New Zealand Wind Speed and Power Potential Forecast: A case of Nelson

Wenyang Lyu
Auckland University of Technology
Auckland, New Zealand
wenyang.lyu@aut.ac.nz

Ramon Zamora
Auckland University of Technology
Auckland, New Zealand
ramon.zamora@aut.ac.nz

Zoey (Shuai) Zhou
Auckland University of Technology
Auckland, New Zealand
zoey.zhou@aut.ac.nz

Tek-Tjing Lie
Auckland University of Technology
Auckland, New Zealand
tek.lie@aut.ac.nz

***Abstract*—This paper introduces a novel framework for wind speed forecasting in Nelson, New Zealand, leveraging meteorological insights alongside advanced deep learning methods. A transformer-based model with multi-head attention, custom cyclical temporal encodings, domain knowledge features, and specialized loss functions is employed to predict 24-hour wind speeds at 10-minute intervals. The resulting forecasts attain a Mean Absolute Error (MAE) of 0.35 m/s and a Mean Squared Error (MSE) of 0.24, aligning competitively with leading forecasting techniques. To further enhance applicability, the model integrates air density calculations for precise wind power estimation using an actual wind turbine power curve, yielding an average power MAE of 13.27 W. The proposed approach shows considerable promise for renewable energy planning toward 100% renewable electricity in New Zealand.**

***Index Terms*—wind speed forecasting, transformer networks, feature engineering, renewable energy, meteorology**

I. Introduction

The increasing reliance on renewable energy sources necessitates accurate forecasting models to enhance grid stability and energy management. Wind energy, in particular, plays a crucial role in the global transition to sustainable power generation. However, the inherent variability of wind speeds presents a major challenge for energy planning, power dispatching, and grid reliability [1]. As wind power penetration grows, accurate short-term and long-term wind speed forecasting becomes critical to mitigate uncertainty and optimize integration into power grids.

Wind speed forecasting methods can be broadly categorized into statistical models, physical models, and machine learning-based models. Traditional statistical approaches, such as Autoregressive Integrated Moving Average (ARIMA) and Support Vector Regression (SVR), have been widely used due to their simplicity and interpretability [2]. However, they struggle to capture the nonlinear and chaotic behavior of wind speed, particularly in highly dynamic environments [3]. Physical models, such as the Weather Research and Forecasting (WRF) model, rely on numerical weather predictions (NWPs) but demand extensive computational resources and precise initial conditions, making them less practical for real-time forecasting [4].

Recent advancements in artificial intelligence (AI) and deep learning have significantly improved wind forecasting accuracy. Long Short-Term Memory (LSTM) networks and Convolutional Neural Networks (CNNs) have been extensively explored for their ability to capture temporal dependencies and extract spatial features from wind datasets [5]–[7]. Hybrid models, which combine traditional statistical techniques with deep learning, have shown promising results in improving short-term wind forecasting accuracy [8], [9]. However, many of these models suffer from limited generalizability across different locations due to variations in terrain complexity, atmospheric conditions, and data availability [10].

Recently, transformer-based architectures have emerged as a powerful alternative for time series forecasting, demonstrating superior performance over traditional sequence models such as LSTMs [11], [12]. Unlike recurrent neural networks (RNNs), transformers use self-attention mechanisms to efficiently capture long-range dependencies and fluctuating temporal patterns, making them well-suited for wind speed prediction [1], [10]. Studies have shown that transformer models outperform conventional deep learning models in terms of accuracy and computational efficiency, particularly for complex meteorological forecasting tasks [9], [13].

While several deep learning-based models have been proposed for wind forecasting, their practical implementation still faces challenges, including hyperparameter tuning, data sparsity, and uncertainty quantification. Many existing models require extensive fine-tuning, which can be computationally expensive and time-consuming. Additionally, missing data and measurement errors in meteorological stations can degrade forecasting accuracy [3]. Bayesian optimization and meta-learning approaches have been explored to automate hyperparameter tuning, improving the adaptability of deep learning models to diverse weather conditions [14].

This study proposes a transformer-based wind speed forecasting framework tailored for Nelson, New Zealand, a region where direct wind speed measurements are unavailable. To compensate for this limitation, meteorological data from nearby cities, Richmond and Greymouth are integrated, allowing the model to capture broader atmospheric influences. The key contributions of this work include:

- *Transformer-Based Forecasting:* A self-attention-based model is implemented to effectively capture both short-term fluctuations and long-term dependencies in wind speed data.
- *Multi-Source Data Fusion:* Richmond wind measurements and Greymouth pressure data are combined to enhance forecasting accuracy in Nelson.
- *Cyclical Feature Encoding:* Time-related features, such as time-of-day and seasonal variations, are encoded using sine-cosine transformations to preserve periodicity.
- *Hyperparameter Optimization:* Optuna is used to systematically tune key model parameters, significantly improving performance over baseline approaches [14].
- *Wind Power Estimation:* The model's wind speed predictions are adapted to real-world wind power calculations by incorporating air density corrections.

By leveraging attention mechanisms, multi-source data fusion, and domain-specific encoding techniques, this study aims to improve wind forecasting accuracy for energy planning and grid management. The following sections describe the dataset, preprocessing techniques, transformer model architecture, hyperparameter tuning strategy, and evaluation results.

II. Feature Engineering and Data Preprocessing

This section describes the process of acquiring, cleaning, and transforming meteorological data to forecast wind speed in Nelson as shown in Fig. 1. Because Nelson does not have its own wind measurements, we rely on records from the nearby town of Richmond, acquired from NIWA's climate-station-observation repository [15], spanning from 2015 to 2025 at a 10-minute interval.

A. Feature Selection

To identify the most informative predictors, we analyze both linear correlations and more general uncertainty-based metrics, such as mutual information, with respect to wind speed. This analysis reveals that rainfall and radiation contribute marginal predictive value and can be safely discarded. In contrast, local atmospheric variables—including pressure, temperature, humidity, and wind direction—are crucial for wind speed prediction.

Following recommendations from meteorological experts, we also incorporate pressure data from Greymouth, a West Coast town, to capture large-scale weather patterns influencing wind formation in Nelson. Since the original Greymouth dataset is recorded at 60-minute intervals, we apply a fill-forward method to expand it to 10-minute intervals, ensuring alignment with Nelson's dataset. Finally, the two datasets are synchronized and merged for further processing.

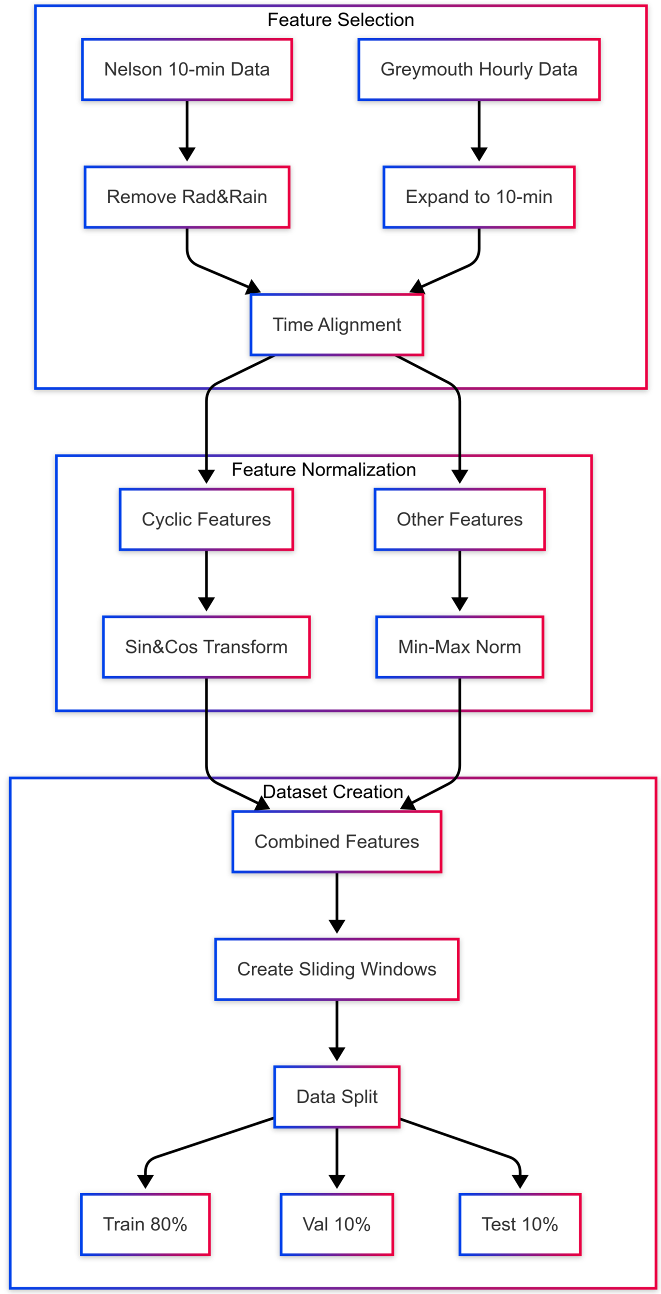

Fig. 1. Feature Engineering and Data Preprocessing Overview.

B. Feature Normalization

Among these features, some exhibit cyclical patterns, such as wind direction, day-of-year, and 10-minute time intervals within each day. To accurately capture these periodic characteristics, we apply cyclical encoding for time and wind direction, as illustrated in Fig. 2.

These variables are transformed into sine and cosine pairs to preserve their circular continuity. For example, the minute-of-day variable m is encoded as:

$$m_{\sin} = \sin\left(\frac{2\pi m}{1440}\right), \quad m_{\cos} = \cos\left(\frac{2\pi m}{1440}\right). \qquad (1)$$

Similar transformations are applied to the day-of-year and wind direction, ensuring that, for instance, $359°$ remains numerically close to $0°$.

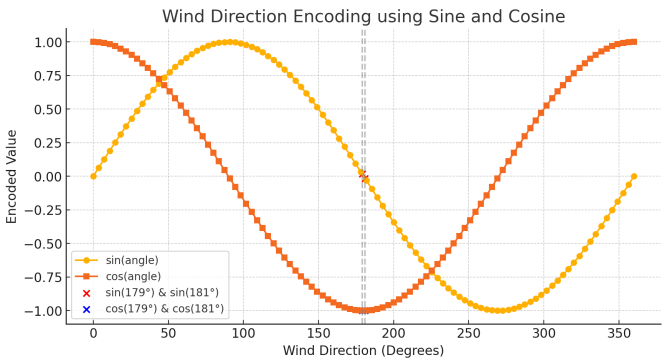

Fig. 2. Wind direction encoding using sine and cosine.

For the remaining features, we apply min-max normalization to scale them into a $[0, 1]$ range:

$$x_{\text{norm}} = \frac{x - x_{\min}}{x_{\max} - x_{\min}}, \qquad (2)$$

where x represents year number, pressure, temperature, humidity, and wind speed.

C. Sliding Window and Dataset Splitting

This study utilizes time-related features, weather variables, and wind speed itself, totaling 12 features over a 10-day period as input to forecast the next 24 hours of wind speed at a 10-minute interval.

To balance computational efficiency and forecasting accuracy, the first 7 days of 10-minute data are aggregated into hourly means, resulting in 24 samples per day. The following 3 days retain the original 10-minute resolution, yielding 144 samples per day to capture short-term variability.

Each input window consists of 7,200 features, corresponding to 12 input variables over 600 time steps (10 days). The output target includes 144 values, representing the next 24 hours of wind speed.

A sliding window approach with a step size of 3 time steps (30 minutes) is applied, meaning each successive window is shifted forward by half an hour. This method preserves temporal dependencies while maximizing the number of training examples.

Finally, the dataset is split into approximately 600,000 examples for training, 75,000 for validation, and 75,000 for testing. The test set is held out to ensure an unbiased evaluation of the model once training concludes.

III. Transformer Model Architecture

Having prepared the dataset with engineered and cyclically encoded features, this study implements a transformer-based neural architecture for wind speed forecasting as shown in Fig. 3. The model leverages self-attention mechanisms to capture both short and long-range dependencies in meteorological time series, which often exhibit periodic patterns and abrupt changes.

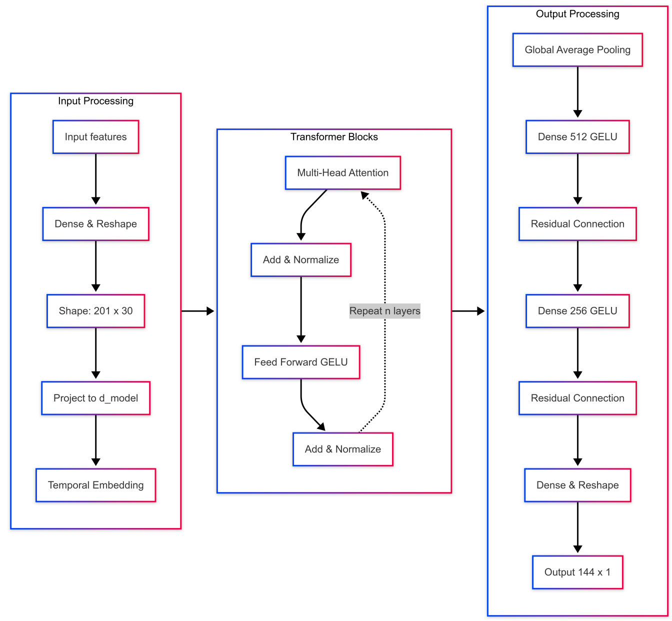

Fig. 3. Multi-head attention transfomer model architecture.

A. Model Design

The architecture consists of three main stages. First, an input projection layer reshapes the high-dimensional input (10-day sliding window) into a lower-dimensional representation. Positional embeddings are added to preserve sequence ordering. Next, multiple transformer blocks process the input, each comprising multi-head self-attention [**?**], feed-forward layers, and residual connections. Finally, a dense output layer produces the wind speed forecast for the next 24 hours at 10-minute intervals.

B. Self-Attention and Feed-Forward Processing

The model employs scaled dot-product attention to capture dependencies across time steps. Multi-head attention enables the network to focus on different temporal patterns simultaneously [**?**]. Each transformer block includes a feed-forward network applied independently to each time step, with layer normalization ensuring stable training.

C. Output Projection and Loss Function

The final output layer maps the learned representations to a forecast vector of size T_{pred} (e.g., 144 for 24-hour predictions at 10-minute intervals). Since wind power depends on the cube of wind speed, a custom loss function is applied, balancing mean squared error with power-based constraints to improve accuracy for energy applications.

D. Hyperparameter Flexibility

The transformer model is designed with flexible hyperparameters, including the model dimension d_m, feed-forward layer size d_{ff}, number of layers, and attention heads. These parameters are tuned to balance computational efficiency and predictive accuracy.

IV. Training with Optuna

The transformer model is trained using a supervised learning approach, where historical wind speed sequences are used to predict future values. The overall training pipeline, illustrated in Fig. 4, consists of data preprocessing, model initialization, optimization, and evaluation. A critical factor influencing the model's performance is the selection of optimal hyperparameters, as they directly affect the model's ability to learn temporal dependencies and generalize effectively.

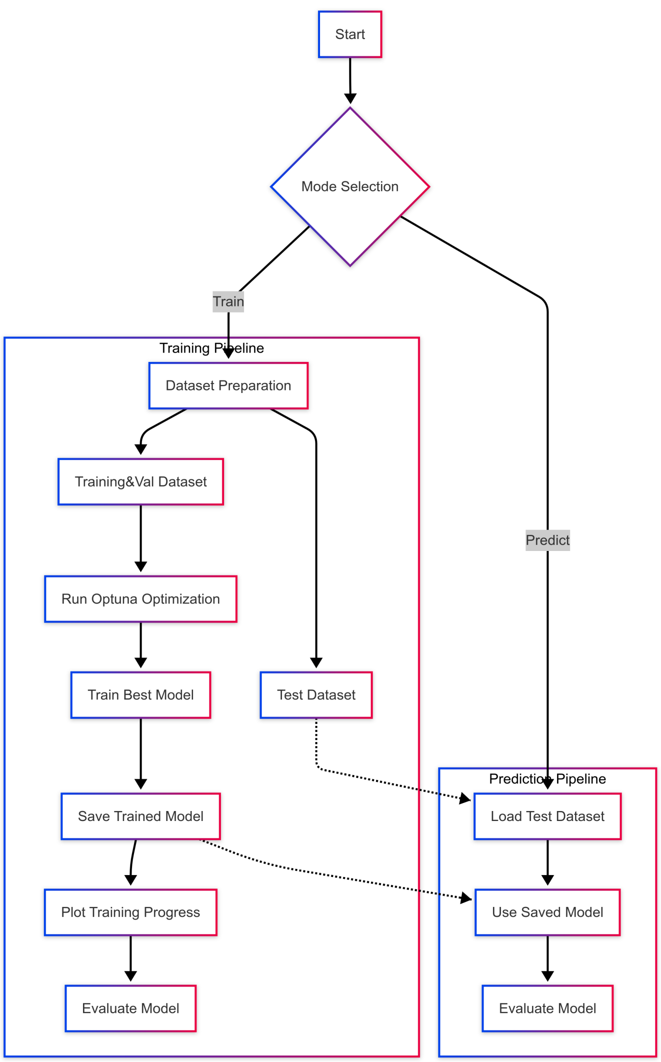

Fig. 4. Training pipeline with Optuna.

To systematically optimize the hyperparameters, this study employs Optuna, a powerful hyperparameter tuning framework. As shown in Fig. 5, Optuna automates the selection process by exploring a predefined search space and refining the choices based on validation performance.

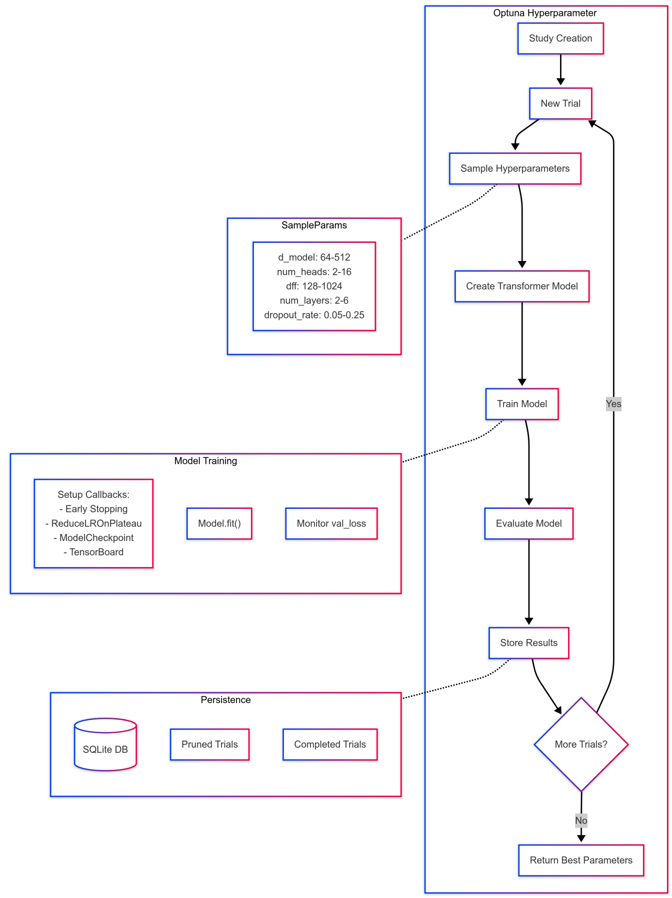

Fig. 5. Optuna optimization process.

A. Hyperparameter Optimization Strategy

Hyperparameter tuning is essential in achieving high forecasting accuracy. Poorly chosen values can lead to overfitting, underfitting, or inefficient training. Optuna defines an objective function that evaluates different hyperparameter configurations and iteratively refines the search space using a probabilistic model. The optimization process consists of four main steps: 1) proposing hyperparameter values, 2) training the model with these parameters, 3) evaluating performance on the validation set, and 4) adjusting future suggestions based on prior results.

The key transformer hyperparameters optimized in this study include:

- Model dimension d_{model} (64–512), number of attention heads (2–16), feed-forward layer size d_{ff} (128–1024), and number of layers (2 or 3).
- Regularization through dropout rate (0.01–0.25).

- Training parameters, including the loss function (`huber`), optimizer (`adamw`), and learning rate (10^{-5} to 10^{-3}, log scale).

To balance exploration and exploitation, the Tree-structured Parzen Estimator (TPE) is employed as the primary search algorithm. Additionally, Optuna's pruning mechanism is applied to terminate underperforming trials early, reducing computational overhead while maintaining search efficiency.

B. Incremental Training and Memory Optimization

During hyperparameter tuning, significant GPU memory constraints were observed due to TensorFlow's memory allocation behavior. To mitigate out-of-memory (OOM) errors, an incremental training strategy is implemented. This method allows Optuna to resume tuning from previously stored checkpoints, optimizing in smaller increments rather than reloading models from scratch. This ensures stable execution across multiple trials and facilitates an efficient search process without requiring excessive GPU resources.

Furthermore, to reduce optimization time, hyperparameter choices with minimal impact on model performance are systematically narrowed. For instance, transformer depth is constrained to 2 or 3 layers, as deeper models show diminishing improvements while increasing computational complexity. Similarly, the learning rate search space is refined to exclude unproductive ranges, optimizing efficiency without compromising accuracy.

C. Final Training Configuration

After conducting extensive hyperparameter optimization, the best-performing configuration is selected for final model training. The chosen parameters are:

- Model architecture: $d_{model} = 384$, $d_{ff} = 1024$, number of heads = 16, and number of layers = 2.
- Dropout rate: 0.0521.
- Optimizer: AdamW (switching to Adam if convergence stalls).
- Training setup: 2000 epochs, batch size of 64, early stopping patience of 60 epochs, and optimizer switching patience of 15 epochs.

V. Model Evaluation

A. Quantitative Performance Comparison

The trained transformer model is evaluated using standard error metrics, including Mean Absolute Error (MAE), and compared with existing forecasting methods.

Recent works in wind speed forecasting exhibit a range of MAEs from 0.05 m/s to beyond 1.0 m/s, depending on forecast horizon, terrain complexity, and data availability. As shown in previous studies (e.g., Table I), our result of 0.35 m/s positions the transformer-based approach in a competitive bracket, especially considering the unique challenge of cross-location data fusion.

TABLE I
Comparative MAE from Selected Literature

Method	Forecast Horizon	MAE (m/s)
EWT-GWO-RELM	10 min	0.0273
MCEEMD-MOSCA-WNN	10/30 min	0.0990
ARIMA/BPNN (Various)	Short-term	0.1260
Ours (Transformer)	**10 min (24 hr horizon)**	**0.35**
EWT-LSTM-ENN	Unspecified	0.51
FN Whale Algorithm	Short-term	1.05

B. Wind Power Adaptation

To assess the model's relevance to wind energy applications, the predicted wind speed is converted into power output using a real-world wind turbine in New Zealand. This study utilizes the *G128/4500 (Gamesa)*, which features a rated capacity of *4.5 MW* and a cut-in speed of *2 m/s*. The standard wind power curve for this turbine, assuming air density at standard conditions, is shown in Fig. 6.

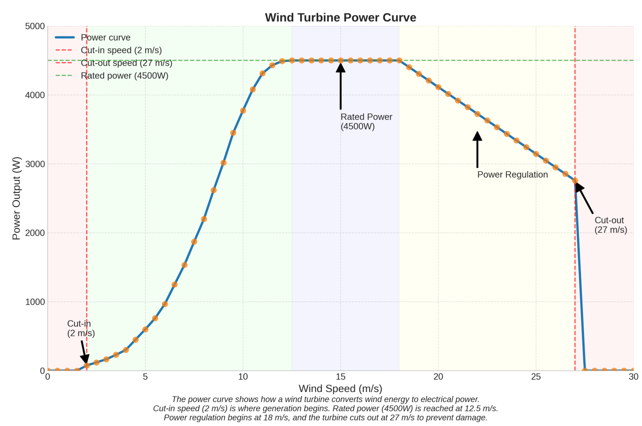

Fig. 6. Wind power curve under standard air density conditions.

Under ideal conditions, wind power is given by:

$$P = \frac{1}{2}\rho A v^3, \tag{3}$$

where ρ is air density, A is the rotor swept area, and v is wind speed. Since air density varies due to temperature, pressure, and humidity, real-world power output differs from standard assumptions.

To account for these variations, the real-world wind power output is adjusted based on the ratio of actual air density to standard air density:

$$P_{\text{real}} = \left(\frac{\rho_{\text{real}}}{\rho_{\text{std}}}\right) P_{\text{std}}, \tag{4}$$

where ρ_{real} is the actual air density, ρ_{std} is the standard air density (typically 1.225 kg/m^3 at sea level), and P_{std} represents the standard power output from the wind turbine's manufacturer-provided power curve.

This adjustment ensures that power predictions reflect real-world atmospheric conditions, improving accuracy in wind energy estimation.

C. Air Density Estimation

Air density plays a crucial role in wind power estimation, as it directly affects the available kinetic energy in the wind. Standard air density at sea level is approximately 1.225 kg/m^3, but it varies with altitude, temperature, humidity, and atmospheric pressure. To ensure accurate wind power calculations, this study applies an air density estimation method based on thermodynamic principles.

The air density ρ is computed using the ideal gas law for moist air:

$$\rho = \frac{P}{RT_v}, \tag{5}$$

where P is the atmospheric pressure, R is the specific gas constant, and T_v is the virtual temperature, which accounts for the effect of humidity. The virtual temperature is defined as:

$$T_v = T(1 + 0.61q), \tag{6}$$

where q is the specific humidity, representing the ratio of water vapor mass to the total mass of air. This correction is necessary because moist air is less dense than dry air at the same temperature and pressure.

To account for the effect of water vapor, the vapor pressure e is computed as:

$$e = \mathrm{RH} \cdot e_s(T), \tag{7}$$

where RH is the relative humidity and $e_s(T)$ is the saturation vapor pressure, commonly approximated as:

$$e_s(T) = 611 \exp\left(\frac{17.27T}{T + 237.3}\right), \tag{8}$$

with temperature T in degrees Celsius and pressure in Pascals. The final formulation for air density, incorporating both dry air and water vapor components, is given by:

$$\rho = \frac{P - e}{R_d T} + \frac{e}{R_v T}, \tag{9}$$

where $R_d = 287.058$ J/kg·K is the gas constant for dry air, and $R_v = 461.495$ J/kg·K is the gas constant for water vapor.

By integrating real-time meteorological data from NIWA climate stations, including temperature, humidity, and pressure, the model dynamically adjusts air density values to match actual atmospheric conditions. This ensures that wind power estimates remain aligned with real-world turbine performance.

D. Wind Power Forecasting Results

Table II and Fig. 7 present the estimated wind power outputs compared to the actual turbine data on a random day, accounting for the real-time air density factor, powered by the *G128/4500 (Gamesa)* model.

TABLE II
WIND SPEED AND POWER OUTPUT METRICS

Metric	Value
Actual Avg Wind Speed	4.60 m/s
Predicted Avg Wind Speed	4.63 m/s
Mean Absolute Error (MAE)	0.50 m/s
Root Mean Square Error (RMSE)	0.69 m/s
Mean Absolute Percentage Error (MAPE)	21.10%
Actual Avg Power Output	1119.39 W
Predicted Avg Power Output	1104.59 W
Absolute Power Error	14.80 W
Relative Power Error	1.32%

VI. Conclusions and Future Work

In this paper, a transformer-based framework for wind speed forecasting in Nelson, New Zealand, was proposed. Due to the absence of direct wind measurements, this study leveraged meteorological data from nearby Richmond and Greymouth, demonstrating that cross-location data fusion—particularly pressure differentials—enhances forecasting accuracy. The key contributions of this study include:

- **Cyclical Encoding of Temporal Features:** Sine-cosine transformations for time-of-day, day-of-year, and wind direction effectively capture the circular nature of these variables, reducing artificial discontinuities. However, this encoding method doubles the number of features, increasing computational complexity, necessitating further evaluation on whether the accuracy gains justify the additional processing overhead.
- **Transformer Architecture with Optimized Hyperparameters:** Multi-head attention layers effectively capture both short- and long-range dependencies. Optuna-based hyperparameter tuning played a crucial role in maximizing model performance, with key parameters such as d_{model}, number of heads, and dropout rate significantly influencing forecast accuracy.
- **Sliding Window and Multi-Resolution Data:** A hybrid input representation, merging a 7-day hourly aggregation with a 3-day 10-minute resolution, proved beneficial for balancing historical trends against short-term fluctuations.
- **Air Density Adjustments for Power Output:** By incorporating temperature, humidity, and pressure data, wind power estimates were refined to align with real-world turbine performance, improving applicability for energy forecasting.

A. Summary of Results

Empirical evaluation on the held-out test set produced a 0.35 m/s MAE and 0.24 MSE in wind speed prediction at a 24-hour horizon (10-minute intervals). When converted to wind power, the error was approximately 13.27 W in mean absolute terms, demonstrating viability for energy forecasting applications. This level of accuracy is competitive with recent literature, despite the added challenge of indirect measurements for Nelson.

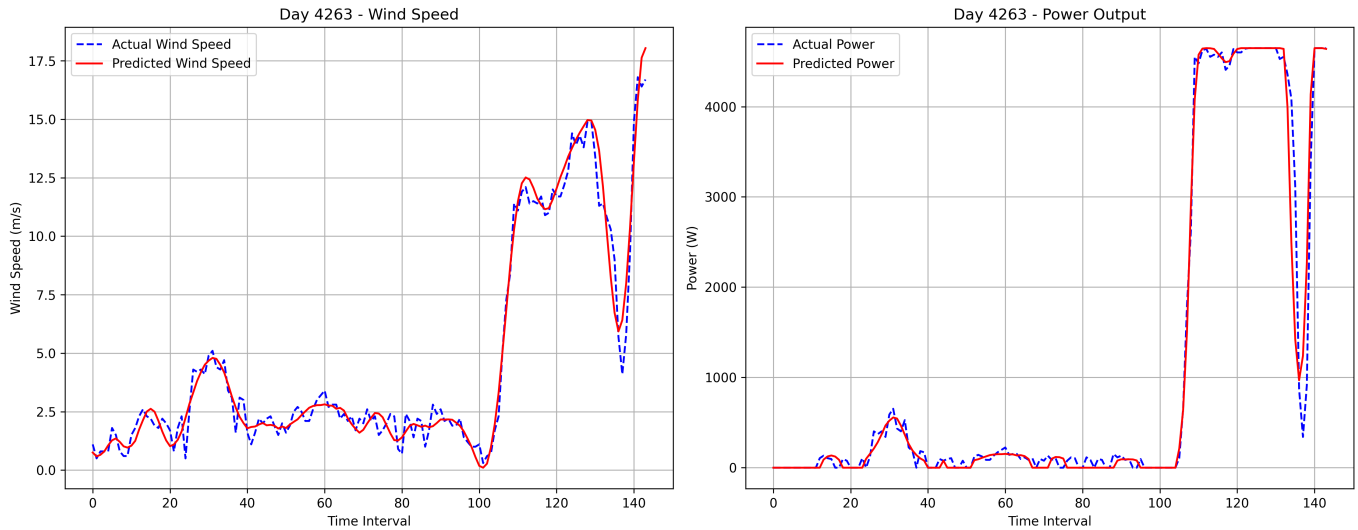

Fig. 7. Comparison of estimated and actual wind power outputs.

B. Limitations

- **Data Availability and Sensor Reliability:** The model assumes continuous data availability from Richmond and Greymouth. Prolonged sensor outages or data inconsistencies could impact reliability.
- **Extreme Meteorological Events:** While the transformer model effectively captures typical daily and seasonal variations, rare extreme weather patterns (e.g., cyclones) are underrepresented in the training data, potentially affecting prediction accuracy in such cases.
- **Computational Overheads:** The transformer's self-attention mechanism, while efficient compared to recurrent architectures for long sequences, requires significant computational resources, particularly during hyperparameter tuning and large sliding window training.

C. Future Directions

While this study demonstrates the feasibility of attention-based architectures for wind speed forecasting, several areas warrant further investigation:

1) **Evaluating Cyclical Encoding Overhead:** The sine-cosine transformation increases the dimensionality of certain features, doubling the number of input variables. Future work should assess whether this additional complexity is justified by improvements in predictive accuracy or if alternative encoding strategies could achieve similar results more efficiently.
2) **Expanding Forecasting to Solar and Wave Energy:** Given the model's structure, it may be possible to adapt this forecasting approach to other renewable energy domains, such as solar irradiation and wave energy, with minimal modifications. Investigating transferability to these energy sources could provide a unified forecasting framework.
3) **Integration with Grid Management Systems:** The next logical step is to integrate this wind forecasting model with energy grid management strategies, optimizing power distribution and storage. Future studies could explore coupling this model with battery storage forecasting or demand-response mechanisms.
4) **Hybrid Physics-Informed Models:** Merging statistical deep learning with physical weather models (e.g., WRF) could improve forecasting robustness, particularly for extreme events.
5) **Advanced Data Assimilation:** Incorporating real-time data from satellites, radar, or additional weather stations could enhance short-term gust prediction and storm detection.
6) **Transfer Learning Across Regions:** Exploring how pre-trained models from data-rich regions can be adapted for data-sparse locations like Nelson could improve forecasting accuracy when local data is scarce.
7) **Probabilistic Forecasting for Uncertainty Management:** Extending the model to output probability distributions instead of point estimates—using techniques such as Monte Carlo dropout or Bayesian transformers—could help energy operators manage supply uncertainties.

Overall, this study highlights the potential of transformer-based architectures, enhanced by domain-driven preprocessing and optimization, for high-resolution wind forecasting in complex, data-limited environments. By refining both the model and its meteorological inputs, future advancements could further support sustainable energy planning in New Zealand and beyond.

References

[1] R. Teixeira, A. Cerveira, E. Pires, and J. Baptista, "Advancing renewable energy forecasting: A comprehensive review of renewable energy forecasting methods," *Energies*, vol. 17, p. 3480, 2024.

[2] S. Aasim, S. Singh, and A. Mohapatra, “Repeated wavelet transform based arima model for very short-term wind speed forecasting,” *Renewable Energy*, vol. 136, pp. 758–768, 2019.

[3] A. T. Eseye, J. Zhang, and D. Zheng, “Short-term photovoltaic solar power forecasting using a hybrid wavelet-pso-svm model,” *Renewable Energy*, vol. 118, pp. 357–367, 2023.

[4] S. Pretto, E. Ogliari, A. Niccolai, and A. Nespoli, “A new probabilistic ensemble method for an enhanced day-ahead pv power forecast,” *IEEE Journal of Photovoltaics*, vol. 12, pp. 581–588, 2022.

[5] Z. Yang and J. Wang, “A hybrid forecasting approach applied in wind speed forecasting based on a data processing strategy and an optimized artificial intelligence algorithm,” *Energy*, vol. 160, pp. 87–100, 2018.

[6] J. Song, J. Wang, and H. Lu, “A novel combined model based on advanced optimization algorithm for short-term wind speed forecasting,” *Applied Energy*, vol. 215, pp. 643–658, 2018.

[7] Q. He, J. Wang, and H. Lu, “A hybrid system for short-term wind speed forecasting,” *Applied Energy*, vol. 226, pp. 756–771, 2018.

[8] Y. Zhang, J. Le, X. Liao, F. Zheng, and Y. Li, “A novel combination forecasting model for wind power integrating least square support vector machine, deep belief network, singular spectrum analysis and locality-sensitive hashing,” *Energy*, vol. 168, pp. 558–572, 2019.

[9] G. Wang, R. Jia, J. Liu, and H. Zhang, “A hybrid wind power forecasting approach based on bayesian model averaging and ensemble learning,” *Renewable Energy*, vol. 145, pp. 2426–2434, 2020.

[10] Y. Xie, C. Li, M. Li, F. Liu, and M. Taukenova, “An overview of deterministic and probabilistic forecasting methods of wind energy,” *iScience*, vol. 26, p. 105804, 2023.

[11] F. Shahid, A. Zameer, and M. Muneeb, “A novel genetic lstm model for wind power forecast,” *Energy*, vol. 223, p. 120069, 2021.

[12] C. Wu, J. Wang, X. Chen, P. Du, and W. Yang, “A novel hybrid system based on multi-objective optimization for wind speed forecasting,” *Renewable Energy*, vol. 146, pp. 149–165, 2020.

[13] H. H. Aly, “A novel deep learning intelligent clustered hybrid models for wind speed and power forecasting,” *Energy*, vol. 213, p. 118773, 2020.

[14] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, “Optuna: A next-generation hyperparameter optimization framework,” in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2019.

[15] National Institute of Water and Atmospheric Research (NIWA), “Climate Station Observations,” 2025, accessed: Mar. 5, 2025. [Online]. Available: