Journal of Physics: Conference Series **3224** (2026) 022024 doi:10.1088/1742-6596/3224/2/022024

Short-Horizon Data-Driven Joint Forecasting of Wind Speed and Direction for Future Aware Wind Farm Control

Abdulbaset Alazhare[1*], Michael F. Howland[2] and Breiffni Fitzgerald[1]

[1] Civil Structural & Environmental Engineering, Trinity College Dublin, Dublin, Ireland
[2] Civil and Environmental Engineering, Massachusetts Institute of Technology, Cambridge, USA

[*]E-mail: alazhara@tcd.ie

Abstract.
Preview-based wind-farm control depends on accurate short-horizon estimates of wind speed and direction. Yet at five-minute resolution, practical models must be low-latency, and must outperform persistence modelling. This paper presents a two-step, control-oriented data-driven forecasting model for one-step-ahead prediction of the wind vector components. In step 1, the study compares a Long short-term memory network (LSTM), a 1-D convolutional neural network (CNN), and a causal temporal convolutional network (TCN) against a one-step persistence baseline, under a unified training and evaluation protocol. In step 2, the study retains the best-performing architecture and enhances the input representation using variational mode decomposition (VMD) and mutual-information (MI) lag analysis, yielding a VMD–MI–TCN model. The study uses three consecutive years of five-minute wind speed and direction from NREL's Wind Resource. Forecasts are evaluated on a held-out test set using wind-speed errors, circular wind-direction errors, and error-based skill scores relative to persistence. Overall, the proposed workflow provides a basis for constructing and benchmarking lightweight wind speed and wind direction forecasting models suitable for real-time control integration.

1. Introduction

Wind farms operate in a turbulent atmospheric boundary layer where inflow direction and speed can change on the order of minutes, yet many supervisory controllers assume persistence over the next decision interval. This assumption simplifies implementation but can degrade wind farm flow control outcomes when the wind direction changes or the wind speed ramps shortly after a control action, leading to misalignment, reduced farm power, and potentially higher loads [1–3]. Recent studies have shown that yaw-based wake steering is effective but sensitive to wind direction variability, underscoring the value of anticipatory information for setpoint selection [4,5]. Furthermore, studies in wake steering and predictive control suggest that even modest reductions in short-horizon direction error can translate into measurable farm-level energy gains, strengthening the motivation for control-oriented forecasting [6]. In line with this, international initiatives such as IEA Wind Task 51 include minute-scale forecasting as one of several work streams aimed at linking improved short-horizon inflow characterisation with operational decision-making [7], reinforcing its importance for forecast-aware wind farm control.

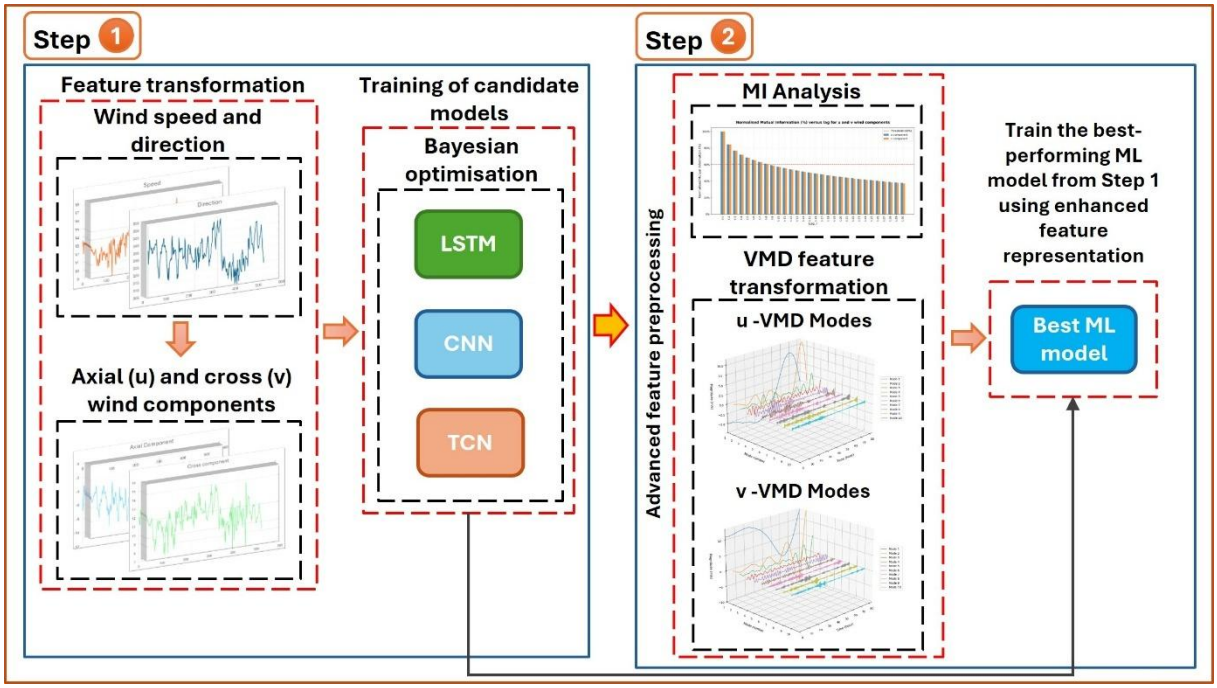

Figure 1. Overview of the proposed two-step wind-vector forecasting workflow.

Earlier wind forecasting studies largely used linear statistical time-series models such as ARMA/ARIMA [8]. More recently, deep-learning sequence models have become common for wind forecasting, where recent reviews of wind forecasting highlight recurrent networks (e.g., LSTM/GRU) and convolutional sequence models as two dominant deep-learning families, and emphasise the importance of consistent benchmarking across forecast horizons [9,10]. Convolutional sequence modelling is particularly attractive because causal, dilated 1-D convolutions can capture long effective temporal context efficiently. Furthermore, temporal convolutional networks (TCNs) implements this idea by stacking causal dilated convolution layers (often within residual blocks) to model sequence-to-sequence or sequence-to-one forecasting problems [11]. A further theme in wind forecasting is the use of hybrid "decomposition-prediction" pipelines to address non-stationarity and multi-scale variability in wind signals. Reviews of ultra-short-term forecasting describe decomposition-based pre-processing as a common route to reduce input complexity and isolate informative components before prediction [12]. Variational mode decomposition (VMD) is one of the most successful signal decomposition algorithm for forecasting tasks as demonstrated by [13] and has been combined with a range of predictors for wind-speed forecasting, including hybrid VMD-based models [13–15].

Nonetheless, at 5-minute one-step horizon – which typically represents the response time of yaw-based wake steering, and therefore, provides information suitable for control-oriented applications – there is limited controlled benchmarking of common deep sequence models against persistence for joint speed-direction prediction, and limited quantification through controlled comparisons of how decomposition plus data-driven lag selection affects skill under a unified protocol. Therefore, the contributions of this paper are as follows: (i) We formulate one-step-ahead (5-minute) joint forecasting of wind speed and direction as a two-component wind-vector regression problem in Cartesian space, avoiding angular discontinuities and enabling consistent learning and evaluation across variables. (ii) We provide a controlled benchmark of multiple contemporary deep-learning sequence models under an identical training protocol, and quantify their value using skill scores relative to a one-step persistence baseline, which is known to be difficult to

outperform at this horizon. (iii) We utilise feature enhancement techniques that combine variational mode decomposition (VMD) for multi-scale representation with mutual-information (MI) analysis for data-driven window-length selection, see Figure 1, together with diagnostics that justify these design choices.

The present study is intended as a benchmarking exercise rather than a deployment validation study. Specifically, the objective is to determine whether machine-learning models can extract short-horizon predictive skill beyond persistence under a controlled and reproducible settings. Validation against within-farm measurement data, where higher-frequency variability, sensor noise, and data-quality issues are more pronounced, is identified as a necessary next step before operational conclusions can be drawn.

2. Data and pre-processing

2.1 Data source

The study uses three consecutive years of five-minute wind speed and wind direction outputs from a Weather Research and Forecasting (WRF) modelling product accessed via the National Renewable Energy Laboratory (NREL) Wind Resource Database (WRDB) [16]. This dataset was chosen to provide a long and continuous wind-speed and wind-direction record at a fixed height, enabling a controlled and reproducible comparison of model architectures and the proposed feature-enhancement strategy while avoiding the need for extensive data cleaning and harmonisation that can introduce additional confounding factors (e.g., data gaps, sensor changes, or height inconsistencies). Numerical weather prediction time series typically represent atmospheric flow at a coarser effective resolution and therefore do not fully capture the small-scale variability that would be present in within-farm measurement data. Consequently, the dataset is used here primarily as a clean and reproducible benchmark for model development and comparison, rather than as a direct representation of operational wind-farm inflow measurements.

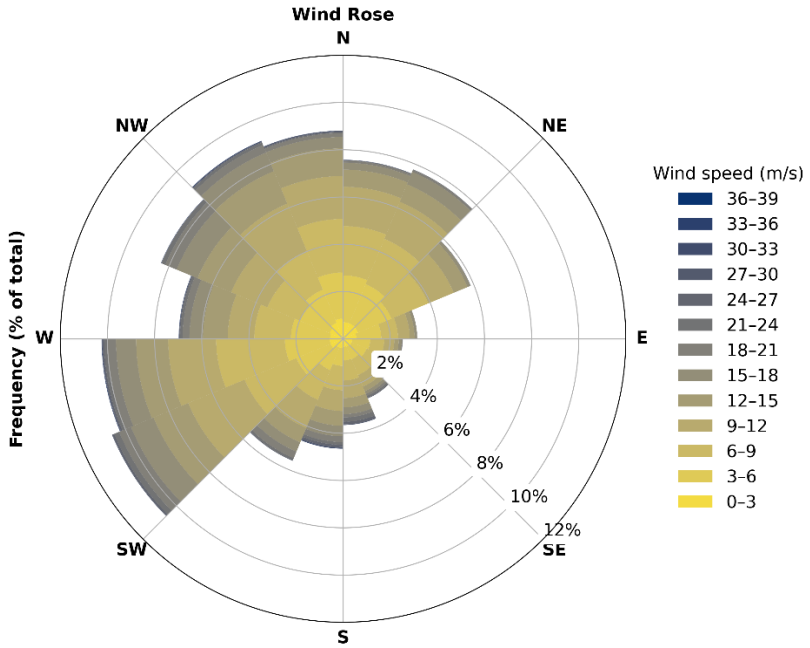

Figure 2. Wind rose of the three-year NREL WRDB WRF dataset at 140 m above mean sea level.

Data are extracted at a single offshore grid point off the coast of Virginia, a zone of active offshore wind development (e.g., Dominion's Coastal Virginia Offshore Wind area), and at a hub-height-relevant level of 140 m above mean sea level. A wind-rose summary of the full three-year record is shown in Figure 2. The distribution is dominated by south-westerly flow, with substantial contributions from westerly and north-westerly sectors.

2.2 Quality control, splitting and normalisation

The selected WRDB time series is continuous over the full three-year period and contains no missing samples; accordingly, no temporal interpolation or gap-filling is required. To preserve strict temporal causality and prevent leakage, the record is split chronologically, with 90% used for training, the next 5% for validation, and the final 5% for testing. This choice also maximises the amount of training data available for the data-hungry one-step-ahead forecasting problem under a unified benchmarking protocol. Across these splits, wind-speed statistics remain broadly similar, with means of 9.21, 8.06, and 9.65 m/s, standard deviations of 5.03, 4.60, and 5.22 m/s, and medians of 8.59, 7.37, and 8.91 m/s for the training, validation, and test sets, respectively. Wind-direction resultant lengths, as defined in [17], are likewise comparable ($\bar{R}$ = 0.28, 0.30, and 0.35), indicating that the validation and test segments remain broadly representative of the full dataset despite their shorter duration. Feature scaling uses z-score normalisation computed exclusively on the training split. More extensive temporal robustness analysis, for example using longer held-out windows, seasonal hold-out strategies, or rolling-origin evaluation, is left as future work.

2.3 Transformation to Cartesian wind components

The native variables are wind speed U (m/s) and wind direction $\emptyset$ (degrees). These are mapped to horizontal Cartesian components (u, v) to enable linear regression while avoiding wrap-around discontinuities in the target. The inverse mapping is later used to interpret forecasts in speed–direction form. Wind direction is recovered from the predicted components using a four-quadrant arctangent and a modulo operation to constrain angles to [0°, 360°). Directional errors are computed using a minimal angular difference to ensure circular consistency.

3. Forecasting formulation and baseline

3.1 Supervised learning via sliding-window embedding

Each training example consists of a fixed-length historical window of normalised feature vectors and a one-step-ahead target. Let the measured wind vector at time index t be $\mathbf{y}_t = [u_t \;\; v_t] \in \mathbb{R}^2$, where u_t and v_t are the horizontal Cartesian wind components. For a fixed window length T, the input sequence used to predict the next step is $\boldsymbol{X}_t = [\boldsymbol{x_{t-T+1}}, \boldsymbol{x_{t-T+2}}, \dots, \boldsymbol{x_t}] \in \mathbb{R}^{\boldsymbol{T\times d}}$ where $\mathbf{x}_t \in \mathbb{R}^d$ is the feature vector at time t. The one-step-ahead forecasting problem is defined as learning a mapping $f_\theta: \mathbb{R}^{T\times d} \rightarrow \mathbb{R}^2$, $\hat{\boldsymbol{y}}_{t+1} = f_\theta(\boldsymbol{X}_t)$, where θ denotes model parameters and $\hat{\mathbf{y}}_{t+1} = [\hat{u}_{t+1}, \hat{v}_{t+1}]^\top$ is the predicted wind vector. Training samples are generated for $t = T, \dots, N-1$, yielding the supervised dataset $\mathcal{D} = \{(\boldsymbol{X}_t, \boldsymbol{y}_{t+1})\}_{t=T}^{N-1}$.

3.2 Persistence baseline

A one-step persistence model which is used as the benchmark. For each time step, persistence assumes the next wind vector equals the most recent observation. This baseline is particularly strong at five-minute horizons and provides an operational reference for reporting skill scores. Mathematically, it is defined as $\hat{y}_{t+1}^{pers} = y_t$.

4. Models and training protocol

4.1 LSTM, CNN, and TCN architectures

This study benchmarks three deep-learning architectures which are Long Short-Term Memory (LSTM), Convolutional Neural Network (CNN), and Temporal Convolutional Network (TCN), under a common forecasting formulation. Each model receives a fixed-length historical window of normalised feature vectors and produces a one-step-ahead prediction of the horizontal wind-vector components, (u, v), jointly. The LSTM is implemented as a sequence-to-vector recurrent model that maps an input window of length T to a single forecast at the next time step. The network processes the input sequentially, updating hidden and cell states through gated recurrences, and the final hidden state (at the end of the window) is used as a compact summary of the historical context. This final hidden state is passed to a linear prediction head to output the two-component vector forecast. Model capacity is controlled through the number of stacked LSTM layers and the hidden-state dimension, while regularisation is applied via dropout between LSTM layers (i.e., across layers rather than across time). The CNN applies one-dimensional convolutions along the temporal axis to extract local patterns within the input window (e.g., short ramps and oscillatory structure). The input is treated as a multi-channel time series (one channel per feature) and passed through a small stack of temporal convolution layers with nonlinear activations. The resulting feature maps are compressed through a pooling operation to obtain a fixed-size representation, which is then passed through a fully connected prediction head to output (u, v). Dropout is applied in the prediction head to reduce overfitting. The TCN is a causal convolutional architecture built from residual blocks containing causal, dilated one-dimensional convolutions. Causality is enforced by left-padding such that the representation used to predict the next step is constructed only from current and past samples within the window. Dilation expands the effective receptive field without requiring deep recurrence, enabling the model to represent longer temporal dependencies while maintaining efficient parallel computation. As with the CNN, the TCN output is passed to a compact fully connected head that produces an unconstrained two-dimensional regression output.

4.2 Training objective, optimisation, and model selection

All the models are trained using a mean-squared error (MSE) objective computed jointly over the two predicted components. Optimisation is performed using Adam, and model selection is based on validation performance using early stopping with a fixed patience. A Reduce-on-Plateau learning-rate scheduler is applied to stabilise training when validation improvement stalls. The loss (objective) function used for optimisation is:

$$\mathcal{L}(\theta) = \frac{1}{N}\sum_{i=1}^{N} \| f_\theta(\boldsymbol{X}^{(i)}) - \boldsymbol{y}^{(i)} \|_2^2$$

Where $\boldsymbol{N}$ is the mini-batch size, $\boldsymbol{\theta}$ denotes the set of trainable model parameters, $\boldsymbol{X^{(i)}} \in \mathbb{R}^{\boldsymbol{T\times d}}$ is the $\boldsymbol{i}$-th input sequence window of length $\boldsymbol{T}$ with $\boldsymbol{d}$ features, $\boldsymbol{y^{(i)}} \in \mathbb{R}^{\boldsymbol{2}}$ is the corresponding one-step-ahead target wind-vector in Cartesian components, $\boldsymbol{f_\theta}(\cdot)$ is the model mapping from the input window to the predicted vector $\boldsymbol{\hat{y}^{(i)} = f_\theta(X^{(i)})}$, and $\|\cdot\|_{\boldsymbol{2}}$ denotes the Euclidean norm. Hyperparameters are tuned separately for the LSTM, CNN, and TCN using Bayesian optimisation via Optuna's Tree-structured Parzen Estimator sampler [18] with pruning to allocate computation efficiently across trials. Each trial is trained with a limited budget during optimisation, and the

best-performing configuration is subsequently re-trained with a longer budget while checkpointing the epoch that achieves the best validation score. The tuned hyperparameters include both architectural and training settings for all of the trained models, see Table 1 below.

4.3 Evaluation metrics

Evaluation is performed on the held-out test set using MAE, RMSE, and MAPE. To compare ML models against the persistence baseline, a skill score is also reported: $\%\, Skill\ Score = (1 - E_{ML}/E_{per}) \times 100$, where E_{ML} and E_{pers} denote the error of the machine-learning model and the persistence forecast, respectively.

5. Results and discussion

5.1 Variational mode decomposition and mode-count selection

The input representation is enhanced by decomposing each wind component into K band-limited modes using variational mode decomposition (VMD), see [19] for information on the mathematical formulation of VMD. To select K (the number of decomposition modes), candidate decompositions are evaluated by examining the centre frequencies of each mode and computing the minimum spacing between adjacent centre frequencies as a function of K. Figure 3 shows that as K increases, mode spacing decreases and then stabilises, indicating diminishing returns. Based on the flattening of the spacing curves between approximately $K = 6$~8, a conservative mode count of $K = 8$ is adopted as a compromise between spectral separation and model parsimony.

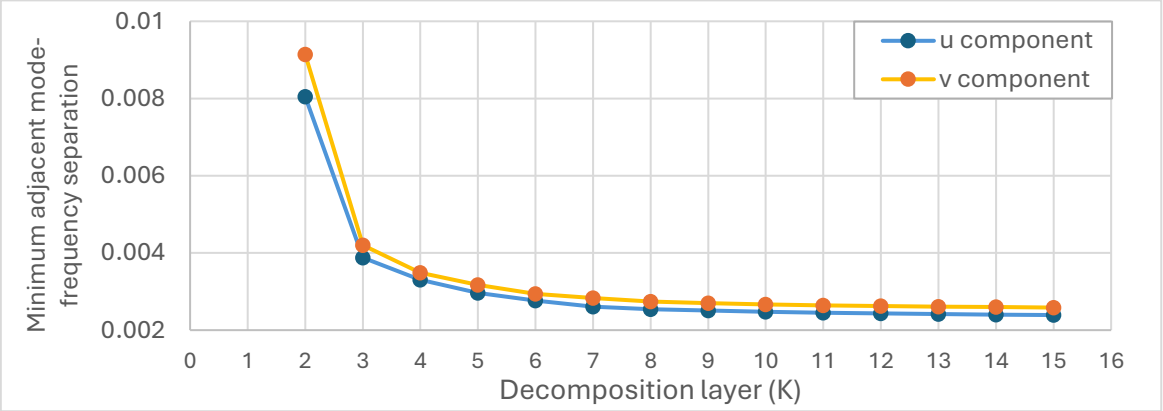

Figure 3. Minimum spacing between adjacent VMD mode centre frequencies versus number of decomposition layers K for u and v.

5.2 Mutual-information lag analysis and window-length selection

Mutual information (MI) between the forecast target and lagged component values is used to select a data-driven input window length, see [20] for information on the mathematical formulation of MI. For lags from 1 to 30 sampling intervals, MI is estimated from the training data using a nearest-neighbour estimator, then normalised by its maximum value. The results of this analysis are shown in Figure 4. There is no agreed mutual-information (MI) threshold in the literature for selecting input sequence length, with reported cut-offs ranging from 25% [21] to 80% [13]. Therefore, the MI threshold in this study is treated as a pragmatic choice to balance information retention against input dimensionality, and to remain broadly consistent with the Step-1 outcome where Bayesian optimisation identified an effective TCN window length of 11, a 60% cut-off on

normalised MI is therefore adopted. For both u and v, MI remains above 60% up to lag 8 and falls below thereafter, thus, the model trained in Step 2 is trained with window length $T = 8$.

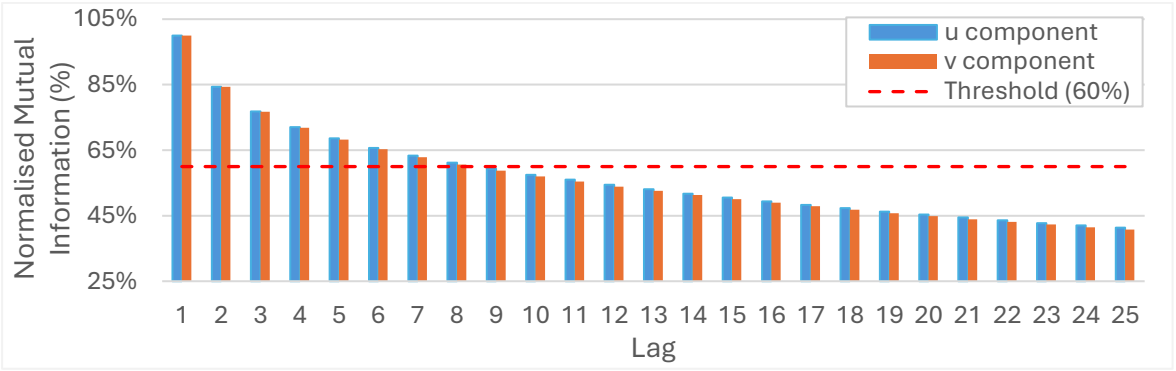

Figure 4. Normalised mutual information versus lag for u and v components with a 60% threshold.

5.3 Selected hyperparameter configurations

Table 1 below summarise the optimised configurations for the LSTM, CNN, TCN, and the VMD–MI–TCN model. The VMD–MI–TCN window length is fixed by MI analysis rather than tuned.

Table 1. Hyperparameter search space and corresponding optimal configurations for the machine learning models evaluated in this study.

	Search space	LSTM	CNN	TCN	VMD–MI–TCN
Number of LSTM layers	$1 \leq N_{lstm} \leq 3$	1	N/A	N/A	N/A
LSTM hidden state size	$32 \leq H_{lstm} \leq 256$	48	N/A	N/A	N/A
No. convolutional layer	$1 \leq N_c \leq 3$	N/A	2	N/A	N/A
No. residual blocks	$1 \leq N_r \leq 3$	N/A	N/A	1	1
No. filters	$32 \leq N_f \leq 256$	N/A	223	88	82
Kernel size K_s	{3,5,7,9}	N/A	3	5	9
FC hidden size	$32 \leq H \leq 256$	N/A	134	254	131
Dropout probability	$0.0 \leq p \leq 0.5$	N/A	0.01	0.011	0.058
Batch size B	$\{64, 128, 256\}$	256	64	64	128
Window length	$5 \leq \mathrm{T} \leq 300$	245	22	11	8
Learning rate	$10^{-5} \leq \eta \leq 10^{-2}$	0.0095	0.00025	10^{-5}	1.8×10^{-4}
VMD modes per component	$1 \leq \mathrm{K} \leq 15$	N/A	N/A	N/A	10

5.4 Overall performance of trained models

Table 2 summarises test-set errors for wind-speed and wind-direction forecasting across the persistence baseline, three deep sequence models (LSTM, CNN, TCN), and the feature-enhanced VMD–

MI–TCN. Across all metrics and both targets, a consistent performance ranking is observed. The Persistence performs worse than both LSTM and CNN, which show comparable performance, but both models are outperformed by TCN, while VMD–MI–TCN achieves the best overall performance. For wind speed, persistence yields MAPE = 2.931 and RMSE = 0.493 m/s. All learned models improve on this baseline, with the largest gains achieved by VMD–MI–TCN (MAPE = 2.210, RMSE = 0.443 m/s), corresponding to 24.25% skill (MAPE) and 10.28% skill (RMSE) relative to persistence. Among the "plain" deep models, TCN is consistently strongest (MAPE = 2.288, RMSE = 0.457 m/s), exceeding the LSTM and CNN on both metrics. For wind direction, persistence produces MAE = 1.795° and RMSE = 3.496°. Again, all deep models improve on persistence, with VMD–MI–TCN providing the best overall direction accuracy (MAE = 1.273°, RMSE = 2.722°), equating to 29.12% skill (MAE) and 22.14% skill (RMSE). TCN is again the strongest of the three base architectures (MAE = 1.358°, RMSE = 2.826°). Two points are notable. First, the improvements are larger for direction than for speed when expressed as skill, indicating that the learning models extract more exploitable short-horizon structure for directional changes than for speed at this one-step horizon. Second, the benefit of adding VMD and MI-based windowing is consistent across targets as VMD–MI–TCN improves upon the already-strong TCN model for both speed (RMSE 0.457 to 0.443 m/s) and direction (RMSE 2.826° to 2.722°), supporting the value of the feature-enhancement step rather than relying solely on architecture choice, which aligns with the findings of [22,23].

From a wind-farm control perspective, the practical value of improved one-step-ahead forecasting lies primarily in providing more reliable short-horizon preview information for supervisory control strategies. In particular, more accurate short-term estimates of wind speed and direction can support improved timing of yaw-based wake steering actions. The consistent improvement over persistence observed here therefore suggests that machine-learning-based predictors can provide useful additional information relative to purely reactive baselines. However, the extent to which this forecasting skill translates into measurable farm-level benefits, such as increased power capture or reduced structural loads, ultimately depends on the design of the closed-loop control system and must be evaluated through dedicated control simulations or field studies.

Table 2. Test-set performance metrics and persistence skill scores for one-step-ahead wind speed and wind direction forecasts.

Target	Metric	Persistence	LSTM	CNN	TCN	VMD–MI–TCN
Speed	MAPE	2.931	2.477	2.557	2.288	2.210
	RMSE	0.493	0.468	0.473	0.457	0.443
	% skill score (MAPE)	N/A	15.48%	12.74%	21.94%	24.25%
	% skill score (RMSE)	N/A	5.06%	4.15%	7.45%	10.28%
Direction	MAE	1.795	1.494	1.418	1.358	1.273
	RMSE	3.496	3.053	2.897	2.826	2.722
	% skill score (MAE)	N/A	16.77%	21.04%	24.39%	29.12%
	% skill score (RMSE)	N/A	12.66%	17.14%	19.18%	22.14%

5.5 Wind speed errors across speed bins

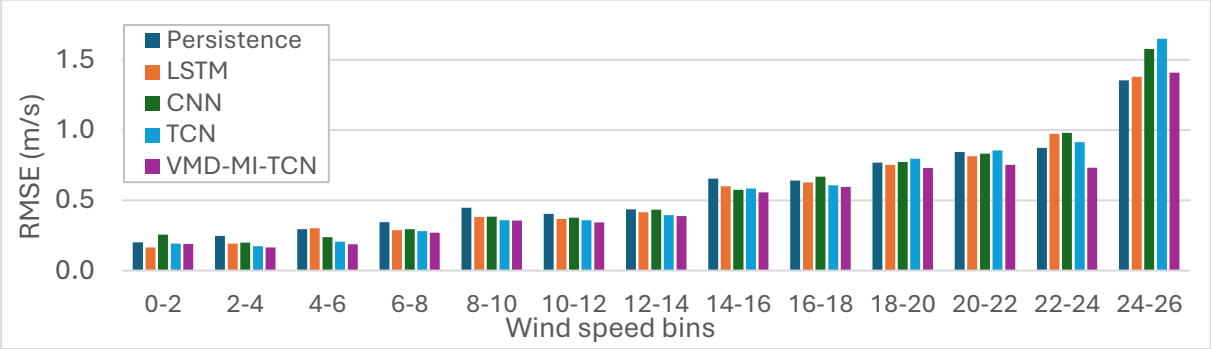

Figure 5. Wind speed RMSE by wind-speed bin.

Figure 5 shows a general trend across models: errors increase with wind speed, with the largest RMSE appearing in the highest-speed bin (24-26 m/s). This pattern could be partially due to the fact that higher wind speeds often coincide with stronger gradients and greater short-term variability in the wind vector. Across low-to-moderate wind speeds, the learned models typically reduce RMSE relative to persistence, and the advantage of TCN and VMD–MI–TCN is consistently visible. However, at the highest speeds the differences between methods narrow and, in some bins, certain learned models can underperform persistence. This is an important outcome as in short-horizon forecasting: extreme bins tend to contain fewer samples, and the conditional dynamics can differ from the bulk regime, making generalisation harder. Notably, VMD–MI–TCN remains close to the better-performing methods in the high-speed tail, aligning with the overall reduction in test-set RMSE reported in Table 2. When combined with the wind rose (Figure 2) the speed-binned view also indicates that the data are concentrated in mid-range speeds, with progressively fewer samples at the highest speeds.

5.6 Wind direction errors across direction bins

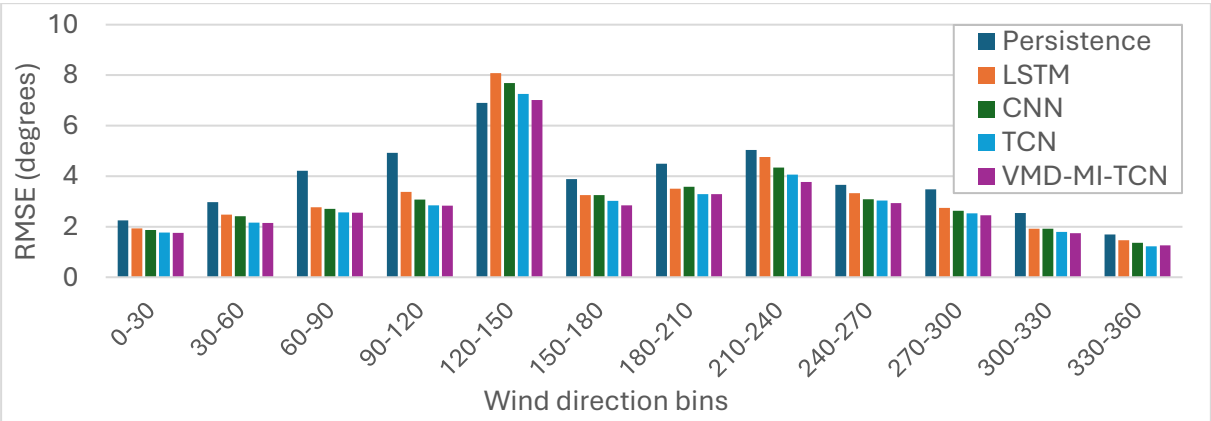

Figure 6. Wind direction RMSE by directional sector.

Figure 6 shows that RMSE varies substantially with wind direction across all models, indicating that short-horizon direction predictability is not uniform across sectors. A pronounced peak occurs in the 120–150° bin for every method, where errors increase markedly relative to

neighbouring bins. In this sector, the learned models fail to outperform the persistence baseline, whereas VMD–MI–TCN remains close to persistence and achieves the best performance in most of the remaining bins. This behaviour suggests that certain directional regimes are intrinsically harder to predict at a one-step horizon, possibly because they exhibit more rapid variability or because they are represented by fewer examples in the training data. The wind rose (Figure 2) supports this interpretation by showing a non-uniform directional distribution, with dominant flow concentrated in specific sectors and comparatively low occurrence in others. In this context, the improvement delivered by VMD–MI–TCN is valuable because by separating large-scale wind-direction behaviour from faster fluctuations, the decomposition-based representation can stabilise learning and reduce overfitting in sparsely represented directional sectors.

6. Conclusion

This paper presented a two-step, machine-learning workflow for one-step-ahead (5-minute) joint forecasting of wind speed and wind direction by predicting the horizontal wind-vector components in Cartesian space. Using three years of 5-minute WRDB data at 140 m and a unified training protocol, the LSTM and CNN improved modestly over persistence, while the causal TCN delivered consistently stronger performance, particularly for wind direction. Variational mode decomposition and mutual-information-based lag selection enhanced the input representation; consequently, the VMD–MI–TCN achieved the best overall accuracy and the highest skill scores relative to persistence for both speed and direction, demonstrating that multi-scale decomposition and data-driven window selection can provide measurable value. In addition, binned analyses highlighted that forecast errors are regime-dependent, with larger direction errors concentrated in specific sectors.

Overall, these findings are encouraging for preview-based wind-farm control applications where improved short-horizon estimates of inflow conditions may enhance supervisory decision-making. However, because the present study relies on WRF-derived WRDB data rather than within-farm measurement data, the results should be interpreted as a methodological benchmark rather than a direct indication of operational forecasting performance. Future work should therefore evaluate the proposed workflow using within-farm observations and investigate how forecasting improvements propagate through closed-loop wind-farm control objectives such as power capture, yaw alignment, and structural loading.

Acknowledgements

This work was supported by Fulbright Ireland under the Fulbright Irish Student - All Disciplines Award, which funded a six-month research visit to the Howland Lab, MIT. The study was also supported by Science Foundation Ireland (SFI) under Grant No. 20/FFP-P/8702.

References

[1] Fleming P, King J, Simley E, Roadman J, Scholbrock A, Murphy P, Lundquist J K, Moriarty P, Fleming K, van Dam J, Bay C, Mudafort R, Jager D, Skopek J, Scott M, Ryan B, Guernsey C and Brake D 2020 Continued results from a field campaign of wake steering applied at a commercial wind farm – Part 2 *Wind Energy Science* **5** 945–58

[2] Fleming P, King J, Dykes K, Simley E, Roadman J, Scholbrock A, Murphy P, Lundquist J K, Moriarty P, Fleming K, van Dam J, Bay C, Mudafort R, Lopez H, Skopek J, Scott M, Ryan B, Guernsey C and Brake D 2019 Initial results from a field campaign of wake steering applied at a commercial wind farm – Part 1 *Wind Energy Science* **4** 273–85

[3] Simley E, Fleming P, Girard N, Alloin L, Godefroy E and Duc T 2021 Results from a wake-steering experiment at a commercial wind plant: investigating the wind speed dependence of wake-steering performance *Wind Energy Science* **6** 1427–53
[4] Simley E, Fleming P, King J and Sinner M 2021 Wake Steering Wind Farm Control With Preview Wind Direction Information *2021 American Control Conference (ACC)* 2021 American Control Conference (ACC) pp 1783–9
[5] Simley E, Fleming P and King J 2020 Field Validation of Wake Steering Control with Wind Direction Variability *J. Phys.: Conf. Ser.* **1452** 012012
[6] Sengers B A M, Rott A, Simley E, Sinner M, Steinfeld G and Kühn M 2023 Increased power gains from wake steering control using preview wind direction information *Wind Energ. Sci.* **8** 1693–710
[7] Anon T51 Minute-Scale Forecasting *IEA Wind TCP*
[8] Gomes P and Castro R 2012 Wind Speed and Wind Power Forecasting using Statistical Models: AutoRegressive Moving Average (ARMA) and Artificial Neural Networks (ANN) *IJSED* **1** 41–50
[9] Wang Y, Zou R, Liu F, Zhang L and Liu Q 2021 A review of wind speed and wind power forecasting with deep neural networks *Applied Energy* **304** 117766
[10] Alves D, Mendonça F, Mostafa S S and Morgado-Dias F 2023 The Potential of Machine Learning for Wind Speed and Direction Short-Term Forecasting: A Systematic Review *Computers* **12** 206
[11] Bai S, Kolter J Z and Koltun V 2018 An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling
[12] Chen Y, Hu X and Zhang L 2022 A review of ultra-short-term forecasting of wind power based on data decomposition-forecasting technology combination model *Energy Reports* **8** 14200–19
[13] Tang Z, Zhao G and Ouyang T 2021 Two-phase deep learning model for short-term wind direction forecasting *Renewable Energy* **173** 1005–16
[14] Wu Q and Lin H 2019 Short-Term Wind Speed Forecasting Based on Hybrid Variational Mode Decomposition and Least Squares Support Vector Machine Optimized by Bat Algorithm Model *Sustainability* **11** 652
[15] Zhao S, Wang Y, Deng J, Chen Z, Yang P and Li Y 2025 A novel hybrid approach for hourly wind speed forecasting based on variational mode decomposition, data feature reconstruction, and machine learning *Journal of Renewable and Sustainable Energy* **17** 033301
[16] Anon 2025 WRDB - Wind Resource Database
[17] Fisher N I 1995 *Statistical Analysis of Circular Data* (Cambridge University Press)
[18] Ozaki Y, Tanigaki Y, Watanabe S, Nomura M and Onishi M 2022 Multiobjective Tree-Structured Parzen Estimator *jair* **73** 1209–50
[19] Dragomiretskiy K and Zosso D 2014 Variational Mode Decomposition *IEEE Transactions on Signal Processing* **62** 531–44
[20] Magrans de Abril I, Yoshimoto J and Doya K 2018 Connectivity inference from neural recording data: Challenges, mathematical bases and research directions *Neural Networks* **102** 120–37
[21] Saffari M, Williams M, Khodayar M, Shafie-khah M and Catalao J P S 2021 Robust Wind Speed Forecasting: A Deep Spatio-Temporal Approach *2021 IEEE International Conference on Environment and Electrical Engineering and 2021 IEEE Industrial and Commercial Power Systems Europe (EEEIC / I&CPS Europe)* 2021 IEEE International Conference on Environment and Electrical Engineering and 2021 IEEE Industrial and Commercial Power Systems Europe (EEEIC / I&CPS Europe) (Bari, Italy: IEEE) pp 1–6
[22] Wu Q and Lin H 2019 Short-Term Wind Speed Forecasting Based on Hybrid Variational Mode Decomposition and Least Squares Support Vector Machine Optimized by Bat Algorithm Model *Sustainability* **11** 652
[23] Taha A, Nazih N and Makeen P 2025 Wind speed prediction based on variational mode decomposition and advanced machine learning models in zaafarana, Egypt *Sci Rep* **15** 15599