Workshop Nº2 – System Design for GEFCom2012 Wind Forecasting

Tomás Cárdenas Benítez – 20221020021
Diego Angelo Ruano Vergara – 20242020278
Sebastián David Trujillo Vargas – 20242020217
Arlo Nicolás Ocampo Gallego – 20221020104

Universidad Distrital Francisco José de Caldas
Facultad de Ingeniería – Ingeniería de Sistemas

October 2025

1 FIRST FINDINGS

The analysis conducted in Workshop Nº1 provided a comprehensive understanding of the forecasting challenge, highlighting the temporal and meteorological nature of the data, the constraints imposed by the evaluation metric, and the presence of sensitive or chaotic behaviors within the wind energy system. The project focuses on the short-term forecasting of normalized wind power generation across seven anonymized wind farms, each characterized by its own time series of hourly observations. These observations are complemented by meteorological forecasts that include the wind's zonal and meridional components (u and v), wind speed (ws), and wind direction (wd) over horizons from one to forty-eight hours ahead.

This dataset enables the study of complex dependencies between physical and environmental variables. Wind power generation is a highly nonlinear process influenced by meteorological fluctuations, atmospheric stability, and turbine behavior. Consequently, accurate forecasting requires models that can capture temporal autocorrelation, cross-variable interactions, and seasonal or diurnal patterns.

The competition evaluates participants using the **Root Mean Square Error (RMSE)**, which measures the deviation between predicted and observed power values. A lower RMSE reflects better predictive performance and model generalization. The benchmark persistence model assumes that the future value equals the last observation, establishing a minimal reference for predictive skill.

Throughout the first analysis, several key constraints were identified. The raw data often contain missing timestamps, misaligned forecasts, or measurement noise due to sensor inaccuracies. These inconsistencies require rigorous preprocessing, including cleaning, temporal alignment, normalization, and imputation. Furthermore, each forecast must be computed exclusively from data available at the time of prediction to maintain causal validity and avoid information leakage.

Finally, sensitivity analysis revealed that small deviations in the meteorological input can produce significant variations in predicted wind power. Such sensitivity highlights the inherently unstable and stochastic nature of atmospheric systems. Therefore, the forecasting framework must integrate mechanisms for robustness, adaptivity, and continuous feedback to handle uncertainty effectively and ensure operational reliability.

—

2 SYSTEM REQUIREMENTS

The system requirements translate analytical findings into measurable objectives that define the structure and behavior of the forecasting framework. The design must ensure that all processes —from data ingestion to evaluation— preserve the temporal and causal integrity of the dataset while maximizing predictive accuracy under operational constraints.

Functional Requirements

- The system must forecast normalized wind power for each of the seven wind farms with high accuracy for 48-hour horizons.
- The predictive models must minimize the **RMSE** relative to the benchmark persistence model by at least ten percent on average.
- The preprocessing pipeline must handle missing or inconsistent data autonomously, ensuring a clean and continuous time series for both training and prediction.
- The system must store and version all processed data, features, and model configurations to guarantee reproducibility.
- Each forecast cycle must be executed automatically at scheduled intervals, producing updated predictions without manual intervention.

Non-Functional Requirements

- **Reliability:** The system must maintain an uptime of 99% and support fallback to persistence forecasts during data or model failures.
- **Scalability:** The architecture must accommodate additional wind farms, longer forecast horizons, or alternative meteorological inputs without restructuring the pipeline.
- **Security:** All stored and transmitted data must be encrypted, and user access must follow role-based permissions with logging of activity.
- **Interpretability:** The models should provide transparency through feature importance scores and visualization of temporal contributions.

- **Adaptivity:** When performance degradation or data drift is detected, retraining must be triggered automatically.

These requirements define a system that is accurate, reproducible, secure, and operationally sustainable, while remaining adaptable to changes in environmental behavior or data quality.

—

3 SYSTEM ARCHITECTURE

The forecasting system is conceived as a modular pipeline that transforms raw meteorological and wind power data into accurate and interpretable forecasts. Its architecture (Figure 1) follows a layered approach where each component executes a distinct yet interconnected function, ensuring clear data flow, observability, and maintainability.

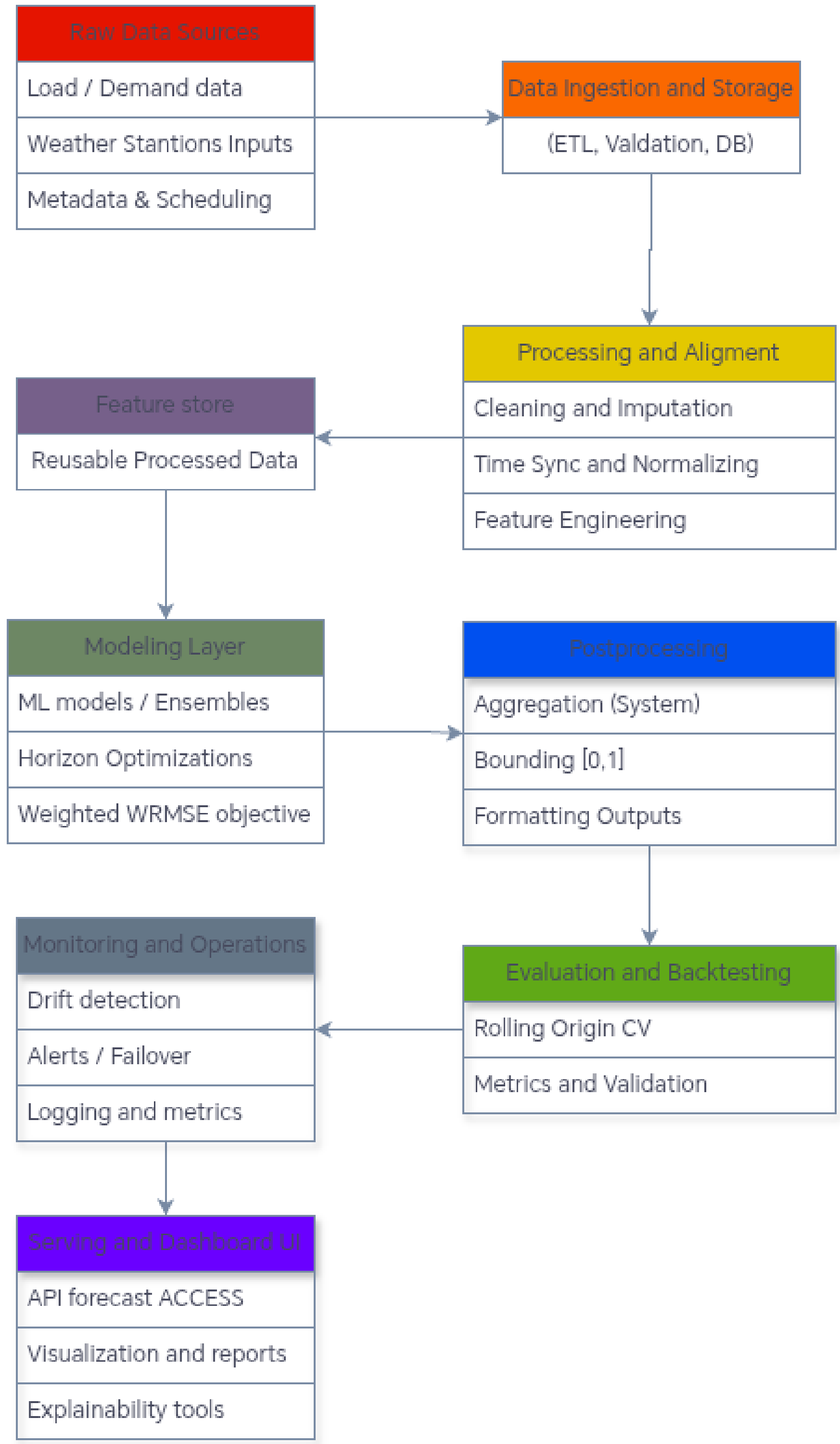

Figure 1: Proposed architecture for the wind power forecasting system.

Core Components

1. Raw Data Sources. This layer collects the core inputs of the system:

- Historical wind power data (*wp1–wp7*) normalized between [0,1].
- Forecasted meteorological variables (*u, v, ws, wd*) for 48 horizons ahead.
- Metadata describing temporal indices, forecast validity, and measurement intervals.

These datasets form the physical foundation for both model training and operational forecasting.

2. Data Ingestion and Storage. This module performs ETL operations, validation, and structured storage. Forecast timestamps are synchronized with their corresponding power observations to maintain temporal coherence. Data are stored in a time-series database or structured CSV repositories, preserving traceability for future analysis.

3. Processing and Alignment. Handles all data refinement processes — cleaning, imputation, and normalization. Additionally, engineered features such as lagged variables, rolling statistics, and trigonometric encodings of wind direction (sin(wd), cos(wd)) are created to capture periodicity and directional effects.

4. Feature Store. A centralized repository that stores processed and feature-enriched data. By versioning transformations and ensuring consistent access during both training and inference, this component prevents feature drift and improves reproducibility.

5. Modeling Layer. The analytical core of the system. Predictive models—such as Gradient Boosted Trees, Random Forests, or Recurrent Neural Networks—are trained to minimize RMSE across forecast horizons. Models may be independent per wind farm or unified multi-output architectures capturing spatial correlations. Ensemble strategies further stabilize predictions and mitigate random variance.

6. Postprocessing. Refines the raw outputs by bounding them to the normalized range [0,1] and structuring them according to the expected submission format (`id, date, wp1{wp7`). This ensures comparability and standardization for evaluation and operational use.

7. Evaluation and Backtesting. Implements rolling-origin cross-validation to emulate real forecasting conditions. RMSE is computed per wind farm and horizon, providing granular insights into temporal performance. Evaluation logs are archived to monitor consistency and retraining outcomes.

8. Monitoring and Operations. Provides real-time system oversight, tracking data drift, error dynamics, and model degradation. When performance falls below predefined thresholds, retraining is initiated automatically. Alert mechanisms and failover strategies enhance robustness.

9. Serving and Dashboard. Outputs the final forecasts via APIs and interactive dashboards. Users can visualize predictions, error trends, and feature importance, fostering operational transparency and human interpretability.

System Qualities

The architecture integrates fundamental system design principles:

- **Modularity:** Independent layers simplify maintenance and experimentation.
- **Feedback Control:** Evaluation metrics directly influence retraining and calibration.

- **Scalability:** Each component supports horizontal growth for larger datasets or additional forecast targets.
- **Resilience:** Fallback mechanisms ensure continuity during failures or unexpected input anomalies.

Together, these features produce a reliable and scientifically coherent forecasting system capable of continuous self-improvement.

—

4 SENSIBILITY AND CHAOS

Wind power forecasting systems exhibit inherent sensitivity to input perturbations due to the turbulent and nonlinear nature of atmospheric dynamics. Even small changes in wind speed or direction can result in substantial fluctuations in predicted power output. This sensitivity grows with the forecast horizon, making longer-term predictions more uncertain and variable.

To address these challenges, the system embeds multiple mechanisms of stability:

- **Data Control:** Anomaly detection filters and interpolates outliers in both meteorological and power data streams.
- **Model Robustness:** Regularized regression and ensemble learning reduce overfitting and variance from random noise.
- **Feedback Monitoring:** Continuous RMSE tracking allows early detection of prediction drift and automatic retraining.
- **Operational Resilience:** Backup persistence models ensure uninterrupted operation during extreme deviations or data loss.

From a systems science perspective, these elements form a feedback-regulated architecture where prediction errors act as signals driving corrective responses. The result is a dynamic equilibrium between adaptability and stability — a cybernetic loop in which the system learns to maintain performance despite chaotic external influences.

—

5 TECHNICAL STACK AND IMPLEMENTATION SKETCH

The technical implementation relies on a robust set of Python-based tools for time-series analysis, feature engineering, and model evaluation.

Programming Language: Python, due to its versatility, readability, and extensive support for data science workflows.

Libraries and Frameworks:

- **pandas, NumPy:** Data handling, cleaning, temporal transformations, and efficient numerical computation.
- **scikit-learn:** Provides regression models, pipelines, and cross-validation utilities for structured experimentation.
- **XGBoost, LightGBM:** Optimized ensemble algorithms that handle nonlinearity and interactions between weather variables.
- **matplotlib, seaborn:** Visualization of wind power trends, feature correlations, and model performance metrics.
- **Jupyter Notebook / Streamlit:** Interactive environments for exploratory analysis and dashboard development.

Implementation Plan:

1. Import and explore the historical wind power and forecasted meteorological datasets.
2. Clean and align timestamps, addressing missing data and ensuring consistent time horizons.
3. Generate engineered features including lagged variables, rolling statistics, and trigonometric encodings.
4. Train predictive models using cross-validation and evaluate using RMSE for each wind farm and horizon.
5. Integrate model selection, persistence benchmarking, and visualization of results.
6. Deploy forecasts through dashboards and APIs for operational access and monitoring.

This implementation strategy guarantees traceability, modular experimentation, and compatibility with continuous integration frameworks, enabling the system to evolve iteratively as new data become available.

—

6 CONCLUSION

The proposed system design offers a comprehensive framework for short-term wind power forecasting using the GEFCom2012 dataset. By combining structured data engineering, modular architecture, and adaptive feedback, the system achieves both scientific rigor and operational practicality.

Its emphasis on interpretability and robustness ensures that the resulting forecasts can inform real-world decision-making processes, such as energy dispatch, grid balancing, and renewable integration strategies. Through continuous monitoring and retraining cycles, the system embodies a cybernetic structure capable of maintaining accuracy within an inherently chaotic environment—transforming variability into learning and uncertainty into adaptive control.

—

References

- Kaggle. (2012). *GEFCom2012 - Wind Forecasting.* `https://www.kaggle.com/competitions/GEF2012-wind-forecasting`
- Hong, T., Pinson, P., & Fan, S. (2014). *Global Energy Forecasting Competition 2012.* International Journal of Forecasting, 30(2), 357–363.
- scikit-learn documentation: `https://scikit-learn.org/`
- XGBoost documentation: `https://xgboost.readthedocs.io/`
- Streamlit documentation: `https://streamlit.io/`