IJMT. Vol. 22, Iss. 2, 2026. (48-59)

International Journal of Maritime Technology

Journal homepage: ijmt.ir

Operational Forecasting for an Offshore Wind Turbine: Benchmarking Data-Driven against Physics-Informed Machine Learning

Kimia Nazarizadeh[1], **Hashem Nowruzi**[2*]

[1] *M.Sc Student, Babol Noshirvani University of Technology, Babol, Iran; k.nazarizadeh@outlook.com*
[2] *Assistant Professor, Babol Noshirvani University of Technology, Babol, Iran; h.nowruzi@nit.ac.ir*

ARTICLE INFO

Article History:
Received: 20 Nov 2025
Last modification: 18 Apr 2026
Accepted: 21 Apr 2026
Available online: 22 Apr 2026

Article type:
Research paper

Keywords:
Floating Offshore Wind Turbine (FWOT)
Physics-Informed Neural Networks (PINN)
Gradient Energy Theory (GET)
Digital Twin
Hybrid ML

ABSTRACT

Accurate forecasting of operational parameters is essential for predictive maintenance and digital twinning of offshore wind turbines. Using a unique dataset from the Levenmouth 7 MW demonstration turbine, we compare a purely data-driven stacked ensemble model (StackedRidge) with a novel physics-informed neural network (GET-PINN) that incorporates the Energy Gradient parameter (K) from Energy Gradient Theory (GET). The StackedRidge model achieves superior predictive accuracy (RMSE = 0.2976, R^2 = 0.9731) for barometric pressure signals. In contrast, the GET-PINN provides valuable physics-aware diagnostics by jointly estimating the flow instability parameter K, supporting the detection of phenomena such as vortex-induced vibrations (VIV), albeit with higher forecasting error. These results highlight the complementary strengths of the two approaches: the stacked ensemble for high-fidelity point forecasting and the GET-PINN for interpretable, physics-guided maintenance decision support in operational wind farm digital twins.

ISSN: 2645-8136

DOI: http://dx.doi.org/10.66224/ijmt.22.2.48

Copyright: © 2026 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license [https://creativecommons.org/licenses/by/4.0/]

[Downloaded from ijmt.ir on 2026-11-08]

1. Introduction

The world is shifting toward sustainable energy, and offshore wind power plays a central role. Offshore wind farms, particularly those in deeper waters, must maximize energy output while keeping operational expenditures (OPEX) low. Predictive maintenance is essential to achieve this balance [1, 2]. Accurate forecasting of structural and environmental parameters is a key element. It can provide early warnings of damaging conditions, such as vortex-induced vibrations (VIV) or aero-hydrodynamic instabilities [3, 4].

The rise of digital twins has created a strong demand for reliable forecasting models. These models must be high-fidelity, causal, and interpretable, and they must run efficiently on edge devices for real-time monitoring [5, 6]. Purely data-driven machine learning (ML) approaches, including tree-based ensembles and deep neural networks, deliver excellent predictive accuracy on time-series data [7, 8]. However, they often behave as "black boxes." They do not incorporate fundamental physical principles. This limitation reduces their reliability and generalizability, especially during novel or extreme events absent from the training data [9, 10]. Physics-Informed Neural Networks (PINNs) offer a solution. They embed physical laws directly into the training process through custom loss functions [11, 12]. This approach is particularly valuable for complex fluid-structure interaction problems in offshore engineering. It enforces physical consistency and improves extrapolation beyond the training data [13, 14]. One such physical principle is the Energy Gradient Theory (GET) and its associated Energy Gradient (*K*) parameter, which is a robust indicator of flow instability and transition to turbulence [15, 16].

A promising remedy comes from Hua-Shu Dou's energy-gradient theory (GET), which links the onset of turbulence directly to a simple parameter *K*.

$$K = \frac{\partial E/\partial y}{\partial E/\partial x} \quad (1)$$

Where E is the total mechanical energy, x is the streamwise coordinate, and y is the transverse direction. The parameter K represents the ratio of the energy gradient in the transverse direction (which tends to amplify disturbances) to the streamwise energy gradient (which resists them through viscous losses). When K exceeds approximately 385, the flow becomes unstable. This triggers turbulent bursts, vortex shedding, and the damaging vibrations and fatigue cycles that commonly affect offshore structures.

The main objective of this study is to benchmark purely data-driven machine learning against physics-informed approaches for real-time operational forecasting of offshore wind turbines, using data from the Levenmouth 7 MW demonstration turbine. The primary novelty lies in developing and evaluating GET-PINN, a compact, edge-deployable Physics-Informed Neural Network that integrates the Energy Gradient Theory (GET) stability parameter K both as an engineered feature and as a physics-based regularization term in the loss function. Unlike conventional PINNs that enforce governing PDEs, this work introduces the use of a local flow-instability criterion (K), highly sensitive to transitional and turbulent phenomena relevant to vortex-induced vibrations (VIV), directly into a time-series forecasting model. By comparing GET-PINN with a high-accuracy stacked ensemble (StackedRidge), the study reveals a clear trade-off: data-driven ensembles achieve superior point-forecasting accuracy for strongly auto-correlated signals, while the physics-informed model provides interpretable diagnostics of flow-induced structural risks. This dual assessment, combined with a reproducible feature-engineering pipeline and an edge-suitable physics-informed loss formulation, lays a practical foundation for hybrid forecasting systems in offshore wind farm digital twins.

2. Literature Review

2.1. Offshore Wind Energy and the Digital Twin Paradigm

Offshore wind energy is a cornerstone of global renewable energy strategies. Operation and maintenance (O&M) accounts for a large share of lifetime costs [17]. Digital twins have become a key technology for optimizing O&M through condition-based monitoring and remaining useful life prediction [5, 18]. A core function of a digital twin is to process real-time operational data from platforms such as the Levenmouth POD to generate accurate forecasts and diagnostics [6].

2.2. Data-Driven Forecasting in Renewable Energy

Machine learning techniques for forecasting in renewable energy systems range from linear models and tree-based ensembles to deep learning models such as LSTMs. Ensemble methods, including stacking, often achieve superior performance by combining multiple base learners. Rigorous temporal

[Downloaded from ijmt.ir on 2026-11-08]

preprocessing, causal feature engineering, and avoidance of data leakage are critical in modern time-series ML practice, protocols followed in this study [7, 8, 19, 20, 21, 22, 23].

2.3. Physics-Informed Machine Learning (PINN)

Purely data-driven models can struggle with physical plausibility. PINNs address this by penalizing violations of physical laws during training [11, 12]. This approach has been adopted in fluid dynamics and renewable energy applications, including turbine wake modeling and structural health monitoring, showing improved performance in data-scarce regimes [13, 14, 24, 25].

2.4. Energy Gradient Theory (GET) and the Parameter (K)

The Energy Gradient parameter K is a validated indicator of laminar-to-turbulent transition and flow-induced instabilities. Its applicability has been demonstrated for complex flows around bluff bodies and oscillating structures relevant to offshore engineering [15, 16, 26, 27], and recent studies have begun incorporating such stability thresholds into data-driven VIV detectors [28, 29]. The present study computes *K* from operational sensor data and employs it as a primary physics-informed feature and training constraint, leveraging its sensitivity as a local, flow-condition–normalized Reynolds-type diagnostic for predicting and mitigating flow-induced failures [1, 5, 6, 11].

Table 1. *K* as a local, flow–condition–normalized Reynolds number [1, 5, 6, 11]

Application	Critical *K*	Observed Phenomenon
Canonical pipe flow	≈385	Laminar-turbulent transition: Flow instability characterized by the breakdown of laminar flow structures into turbulent eddies under adverse pressure gradients.
Offshore mooring–line VIV	>390	Vortex-induced fatigue in cables: Lock-in phenomenon causing resonant oscillations that lead to material fatigue and structural failure.
Jacket-joint cracking under waves	≈380	Stress concentrations from turbulent eddies: Cyclic loading from turbulent wake structures leading to stress concentration and fatigue cracking at tubular joints.
Pitch-bearing wear in storm conditions	>400	Aeroelastic instabilities in blade bearings: Low-amplitude oscillatory movements under starved lubrication conditions leading to false brinelling and wear.

2.5. Hybrid Modelling for Offshore Systems

Hybrid or grey-box modeling, which integrates physical principles with data-driven approaches, is increasingly recognized as an effective strategy for complex offshore systems [30, 31]. In offshore renewable energy, this is exemplified by coupled modeling of wind and wave energy converters [1, 32] and the fusion of computational fluid dynamics (CFD) with ML [33, 34]. The present study contributes to this trend by combining a data-driven LSTM with a physics-derived regularizer based on the GET parameter K, resulting in a compact and interpretable tool suitable for operational forecasting and physics-aware diagnostics in offshore wind digital twins.

3. Methodology

Data were obtained from the Platform for Operational Data (POD) of the Levenmouth 7 MW demonstration turbine. This section describes the dataset, preprocessing steps, feature engineering, modeling approaches, training protocols, and performance metrics. All procedures were designed with chronological ordering to ensure causality and reproducibility.

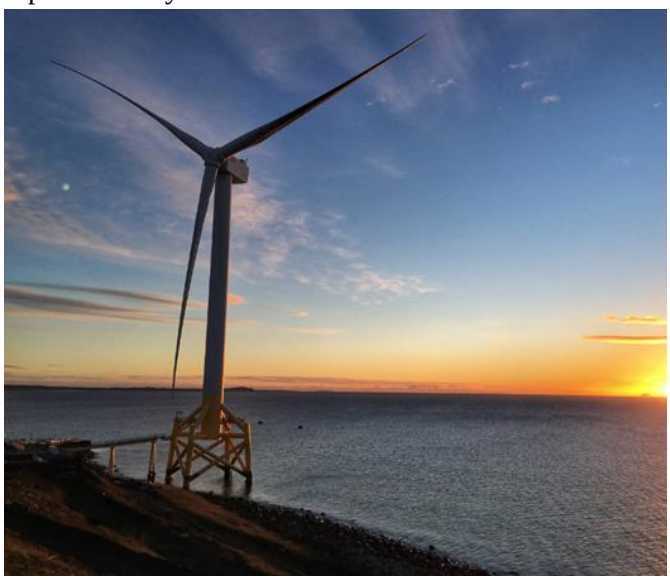


Figure 1. Levenmouth 7MW offshore wind demonstration turbine

3.1. Data Preparation and Feature Engineering

To capture both short-term fluctuations and mid-term trends, rolling statistics were calculated over 10-second and 60-second windows for each primary channel. In addition, the Energy Gradient parameter *K* was engineered to encode the local propensity for flow-induced instability. At each timestamp, the total mechanical energy *E* was estimated from pressure and kinetic contributions, and spatial gradients $\partial E/\partial x$ and $\partial E/\partial y$ were approximated by finite differences across paired sensor locations. The ratio *K* was then computed, and a binary indicator for $K>385$ was appended to the feature set. Table 2 summarizes all

[Downloaded from ijmt.ir on 2026-11-08]

input variables, including primary sensor means, rolling statistics, and engineered GET features.

Table 2. Input variables and engineered GET features

Feature Name	Type	Unit	Sensor Domain	Relevance
Barometer_2_Mean	Target Variable	hPa	Environmental	Proxy for pressure-related effects
Vibration_1_Mean	Input Feature	mm/s	Structural	Indicator of oscillatory loads
Vibration_2_Mean	Input Feature	mm/s	Structural	Redundant load sensor
Displacement_1_Mean	Input Feature	mm	Structural	Captures fatigue & motion range
Displacement_2_Mean	Input Feature	mm	Structural	Cross-check motion data
WindSpeed_1_Mean	Input Feature	m/s	Environmental	VIV excitation source
WaveHeight_Mean	Input Feature	m	Environmental	Hydrodynamic loading
Temp_1_Mean	Input Feature	°C	Environmental	Material stress sensitivity
EnergyGradient_K	Engineered	Dimensionless	Derived from GET	Flow instability indicator (K>385)
K_Threshold_Exceeded	Binary Label	-	Engineered	Used for classification

3.2. Preprocessing and quality control

Preprocessing emphasized causality and data integrity. The raw Excel records were parsed using the StartTime column as timestamps and reindexed to a consistent 10-minute resolution by averaging. Linear interpolation was applied only to short gaps (≤6 consecutive rows), while physically implausible values (e.g., negative standard deviations) were flagged as NaN and handled selectively. Target anomalies were winsorized using the 1st and 99th percentiles computed exclusively on the training partition, with identical bounds applied to the test set to prevent leakage. Numeric features were cast appropriately, and any columns with persistent sensor failures were excluded. A chronological 80/20 split defined the training and held-out test partitions, reflecting realistic operational deployment scenarios.

3.3. Feature Engineering

Feature engineering was designed to (a) capture short-term autocorrelation (autoregressive features), (b) represent local turbulence and sensor disagreement, and (c) incorporate a physics-informed GET indicator.

3.3.1. Temporal and (Autoregressive) AR features

Causal moving averages and standard deviations were computed over 1-hour (6 steps) and 6-hour (36 steps) lookback windows and shifted forward by one interval to preserve causality. Autoregressive lags of the target variable were also included. These features enable the models to exploit the strong autocorrelation observed in the barometric pressure signal.

3.3.2. Wind decomposition and turbulence proxies

Wind speed and direction measured by the tower anemometer were decomposed into Cartesian (u, v) components. The mean standard deviation across available anemometers was calculated, from which a turbulence intensity proxy was derived (with a small constant added in the denominator to avoid division by zero). These proxies help capture local flow variability relevant to vortex-induced vibrations (VIV).

3.3.3. GET (Energy Gradient) feature

The core physics-informed feature is the Energy Gradient parameter *K*, estimated from barometric pressure and paired anemometer wind speeds. Total mechanical energy per unit volume was approximated from pressure and kinetic contributions, with air density derived from the ideal gas law using measured temperature. Transverse and streamwise energy gradients were computed via finite differences, adopting a lateral sensor separation of $\Delta y \approx 2.5$ m (consistent with typical nacelle-mounted redundant anemometers) and a dynamic streamwise distance based on local wind speed and the 10-minute sampling interval.

The final *K* value was obtained as the ratio of transverse to streamwise energy gradients. A binary indicator for $K > 385$ (lagged by one step to remain causal) was added following the established GET instability criterion. This feature provides a sensitive, local diagnostic of flow instability and potential VIV risk.

For reproducibility, the detailed mathematical formulations of all features, including rolling statistics, wind decomposition, turbulence proxies, energy

[Downloaded from ijmt.ir on 2026-11-08]

approximations, gradient calculations, and the final definition of *K*, are provided in Appendix A.

3.3.4. Feature Selection and Scaling

An automatic procedure was applied to reduce dimensionality while retaining physically meaningful predictors. Numeric features (excluding timestamps, the raw target, and same-timestamp target-derived variables) with more than one unique value formed the candidate pool. Highly collinear features were removed using pairwise absolute Pearson correlations on the training partition (threshold $|r| > 0.95$), retaining the feature with higher variance in each pair. Remaining candidates were ranked by a composite score combining absolute Pearson correlation with the target anomaly and non-parametric mutual information. The top 12 features were retained and standardized using a robust scaler prior to model training.

3.4. Models and training

Three classes of models were developed and compared in the experimental pipeline. After empirical diagnostics, the final evaluation focused on the stacked Ridge surrogate and the GET-PINN.

3.4.1. Stacked linear surrogate (StackedRidge)

A two-level stacking architecture was employed to combine the strengths of tree-based and linear models while maintaining low computational cost. At level-0, a Random Forest regressor and a Ridge linear regressor were trained on the scaled top features. At level-1, a Ridge meta-learner combined the out-of-fold (OOF) predictions from the base models, using a small temporal holdout (last 20% of the training partition) to prevent future information leakage. Hyperparameters for the Random Forest (e.g., n_estimators ≈ 150–200, max_depth ≈ 10) were selected for robustness. This design produces a compact, interpretable surrogate well-suited for edge deployment in operational digital twins.

3.4.2. GET-PINN (physics-informed tiny LSTM)

The GET-PINN is a compact physics-informed neural network designed to jointly predict the primary target anomaly and the physics-derived Energy Gradient parameter **K**, with an optional binary instability alert. The architecture consists of:

- Input: Sliding windows of length S (typically S = 6, corresponding to 1 hour) of preprocessed feature vectors, including rolling statistics and the engineered **K** feature.
- Sequence encoder: A lightweight LSTM layer (or two stacked layers in larger variants) with hidden size H = 32 to capture temporal dependencies.
- Output heads: Two dense layers — one for regression of the target anomaly and one for regression of the **K** parameter. An optional sigmoid-activated unit can predict a binary instability flag when classification is required.

All input features were standardized using a robust scaler. The model was trained by minimizing a composite loss function that balances regression accuracy with physics consistency (detailed formulation provided in Appendix A). Weighting coefficients were selected via coarse grid search to prioritize accurate anomaly forecasting while enforcing physical regularization.

Optimization was performed using the Adam optimizer (learning rate 0.001) with mini-batch size 32. Early stopping and model checkpointing were applied on a temporal validation holdout, with typical convergence in 20–30 epochs. The compact design (hidden size H = 32) ensures low inference latency, making the GET-PINN suitable for near-real-time execution on resource-constrained edge devices.

Both models used the same preprocessed feature set. Table 3 summarizes the key characteristics of the models compared in this study, including their purpose, architecture, and trade-offs. Figure 2 illustrates the GET-PINN architecture and data flow.

Table 3: Selected ML models for comparison

Model	Purpose	Core Architecture	Key Parameters
Persistence	Causal baseline	Identity (lag-1 shift)	-
StackedRidge	High-accuracy deployment surrogate	Two-level stacking (RF + Ridge → Ridge meta-learner)	RF: n_estimators ≈ 150–200, max_depth ≈ 10; Ridge α = 1
GET-PINN	Physics-aware forecasting and diagnostics	Tiny LSTM encoder with dual regression heads (target + K)	LSTM hidden size = 32, sequence length = 6, λ_K ≈ 0.2

[Downloaded from ijmt.ir on 2026-11-08]

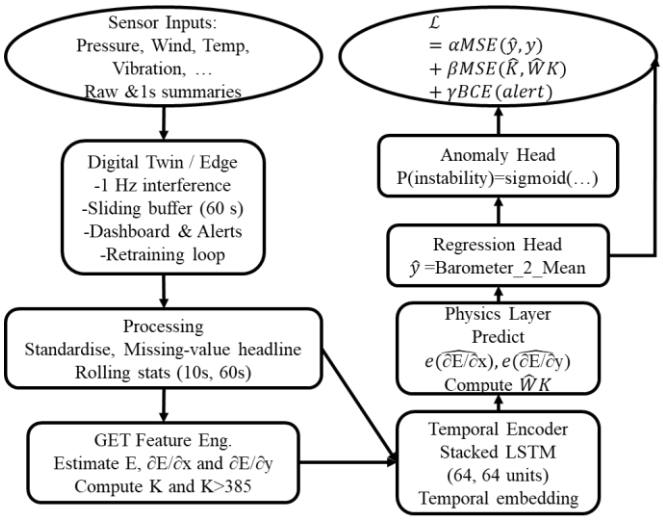


Figure 2: Illustrates the GET-PINN architecture and data flow.

3.5. Validation, hyperparameter selection and Performance metrics

Validation, hyperparameter tuning, and performance evaluation were conducted jointly under a strictly chronological protocol to simulate real-world deployment. The final 20% of the dataset formed the held-out test set, while a smaller temporal holdout (approximately 10–20% of the training data) was used for hyperparameter optimization and early stopping to prevent future information leakage. Hyperparameter searches remained lightweight, relying on small grids or conservative defaults (e.g., tuning n_estimators and max_features for Random Forest). Additional stability was verified using TimeSeriesSplit or simple chronological holdouts, with validation RMSE on the standardized target anomaly serving as the primary selection criterion.

Performance was reported on the unscaled physical target (Barometer_2_Mean) using standard regression metrics: Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and the coefficient of determination (R^2). Detailed definitions of these metrics are provided in Appendix A. Residual diagnostics—including histograms, Q–Q plots, and time-series analysis of residuals (see Figure 3)—were examined for bias, heteroscedasticity, and temporal clustering. For the GET-PINN, a secondary evaluation of the predicted K parameter was performed, with threshold exceedance scored using precision and recall where applicable. Final hyperparameters for the StackedRidge, Random Forest, Ridge base learners, and GET-PINN are summarized in Table 3. The overall methodological framework is illustrated in Appendix

B.

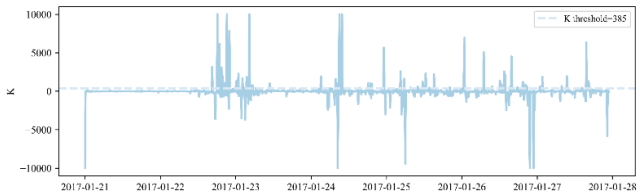


Figure 3: Energy-Gradient K (time series)

4. Results and Discussion

4.1. Summary of numerical results

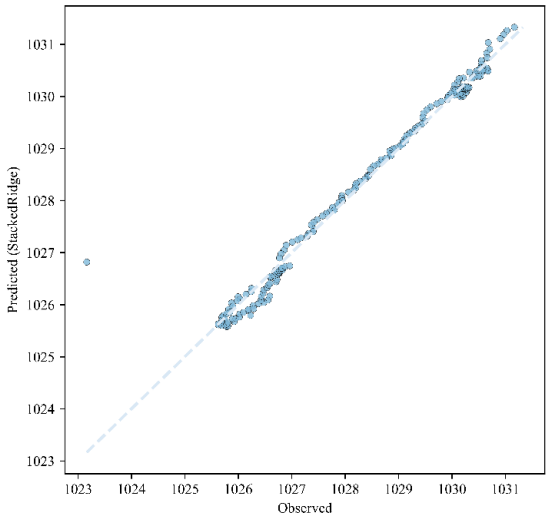
The models were evaluated on the held-out chronological test partition. Two simple baselines were computed for reference: a persistence forecast (previous timestep) and a causal rolling mean (one-hour rolling mean shifted forward by one interval). The key results are summarised in Table 4.

Table 4: Comparison of the models' metrics

Model/baseline	RMSE	MAE	R^2
Persistence baseline	0.3567	,	,
Rolling-mean baseline (1 h, causal)	0.3080	,	,
StackedRidge (stacked Ridge on RF+GB)	**0.2976**	**0.1353**	**0.9731**
GET-PINN (tiny LSTM + GET)	0.8779	0.7203	0.7679

All error metrics are in the same units as the target (Barometer_2_Mean). The stacked linear model (StackedRidge) achieved the best predictive performance by a clear margin: it reduced RMSE relative to the rolling-mean baseline (0.308 → 0.298) and obtained an R^2 of 0.9731 on the test set. The GET-PINN model produced a substantially larger RMSE and MAE, with R^2 = 0.7679.

(a)



[Downloaded from ijmt.ir on 2026-11-08]

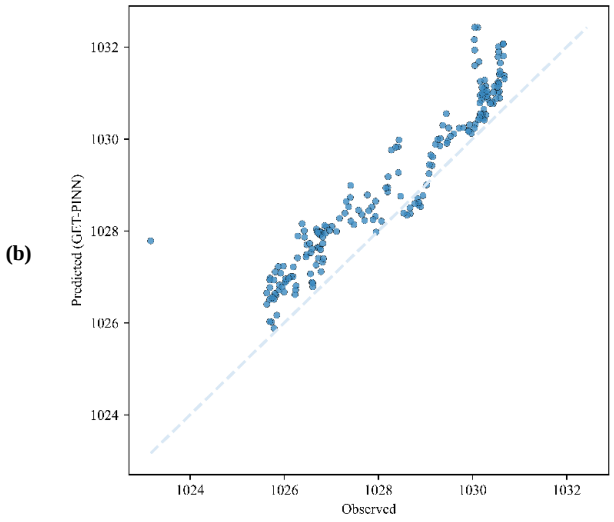


Figure 4: (a) Prediction vs. Observation: StackedRidge, and (b) GET-PINN

The scatter plots in Figure 4 illustrate that StackedRidge predictions align closely with the 45° line, indicating tight agreement and low error. The GET-PINN predictions show greater dispersion, particularly at higher values. Overall, StackedRidge delivers superior pointwise accuracy and consistency, while the GET-PINN, although more physically interpretable, exhibits larger variance and would benefit from further tuning for comparable numerical performance.

4.2. Interpretation and diagnostics

The strong performance of StackedRidge can be attributed to several dataset characteristics. The target (Barometer_2_Mean) is strongly autocorrelated and closely tracked by the co-located Barometer_1_Mean series. This makes the forecasting task effectively an autoregressive/sensor-agreement problem that linear combinations of strong base learners can approximate with high accuracy. The inclusion of engineered autoregressive lags and causal rolling features provided the stacked model with direct access to the most predictive signals, which the Ridge meta-learner weighted optimally. Additionally, Ridge regularization at the meta-level prevented overfitting and ensured robust predictions on the chronological test set.

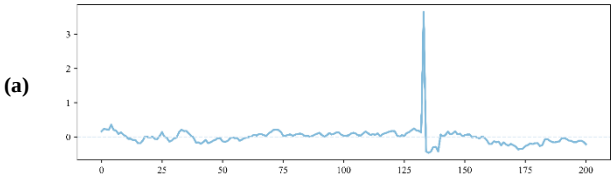
The GET-PINN was designed as a physics-informed sequence model that jointly predicts the target anomaly and learns the GET parameter K as an auxiliary task. Several factors likely explain its comparatively weaker numerical performance in this study:

1. The target is dominated by slowly varying pressure with high sensor redundancy. A large fraction of variance is trivially captured by lag features and sensor agreement, which linear models exploit effectively.
2. Noise in the engineered K parameter, arising from sparse sensor pairs, assumed geometry ($\Delta y \approx 2.5$ m), and finite-difference approximations, can introduce a suboptimal trade-off in the composite loss.
3. The compact LSTM architecture with short sequences (S = 6) may limit capture of longer-term dynamics, and the modest dataset size (~1k rows) is challenging when fitting both regression and physics outputs simultaneously.
4. The chosen loss weighting may have placed relatively high emphasis on physics consistency given the inherent noise in the K supervision signal.

The derived K values exhibited substantial variability (standard deviation ≈ 130–150, coefficient of variation ≈ 0.55–0.65), with typical stable-condition values ranging from 50 to 300 and intermittent spikes above the critical threshold of 385. This noise in the physics signal likely contributed to the higher forecasting error of the GET-PINN. These results do not indicate that the GET-PINN approach is inherently inferior; rather, the current experimental conditions favored the lightweight stacked ensemble for this particular target.

4.3. Residual and error structure

Residual diagnostics (Figure 5) show that StackedRidge residuals are tightly concentrated around zero with low variance, consistent with its excellent RMSE and MAE. In contrast, GET-PINN residuals display heavier tails and more pronounced time-clustered deviations, suggesting occasional systematic deviations during transient periods. It is recommended that these episodes be cross-referenced with *K(t)* exceedance periods to assess whether the GET-PINN still captures physical instability signals despite higher pointwise error.



[Downloaded from ijmt.ir on 2026-11-08]

(b)

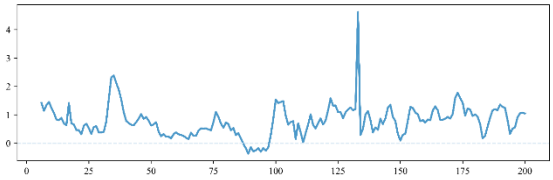
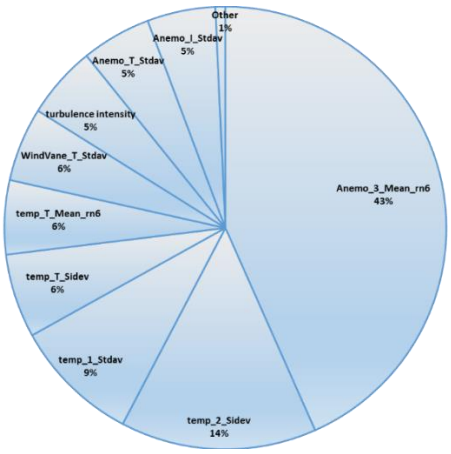


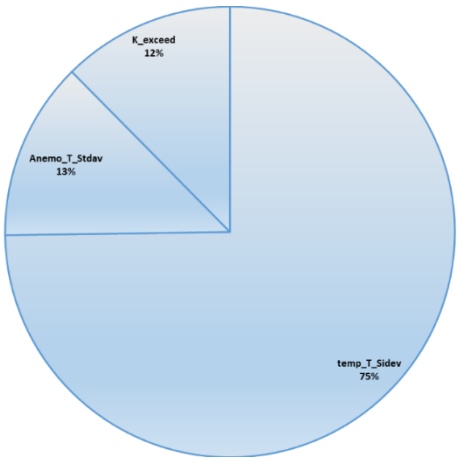
Figure 5: (a) Residuals over test index: StackedRidge, and (b) GET-PINN

4.4. Feature Importance and Practical implications for predictive maintenance

Feature importance analysis (Figure 6) highlights clear differences between the models. StackedRidge distributes importance broadly across sensors (e.g., Anemo_3_Mean_rs6 contributing ~41.7%, with substantial contributions from temperature standard deviation and turbulence proxies), reflecting a balanced multi-sensor approach. The GET-PINN, however, concentrates heavily on temp_T_Stdev (~74.7%), with secondary contributions from Anemo_T_Stdev and K_exceed (~12–13% each), indicating greater sensitivity to dominant features and physics signals.



(a)



(b)

Figure 6: (a) Feature Importance Pie Charts: StackedRidge, and (b) GET-PINN

Practically, the two models are complementary. StackedRidge is ideal for high-accuracy short-term forecasting and alarm generation with minimal computational cost, making it well-suited for edge deployment. The GET-PINN provides valuable physics-aware diagnostics, particularly for assessing VIV risk through estimated K values, even if its point forecasting accuracy is lower. A hybrid strategy is therefore recommended: use StackedRidge for precise point forecasts and quick alerts, while employing the GET-PINN in parallel for interpretable, physics-guided maintenance decision support within the digital twin framework.

4.5. Robustness checks and ablation suggestions

To strengthen the conclusions and further improve GET-PINN performance, the following experiments are recommended:To strengthen the conclusions and improve GET-PINN performance, the following experiments are recommended:

1. Refine K estimation using actual sensor geometries (precise Δy), alternative numerical derivatives, denoising techniques (low-pass filtering or wavelet shrinkage), and sensor fusion. Quantify K noise via signal-to-noise ratio (SNR) before using it as a supervision signal.
2. Perform a systematic grid search on PINN loss weights (α for regression, β for K-MSE, γ for binary exceedance) and select operating points from the Pareto front balancing RMSE and K-MSE.
3. Test longer sequences (S = 12–24), alternative architectures (GRU or Transformer), and transfer learning or pretraining strategies.
4. Conduct ablation studies: train GET-PINN without the K loss term and evaluate StackedRidge without lag/rolling features.
5. Incorporate uncertainty quantification (quantile regression for stacking; dropout or Bayesian methods for PINN) to support maintenance scheduling.
6. Perform operational validation on out-of-sample data containing extreme events, faults, and maintenance records.

5. Conclusion

This study introduces a practical hybrid framework for operational forecasting in offshore wind turbines, combining advanced feature engineering with both

[Downloaded from ijmt.ir on 2026-11-08]

data-driven and physics-informed machine learning. Using high-frequency data from the Levenmouth 7 MW demonstration turbine, the analysis reveals two complementary strengths with direct relevance to digital twin development.

The stacked ensemble model (StackedRidge) delivered exceptional point-forecasting performance, achieving an RMSE of 0.2976 and R^2 of 0.9731 for barometric pressure. These results demonstrate that, for strongly autocorrelated signals with high sensor redundancy, carefully engineered features paired with ensemble methods can capture nearly all predictable variance while remaining lightweight and edge-deployable.

In parallel, the novel GET-PINN provides valuable physics-aware diagnostics by jointly estimating the Energy Gradient parameter **K**—a sensitive indicator of flow instability and vortex-induced vibration (VIV) risk. Although its numerical accuracy is lower, the model transforms forecasting from a black-box prediction into an interpretable tool that reveals *why* damaging conditions may arise.

The optimal strategy is therefore not to select one approach over the other, but to deploy them together: StackedRidge for high-fidelity, low-latency forecasting and real-time alarms, and GET-PINN as a parallel physics-diagnostic module that flags periods where **K** exceeds the critical threshold of 385. This hybrid system merges data-driven precision with physical insight, delivering a more robust and trustworthy decision-support tool for predictive maintenance in offshore wind digital twins.

While the Levenmouth dataset offers unique real-world value, its modest size and the approximations inherent in deriving **K** from sparse sensors limit generalizability to extreme events. Future work will refine **K** estimation using precise sensor geometries, optimize PINN loss weighting, and validate the hybrid framework on larger, multi-turbine datasets containing severe conditions. These advances will further strengthen the operational readiness of physics-informed digital twins for next-generation offshore wind farms.

6. Author Contribution

Kimia Nazarizadeh – Contributed to conceptualization, methodology (development of the StackedRidge and GET-PINN modelling approaches), software implementation, data curation and preprocessing of the Levenmouth 7 MW demonstration turbine dataset, formal analysis, validation, visualization, writing the original draft, and review and editing.

Hashem Nowruzi – Contributed to conceptualization, methodology) theoretical guidance and Energy Gradient Theory integration). Validation of physics-informed outputs and L parameter estimation, writing, review and editing, and supervision.

All authors have read and approved the final manuscript and agree with the order of authorship.

7. Data Availability Statement

Authors should specify whether the data are shareable, where they can be accessed, or, if there are any restrictions, the reasons for those limitations.

8. Publisher's Note

Any views and information expressed in this publication are those of the authors and do not necessarily reflect the views of the publisher

9. References

1- D. Zhang, Y. Si, H. Liu, L. Wang, H. Huang, Y. Sun, and J. Li, "*A coupled numerical framework for hybrid floating offshore wind turbine and oscillating water column wave energy converters*," Energy Conversion and Management, vol. 267, 2022.

2- S. Feng, L. Song, J. Zhou, Z. Yang, Y. S. Choo, T. Sun, and S. Wang, "*Multi-Scale CNN for Health Monitoring of Jacket-Type Offshore Platforms with Multi-Head Attention Mechanism*," Journal of Marine Science and Engineering, 2025.

3- A. Ijaz and S. Manzoor, "*Vortex induced vibration prediction through machine learning techniques*," AIP Advances, vol. 14, no. 11, Art. no. 115025, Nov. 2024.

4- D. Wan et al., "*Floating Offshore Wind Farm Projects in China: Part I — Coupled Aero-Hydro-Elastic Behaviors*," in Proceedings of the ISOPE International Conference, 2024.

5- K. Y. H. Lim, P. Zheng, and C.-H. Chen, "*A state-of-the-art survey of Digital Twin: techniques, engineering product lifecycle management and business innovation perspectives*," Journal of Intelligent Manufacturing, vol. 31, no. 6, pp. 1313–1337, Aug. 2020. DOI: 10.1007/s10845-019-01512-w.

6- A. Fuller, Z. Fan, C. Day, and C. Barlow, "*Digital Twin: Enabling Technologies, Challenges and Open Research*," IEEE Access, 2020. (review article / IEEE Access special issue)

7- M. Masoumi, "*Machine learning solutions for offshore wind farms: a review of applications and impacts*," Journal of Marine Science and Engineering, vol. 11, no. 10, Art. no. 1855, 2023. DOI: 10.3390/jmse11101855.

8- N. Wang and Z. Li, "*A stacking-based short-term wind power forecasting method by CBLSTM and ensemble learning*," Journal of Renewable and Sustainable Energy, vol. 14, no. 4, 2022.

9- J. Willard, X. Jia, S. Xu, M. Steinbach, and V. Kumar, "*Integrating Physics-Based Modeling with Machine Learning: A Survey*," arXiv preprint arXiv:2003.04919, 2020.

10- A. Karpatne, G. Atluri, J. F. Faghmous, et al., "*Theory-Guided Data Science: A New Paradigm for Scientific Discovery from Data*," IEEE Transactions on Knowledge and Data Engineering, vol. 29, no. 10, pp. 2318–2331, Oct. 2017.

11- M. Raissi, P. Perdikaris, and G. E. Karniadakis, "*Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations*," Journal of Computational Physics, vol. 378, pp. 686–707, 2019.

[Downloaded from ijmt.ir on 2026-11-08]

12- G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang, "*Physics-informed machine learning*," Nature Reviews Physics, vol. 3, no. 6, pp. 422–440, 2021. DOI: 10.1038/s42254-021-00314-5.
13- S. Cai, Z. Wang, S. Wang, P. Perdikaris, and G. E. Karniadakis, "*Physics-Informed Neural Networks for Heat Transfer Problems*," Journal of Heat Transfer, vol. 143, no. 6, 060801, 2021. DOI: 10.1115/1.4050542.
14- L. Wang et al., "*Dynamic wake field reconstruction of wind turbine through physics-informed neural network and sparse LiDAR data*," Energy, 2024; DOI: 10.1016/j.energy.2024.130401.
15- H.-S. Dou, "*The mechanism of flow instability and transition to turbulence*," International Journal of Non-Linear Mechanics, vol. 41, pp. 512–517, 2006.
16- H. Nowruzi, H. Ghassemi, and S. S Nourazar. "*Study of the effects of aspect ratio on hydrodynamic stability in curved rectangular ducts using energy gradient method*." Engineering Science and Technology, an International Journal , Vol. 23, no. 2, 2020.
17- R. Perveen, N. Kishor and S. R. Mohanty, "*Off-shore wind farm development: Present status and challenges*," Renewable and Sustainable Energy Reviews, vol. 29, pp. 780–792, 2014. DOI: 10.1016/j.rser.2013.08.108.
18- X. Zhang, J. Tao, and A. Noshadravan, "*Probabilistic digital twin for reliability-based maintenance optimization of offshore wind turbines*," Renewable Energy, vol. 256, Art. 123777, 2025/2026 (publisher listing shows article in Vol. 256).
19- Z. Liu, H. Guo, Y. Zhang, and Z. Zuo, "*A comprehensive review of wind power prediction based on machine learning: models, applications, and challenges*," Energies, vol. 18, no. 2, 2025.
20- J. Zhang, F. Luo, X. Quan, Y. Wang, and C. Zhang, "*Improving wave height prediction accuracy with deep learning*," Ocean Modelling, vol. 188, Art. 102312, Apr. 2024.
21- A. T. Nguyen, Y. Ahn, S. Park, S. Park, and D. H. Pham, "*Meta-learning regression framework for energy consumption prediction in retrofitted buildings: A case study of South Korea*," Journal of Building Engineering, vol. 96, 110403, 2024. DOI: 10.1016/j.jobe.2024.110403.
22- A. Alexandrov, K. Benidis, M. Bohlke-Schneider, V. Flunkert, J. Gasthaus, T. Januschowski, D. C. Maddix, S. Rangapuram, D. Salinas, J. Schulz, L. Stella, A. C. Türkmen, and Y. Wang, "*GluonTS: Probabilistic and Neural Time Series Modeling in Python*," Journal of Machine Learning Research, vol. 21, 2020.
23- J. A. Miller et al., "*A survey of deep learning and foundation models for time series forecasting*," arXiv preprint arXiv:2401.13912, Jan. 2024.
24- S. L. Brunton, B. R. Noack and P. Koumoutsakos, "*Machine learning for fluid mechanics*," Annual Review of Fluid Mechanics, vol. 52, pp. 477–508, 2020.
25- E. J. Cross, S. J. Gibson, M. R. Jones, D. J. Pitchforth, S. Zhang, and T. J. Rogers, "*Physics-informed machine learning for structural health monitoring*," in Structural Health Monitoring Based on Data Science Techniques, Springer, Cham, 2021. (book chapter)
26- M. Xiao, H.-S. Dou, C. Wu, Z. Zhu, X. Zhao, S. Chen, H. Chen and Y. Wei, "*Analysis of vortex breakdown in an enclosed cylinder based on the energy gradient theory*," European Journal of Mechanics — B/Fluids, vol. 71, pp. 66–77, 2018. DOI: 10.1016/j.euromechflu.2018.03.013.
27- A. Chizfahm and R. D. K. Jaiman, "*Data-driven stability analysis and near-wake jet control for the vortex-induced vibration of a sphere*," Physics of Fluids, vol. 33, no. 4, 044104, Apr. 2021.
28- 임도형 (Im Do-hyung), "*Deep learning-based detection technology for vortex-induced vibration of a ship's propeller*," Ph.D. diss., Seoul National University, 2022.
29- A. P. Mentzelopoulos, J. del Águila Ferrandis, S. Rudy, T. Sapsis, M. S. Triantafyllou, and D. Fan, "*Data-driven prediction and study of vortex-induced vibrations by leveraging hydrodynamic coefficient databases learned from sparse sensors*," Ocean Engineering, vol. 266, 112833, 2022. DOI: 10.1016/j.oceaneng.2022.112833.
30- D. Zhang, C. Tan, A. K., and co-authors, "*Hybrid physics-based and data-driven modeling for bioprocess online simulation and optimization*," Biotechnology and Bioengineering, vol. 116, no. 11, pp. 2800–2814, Nov. 2019. DOI available via the publisher.
31- J. Hauth, *Grey-box modelling for nonlinear systems*, Ph.D. dissertation, Technische Universität Kaiserslautern, 2008.
32- M. J. Muliawan, M. Karimirad and T. Moan, "*Dynamic response and power performance of a combined spar-type floating wind turbine and coaxial floating wave energy converter*," Renewable Energy, vol. 50, pp. 47–57, 2013.
33- X. Zhou, J. Zhang, K. Feng, Z. Qiao, Y. Wang, and L. Shi, "*Machine-learning-assisted design of flow fields for proton exchange membrane fuel cells*," Journal of Power Sources, 2025, Art. 235753. DOI: 10.1016/j.jpowsour.2024.235753.
34- J. Tao and G. Sun, "*Application of deep learning based multi-fidelity surrogate model to robust aerodynamic design optimization*," Aerospace Science and Technology, vol. 92, pp. 722–737, Jul. 2019. DOI: 10.1016/j.ast.2019.07.002.
35- Offshore Renewable Energy Catapult, "*Levenmouth Research Wind Turbine Data: Turbine Information — Dimensions and Data*," Supergen Wind Hub / ORE Catapult technical sheet, 2016.
36- R. S. Hunter, B. M. Pedersen and T. F. Pedersen (eds.), *Recommended Practices for Wind Turbine Testing and Evaluation — Volume 11*: Wind Speed Measurement and Use of Cup Anemometry, IEA Wind Annex XI, 1st ed., 1999 (2nd print 2003).

Appendix A: Supplementary Equations

This appendix provides the complete mathematical formulations for the feature engineering steps and model components described in Section 3. All equations are presented for reproducibility.

A.1 Temporal and Autoregressive Features

For a generic sensor signal $x(t)$, the following causal features were computed:

[Downloaded from ijmt.ir on 2026-11-08]

- Causal rolling mean over a window of size *w*:

$$\overline{x}(t)\frac{1}{w}\sum_{i=1}^{w}x(t-i) \quad \text{(A.1)}$$

- Causal rolling standard deviation over a window of size *w*:

$$\sigma_x(t)=\sqrt{\frac{1}{w}\sum_{i=1}^{w}(x(t-i)-\overline{x}(t))^2} \quad \text{(A.2)}$$

- Autoregressive lags for the target variable *y(t)*:

$$y_{lag-k}(t)=y(t-k), \qquad k=1,2,3,\ldots \quad \text{(A.3)}$$

These features were calculated over 1-hour (6 steps) and 6-hour (36 steps) lookback windows and shifted forward by one interval to preserve causality.

A.2 Wind Decomposition and Turbulence Proxies

Wind speed and direction from the tower anemometer were decomposed into Cartesian components:

$$u(t)=WindSpeed(t).cos\big(\theta(t)\big),$$
$$v(t)=WindSpeed(t).sin\big(\theta(t)\big) \quad \text{(A.4)}$$

where *θ(t)* is the wind vane direction in radians.

The mean anemometer standard deviation across sensors was computed as:

$$\sigma_{anemo}(t)=\frac{1}{N}\sum_{i=1}^{N}\sigma_i\,(t) \quad \text{(A.5)}$$

A turbulence intensity proxy was then formed as:

$$TI(t)=\frac{\sigma_{anemo}(t)}{WindSpeed(t)+\epsilon}, \qquad \epsilon=0.01\ m/s \quad \text{(A.6)}$$

(to avoid division by zero).

A.3 Energy Gradient (GET) Feature

The total mechanical energy per unit volume at a sensor location was approximated as:

$$E(t)=p(t)+\frac{1}{2}\rho(t)u(t)^2 \quad \text{(A.7)}$$

Where *p(t)* is absolute pressure (Pa) and *u(t)* is local wind speed.

Air density was estimated using the ideal gas law:

$$\rho(t)=\frac{P(t)}{R.T(t)} \quad \text{(A.8)}$$

with (R=287.058 J·kg⁻¹·K⁻¹) (specific gas constant for dry air) and *T(t)* the absolute temperature (*K*).

The transverse energy gradient was approximated via finite differences across laterally separated anemometers:

$$\frac{\partial E}{\partial y}\approx\frac{E_2(t)-E_1(t)}{\Delta y}, \qquad \Delta y\approx 2.5 \quad \text{(A.9)}$$

The streamwise gradient used a temporal proxy scaled by local wind speed:

$$\frac{\partial E}{\partial x}\approx\frac{E(t)-E(t-\Delta t)}{v(t).\Delta t} \quad \text{(A.10)}$$

where Δt=600 s (10-minute sampling interval) and *v(t)* is the tower anemometer mean wind speed.

The Energy Gradient parameter ***K*** was then defined as the ratio:

$$K(t)=\frac{\partial E/\partial y}{\partial E/\partial x} \quad \text{(A.11)}$$

A binary threshold indicator was created as:

$$K_{exceeded}(t)=\begin{cases}1 & if\ K(t-1)>385\\ 0 & otherwise\end{cases} \quad \text{(A.12)}$$

(lagged by one step to remain causal).

A.4 Composite Loss Function for GET-PINN

The GET-PINN was trained by minimizing the following composite loss:

$$\mathcal{L}=\alpha.MSE_{target}+\beta.MSE_K+\gamma.BCE_{instability\ (optional)} \quad \text{(A.13)}$$

where

$$MSE_{target}=\frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i-y_i)^2 \quad \text{(A.14)}$$

$$MSE_K=\frac{1}{N}\sum_{i=1}^{N}(\hat{K}_i-K_i)^2 \quad \text{(A.15)}$$

and BCE denotes binary cross-entropy for the optional instability classification head. Typical weighting values were *α=1.0*, *β≈0.2*, with *γ* applied only when the classification objective was enabled.

A.5 Performance Metrics

For completeness, the regression metrics reported in the main text are defined as:

$$RSME=\sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i-y_i)^2} \quad \text{(A.16)}$$

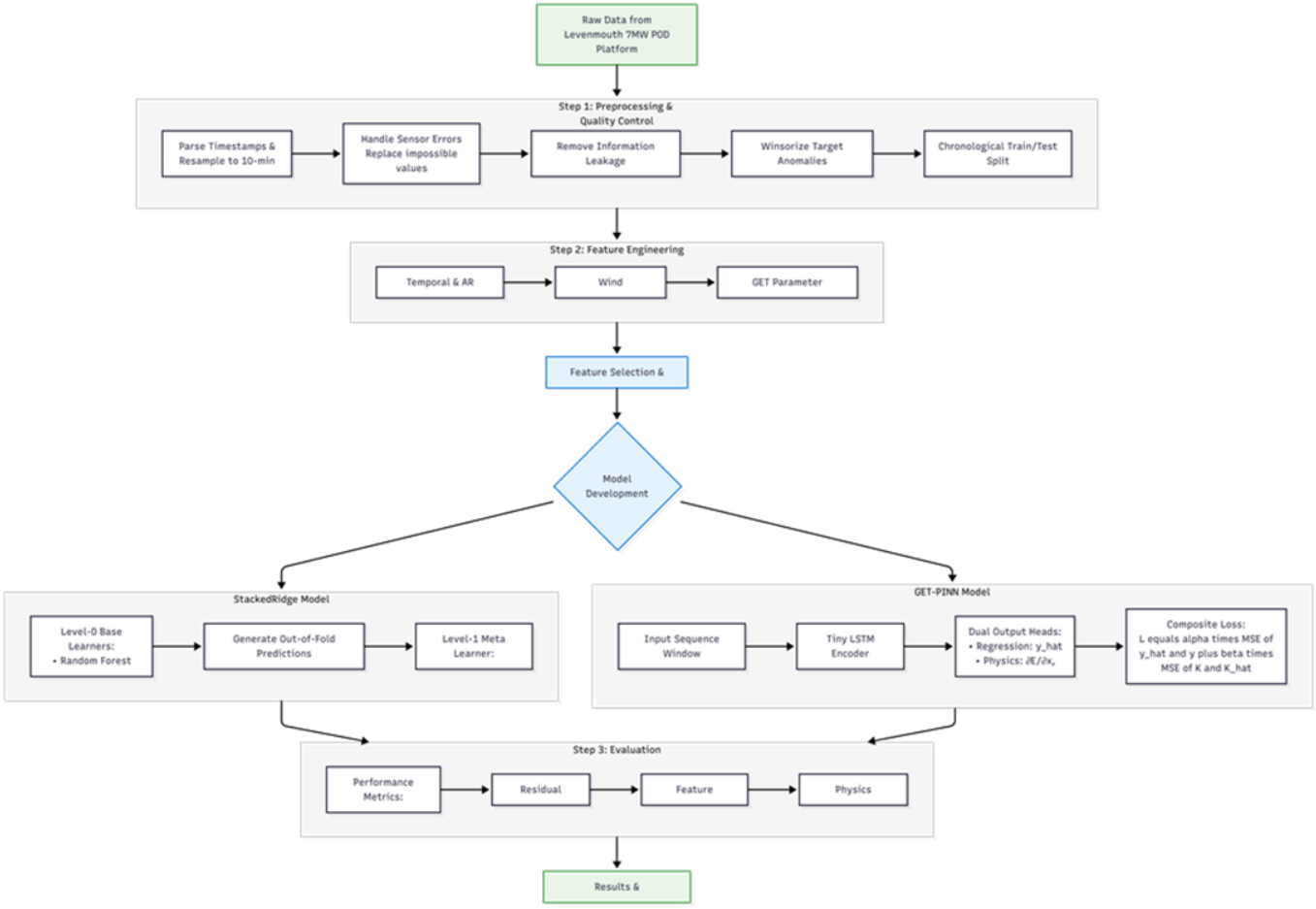
$$MAE=\frac{1}{N}\sum_{i=1}^{N}|\hat{y}_i-y_i| \quad \text{(A.17)}$$

$$R^2=1-\frac{\sum_i(\hat{y}_i-y_i)^2}{\sum_i(y_i-\bar{y})^2} \quad \text{(A.18)}$$

Appendix B: Methodological Framework

This appendix presents the overall methodological framework for hybrid data-driven and physics-informed forecasting of offshore wind turbine operational parameters developed in this study.

[Downloaded from ijmt.ir on 2026-11-08]



[Downloaded from ijmt.ir on 2026-11-08]