wake spreads downstream. The R_r is the rotor radius of upstream turbine measured in meters (m). A complicated but essential phenomenon in wind energy production is multiple partial interferences of wake effects in wind turbine arrays. This phenomenon is known as wake interference, which occurs when wind passes over a turbine array and interacts with the wake produced by each turbine.

Nevertheless, when turbines are placed in irregular configurations or many rows, the wake impact becomes more intricate. To maximize wind farm performance and efficiency, researchers work to understand and minimize the effects of multiple partial interference, which occurs when the wake of one turbine partially interferes with several neighboring turbines, causing variations in wind speed and turbulence levels across the array.

Figure 1

Wake effect at the Horns Rev offshore wind farm in Denmark [6]

Through advanced modelling techniques, researchers attempt to unravel the complex dynamics of multiple partial interferences, offering insights that can inform improved turbine placement strategies and operational practices. In the end, resolving the challenges posed by multiple partial interferences holds the key to unlocking the full potential of wind energy as a sustainable and dependable power source.

The wake speed is calculated using equation (1), considering the influence of the two wind turbine rotors [7].

$$u_{i+1} = u_i \cdot \left[1 - \frac{1-\sqrt{1-C_T}}{\left(1+\alpha\frac{X}{R_r}\right)^2}\right] \quad (1)$$

Where, u_0= Mean wind speed or the free stream wind speed and u_0 = 12 m/s,

i = 0, 1, . . . , 20.

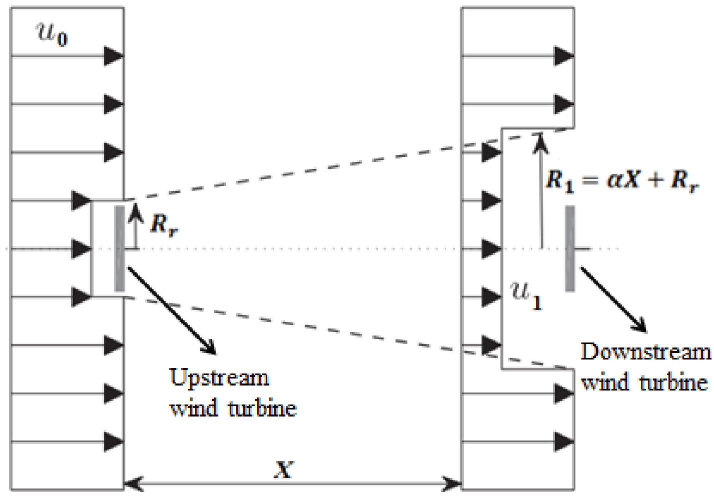

Figure 2
Wake model scheme [7]

The thrust coefficient (C_T) can be calculated by equation (2) [8]:

$$C_T = 4 \cdot a \cdot (1 - a) \tag{2}$$

where, a= Axial induction factor

In the Jensen wake model, the axial induction factor 'a' is usually assigned a constant value of $a = \frac{1}{3}$ to simplify calculations, especially in analytical wind farm design, corresponding to the Betz limit (maximum theoretical efficiency).

The downstream wake radius (R_1) is related to downstream distance of the turbine (X) and rotor radius of upstream turbine (R_r) as represented in equation (3) [8]:

$$R_1 = \alpha X + R_r \tag{3}$$

Where, α is wake decay coefficient and it can be calculated using equation (3), (4), and (5) [8]:

$$R_1 = R_r \cdot \sqrt{\frac{1-a}{1-2a}} \tag{4}$$

$$\alpha = \frac{0.5}{\ln\left(\frac{Z}{Z_o}\right)} \tag{5}$$

Where, Z is the hub height, while the surface roughness is represented by Z_o. Surface roughness has a different value in each field. In terms of flat terrain, the value for Z_o=0.3.

The values for the wake parameters are listed below,

$X = \{100, 200, 300, 400, 500\}$ m

$R_r = \{10, 20, 30, 40\}$ m

u_0 = 12 m/s

a = 0.326795

α = 0. 09437

i = 0, 1, . . . , 20

In this research, machine learning algorithms are used to estimate the wake effect in wind farms. There are two main effects of wake in a wind farm [9]:

(i) A reduction in the wind speed leading to a reduction in the energy produced by the wind farm.

(ii) An increase in wind turbulence leading to an increase in dynamic mechanical loading in downwind turbines.

To optimize wind turbine production and production lifetime, these wakes must be considered while developing wind farms. As a result, numerous models – many of which are intricate and numerical – have been created to take these wakes into account.

2 Literature Review

The wake disturbances caused by upstream turbines to the wind flow reaching downstream turbines emerge as a significant barrier that significantly reduces the effectiveness of wind energy extraction in wind farms [10] [11]. The wake effect reduces the available wind speeds for downstream turbines due to turbulence, which raises mechanical stress and reduces energy output [12]. Historically, wake effect modelling has been based on empirical data and analytical models such as Jensen's wake model. Despite being crucial for understanding wake dynamics, these models usually fall short in capturing the complex, nonlinear interactions that take place in a wind farm environment [13-15]. According to the research, more accurate and flexible modelling approaches are greatly needed to predict wake effects in a variety of operational settings.

Scientists have put a lot of effort into developing wind energy technology to increase efficiency and preserve environmental sustainability. The creation of complex models and the application of state-of-the-art technologies have significantly aided in the precision optimization of wind energy systems [16-18]. These models aim to ensure wind turbine longevity, while optimizing energy extraction. Computational fluid dynamics (CFD) and other advanced simulation methods are commonly their foundation [19]. In recent years, the application of machine learning models and algorithms has emerged as a disruptive force in the study of wind energy. Because of the potential for increased energy output, researchers have been thorough in their investigation of the incorporation of machine learning into various aspects of wind farm operations. Wake effect estimations and other applications rely on machine learning models because of their critical ability to identify complex patterns and adapt to changing conditions. This review of the literature looks closely at the many approaches and advancements in wake effects estimation in wind farms.

To estimate wake velocity and to minimize loss, researchers have used machine learning algorithms, including Support Vector Regression (SVR), Artificial Neural Networks (ANN), Extreme Gradient Boosting (XGBoost), etc. The long Short-Term Memory (LSTM) model is also used to capture the wake effects in wind farms. Support Vector Regression is applied to tasks involving regression and classification. The SVM model is used to evaluate the wake effects in wind farms because it can capture both linear and non-linear relationships in data. An SVR model's performance is influenced by a few factors, including function type, gamma, epsilon, and C. The most helpful parameter, function type, is selected based on the characteristics of the incoming data. Gamma and C parameters are used to protect the model from over-fitting and under-fitting problems [20] [21].

The wake effect of turbines in wind farms is calculated employing numerous numerical models to enhance energy production. There are several issues and challenges with the numerical models. Among its weaknesses and limitations are meticulous turbulence modelling, the number of grid points along the length of an object, computational density, and model verification [22] [23]. Numerical models are computationally intensive models, like CFD simulations, and solving such models is a computationally intensive process and time consuming [24]. High grid resolution is essential for achieving more accurate and reliable simulations; however, it often increases the demand for computational resources. In the real world, the accuracy of the results is impacted by grid resolution compromises brought on by a lack of processing capability. Turbulence models are used in CFD simulations and have limitations in predicting accurate wake turbulence in wind turbines. Due to this limitation, a discrepancy arises between simulated and real-world wake behaviors. There are validation challenges in numeric models against real world data, such as in complex terrain. This challenge in validation makes it difficult to reliability of model as well as it is uncertain to ensure the accuracy of a model's prediction in diverse and changing environmental conditions [25-28].

Kinetic models also have limitations and gaps like numerical models. Some of these gaps and limitations in kinetic models to estimate wind effect in wind farms are empirical nature, simplified representation, limited physics, and modelling calibration. Kinetic models are empirical nature style models. These models are relying on the empirical relationships derived from datasets and use limited datasets. Kinetic models are theory-driven, not data-driven. Unlike machine learning algorithms, kinetic models do not learn from data. They often perform well on datasets that reflect their underlying assumptions, but struggle to generalize to new or more complex datasets. Kinetic models have another gap, which is providing a simplified representation of wake effects and not capturing the complex pattern that affects the accuracy of results of a complex wind farm or a wind farm that has unique and specific atmospheric conditions. Kinetic models are based on physics-based models, but these models sometimes do not capture the physics of complex interaction of wake effects in a wind farm. This physics related limitation can create a big hindrance to the ability of these models to predict certain phenomena with accuracy. Kinetic models need calibration to account for some sort of specific conditions. Due to this limitation, these models face challenges while extrapolating the parameters of these models to a different location [29] [30].

3 Materials and Methods

3.1 Dataset Formulation and Description

The dataset encompasses 4 columns (3 input and one output variable), each variable having 1000 non-null entries. Column one, denoted by "u_{i+1}" (Row) is for a row-associated integer variable, while" X-(Distance)" shows distance as an integer and "R_r-(Rotor Radius)" records the rotor radius. The metric wake speed receives a more precise outcome and is stored in floating-point form (float64) under the "Wake Speed" column.

The following points provide key insights into the distribution of the dataset. The dataset contains 1000 entries for each variable: u_{i+1} (Row), X-(Distance), R_r-(Rotor Radius) and Wake Speed:

- The mean values for u_{i+1}-(Row), X-(Distance), R_r-(Rotor Radius), and Wake Speed are 500.5, 254.6, 25.47, and 0.00575, respectively.
- The minimum values range from 0 for X-(Distance) and Wake Speed to 1 for u_{i+1}- (Row) and 10 for R_r-(Rotor Radius).
- The 25th, 50th (median), and 75th percentiles provide insight into data spread, with Wake Speed mostly concentrated at 0 until the maximum value of 0.90821.
- Standard deviations show that u_{i+1}-(Row) and X-(Distance) have wider variability compared to R_r-(Rotor Radius) and Wake Speed.

The machine learning algorithms were implemented in Python on a high-performance workstation equipped with an Intel Core i7 processor, 32 GB RAM, and NVIDIA RTX 2060 GPU. A total of 1000 samples were analyzed to ensure computational accuracy and reliability. The descriptive statistics of the data are shown in Figure 3.

	Ui+1-(Row)	X-(Distance)	Rr-(RotorRadius)	WakeSpeed
count	1000.000000	1000.000000	1000.000000	1000.000000
mean	500.500000	254.600000	25.470000	0.005754
std	288.819436	116.328725	11.615249	0.052976
min	1.000000	0.000000	10.000000	0.000000
25%	250.750000	100.000000	10.000000	0.000000
50%	500.500000	200.000000	20.000000	0.000000
75%	750.250000	400.000000	40.000000	0.000000
max	1000.000000	400.000000	40.000000	0.908210

Figure 3
Descriptive Stats of the dataset

3.2 Methodology

Figure 4 depicts the research methodology adopted in the present study. The dataset is generated from the Jensen wake model by setting values of the wake model parameters. Then the dataset is divided into training and testing data. The machine learning algorithms are implemented in Python and trained using Keras Tuner library. The trained model is then tested on the testing data, and the values of key performance indicators are recorded to compare the results. The best performing algorithm is selected to estimate the wake speed and further optimize the layout of wind farm or adjust the yaw angle of upstream wind turbines.

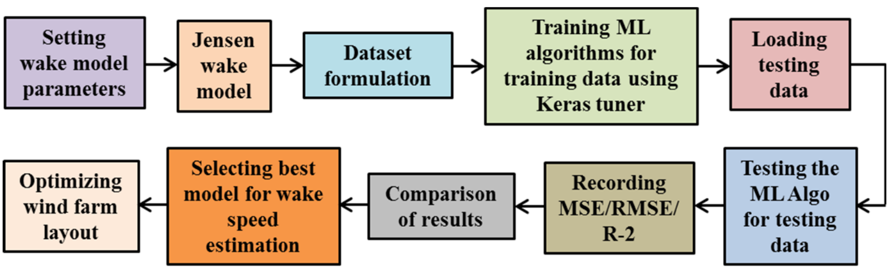

Figure 4
Schematic Diagram of Research Methodology

4 Experimental Results

The “Keras Tuner” was used for training the machine learning algorithms rather than manual tuning. The “Keras Tuner” is a library that automates the process of finding the best hyperparameters for the implemented model. “Keras Tuner” uses different search algorithms such as, Random Search, Hyperband, and Bayesian optimization, to explore the defined search space of hyperparameters and identify the optimal parameters for the model. Therefore, the best possible results for each algorithm are presented with the optimized value of hyperparameters.

The experimental results of machine learning algorithms are compared based on Mean Absolute Error (MAE), Mean Square Error (MSE), Root Mean Square Error (RMSE) and R2 score as follows:

$$MAE = \frac{\sum_{i=1}^{n}|y_i - x_i|}{n} \tag{6}$$

$$MSE = \frac{\sum_{i=1}^{n}(y_i - x_i)^2}{n} \tag{7}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(y_i - x_i)^2}{n}} \tag{8}$$

$$R2 = 1 - \frac{\sum_{i=1}^{n}(y_i - x_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{9}$$

Where, y_i= Actual values

x_i= Predicted value

$\bar{y}$ = Mean of actual values

n= Number of data points

4.1 Artificial Neural Network (ANN)

Two hidden layers and one output layer were part of the artificial neural network (ANN) model configuration. To get better results, the Keras Tuner was used to further fine-tune the model. The best configuration was found by experimenting with a few parameters, leading to the following ideal hyperparameters:

Hidden Layers: (480, 480)

Max Iterations: 1000

Optimizer: Adam

Learning Rate: 0.001

Activation: ReLU

Loss Function: Mean Squared Error

For the given regression task, this arrangement was shown to be the most effective. The ANN model has two hidden layers, each with 480 neurons. For training, the model uses the Adam optimizer and a ReLU activation function, the optimal results produced by ANN are shown in Table 1.

Table 1
Performance of Artificial Neural Networks (ANN)

Metric	R2 score	Testing MSE $(m/s)^2$	Testing RMSE (m/s)	Testing MAE (m/s)
Value	0.39	4.35E-05	0.0066	0.0038

4.2 Random Forest Regression (RFR)

The random forest model was built using the SKLEARN package. The GridSearchCV function was used to improve the predictive accuracy of the model. During the tuning process, the following parameters were altered: n estimators, which determines the number of trees; maximum depth of the trees; min-samples-split, which indicates the minimum number of samples required to split an internal node; min-samples-leaf, which indicates the minimum number of samples required to create a terminal node; and bootstrapping. The following parameters were the most effective for the model.

n Estimators: 100

Min Samples split: 2

Min Samples leaf: 1

Bootstrap: True

Max depth: 10

The random forest model of 100 decision trees yields a perfect match with an R2 score of 0.9905, as shown in Table 2, using a minimum sample split of two and one leaf.

Table 2
Performance of Random Forest Model

Metric	R2 score	Testing MSE $(m/s)^2$	Testing RMSE (m/s)	Testing MAE (m/s)
Value	0.9905	1.26E-06	0.0011	9.60E-05

4.3 Decision Tree Regression (DTR)

The SKLEARN package was utilized to develop the decision tree model. The GridSearchCV function was used to improve and fine-tune the model's forecast accuracy. The tuning process was completed by adjusting parameters such as max depth, which establishes the maximum depth of the tree; min-samples-split, which indicates the minimum number of samples required to split an internal node; and min-samples-leaf, which indicates the minimum number of samples needed to create a terminal node. The model performed best with the following settings.

Max Depth: 20

Min Samples split: 2

Min Samples leaf: 1

Max Features: sqrt

Random State: 42

The decision tree model partitions the data by minimizing the mean squared error and auto optimizing the maximum depth as shown in Table 3.

Table 3
Performance of Decision Tree Regression

Metric	R2 score	Testing MSE $(m/s)^2$	Testing RMSE (m/s)	Testing MAE (m/s)
Value	0.9646	4.75E-06	0.0021	0.0001

4.4 Support Vector Regression (SVR)

The support vector regression model was implemented using the sklearn library. To enhance the model's predictive performance, the GridSearchCV function was employed for hyperparameter tuning. An expanded parameter grid was defined, varying key parameters such as the kernel type, the regularization parameter C (with an extended range), the epsilon value, and the gamma parameter for non-linear kernels. The cross-validation process was increased to 5 folds to ensure robust evaluation. After fitting the GridSearchCV to the training data, the best hyperparameters were identified as:

Kernel: rbf

C: 1000

Epsilon: 0.01

Gamma: scale

The linear kernel is used in the Support Vector Regression (SVR) model to find the optimal line for regression and concurrently employ a regularization parameter C = 0.1 to manage the trade-off between reducing the error and keeping model simplicity, and an epsilon value of 0.01 to specify the margin of tolerance where there is no penalty for errors within this margin. The results produced by SVM is shown in Table 4.

Table 4

Performance of Support Vector Machine

Metric	R2 score	Testing MSE $(m/s)^2$	Testing RMSE (m/s)	Testing MAE (m/s)
Value	0.2557	0.0001	0.01	0.0072

4.5 Adaptive Neuro-Fuzzy Inference System (ANFIS)

The ANFIS model is developed by setting the type of input membership functions, epochs, learning method, and learning rates as hyperparameters. The optimal results produced by ANFIS are shown in Table 5.

Number of Inputs: 3

Type of Input Membership Functions: Gaussian

Number of Rules: 5

Learning Rate: 0.001

Epochs: 100

Table 5

Performance of Adaptive Neuro-Fuzzy Inference System (ANFIS)

Metric	R2 score	Testing MSE $(m/s)^2$	Testing RMSE (m/s)	Testing MAE (m/s)
Value	0.05	0.0031	0.0558	0.0173

The ANFIS model design includes fuzzy logic and neural networks; it makes use of Gaussian membership functions and 5 fuzzy rules for forecasting.

All 5 models show different results. The comparison of the results of these models is given in Table 6.

Table 6

Comparison of Machine Learning Models Performance

Model	R2 score	Testing MSE $(m/s)^2$	Testing RMSE (m/s)	Testing MAE (m/s)
Random Forest Regression	0.9905	1.26E-06	0.0011	9.60E-05

Decision Tree Regression	0.9646	4.75E-06	0.0021	0.0001
Artificial Neural Network	0.39	4.35E-05	0.0066	0.0038
Support Vector Regression	0.2557	0.0001	0.01	0.0072
Adaptive Neuro-Fuzzy Inference System	0.05	0.0031	0.0558	0.0173

Random Forest and Decision Tree Regression: Random forest performs better than decision tree with a higher R2 score of 0.9905 vs 0.9646, MSE of 1.26E-06 vs 4.75E-06, RMSE of 0.0011 vs 0.0021 and MAE of 9.60E-05 vs 0.0001.

Random Forest and Support Vector Regression: Random forest model performs significantly better than support vector regression with an R2 score of 0.9905 vs 0.2557, MSE of 1.26E-06 vs 0.0001, RMSE of 0.0011 vs 0.01 and MAE of 9.60E-05 vs 0.0072.

Random Forest and Adaptive Neuro-Fuzzy Inference System: Random forest again performed better than ANFIS as random forest has R2 score of 0.9905 while ANFIS has 0.05, which is poorer than random forest. The MSE, RMSE, and MAE achieved by ANFIS are 0.0031, 0.0558, and 0.0173.

Random Forest and Artificial Neural Networks (ANNs): Random forest performed better in R2-score than ANN. Random Forest has an R2 score of 0.9905 and ANN has 0.39. The MSE, RMSE, and MAE achieved by ANN are 4.35E-05, 0.0066, and 0.0038.

Figure 5 provides the visual performance of the evaluation metric R2 score of Random Forest, Decision tree, ANN, ANFIS, and SVR models' performance.

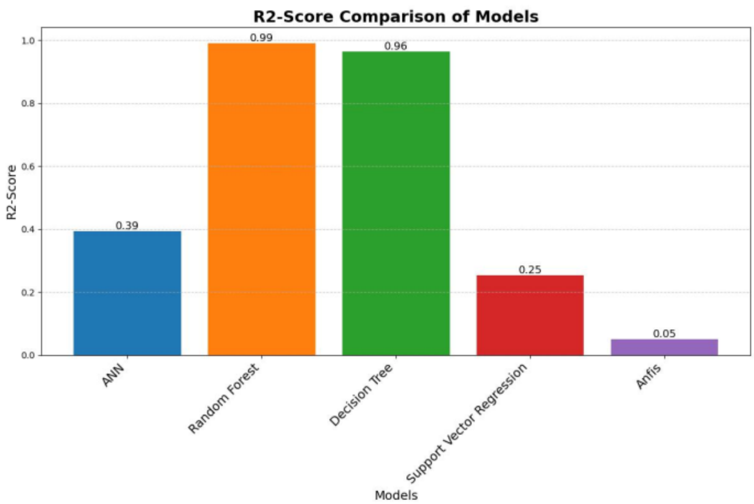

Figure 5
R2-Score Comparison

The graph of R2 scores highlights the remarkable accuracy of the Random Forest regression model compared to other methods analyzed in the study. Random Forest achieves an impressive R2 score of 0.9905, showcasing its strong predictive capability and reliability in estimating the wake effect in wind farms. The Decision Tree model, while performing relatively well with an R2 score of 0.9646, does not match the precision of Random Forest. This outcome underscores the advantage of Random Forest's ensemble approach, which leverages the strength of multiple decision trees for improved accuracy. The Support Vector Machine (SVM) and Adaptive Neuro-Fuzzy Inference System (ANFIS) models exhibit much weaker performance, with R2 scores of 0.2557 and 0.05, respectively. These results suggest that SVM struggles to generalize effectively to the data, while ANFIS is not well-suited for this type of prediction task, yielding particularly poor results. The Artificial Neural Network (ANN) model, with an R2 score of 0.39, demonstrates limited predictive ability compared to Random Forest. This performance gap highlights the challenges faced by ANN in achieving high accuracy without substantial optimization of parameters and sufficient training data.

Figure 6 illustrates that the Random Forest model consistently delivers the highest predictive accuracy, making it the most suitable algorithm for modelling the wake effect in wind farms.

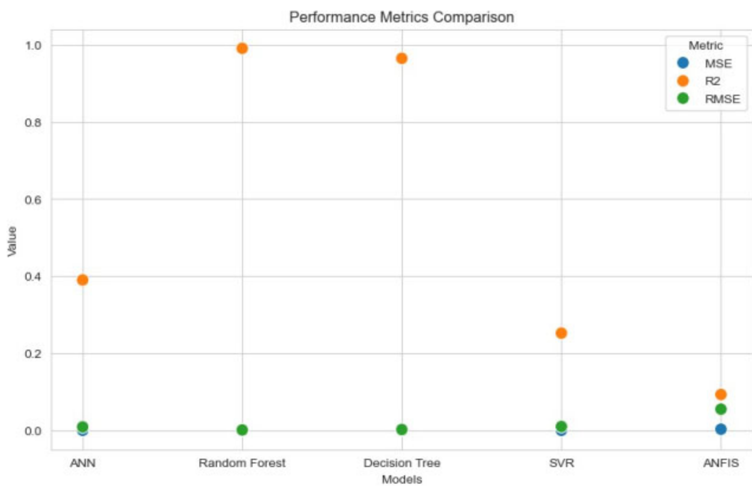

Figure 6
Performance Metrics Comparison

The combination of these metrics reinforces the conclusion that Random Forest consistently delivers the best performance across all evaluation criteria. It achieves the lowest MSE, RMSE, and MAE, along with the highest R2 score, showcasing its accuracy and robustness. Decision Tree performs moderately well, while models like SVM, ANFIS, and ANN demonstrate weaker predictive abilities with higher errors and lower R2 scores. These results confirm the suitability of Random Forest for estimating wake effects in wind farms.

Conclusions and Future Work

In order to optimize the power produced by wind farms, it is very important to estimate the wake speed, due to upstream wind turbines. The wake is referred to as the turbulent airflow behind the wind turbine rotor after it extracts the energy from the wind. The rotor of the downstream wind turbine faces this wake (turbulent and reduced in speed) affecting its' output power and its' service life. Therefore, it is very important to estimate the wake speed in wind farms. The Jensen model is widely used to estimate the wake effect of wind farms. Despite the complex and turbulent nature of the wake, the Jensen model simplifies it into an ideal cone-shaped wake. Jensen's wake model requires less computation than computational fluid dynamics (CFD) simulations. In this study, the Jensen wake model is used to computationally generate the dataset for the wind turbine wake. Then, the well-known five machine learning algorithms are implemented to intelligently estimate the wake speed in a wind farm. The simulations are done in Python, and machine learning algorithms are trained for an optimized set of hyperparameters using the "Keras Tuner" library. The results of implemented machine learning algorithms are compared based on four performance indicators, such as R2 score, MSE, RMSE and MAE. It is observed that the random forest regression outperformed other algorithms in accurately estimating the wake effect. The estimated wake speed can be used to schedule the yaw angle for upstream wind turbines to divert the wake away from the rotor of the downstream wind turbine, which can improve the overall power-capturing capacity of the wind farm.

The detailed examination of five distinct machine learning models yields invaluable insights into wind farm wake effect estimation. In terms of wake impact prediction, each machine learning model has unique characteristics and performance metrics that highlight its benefits and drawbacks. The complexity of the wake effect estimation is highlighted by the observed heterogeneity among multiple machine learning models. Each model employs different algorithms and methodologies, resulting in varying degrees of accuracy and forecasting capabilities. This demonstrates the importance of selecting a suitable model based on the wind farm's specific requirements and characteristics. The constraints of the wind farm estimation task, the dataset, and other considerations determine which model is best for estimating the wake effects in a wind farm. Strong prediction abilities are demonstrated by the ANFIS, ANN, and SVR models to estimate wake effects on this dataset. Random forest model stands first among all of them in estimating wake effects. It shows an exceptional predictive power, with an R2 score of 0.9905, which is the highest of all. Its' treelike structure enables it to capture complex relationships from the data and makes it a robust choice for estimating wake effects in wind farms.

The Decision tree model performed very well. However, it slightly overperformed by the Random forest model, but not very bad, even better than the other 3 models with a 0.9646 R2 score. Its simplicity and easy interpretability make it an attractive option for the estimation of wake effects in wind farms. The support vector machine also performed well at the R2 score, with a value of 0.2557. ANFIS and ANN show mixed results in estimating wake effects. As the R2 score in ANFIS is 0.05, while ANN achieves 0.39.

Table 7
Comparison of proposed work with the literature

Ref	**Year**	**Model Used**	**R2 score**	**MSE**	**RMSE**
Proposed work	-	Random Forest Model	0.9905	1.26E-06	0.0011
[31]	2023	ANN-Jesen wake model	0.93	-	0.06
[32]	2022	Data Driven Analytical Wake Model	0.8522	-	0.046
[33]	2022	CNN-LSTM	-	0.838	0.915
[34]	2022	Gaussian process modelling	0.83	-	-
[8]	2016	Extreme Learning Machine coupled with Wavelet Transform	0.9956	-	0.269

In Table 7, the results of the present research work have been compared with the results published in the literature. The results have been compared on the basis of performance indicators recorded in the literature [35]. It has been observed from Table 7 that the proposed Random Forest Model produced better results than the models applied in the literature.

In the future, the proposed study can also be further utilized to deploy real-time sensors integrated with the machine learning based control mechanism to intelligently adjust the yaw angle of the wind turbines, considering the wind regime and the local terrain. The proposed study can be further implemented by intelligently optimizing the layout of wind turbines by avoiding direct alignment of wind turbines and maintaining an adequate distance of about 5-9 rotor diameters between the turbines [36].

Future investigations into the assessment of wake impacts in wind farms could go in a few different areas [37]. Expanding the quantity of the dataset used for analysis may increase the accuracy of predictive models and provide a more complete understanding of wake dynamics [38]. Scholars can contribute to the development of more accurate and reliable forecast models for wind farm management and optimization [39] [40]. Moreover, other advanced wake models can be implemented to investigate the impact of terrain on wake characteristics, especially complex terrain, as it significantly affects the wind direction and velocity.

Acknowledgements

The authors extend their appreciation to Taif University, Saudi Arabia, for supporting this work through project number (TU-DSPP-2024-34). This research was funded by Taif University, Saudi Arabia, Project N0. (TU-DSPP-2024-34). This work was also supported by the Ministry of Education, Science, Research and Sport of the Slovak Republic and the Slovak Academy of Sciences under the contract no. VEGA 1/0627/24.

References

[1] Sun, H., Gao, X. and Yang, H., A review of full-scale wind-field measurements of the wind-turbine wake effect and a measurement of the wake-interaction effect. Renewable and Sustainable Energy Reviews, 2020, 132, p. 110042, https://doi.org/10.1016/j.rser.2020.110042

[2] Yang, K., Kwak, G., Cho, K. and Huh, J., Wind farm layout optimization for wake effect uniformity. Energy, 2019, 183, pp. 983-995, https://doi.org/10.1016/j.energy.2019.07.019

[3] El-Ahmar, M. H., El-Sayed, A. H. M. and Hemeida, A. M., December. Evaluation of factors affecting wind turbine output power. In 2017 Nineteenth International Middle East Power Systems Conference (MEPCON) 2017 (pp. 1471-1476) IEEE https://doi.org/10.1109/MEPCON.2017.8301377

[4] Houck, D. R., Review of wake management techniques for wind turbines. Wind Energy, 2022, 25(2), pp. 195-220, https://doi.org/10.1002/we.2668

[5] Placidi, M., Hancock, P. E. and Hayden, P., Wind turbine wakes: experimental investigation of two-point correlations and the effect of stable thermal stability. Journal of Fluid Mechanics, 2023, p. A30, https://doi.org/10.1017/jfm.2023.631

[6] Hasager, C. B., Nygaard, N. G., Volker, P. J., Karagali, I., Andersen, S. J. and Badger, J., Wind farm wake: The 2016 Horns Rev photo case. Energies, 2017, 10(3), p. 317, https://doi.org/10.3390/en10030317

[7] Peña, A., Réthoré, P. E. and van der Laan, M. P., On the application of the Jensen wake model using a turbulence-dependent wake decay coefficient: the Sexbierum case. Wind Energy, 2016, 19(4), pp. 763-776, https://doi.org/10.1002/we.1863

[8] Mladenović, I., Marković, D., Milovančević, M. and Nikolić, M., Extreme learning approach with wavelet transform function for forecasting wind turbine wake effect to improve wind farm efficiency. Advances in Engineering Software, 2016, 96, pp. 91-95, http://dx.doi.org/10.1016/j.advengsoft.2016.02.011

[9] Shakoor, R., Hassan, M. Y., Raheem, A. and Wu, Y. K., Wake effect modeling: A review of wind farm layout optimization using Jensen′ s model.

Renewable and Sustainable Energy Reviews, 2016, 58, pp. 1048-1059, https://doi.org/10.1016/j.rser.2015.12.229

[10] Yang, S., Deng, X. and Li, Q., A joint optimization framework for power and fatigue life based on cooperative wake steering of wind farm. Energy, 2025, p. 134849, https://doi.org/10.1016/j.energy.2025.134849

[11] Baxter, J., Walker, C., Ellis, G., Devine-Wright, P., Adams, M. and Fullerton, R. S., Scale, history and justice in community wind energy: An empirical review. Energy Research & Social Science, 2020, 68, p. 101532, https://doi.org/10.1016/j.erss.2020.101532

[12] Song, J., Wang, L., Xin, Z. and Wang, H., Wake field prediction of a wind farm based on a physics-informed neural network with different spatiotemporal prediction performance improvement strategies. Theoretical and Applied Mechanics Letters, 2025, p. 100577, https://doi.org/10.1016/j.taml.2025.100577

[13] Rezaeiha, A., Montazeri, H. and Blocken, B., On the accuracy of turbulence models for CFD simulations of vertical axis wind turbines. Energy, 2019, 180, pp. 838-857, https://doi.org/10.1016/j.energy.2019.05.053

[14] Songyue, L. I. U., Qiusheng, L. I., Bin, L. U. and Junyi, H. E., Prediction of offshore wind turbine wake and output power using large eddy simulation and convolutional neural network. Energy Conversion and Management, 2025, 324, p. 119326, https://doi.org/10.1016/j.enconman.2024.119326

[15] Sun, H. and Yang, H., Study on three wake models' effect on wind energy estimation in Hong Kong. Energy Procedia, 2018, 145, pp. 271-276, https://doi.org/10.1016/j.egypro.2018.04.050

[16] Luo, Z., Wang, L., Xu, J., Wang, Z., Yuan, J. and Tan, A. C., A reduced order modeling-based machine learning approach for wind turbine wake flow estimation from sparse sensor measurements. Energy, 2024, 294, p. 130772, https://doi.org/10.1016/j.energy.2024.130772

[17] Gafoor CTP, A., Kumar Boya, S., Jinka, R., Gupta, A., Tyagi, A., Sarkar, S. and Subramani, D. N., A physics-informed neural network for turbulent wake simulations behind wind turbines. Physics of Fluids, 2025, 37(1) https://doi.org/10.1063/5.0245113

[18] Shamshirband, S., Petković, D., Hashim, R., Motamedi, S. and Anuar, N. B., An appraisal of wind turbine wake models by adaptive neuro-fuzzy methodology. International Journal of Electrical Power & Energy Systems, 2014, 63, pp. 618-624, https://doi.org/10.1016/j.ijepes.2014.06.022

[19] Li, B., Ge, M., Li, X. and Liu, Y., A physics-guided machine learning framework for real-time dynamic wake prediction of wind turbines. Physics of Fluids, 2024, 36(3) https://doi.org/10.1063/5.0194764

[20] Witha, B., CFD-Type Wake Models. Handbook of Wind Energy Aerodynamics, 2022, pp. 1001-1038, https://doi.org/10.1007/978-3-030-05455-7_51-1

[21] Zhang, B., Luo, W., Luo, Z., Xu, J. and Wang, L., Effectiveness of wake control optimization for multiple in-line wind turbines by combinatorial machine learning wake model. Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science, 2024, 238(5), pp. 1280-1294, https://doi.org/10.1177/09544062231185516

[22] Stevens, R. J., Martínez-Tossas, L. A. and Meneveau, C., Comparison of wind farm large eddy simulations using actuator disk and actuator line models with wind tunnel experiments. Renewable energy, 2018, 116, pp. 470-478, https://doi.org/10.1016/j.renene.2017.08.072

[23] Sun, S., Cui, S., He, T. and Yao, Q., An integrated deep neural network framework for predicting the wake flow in the wind field. Energy, 2024, 291, p. 130400, https://doi.org/10.1016/j.energy.2024.130400

[24] Stevens, R. J., Martínez-Tossas, L. A. and Meneveau, C., Comparison of wind farm large eddy simulations using actuator disk and actuator line models with wind tunnel experiments. Renewable energy, 2018, 116, pp. 470-478, https://doi.org/10.1016/j.renene.2017.08.072

[25] van Binsbergen, D., Daems, P. J., Verstraeten, T., Nejad, A. R. and Helsen, J., Hyperparameter tuning framework for calibrating analytical wake models using SCADA data of an offshore wind farm. Wind Energy Science, 2024, 9(7), pp. 1507-1526, https://doi.org/10.5194/wes-9-1507-2024

[26] Abkar, M. and Dabiri, J. O., Self-similarity and flow characteristics of vertical-axis wind turbine wakes: an LES study. Journal of Turbulence, 2017, 18(4), pp. 373-389, https://doi.org/10.1080/14685248.2017.1284327

[27] Kavousi-Fard, A., Dabbaghjamanesh, M., Sheikh, M. and Jin, T., A novel deep learning based digital twin model for mitigating wake effects in wind farms. Renewable Energy Focus, 2025, p. 100686, https://doi.org/10.1016/j.ref.2025.100686

[28] Krokavec D., Filasová A., Data Driven, Additive Fault Detection for Wind Turbines. ACTA POLYTECHNICA HUNGARICA, 2023, pp. 83-96, https://doi.org/10.12700/APH.20.11.2023.11.6

[29] Fleming, P., King, J., Simley, E., Roadman, J., Scholbrock, A., Murphy, P., Lundquist, J. K., Moriarty, P., Fleming, K., van Dam, J. and Bay, C., Continued results from a field campaign of wake steering applied at a commercial wind farm: Part 2. Wind Energy Science Discussions, 2020, pp. 1-24, https://doi.org/10.5194/wes-5-945-2020

[30] Diahoychenko, I., Yevtushenko, I., Kolcun, M., Conka, Z., Zahorodnia, T., Vasyleha, P., Demand-Supply Balancing in Energy Systems with High Photovoltaic Penetration, using Flexibility of Nuclear Power Plants. ACTA

POLYTECHNICA HUNGARICA, 2023, pp. 115-135, https://doi.org/10.12700/APH.20.11.2023.11.8

[31] Pujari, K. N., Miriyala, S. S. and Mitra, K., Jensen-ANN: A machine learning adaptation of jensen wake model. IFAC-PapersOnLine, 2023, 56(2), pp. 4651-4656, https://doi.org/10.1016/j.ifacol.2023.10.979

[32] Nai-Zhi, G., Ming-Ming, Z. and Bo, L., A data-driven analytical model for wind turbine wakes using machine learning method. Energy Conversion and Management, 2022, 252, p. 115130, https://doi.org/10.1016/j.enconman.2021.115130

[33] Park, R. J., Kim, J. H., Yoo, B., Yoon, M. and Jung, S., Verification of prediction method based on machine learning under wake effect using real-time digital simulator. Energies, 2022, 15(24), p. 9475, https://doi.org/10.3390/en15249475

[34] Ashwin Renganathan, S., Maulik, R., Letizia, S. and Iungo, G. V., Data-driven wind turbine wake modeling via probabilistic machine learning. Neural Computing and Applications, 2022, pp. 1-16, https://doi.org/10.1007/s00521-021-06799-6

[35] Medved, D., Kolcun, M., Conka, Z., Hocko, P., Novák, M., Simulation of Faults in the Power System and Their Elimination in the EMTP-ATP. 7th International Scientific Symposium on Electrical Power Engineering (ELEKTROENERGETIKA), 2013, pp. 484-487, ISBN: 978-80-553-1442-6

[36] Kolcun, M., Gawlak, A., Kornatka, M., Conka, Z., Active and Reactive Power Losses in Distribution Transformers. ACTA POLYTECHNICA HUNGARICA, 2020, pp. 161-174, https://doi.org/10.12700/APH.17.1.2020.1.9.

[37] Kumar, B.; Szepesi, G.; Čonka, Z.; Kolcun, M.; Péter, Z.; Berényi, L.; Szamosi, Z. Trendline Assessment of Solar Energy Potential in Hungary and Current Scenario of Renewable Energy in the Visegrád Countries for Future Sustainability. *Sustainability* **2021**, *13*, 5462, https://doi.org/10.3390/su13105462

[38] Diahoychenko, I., Yevtushenko, I., Kolcun, M., Conka, Z., Zahorodnia, T., Vasyleha, P., Demand-Supply Balancing in Energy Systems with High Photovoltaic Penetration, using Flexibility of Nuclear Power Plants. ACTA POLYTECHNICA HUNGARICA, 2023, pp. 115-135, https://doi.org/10.12700/APH.20.11.2023.11.8

[39] Javed, M. Y., Asghar, A. B., Naveed, K., Nasir, A., Alamri, B., Aslam, M., Al-Ammar, E. A., Conka, Z., Improving the efficiency of photovoltaic-thermoelectric generator system using novel flying squirrel search optimization algorithm: Hybrid renewable and thermal energy system (RTES) for electricity generation. PROCESS SAFETY AND

ENVIRONMENTAL PROTECTION, 2024, pp. 104-116, https://doi.org/10.1016/j.psep.2024.04.093

[40] Diahoychenko, I., Chuprun, A., Conka, Z., Assessment and mitigation of the influence of rising charging demand of electric vehicles on the aging of distribution transformers. ELECTRIC POWER SYSTEMS RESEARCH, 2023, pp. 1-13, https://doi.org/10.1016/j.epsr.2023.109455