# DEVELOPMENT OF AI-BASED MICROGRID RENEWABLE GENERATION FORECASTING

**A Thesis Submitted**
**in Partial Fulfillment of the Requirements**
**for the Degree of**

## DOCTOR OF PHILOSOPHY

**by**


## POONAM

**(Roll No. 2K20/PHDEE/506)**

**Under the Supervision of**

**Prof. Mini Sreejeth**

**&**

**Prof. M. M Tripathi**

**Department of Electrical Engineering**

## DELHI TECHNOLOGICAL UNIVERSITY

**(Formerly Delhi College of Engineering)**
**Main Bawana Road, Delhi - 110042. INDIA**


**November, 2025**

©DELHI TECHNOLOGICAL UNIVERSITY, DELHI-2025
ALL RIGHTS RESERVED

# DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)
Shahbad Daulatpur, Main Bawana Road, Delhi-42

## CANDIDATE'S DECLARATION

I hereby certify that the work which is being presented in the thesis entitled **DEVELOPMENT OF AI-BASED MICROGRID RENEWABLE GENERATION FORECASTING** in partial fulfillment of the requirements for the award of the Degree of Doctor of Philosophy and submitted in the Department of Electrical Engineering of the Delhi Technological University is an authentic record of my own work carried out during the period from January, 2021 to November, 2025 under the supervision of Prof. Mini Sreejeth and Prof. M. M Tripathi, Department of Electrical Engineering.

The matter presented in this thesis has not been submitted by me for the award of any other degree of this or any other Institute.

**(Poonam)**

This is to certify that the student has incorporated all the corrections suggested by the examiners in the thesis and the statement made by the candidate is correct to the best of our knowledge.

Signature of Supervisors

Signature of External Examiner

# DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)
Shahbad Daulatpur, Main Bawana Road, Delhi-42

## CERTIFICATE

Certify that **Poonam** (2K20/PHDEE/506) has carried out the work presented in this thesis entitled "**DEVELOPMENT OF AI-BASED MICROGRID RENEWABLE GENERATION FORECASTING**" for the award of the degree of **Doctor of Philosophy** from Department of Electrical Engineering, Delhi Technological University, Delhi, under our supervision.

The thesis embodies the results of original work, and the studies have been carried out by the student herself. The contents of the thesis do not form the basis for the award of any other degree to the candidate or to anybody else from this or any other University/Institute.

(Prof. Mini Sreejeth)
Supervisor

(Prof. M. M. Tripathi)
Co-Supervisor

**Date**:

# Abstract

The growing integration of renewable energy resources into modern microgrids has intensified the need for accurate, scalable, and uncertainty-aware forecasting frameworks capable of operating under dynamic meteorological conditions. Wind and solar generation exhibit strong intermittency, non-stationarity, and complex spatial dependencies, making conventional statistical and deep learning models insufficient for real-time operational decision-making. This thesis develops a comprehensive suite of artificial intelligence–based methodologies that address multi-horizon deterministic forecasting, ensemble-based prediction refinement, uncertainty quantification, and cross-source spatio-temporal modeling tailored for microgrid applications.

The research begins with a systematic review of existing renewable forecasting paradigms, emphasizing the evolution from classical physical and statistical models to modern machine learning, deep learning, ensemble learning, and probabilistic approaches. Critical research gaps are identified in multi-scale temporal modeling, hybridization strategies, uncertainty calibration, and the treatment of spatial correlations in hybrid microgrid settings. Motivated by these insights, the first methodological contribution introduces a hybrid deep learning pipeline that integrates PCA- and EMD-based feature engineering with an attention-augmented LSTM model and XGBoost residual refinement. This framework significantly improves short-term wind power forecasting accuracy across multiple real-world datasets from Tamil Nadu, India.

Building upon these findings, the thesis proposes the Confined Attention-enabled LSTM (CAELSTM) architecture, which explicitly separates localized short-term dynamics from long-term periodic components via a dual-branch temporal modeling strategy. The confined-attention mechanism restricts the receptive field to the most informative temporal windows, while a parallel periodic extraction module captures diurnal and multi-scale variations. Extensive experiments demonstrate superior accuracy and robustness compared to classical LSTM variants and state-of-the-art hybrid models. To enhance

multivariate forecasting reliability, the research advances an ensemble-based framework combining CEEMDAN decomposition, Stacked GRUs, and a hybrid bagging–boosting strategy. This design effectively mitigates noise, reduces variance, and strengthens generalization under diverse operating regimes, providing a highly stable forecasting alternative for microgrid supervisory control.

Recognizing the limitations of deterministic forecasting in operational environments, the thesis develops a Physics-Informed Adaptive Conformal Prediction (PI-ACP) approach for uncertainty quantification. The proposed method integrates regime-aware nonconformity scoring, adaptive calibration via exponentially weighted residuals, and physics-based generation constraints to ensure reliable, distribution-free predictive intervals that remain consistent under temporal drift and wind ramp events. An extended multivariate formulation enables joint multi-horizon interval forecasting for improved operational risk assessment.

The final contribution presents a unified spatio-temporal transformer architecture designed for hybrid solar–wind microgrids. The model employs dual-spatial attention to capture geographical and meteorological couplings across distributed nodes, and dual-phase temporal attention to learn both short-term fluctuations and long-term seasonal trends. Coupled with a PI-ACP layer, the framework delivers physically consistent probabilistic forecasts suitable for real-time microgrid deployment. Validation on a six-node hybrid microgrid in the Khavda–Bhuj region demonstrates significant advancements in accuracy, generalization, and probabilistic reliability relative to deterministic, quantile-based, and diffusion-based baselines.

Overall, the thesis establishes an integrated forecasting ecosystem combining hybrid deep learning, attention mechanisms, ensemble learning, adaptive probabilistic calibration, and spatio-temporal transformers. The developed methodologies collectively advance renewable generation forecasting for intelligent microgrid operations, supporting improved stability, planning, and decision-making in renewable-rich power systems.

# Acknowledgements

First and foremost, I would like to express my sincere gratitude to my respected advisors, **Prof. Mini Sreejeth** and **Prof. M. M. Tripathi**, Department of Electrical Engineering, Delhi Technological University, for their constant guidance, support, and encouragement throughout my PhD journey. I feel privileged to have worked under their supervision.

I am also thankful to **Prof. Prateek Sharma** (Vice-Chancellor) and **Prof. Rinku Sharma** (Dean – Academic PG) for providing a supportive research environment at Delhi Technological University. I extend my gratitude to all faculty members and staff for their cooperation and valuable support.

I owe my deepest gratitude to my PhD research committee for their valuable comments and inputs: **Prof. Rachana Garg**, Department of Electrical Engineering and **Dr. Amrish K. Panwar**, Associate Professor, Department of Applied Physics.

I would like to express my heartfelt gratitude to my husband, **Manjeet Kataria**, for his unwavering support, patience, and belief in me throughout this journey. His encouragement has been my strength during challenging times. I am especially thankful to my son, **Mivaan Kataria**, whose presence brought immense joy and motivation; children support in ways one could never truly imagine. I also extend my sincere thanks to my in-laws for their constant encouragement and understanding.

I am deeply thankful to my friends, both within and outside DTU, for their support, companionship, and encouragement during this journey.

Finally, I dedicate this thesis to my beloved father, **Late Rajbir Singh**, whose values, sacrifices, and blessings have always guided me. His inspiration continues to be the foundation of all my achievements.

**Date**: Poonam

# Contents

Abstract iv
Acknowledgements vi
Contents vii
List of Figures xii
List of Tables xv

**1 Introduction 1**
1.1 Background and Motivation 1
1.1.1 Global Energy Transition and the Role of Microgrids 1
1.1.2 India's Renewable Energy Transition 3
1.1.3 Emergence of AI-based Forecasting 4
1.2 Problem Statement 6
1.3 Research Objectives 8
1.4 Structure of the Thesis 9

**2 Literature Review 10**
2.1 Introduction 10
2.2 Overview of AI-Based Renewable Generation Forecasting Techniques 11
2.2.1 Physical Approaches 16
2.2.2 Traditional Statistical Approaches 17
2.2.3 Intelligent models 18
2.2.4 Hybrid models 18
2.3 Deterministic Renewable Generation Forecasting Techniques 20
2.3.1 Machine Learning techniques 20
2.3.2 Deep Learning techniques 24
2.3.3 Hybrid forecasting Methods 26
2.4 Optimization Frameworks for Forecasting Models 31
2.4.1 Meta-Heuristic Optimization Structure 33

2.4.2 Different Optimization Algorithms 33
2.5 Uncertainty-Aware and Spatio-Temporal Forecasting Techniques 36
2.5.1 Probabilistic Forecasting Approaches 36
2.5.2 Conformal Prediction-Based Uncertainty Quantification 37
2.5.3 Spatio-Temporal Forecasting Techniques 41
2.6 Forecast Evaluation metrics 41
2.6.1 Evaluation of Deterministic Forecasts 43
2.6.2 Evaluation of Probabilistic Forecasts 45
2.7 Summary and Research Gaps 48

**3 Hybrid Deep Learning model for Wind Power Forecasting 53**
3.1 Introduction 53
3.2 Data Pre-processing and Feature Engineering 54
3.2.1 Data Description 54
3.2.2 Aligning NWP data according to wind turbine hub height 55
3.2.3 Data preprocessing of NWP Data: Principal Component Analysis 58
3.2.4 Data preprocessing of NWP Data: Empirical Mode Decomposition 59
3.3 Proposed Integrated Framework 61
3.3.1 Hybrid Deep Learning Model with Self-Attention Mechanism 61
3.3.2 Bayesian Hyperparameter Optimization 65
3.4 Forecast Evaluation and Results 68
3.4.1 Evaluation Metrics 68
3.4.2 Comparison with Baselines 69
3.4.3 Comparison with other feature engineering approaches 70
3.4.4 Comparison with Residual error correction techniques 73
3.4.5 Ablation Study 77
3.5 Summary 79

**4 Confined-attention enabled Deep Learning model for Wind Power Forecasting 81**
4.1 Introduction 81
4.2 Methodology 82

4.2.1 Data preparation and feature engineering 82
4.2.2 LSTM with Confined Attention Mechanism 85
4.2.3 Proposed WPF Framework 90
4.3 Experiments and Results 94
4.3.1 Training Configuration 95
4.3.2 Datasets 97
4.3.3 Baselines for Comparison 98
4.3.4 Comparative experiments with sensitivity analysis 99
4.4 Discussion 105
4.4.1 Ablation Study 106
4.4.2 Statistical Analysis 107
4.5 Summary 109

**5 Multivariate Wind Power forecasting using Ensemble techniques 111**
5.1 Introduction 111
5.2 Methodology 112
5.2.1 Complete Ensemble Empirical Mode Decomposition with Adaptive Noise (CEEMDAN) 112
5.2.2 Multivariate CEEMDAN (MCEEMDAN) 114
5.2.3 Stacked Gated Recurrent Units (SGRU) Networks 114
5.3 Proposed Hybrid Ensemble Learning System 117
5.3.1 Data Preprocessing Stage 117
5.3.2 Ensemble Learning Stage 119
5.3.3 Prediction Error Correction Stage 121
5.4 Data Description and performance evaluation metrics 123
5.4.1 Datasets 123
5.4.2 Performance evaluation metrics 125
5.4.3 Diebold-Mariano (DM) Test 126
5.4.4 Gray Relational Degree (GRD) Test 126
5.5 Experimental Evaluation 127
5.5.1 Experimental settings 128
5.5.2 Comparison with State-of-the-art methods 132
5.5.3 Statistical Analysis 140
5.5.4 Stability analysis 141

5.6 Discussion 141
5.6.1 Computational Analysis 141
5.6.2 Actionable findings and implications for distributed and high-performance systems 145
5.7 Summary 146

**6 Probabilistic Forecasting and Uncertainty Quantification for Wind Power 149**
6.1 Introduction 149
6.2 Methodology 150
6.2.1 Data Preprocessing with CEEMDAN 150
6.2.2 Conformalized GRU Networks 152
6.2.3 Conformal prediction for uncertainty quantification 153
6.2.4 Proposed PI-ACP Framework 156
6.3 Numerical Analysis 162
6.3.1 Datasets and Pre-processing 162
6.3.2 Deterministic Forecasting 164
6.3.3 Probabilistic Forecasting 168
6.3.4 Computational Complexity and Runtime Analysis 173
6.4 Discussion 174
6.4.1 Linking point accuracy to interval quality 175
6.4.2 Calibration stability and physical feasibility 175
6.4.3 Generalisation and the calibration sharpness trade-off 175
6.4.4 Regime awareness as a safety valve 176
6.4.5 Scalability for real-time operation. 176
6.4.6 Limitations 176
6.5 Summary 176

**7 Spatio-temporal analysis for Renewable generation forecasting 179**
7.1 Introduction 179
7.2 Case Study and Dataset Description 181
7.2.1 Meteorological context and dataset characteristics 183
7.2.2 Preprocessing and graph construction 185
7.2.3 Quality control and data statistics 187
7.3 Methodology 188

7.3.1 Preliminaries and Problem Definition 189
7.3.2 Spatio–Temporal Transformer Architecture 190
7.3.3 Physics–Informed Adaptive Conformal Prediction (PI–ACP) 193
7.3.4 Computational Complexity 194
7.4 Experimental Setup 196
7.4.1 Dataset and Preprocessing 196
7.4.2 Evaluation Metrics 196
7.4.3 Baseline Methods 198
7.4.4 Implementation Details 199
7.5 Results and Discussion 199
7.5.1 Deterministic Forecasting Performance 200
7.5.2 Probabilistic Forecasting Reliability 202
7.5.3 Spatio-Temporal Correlation Learning 206
7.5.4 Cross-Source Generalization and Complementarity 206
7.5.5 Computational Efficiency and Scalability 207
7.5.6 Operational Impact Analysis 207
7.5.7 Ablation Studies 208
7.5.8 Limitations and Future Work 210
7.6 Conclusion 211

**8 Conclusion and Future Scope 213**
8.1 Conclusion 213
8.2 Future Scope 215

**List of Publications 216**

**Bibliography 218**

# List of Figures


1.1 Global renewable energy capacity growth between 2010 and 2024. 2
1.2 Global electricity generation by source in 2024. 3

2.1 Schematic diagram of reviewing methodology 13
2.2 Classification of Wind speed/power forecasting 14
2.3 General flowchart of Physical approaches for WPF 16
2.4 General flowchart of Statistical approaches for WPF 17
2.5 Training and selection of an appropriate AI model 18
2.6 Different machine learning based methods 21
2.7 ANN and RNN architectures. 22
2.8 Ensemble Learning and ELM architectures. 23
2.9 Ensemble Learning and ELM architectures. 24
2.10 Flowchart of weight based hybrid approaches. 26
2.11 Schematic depicting the combined approaches including data preprocessing methods 29
2.12 Schematic depicting the combined approaches including parameter selection and optimization 29
2.13 Schematic of combined approaches including error processing techniques 30
2.14 General architecture of series hybrid model 31
2.15 Generalized architecture of parallel hybrid model 32
2.16 Overall framework of metaheuristic algorithm 34

3.1 NWP values of wind speed for Tuticorin wind farm at 10, 50 and 80m . 56
3.2 Proposed scheme of the feature engineering and data preprocessing. 57
3.3 PCA feature extraction heatmap for (a) wind speed and (b) wind direction. 59
3.4 IMF's for Wind direction and wind speed at 80m. 60

3.5 Integrated framework for wind power forecasting: (1) Feature extraction via PCA and EMD, (2) Hybrid LSTM-attention model for temporal modeling, (3) XGBoost-based residual refinement. 62
3.6 Hybrid architecture: (1) LSTM layers with self-attention, (2) Dense layers for initial predictions, (3) XGBoost residual correction. 63
3.7 Self attention and Multi-head attention mechanism. 65
3.8 Comparison of Various Hyperparameter Optimization Algorithms. 67
3.9 Site-1 baseline comparison. 71
3.10 Site-2 baseline comparison. 71
3.11 Site-3 baseline comparison. 72
3.12 Site-4 baseline comparison. 72
3.13 Coefficient of determination assessment results using the proposed approach. 76

4.1 LSTM with Confined Attention mechanism. 87
4.2 CAELSTM architecture. 88
4.3 Schematic representation of the proposed WPF methodology. 91
4.4 Correlation of one timestamp with previous timesteps. 96
4.5 Wind power generation forecast for (a) Dataset-1 (b) Dataset-2. 100
4.6 Monthly average wind characteristics comparison between Two Datasets, illustrating distinct seasonal and climatic wind variability. 104
4.7 Distribution of wind speeds for Dataset-1 and rotor speeds for Dataset-2, highlighting differences in variability and frequency distribution between the two locations. 104
4.8 Attention weight distribution across temporal steps in the CAELSTM model. 105

5.1 Fundamental structure of a classic Gated Recurrent Units (GRU) network. 115
5.2 Basic architecture of a Stacked Gated Recurrent Units (SGRU) network. 116
5.3 Detailed view of the Data Preprocessing. 118
5.4 Detailed view of the Second stage of the Proposed Hybrid Ensemble Learning System. 120
5.5 Detailed view of the Third stage of the Proposed Hybrid Ensemble Learning System. 121

5.6 The wind power data of two datasets: (a) Dataset-1, (b) Dataset-2. 124
5.7 Data decomposition of input variables using CEEMDAN for Dataset-1. 131
5.8 Data decomposition of input variables using CEEMDAN for Dataset-2. 131
5.9 Visualization of baseline comparison results for Dataset-1 134
5.10 Visualization of baseline comparison results for Dataset-2 135
5.11 Multi-step forecast results using proposed model for Dataset-1. 137
5.12 Multi-step forecast results using proposed model for Dataset-2. 138
5.13 Stacked bar chart of multi-step forecasting errors (MAE and RMSE) across different models on Dataset-1 and Dataset-2. 139
5.14 Error distribution plot for proposed hybrid ensemble learning system against baselines. 144

6.1 Schematic for the proposed PI-ACP framework. 158
6.2 Statistical variability of wind turbine operating parameters for (a) Dataset-1 and (b) Dataset-2. 163
6.3 CEEMDAN decomposition results for (a) Dataset-1 and (b) Dataset-2, showing separation of high-frequency and low-frequency components. 166
6.4 Wind-power deterministic forecast with residuals for Dataset-1 using the CEEMDAN–GRU backbone. 167
6.5 Wind-power deterministic forecast with residuals for Dataset-2 using the CEEMDAN–GRU backbone. 167
6.6 Probabilistic forecasts using the proposed PI-ACP framework with 90% confidence level for Dataset-1. 172
6.7 Probabilistic forecasts using the proposed PI-ACP framework with 90% confidence level for Dataset-2. 172

7.1 Spatial representation of the hybrid microgrid nodes. Edge width is proportional to the fused adjacency weight $\tilde{\mathbf{A}}_f$ used by the spatial attention module. 184
7.2 Proposed probabilistic framework for hybrid renewable forecasting 189
7.3 Prediction intervals at 90% confidence level for Cluster A 204
7.4 Prediction intervals at 90% confidence level for Cluster B 204
7.5 Prediction intervals at 90% confidence level for Cluster C 205
7.6 Ablation study of the proposed framework across all clusters 208

# List of Tables


2.1 Summary of recent literature reviews. 11
2.2 Time horizon based classification for wind forecasting techniques 12
2.3 Brief comparison of widely used wind power forecasting techniques 15
2.4 WPF models based on different forecasting approaches 19
2.5 Summary of hybrid approaches for wind power forecasting 27
2.6 Summary of Optimization methods for parameter selection 34
2.7 Brief comparison of probabilistic renewable generation forecasting methods 38
2.8 Brief comparison of conformal prediction approaches for renewable generation forecasting 40
2.9 Brief comparison of spatio-temporal forecasting models for renewable generation 42
2.10 Forecast performance indices for deterministic predictions 45
2.11 Forecast performance indices for probabilistic predictions 49

3.1 Statistical Description of NWP Data for Tuticorin Wind Farm. 54
3.2 Statistical Description of Wind Power Data for Tuticorin Wind Farm. 55
3.3 Wind Power Forecasting Results Using NWP Data at Different Heights. 55
3.4 Hyperparameter settings outcome for LSTM model. 68
3.5 Performance comparison of benchmark methods for Tuticorin wind farm (Site-1 and Site-2). 69
3.6 Performance comparison of benchmark methods for Tuticorin wind farm (Site-3 and Site-4). 70
3.7 Performance comparison of feature engineering methods (Site-1 and Site-2). 74
3.8 Performance comparison of feature engineering methods (Site-3 and Site-4). 74

3.9 Residual correction method comparison (Site-1 and Site-2). 75
3.10 Residual correction method comparison (Site-3 and Site-4). 75
3.11 Ablation study: Impact of key components on forecasting performance (Site-1 and Site-2). 77
3.12 Ablation study: Impact of key components on forecasting performance (Site-3 and Site-4). 78

4.1 Comparison of Baselines without Confined Attention Mechanism for two datasets 98
4.2 Sensitivity Analysis for Dataset-1 101
4.3 Sensitivity analysis for Dataset-2. 102
4.4 Statistical comparison of wind characteristics. 103
4.5 Performance of Different Variants of CAELSTM for two datasets 106
4.6 Paired t-test results on prediction values for Dataset-1 108
4.7 Paired t-test results on prediction values for Dataset-2 108

5.1 Dataset Description 125
5.2 Parameter settings of SGRU for each IMF 127
5.3 Hyper-parameters of different forecasting models 129
5.4 Multi-step MAE of different models for Dataset-1 and Dataset-2 at various forecast horizons 132
5.5 Multi-step RMSE of different models for Dataset-1 and Dataset-2 at various forecast horizons 133
5.6 Results of DM Test and GRD Test for Dataset-1 141
5.7 Results of DM Test and GRD Test for Dataset-2 141
5.8 Stability analysis based on the number of predicted errors within different absolute error ranges for Dataset-1 142
5.9 Stability analysis based on the number of predicted errors within different absolute error ranges for Dataset-2 142
5.10 Computational complexity and runtime per epoch for each forecasting model on Dataset-1 143

6.1 Detailed Dataset Characteristics and Partitioning 162
6.2 Impact of CEEMDAN Preprocessing on Deterministic Forecasts 166
6.3 Key Model and Training Parameters 168

6.4 Training-set probabilistic performance of PI-ACP. 170
6.5 Dataset-1 test-set performance against state-of-the-art probabilistic methods. 170
6.6 Dataset-2 test-set performance against state-of-the-art probabilistic methods. 171
6.7 Regime-specific performance on *Dataset-1* at the 80 % confidence level. 171
6.8 Multi-Step Forecasting Performance of PI-ACP at 90% Confidence Level 173
6.9 Average CWC Scores for Multi-Step Forecasting 173
6.10 End-to-end execution time (s). 174

7.1 Spatial Node-Level Details of the Hybrid Microgrid System 182
7.2 Spatial adjacency matrix $\mathbf{A}_{\text{geo}}$ (rounded values, $\sigma_d = 50$ km). 183
7.3 Statistical characteristics of normalized dataset features (test split). 187
7.4 Benchmark comparison of multi-step renewable forecasting for Cluster A 201
7.5 Benchmark comparison of multi-step renewable forecasting for Cluster B 202
7.6 Benchmark comparison of multi-step renewable forecasting for Cluster C 203
7.7 Probabilistic forecasting performance (95% prediction intervals) 205
7.8 Computational efficiency comparison 207
7.9 Ablation study of the proposed PI–ACP framework for 12-step forecasting across all clusters 209

# Chapter 1

# Introduction

This chapter introduces the thesis topic and outlines the motivation, challenges, and objectives that guided this research. It provides a comprehensive background on renewable generation forecasting in microgrids and presents the rationale for developing an Artificial Intelligence (AI) - based forecasting framework. The chapter concludes with a summary of the thesis structure.

## 1.1 Background and Motivation

### 1.1.1 Global Energy Transition and the Role of Microgrids

The 21st century is witnessing an accelerated global energy transition, driven by the dual need to mitigate climate change and meet the growing energy demands of expanding economies. Historical dependence on fossil fuels has led to rising greenhouse gas emissions and severe environmental consequences, creating an urgent need for a clean and sustainable energy model. This transformation is most visible in the unprecedented growth of renewable energy, especially solar photovoltaic (PV) and wind power. According to the *World Energy Outlook 2025* published by the International Energy Agency (IEA) [1], global electricity demand rose by over 2% in 2024, which is well above the long-term average. Renewables met most of this increase through record capacity additions. Electricity generation from renewables expanded by nearly 700 TWh in 2024, covering around 60% of the year's increase in global electricity demand. As shown in Fig. 1.1, the combined global renewable energy capacity surpassed 4,400 GW by the end of 2024, with solar PV and wind power emerging as the dominant contributors to this surge. Under the Net Zero Emissions (NZE) scenario, the combined capacity of solar and

wind is projected to triple from 2022 levels by 2030, consistent with the COP28 global pledge to reach nearly 11 TW of installed renewable capacity by 2030.

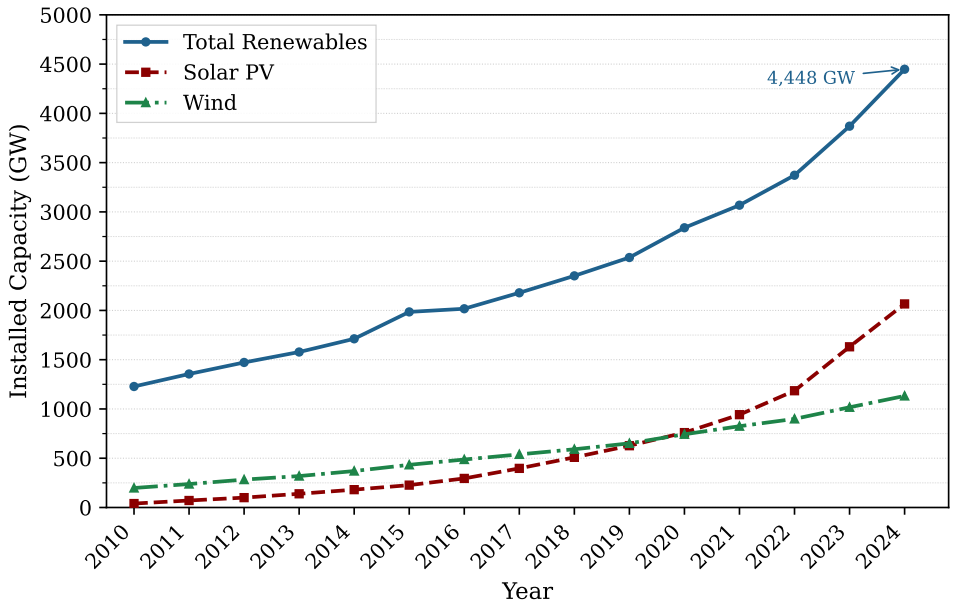



FIGURE 1.1: Global renewable energy capacity growth between 2010 and 2024.

The corresponding distribution of global electricity generation by source for 2024 is presented in Fig. 1.2. Renewables collectively contributed approximately one-third of total generation (34.0%), surpassing coal (33.0%) for the first time in history. Solar PV and wind together accounted for a significant portion of this share, showing their expanding influence in the global energy mix. Amid this accelerating transition, integrating renewable energy resources at the distribution level has brought the concept of the *microgrid* to prominence. A microgrid is a localized, self-sufficient energy system that can operate either in conjunction with the main grid (grid-connected mode) or independently (islanded mode). By bringing generation closer to consumption, microgrids enhance local reliability, improve energy efficiency, and are ideally suited for integrating a high volume of local renewable energy sources (RES). However, the effective operation of a renewable-rich microgrid hinges on a significant challenge: the inherent intermittency of sources like wind and solar power. Unlike conventional thermal generators that can be dispatched on demand, the output of renewable generators

depends on fluctuating weather conditions, which introduces uncertainty in power balance and control.

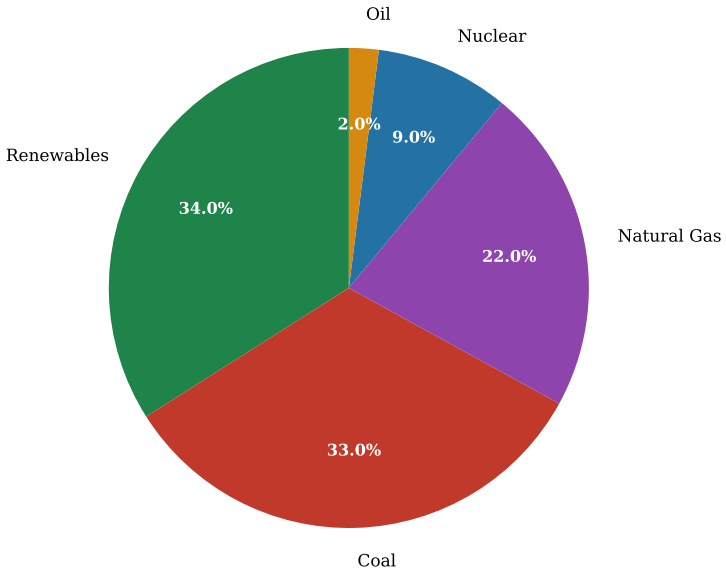



FIGURE 1.2: Global electricity generation by source in 2024.

### 1.1.2 India's Renewable Energy Transition

India has emerged as a key player in this global clean-energy transformation. Supported by strong policy frameworks and increasing investment, the country has already achieved its 50% non-fossil power capacity target in 2025, five years ahead of schedule. As reported in the *IEA World Energy Outlook 2025* [1] and by the Ministry of New and Renewable Energy (MNRE) [2], non-fossil sources now account for approximately 60% of India's installed power capacity and are projected to reach 70% by 2035. Solar PV has become the main driver of new capacity additions, attracting more than USD 113 billion in cumulative investment over the past decade and surpassing investments in fossil-fuel power.

Consistent with these achievements, India maintains a national commitment to achieve net-zero emissions by 2070 and to reach 500 GW of non-fossil capacity by 2030. This ambitious target implies that non-fossil sources will comprise roughly 60% of the nation's total installed capacity by the end of the decade. Such large-scale integration of renewables highlights the growing importance of accurate,

AI-based forecasting frameworks to maintain microgrid stability, ensure optimal scheduling, and mitigate uncertainty in renewable generation.

This accelerating shift, both globally and nationally, reinforces the strategic importance of developing advanced AI-based forecasting and control frameworks to ensure reliability, interpretability, and operational resilience in renewable-dominated microgrids. This challenge gives rise to the central theme of this thesis: **microgrid renewable generation forecasting**. This is the process of using historical data, meteorological information, and advanced computational models to predict the amount of electricity that will be generated by renewable assets within a microgrid at a future point in time. Accurate forecasting is the foundation of intelligent microgrid operation. It informs critical decisions across various time horizons, from real-time power dispatch and economic scheduling of storage to planning for unit commitment and maintenance. An effective forecast allows a microgrid's Energy Management System (EMS) to proactively manage its resources, ensuring stability, minimizing reliance on costly backup power, and optimizing economic performance.

Beyond simple temporal prediction, spatio-temporal analysis has emerged as a key dimension in renewable generation forecasting. Renewable resources such as wind and solar are inherently distributed across wide geographic areas, and their outputs are strongly influenced by local weather dynamics and terrain. Spatio-temporal forecasting techniques capture both the *time-varying patterns* and the *spatial correlations* among multiple generation sites, such as neighboring solar farms or wind turbines. This enables a more holistic understanding of how power generation evolves across a region. By integrating spatial dependencies (for example, cloud movements or wind fronts) with temporal evolution, such models significantly enhance forecast accuracy and resilience. In the context of microgrids, where distributed energy resources are often clustered but meteorologically diverse, spatio-temporal modeling becomes particularly valuable for coordinating distributed assets and improving overall power balance reliability.

### 1.1.3 Emergence of AI-based Forecasting

Renewable generation forecasting has traditionally relied on classical statistical and physical models, such as autoregressive moving average (ARMA), autoregressive

integrated moving average (ARIMA), and numerical weather prediction (NWP)-based regression models. While these methods perform reasonably well under stable conditions, they often fail to capture nonlinear dependencies, temporal non-stationarity, and regime shifts inherent in renewable generation data. Their limited adaptability and reliance on linear assumptions restrict their accuracy during rapidly changing weather events, thereby constraining their operational usefulness in microgrids.

To overcome these limitations, researchers have increasingly turned to Artificial Intelligence (AI)-based models, leveraging machine learning and deep learning architectures to uncover complex nonlinear relationships among meteorological and electrical variables. AI models-such as artificial neural networks (ANNs), support vector machines (SVMs), and recurrent neural networks (RNNs)-have demonstrated remarkable capabilities in handling high-dimensional, nonlinear, and dynamic data. However, despite their flexibility and accuracy, AI-based models introduce their own challenges, including overfitting, data dependency, model interpretability, and uncertainty quantification. Most importantly, purely deterministic AI models provide only point estimates, which do not adequately express the uncertainty critical for risk-aware decision-making in microgrid operations.

To address the inherent uncertainty in renewable generation, the forecasting community has progressively evolved toward probabilistic, hybrid, and ensemble-based approaches. Probabilistic forecasting models-such as quantile regression, Bayesian networks, and conformal prediction-offer prediction intervals or distributions instead of single-point estimates, enabling grid operators to assess risks and allocate reserves more effectively. Hybrid and ensemble strategies further enhance robustness by combining the strengths of different learning paradigms, such as integrating physical knowledge with data-driven inference or merging multiple learners to achieve bias–variance balance. Moreover, the increasing spatial deployment of renewable energy installations has necessitated the development of spatio-temporal forecasting frameworks that account for the interdependence of multiple geographically distributed sites and their evolving temporal dynamics.

In this context, the present research is motivated to develop a comprehensive AI-based framework for microgrid renewable generation forecasting that unifies

the strengths of deterministic and probabilistic learning, leverages hybrid and ensemble mechanisms, and captures multi-site spatio-temporal dependencies. The proposed framework is designed to be four-dimensional-incorporating **deterministic forecasting** for accuracy, hybrid and ensemble learning for robustness, multi-variate and multi-horizon prediction for scalability, and probabilistic spatio-temporal modeling for uncertainty quantification and spatial coherence. This integrated approach aims to deliver a next-generation forecasting architecture that enhances reliability, interpretability, and operational value for microgrid energy management systems under dynamic weather regimes.

## 1.2 Problem Statement

Despite substantial progress in renewable generation forecasting, existing methodologies remain inadequate for ensuring reliable and adaptive operation of microgrid systems under dynamic weather conditions. Classical statistical and physical models, such as ARIMA, SARIMA, and numerical weather prediction (NWP)-based approaches, rely on linearity and stationarity assumptions that fail under real-world variability. These models exhibit poor adaptability to rapid changes in meteorological parameters, resulting in increased forecasting errors during transient events such as cloud movements and wind ramping.

In general, renewable generation forecasting can be formulated as a supervised learning problem, where the objective is to learn a nonlinear mapping as defined in Eq. (1.1):

$$\hat{P}_{t+h} = \mathcal{F}(\mathbf{x}_t, \mathbf{x}_{t-1}, \ldots, \mathbf{x}_{t-k}; \boldsymbol{\theta}), \tag{1.1}$$

where $\hat{P}_{t+h}$ denotes the forecasted renewable generation at a future time horizon $h$, $\mathbf{x}_t$ represents the input feature vector (meteorological and historical power data), $k$ is the temporal window length, and $\boldsymbol{\theta}$ represents the learnable parameters of the forecasting model $\mathcal{F}(\cdot)$.

Conventional deterministic models estimate only a single-point value $\hat{P}_{t+h}$ by minimizing a point-based loss function such as the mean squared error (MSE) given in Eq. (1.2):

$$\mathcal{L}_{\text{det}} = \frac{1}{N} \sum_{i=1}^{N} (P_{t+h}^{(i)} - \hat{P}_{t+h}^{(i)})^2, \tag{1.2}$$

where $P_{t+h}^{(i)}$ denotes the true generation for the $i^{\text{th}}$ sample. While effective under stable conditions, this formulation neglects uncertainty propagation and fails to provide confidence bounds necessary for decision-making in microgrids.

Probabilistic forecasting extends this problem to estimating the conditional distribution of future power output, as shown in Eq. (1.3):

$$p(P_{t+h} \mid \mathbf{x}_t) = \mathcal{F}_{\text{prob}}(\mathbf{x}_t; \boldsymbol{\theta}), \tag{1.3}$$

where $\mathcal{F}_{\text{prob}}$ generates a predictive density or quantile interval instead of a point forecast. However, many existing probabilistic models lack calibration, interpretability, and physical feasibility constraints.

Furthermore, in microgrids with multiple distributed renewable sites, spatial correlations must be explicitly modeled. The joint forecasting problem can be represented as the spatio-temporal function in Eq. (1.4):

$$\hat{\mathbf{P}}_{t+h} = \mathcal{G}(\mathbf{X}_{t:t-k}, \mathbf{A}; \boldsymbol{\theta}), \tag{1.4}$$

where $\mathbf{X}_{t:t-k}$ is a multivariate time series representing multiple sites over $k$ past time steps, and $\mathbf{A}$ denotes the spatial adjacency matrix capturing inter-site dependencies such as wind field propagation or cloud motion. Ignoring these dependencies results in degraded forecasting accuracy and reduced microgrid coordination.

Finally, the forecasting model must satisfy physical feasibility constraints to ensure operational consistency, as expressed in Eq. (1.5):

$$0 \leq \hat{P}_{t+h} \leq P_{\text{rated}}, \tag{1.5}$$

where $P_{\text{rated}}$ represents the rated capacity of the renewable source. Many data-driven models neglect such physical constraints, leading to unrealistic or infeasible power predictions.

In summary, existing deterministic, probabilistic, and hybrid approaches are often developed in isolation and fail to holistically capture nonlinear, non-stationary, and multi-scale dynamics across spatially distributed renewable assets. Therefore, there is a critical need for an adaptive, physically consistent, and uncertainty-aware forecasting framework that unifies deterministic, probabilistic, and spatio-temporal learning paradigms. Such a framework should effectively

model the intrinsic variability of renewable generation while ensuring high forecasting accuracy, reliability, and interpretability for real-time microgrid operations.

## 1.3 Research Objectives

To address the aforementioned challenges and advance the state of renewable generation forecasting in microgrid environments, this research is structured around the following four overarching objectives:

- **RO-1: Multi-horizon Renewable Generation Forecasting:**
  To develop and evaluate data-driven models capable of delivering accurate renewable generation forecasts across multiple temporal horizons. The objective is to establish forecasting schemes that can effectively support both short-term operational control and long-term planning in microgrid systems, ensuring consistent predictive performance under varying time scales and weather dynamics.

- **RO-2: Ensemble Learning for Forecasting Robustness:**
  To enhance the robustness and generalization of renewable generation forecasts through ensemble learning methodologies. This involves integrating diverse machine learning and signal decomposition techniques within hybrid frameworks to exploit complementary model strengths and reduce prediction uncertainty across dynamic operating conditions.

- **RO-3: Probabilistic Forecasting and Uncertainty Quantification:**
  To formulate a probabilistic forecasting framework that quantifies prediction uncertainty and provides calibrated predictive intervals in addition to point estimates. The objective is to enable risk-aware decision-making in renewable-rich microgrids by incorporating both parametric and non-parametric approaches for uncertainty modeling and calibration.

- **RO-4: Spatio-temporal Forecasting and Multi-site Correlation Modeling:**
  To design an integrated spatio-temporal forecasting framework that captures the spatial dependencies and temporal evolution of geographically

distributed renewable energy sources. This objective aims to improve system-level situational awareness and coordination in microgrids by leveraging spatial learning architectures capable of representing inter-site meteorological interactions.

## 1.4 Structure of the Thesis

The remainder of this thesis is organized as follows: Chapter 2 provides a comprehensive literature review on renewable generation forecasting, including multi-horizon prediction, ensemble learning, probabilistic methods, and spatio-temporal modeling. Chapter 3 presents the first major methodological contribution by introducing a hybrid deep learning framework that integrates feature engineering, attention-augmented LSTMs, and residual refinement for short-term wind power forecasting. Chapter 4 extends this work through a confined-attention enabled deep learning architecture that captures both localized short-term dynamics and long-term periodic patterns. Chapter 5 develops a multivariate ensemble forecasting framework based on CEEMDAN decomposition, stacked GRUs, and hybrid bagging–boosting fusion to enhance robustness and stability. Chapter 6 introduces the Physics-Informed Adaptive Conformal Prediction (PI-ACP) framework for reliable, physically consistent uncertainty quantification under dynamic atmospheric conditions. Chapter 7 proposes a unified spatio-temporal transformer architecture for hybrid solar–wind microgrid forecasting, incorporating dual-spatial and dual-phase temporal attention with physics-informed probabilistic calibration. Finally, Chapter 8 concludes the thesis by summarizing key findings, contributions, and directions for future research.

# Chapter 2

# Literature Review

## 2.1 Introduction

This chapter presents a comprehensive review of existing literature on renewable generation forecasting with a particular focus on microgrid applications. It aims to consolidate recent developments in multi-horizon prediction, ensemble-based learning, probabilistic forecasting, and spatio-temporal modeling to establish a clear understanding of current trends and research gaps. The review is organized to directly correspond with the four research objectives defined in this study. The first objective explores models capable of delivering accurate forecasts across multiple temporal horizons, ensuring the applicability of predictions for both short-term operational control and long-term energy planning. The second objective investigates ensemble learning and hybridization techniques designed to enhance the robustness, accuracy, and generalization capability of forecasting frameworks under varying environmental regimes. The third objective examines probabilistic forecasting methods that quantify uncertainty through predictive intervals and probabilistic scores, enabling risk-aware decision-making in renewable-rich systems. The fourth objective focuses on spatio-temporal modeling approaches that capture spatial correlations and inter-site dependencies among distributed renewable energy sources, which are vital for improving situational awareness and coordinated microgrid management.

In addition to these thematic areas, the chapter discusses the role of meta-heuristic optimization algorithms for parameter tuning, model calibration, and performance enhancement of AI-based forecasting frameworks. It also reviews deterministic and probabilistic evaluation metrics that serve as benchmarks for

assessing forecasting accuracy and uncertainty quantification. The outcome of this chapter is a consolidated research landscape that identifies the strengths and limitations of existing forecasting methodologies. This synthesis provides the rationale for the development of an integrated AI-based multi-horizon forecasting framework that combines ensemble learning, probabilistic modeling, and spatio-temporal correlation analysis to advance renewable generation forecasting for intelligent microgrid applications.

**Table 2.1** Summary of recent literature reviews.

| Reference | Year | Main contribution |
|---|---|---|
| [3] | 2026 | Comparison of statistical and machine learning based models is provided. Focus was on short-term forecasts. Presented a thorough analysis of the duration of the data utilized and a comparison of performance metrics. |
| [4] | 2025 | Evaluated wind power probabilistic forecasting. Focus was on short-term forecasts. Analysis of type of input data used is also provided. |
| [5] | 2025 | Hybrid prediction models based on deep learning has been reviewed. Provided performance comparison of hybrid deep learning based models. |
| [6] | 2024 | Comparison of different deterministic and probabilistic models has been provided. |
| [7] | 2024 | Presented comprehensive review of meta-huristic algorithms. In addition, a quantitative study of their advantages, disadvantages, computational costs, and forecasting precision is presented. |
| [8] | 2022 | A comprehensive examination of forecast evaluation techniques for short-term Wind Power Forecasting (WPF) has been presented. The robustness of performance evaluation metrics is further assessed by a numerical examination of hybrid models based on decomposition. |
| [9] | 2021 | AI based hybrid models has been reviewed. A comparison is made based on temporal resolution, parameters employed, accuracy, and research limitations. |
| [10] | 2021 | The various forms and topologies of hybrid models have been reviewed. |
| [11] | 2020 | A brief assessment of key energy forecasting articles was provided, as well as a summary of research trends and a discussion of the value of reproducible research. |
| [12] | 2020 | Reviewed physical, statistical, and hybrid methods. The emphasis was on the elements that influence predictive modeling accuracy and computational time. |
| [13] | 2020 | Presented an extensive review of existing forecasting approaches, current advances and hybrid forecasting models. |

## 2.2 Overview of AI-Based Renewable Generation Forecasting Techniques

With the increasing integration of renewable energy into microgrids and bulk power systems, forecasting research has expanded at an unprecedented pace. To assess this progression and identify major research directions, a comprehensive literature search was conducted using the Web of Science database with the query: TS=("Wind forecasting" OR "Wind Power Forecasting (WPF)" OR "Wind speed forecasting" OR "Wind prediction"). Table 2.1 summarizes the key points from recent reviews. A total of 3762 publications were identified, showing a sharp and

consistent rise in activity between 2011 and 2023. Since the first article on the prediction of wind power was published [14], many forecasting strategies have been put out that use past wind speed and power to predict 60 wind power output in wind farms [15]. These methods can be classified into mainly four groups:

- Physical forecasting methods [16]
- Traditional statistical forecasting methods [17]
- Artificial Intelligence (AI)-based forecasting methods [18]
- Hybrid forecasting methods [19]

Each strategy aims to decrease computational costs while increasing the accuracy of wind power forecast. So far, many studies have been conducted to analyze various wind forecasting systems to present classification of different time scales in Table 2.2.

**Table 2.2** Time horizon based classification for wind forecasting techniques

| Time horizon | Scale | Applications |
|---|---|---|
| Long-term [20] | More than 1-week | • Maintenance Scheduling to Obtain Optimal Operating Cost<br>• Operations management |
| Medium-term [21] | 5 to 7-days ahead | • Decisions on unit commitment<br>• Reserve requirement |
| Short-term [22] | 1-hour to 1 or 2 days ahead | • Economic Load Dispatch Planning<br>• Load sharing Decisions<br>• Operational Security in day-ahead electricity markets |
| Very short-term [23] | Few minutes to 1 hour ahead | • Regularty Actions<br>• Electricity Market Clearing<br>• Real-time grid operations |

Keeping in mind the mentioned reviews in Table 2.1, the present review has been carried out in two stages, as indicated in Fig. 2.1. The intricate procedure of electric power scheduling encompasses both the generation and forecasting of electrical power. Prior to the advent of renewable energy sources, conventional

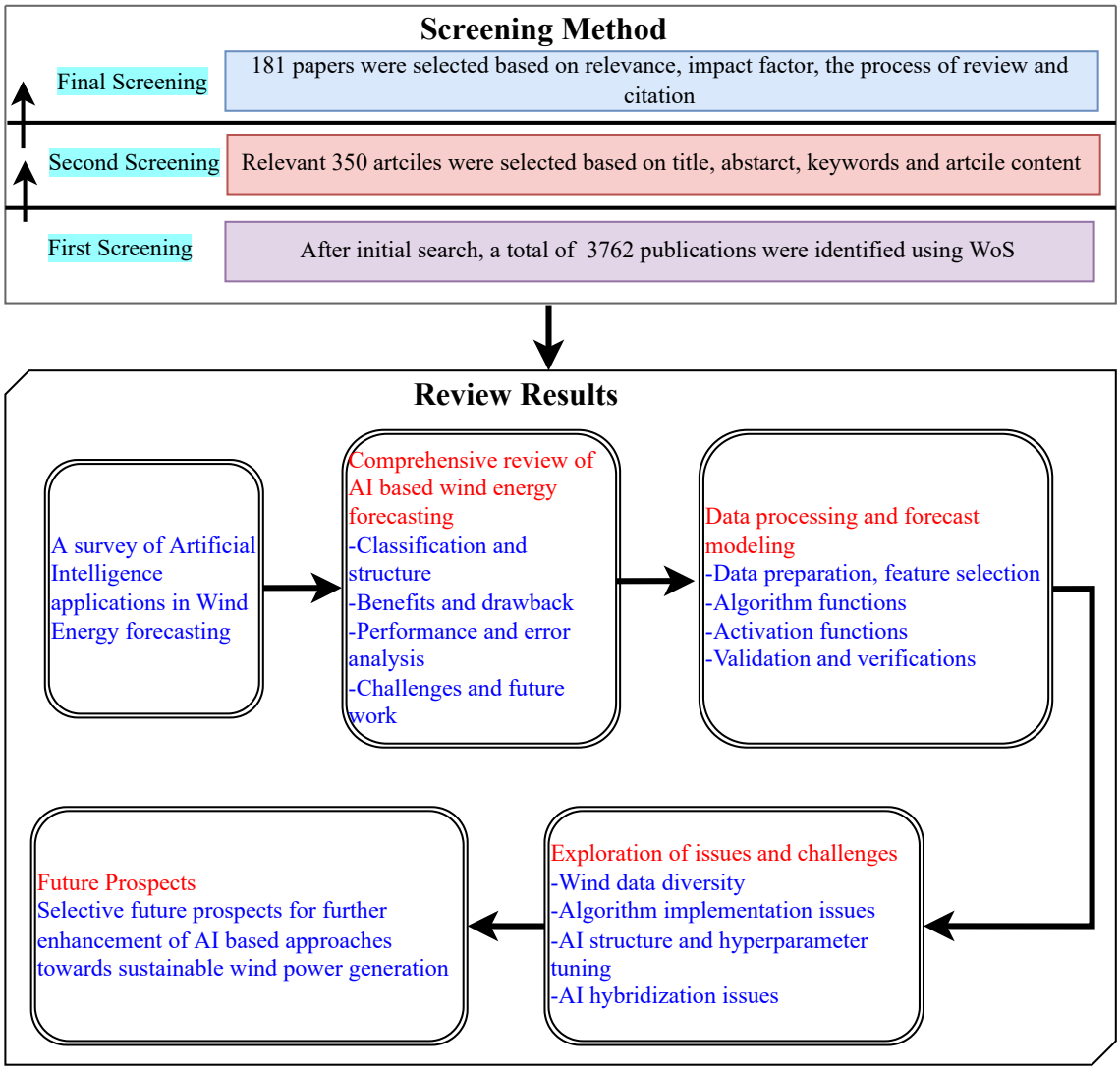



FIGURE 2.1: Schematic diagram of reviewing methodology

sources such as thermal and nuclear power plants predominantly generated electricity. These plants offered the advantage of manageable and controllable production within the constraints of the plant's ratings. In such a context, load forecasting took precedence, with generation scheduled to fulfill the load demand, thereby minimizing the need for generation forecasting. However, with the emergence of renewable energy production, which is inherently dependent on natural factors and challenging to control, generation forecasting has become crucial. In this scenario, the prediction of generation ahead of the load schedule

becomes vital. As research shifts focus towards enabling higher penetration of renewable energy, generation forecasting is gaining prominence. One critical aspect to consider in generation forecasting is the forecast period. The categorization of wind forecasting techniques by time scale can be nebulous.

Nevertheless, Table 2.2 presents a classification of time scales along with the application of specific time horizons in the context of the electrical system [24].

Since wind speed and power forecasting share a lot in common, both are discussed with an emphasis on the model principle. The literature describes two approaches for forecasting wind energy. The first technique involves first predicting wind speed and then forecasting wind power based on a wind power curve [25]; this shows the power that can be produced at different wind speed values. The second technique entails creating wind power forecasts using past wind data. Therefore, researchers have focussed on wind speed and power forecasts. Numerous approaches and techniques have been used to predict wind speed and power. According to several classification criteria, these can be divided into a wide range of groupings, as shown in Fig. 2.2.

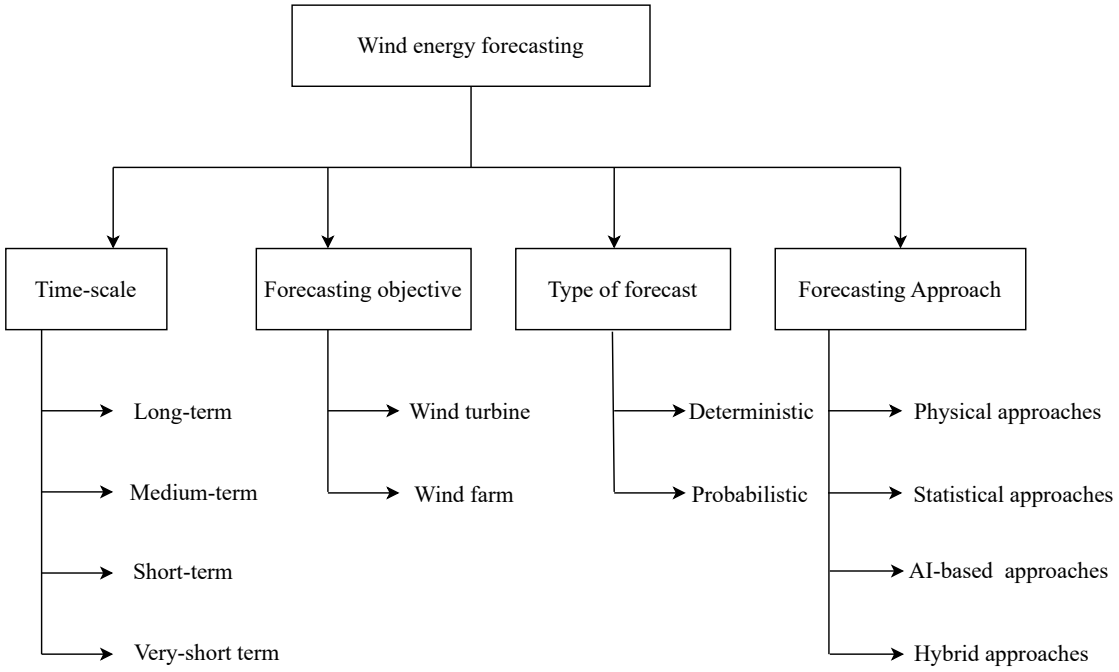



FIGURE 2.2: Classification of Wind speed/power forecasting

**Table 2.3** Brief comparison of widely used wind power forecasting techniques

| Forecasting approach | Advantages | Disadvantages |
|---|---|---|
| Time-series models [26] | • Easy to implement with widely available tools<br>• Capable of correcting local trends<br>• Simple model structure<br>• Can provide confidence intervals | • Require large historical datasets<br>• Limited ability to model nonlinear behavior<br>• Model structure selection can be difficult |
| Numerical Weather Prediction (NWP) models [27] | • Suitable for long-term forecasting horizons | • Weak at capturing small-scale/local phenomena<br>• Not suitable for short-term forecasting<br>• High computational cost and runtime |
| Artificial Neural Network-based models [28] | • Learn complex patterns from data<br>• Do not require explicit mathematical formulation<br>• Adapt well to online measurements<br>• High tolerance to measurement noise | • Require extensive training data<br>• Training can be time-consuming |
| Support Vector Machines-based models [29] | • Strong generalization performance | • Sensitive to parameter tuning<br>• Long training time<br>• Optimization can be complex |

Table 2.3 delineates the merits and demerits of various forecasting methods. The models can also be divided into two groups based on their goal of forecasting: wind turbine forecasting and wind farm forecasting. The former can be used to demonstrate the amount of power that a single wind turbine produces [30]. The latter forecasts electricity output of an entire wind farm by combining data from numerous wind turbines [31]. The latter strategy is more challenging of the two.

Based on the nature of the forecast, models can be partitioned into deterministic and probabilistic types [32]. Due to the intricate nature of the environment, deterministic models, which yield only point forecasts [33], are

limited in their forecasting performance and may yield imprecise forecasts. In contrast to point predictions, probabilistic models encapsulate uncertainty in prediction intervals (PI) and provide decision-makers with more comprehensive information [33–35]. Uncertainty prediction is bifurcated into non-parametric and parametric techniques in the literature [36].

Further, an overview of the forecasting techniques and their generalized flowcharts is provided in this section.

### 2.2.1 Physical Approaches

The most often used methods for predicting wind are physical models. To determine the local wind power, speed, and direction, these models often incorporate a wide range of meteorology data, including orography, surface roughness, obstructions, pressure, and temperature near the wind farm [37]. Physical models include the Weather Research and Forecasting Model (WRF) and Numerical Weather Prediction (NWP). A generalized flowchart of these models is depicted in Fig. 2.3. Physical methods are highly susceptible to the initial conditions, which could cause variations in wind speed measurements [38]. They are hence commonly integrated with other models to improve prediction accuracy [12].

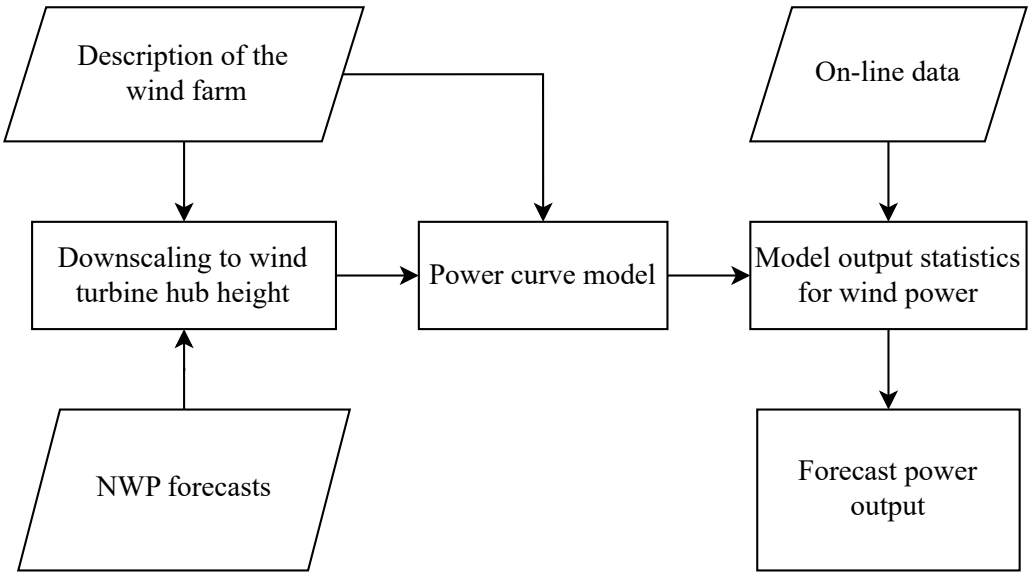



FIGURE 2.3: General flowchart of Physical approaches for WPF

### 2.2.2 Traditional Statistical Approaches

Statistical models use historical data to perform predictions and are unaffected by atmospheric conditions. Time series models, Gray models, Markov-regime switching models, stochastic differential equations, and quantile regression (QR) are some examples of statistical methods. Among these, time series methods such as autoregressive (AR), autoregressive–moving-average (ARMA), autoregressive integrated moving average (ARIMA), and fractionally integrated ARMA (FARIMA) are frequently utilized. These approaches establish a precise correlation between inputs (like NWP data, historical data, geographic data, etc.) as depicted in Fig. 2.4. These approaches assume that time series have linear structures; therefore these techniques cannot predict accurately for non-linear data like wind time series data. Also, statistical models perform well for short and medium-term time horizons [17].

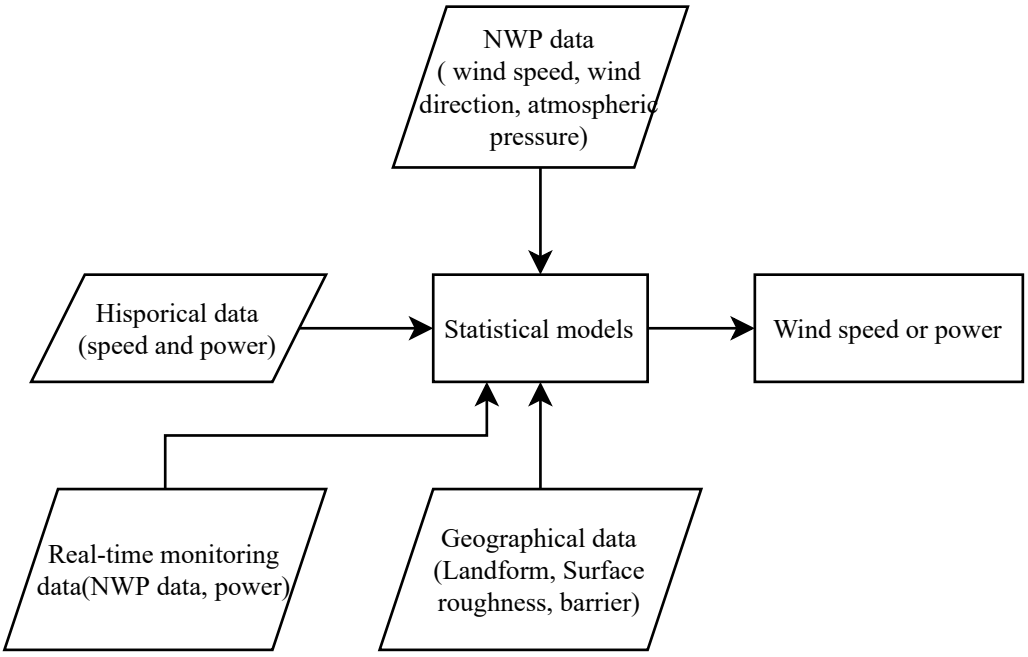



FIGURE 2.4: General flowchart of Statistical approaches for WPF

### 2.2.3 Intelligent models

When time series data are non-stationary and irregular, it is reasonable to use intelligent forecasting algorithms. Intelligent models are widely employed in many forecasting applications because they can forecast output based on historical data and the behaviour of data sets. Support Vector Machine (SVM), Extreme Learning Machine (ELM), ensemble learning, Recurrent Neural Network (RNN), Stacked Auto Encoder, Convolutional Neural Network (CNN), Random Forest, Deep Belief Network (DBN), Generative Adversarial Network (GAN), and hybrid ensemble learning techniques are some of the AI-based models that are widely employed in wind power prediction. Fig. 2.5 depicts the selection and training of an AI model.

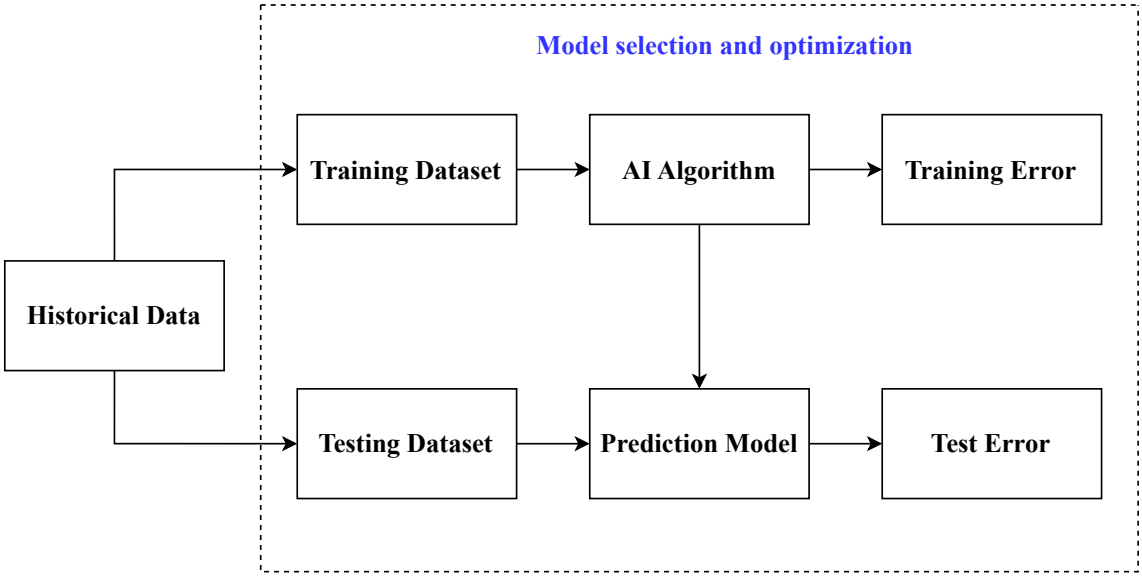



FIGURE 2.5: Training and selection of an appropriate AI model

### 2.2.4 Hybrid models

Owing to their superior prediction precision and diminished uncertainty, hybrid models, which amalgamate the benefits of diverse forecasting models, have found extensive application in forecasting domains. Despite their unique attributes, standalone models often exhibit limitations in modeling and prediction accuracy. Furthermore, the use of a singular model heightens the risk of inaccurate modeling. Table 2.4 clusters various wind energy models based on different approaches.

**Table 2.4** WPF models based on different forecasting approaches

| Forecasting Approach | Forecasting model |
|---|---|
| 1. Physical approaches [39] | • 1.1 Weather Research and forecasting model(WRFM)<br>• 1.2 Numerical Weather Prediction (NWP) |
| 2. Traditional statistical approaches [40] | • 2.1 Time series models<br>• 2.2 Gray models<br>• 2.3 Quantile regression<br>• 2.4 Stochastic differential equations<br>• 2.5 Markov regime-switching model |
| 3. Machine Learning approaches [41] | • 3.1 Artifical Neural Networks(ANN's)<br>• 3.2 Support Vector Machines(SVM's)<br>• 3.3 Extreme Learning Machines(ELM's)<br>• 3.4 Recursive Neural Network(RNN)<br>• 3.5 Random forest<br>• 3.6 Ensemble Learning |
| 4. Deep Learning approaches [42] | • 4.1 Deep Belief Network(DBN)<br>• 4.2 Convolution Neural Network(CNN)<br>• 4.3 Generative Adversial Network(GAN) |
| 5. Hybrid approaches [43] | • 5.1 Weighting based hybrid approaches<br>• 5.2 Data processing based hybrid approaches<br>• 5.3 Parameter selection and optimization based hybrid approaches<br>• 5.4 Component combination based hybrid approaches |

## 2.3 Deterministic Renewable Generation Forecasting Techniques

"Artificial Intelligence (AI) is a branch of computer science by which intelligent machines can be created that behave like humans, think like humans, and make decisions" [44]. AI algorithms excel at processing and interpreting information, converting it into actionable knowledge. This proficiency renders AI indispensable in decision-making, speech recognition, and heuristic search applications [45]. In recent times, AI algorithms have demonstrated promising outcomes in the realm of renewable energy forecasting. Fuzzy logic, Deep Learning (DL), and Machine Learning (ML) are three prevalent AI approaches. This section delves into the application of ML and DL in wind power prediction. The fundamental concepts and structures of AI techniques are further elucidated below.

### 2.3.1 Machine Learning techniques

Machine Learning (ML) algorithms, an essential element of AI, are used to analyze data behavior and frequently execute specific learning and logical reasoning abilities without precise instructions [46]. These algorithms have proven instrumental in diverse domains such as web search, speech recognition, autonomous navigation, and human genome identification. The survey reveals that Artificial Neural Networks (ANNs) [47], Support Vector Machine (SVM) [48], Extreme Learning Machine (ELM), Recurrent Neural Network (RNN) [49], ensemble learning, and Random Forest [50] are among the most frequently employed ML techniques in wind energy forecasting. The classification of ML-based models is depicted in Fig. 2.6.

#### Artificial Neural Network

The term "ANN" is inspired by the biological neural networks that play a crucial role in shaping the human brain's structure. Similar to the human brain, ANNs consist of neurons, also referred to as nodes, interconnected across various network layers. Each neuron receives data from its input nodes, processes it, determines its activation function, and transmits its response to other neurons through the

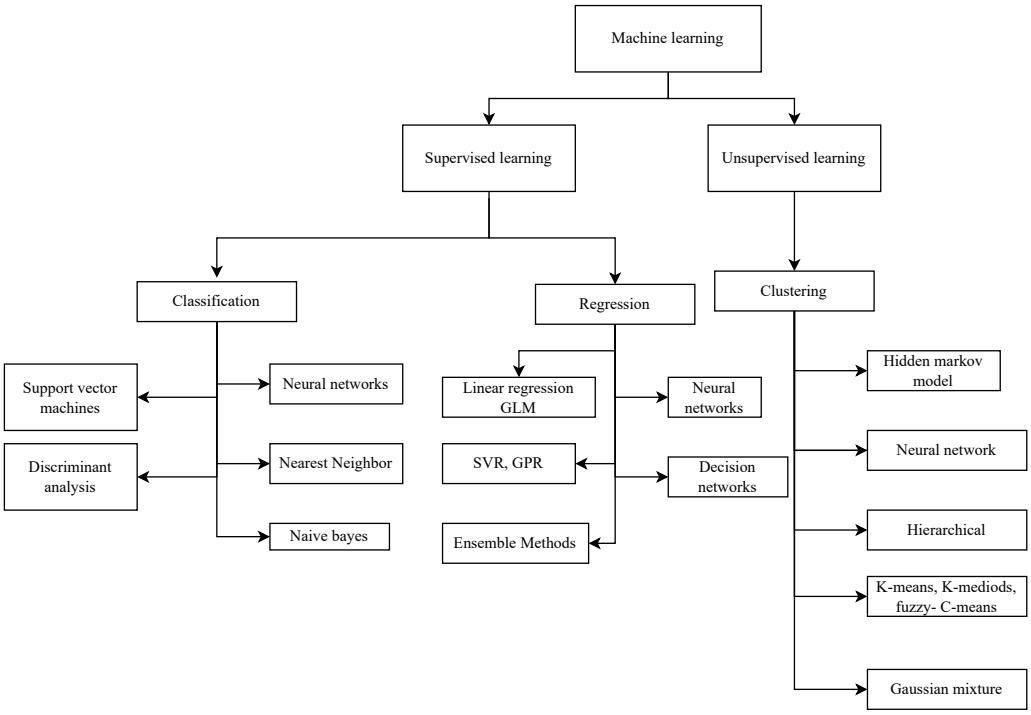



FIGURE 2.6: Different machine learning based methods

output node [47]. Fig. 2.7 (a) illustrates a standard neural network model. ANNs are capable of approximating diverse data nonlinearities. They serve as versatile computational frameworks for simulating a wide range of nonlinear scenarios. The model construction technique does not require any prior model form assumptions. Instead, the network model is determined mainly by the features of the data. The prevalent neural network structures include (i) Feed-forward multilayer perceptron (MLP), (ii) Generative Adversarial Network (GAN), (iii) Deep Belief Network (DBN), (iv) Recurrent Neural Network (RNN), and (v) Convolutional Neural Network (CNN) [51]. Numerous research articles have been published that utilize feed-forward neural networks for wind energy predictions [52, 53].

## Recursive Neuron Network

RNN refers to a family of ANNs that include recurrent and recursive variants [54]. Inter-neurons in a time-recursive neural network form feedback connections. The structural recursive neural network recursively builds a deep neural network [49]. The study in [54] used an online recursive prediction error (RPE) learning

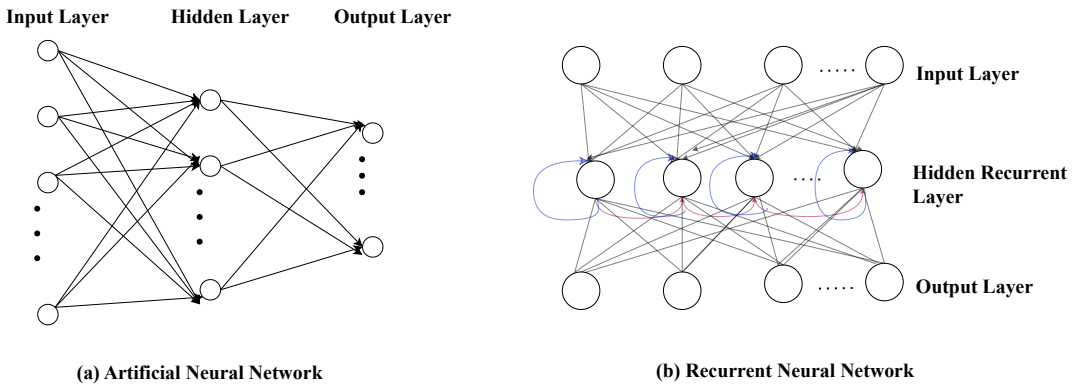



FIGURE 2.7: ANN and RNN architectures.

algorithm and found that a trained RNN predicted wind speed and power better than the ARIMA model based on meteorological data [54, 55]. Fig. 2.7 (b) shows the structure of RNN.

## Ensemble Learning

Ensemble learning merges various weak learners into a stronger predictor by combining outputs from multiple ML models to improve prediction accuracy [56]. From the perspective of sample diversity, ensemble learning focuses on enhancing the learning capability. Algorithms such as bagging and boosting [57] are common ensemble methods. In bagging, training samples are chosen randomly from the original dataset, and multiple models are trained; the rolling average or majority voting produces the final prediction. Boosting adjusts sample weights iteratively to focus on higher-error samples and thus improves model performance [58]. The ensemble mechanism is illustrated in Fig. 2.8 (a).

## Extreme Learning Machine

The Extreme Learning Machine (ELM) is an advancement over the traditional single-hidden-layer feedforward neural network. ELM sets hidden layer neuron thresholds and input layer connection weights at random and does not change them during training. Its simple structure, fast training time, and improved generalization have led to its use in wind power prediction studies. Random ELM

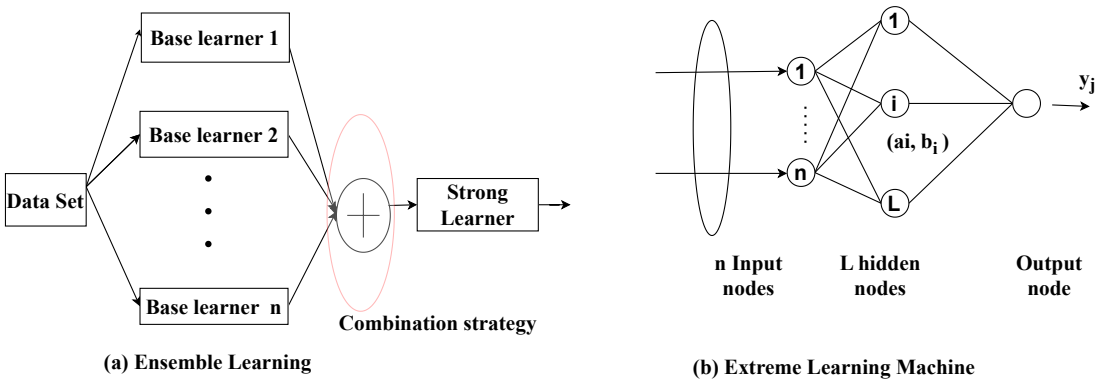



FIGURE 2.8: Ensemble Learning and ELM architectures.

input settings impact prediction results. Some researchers have modified ELM's input parameters to improve prediction accuracy. For example, [59] employed a Genetic Algorithm (GA) to optimize ELM's input weights; [60] and [61] used artificial fish swarm and salp swarm algorithms, respectively. Fig. 2.8 (b) shows the basic diagram of ELM.

## Support Vector Machine

Cortes and Vapnik proposed the Support Vector Machine (SVM) in 1995 [62]. SVM is a type of supervised ML algorithm which can be utilized for both classification and regression. SVM identifies a hyperplane that separates two data types. This hyperplane consists of nothing more than a two-dimensional line. SVM plots each data point in the dataset in an N-dimensional space [48]. The optimal separating hyperplane is determined using quadratic programming techniques.

Support Vector Regression (SVR) is used for regression tasks. A kernel is a measure of similarity between two data points. In a kernelized SVM, the kernel function indicates the degree of resemblance between two data points in the original feature space and points in the transformed feature space. Two prevalent kernel functions are the radial basis function kernel and the polynomial kernel [63]. The structure of the SVM is shown in Fig. 2.9 (a).

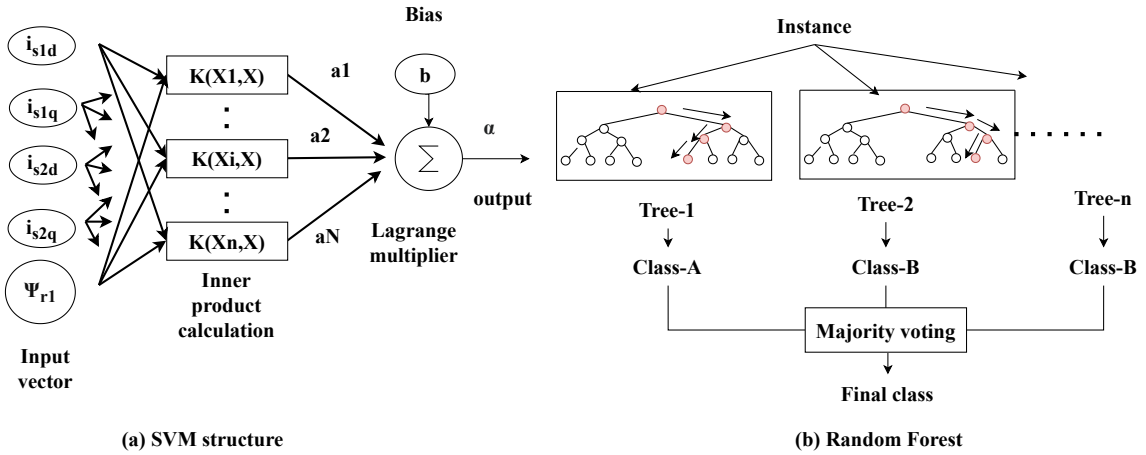



FIGURE 2.9: Ensemble Learning and ELM architectures.

## Random forest

Random Forest (RF) is a supervised learning algorithm that uses randomization, ensemble methodology, and multiple decision trees trained with bagging techniques [64]. The most significant benefits of using RF are feature importance estimation, pattern discovery, outlier detection, and interpretability. RF is frequently utilized for predicting wind energy due to its simplicity and flexibility. Fig. 2.9 (b) shows the architecture of Random Forest.

### 2.3.2 Deep Learning techniques

Deep Learning (DL) is a subset of ML built on ANN architectures that attempt to learn hierarchical features similarly to the human brain [65]. DL employs multiple hidden layers to convert low-level features into abstract high-level representations [66]. The main benefits of DL include (1) the ability to learn representations without strong supervision, (2) improved generalization when large datasets are available, and (3) reduced reliance on manual feature engineering [67]. DL-based frameworks for forecasting wind power have shown strong accuracy in numerous studies. Prominent DL architectures include Convolutional Neural Networks (CNN), Stacked Auto-Encoders (SAE), Deep Belief Networks (DBN), and Generative Adversarial Networks (GAN).

### Stacked Auto-encoder

A Stacked Auto-Encoder (SAE) is a deep neural network composed of multiple autoencoders stacked sequentially [68]. An autoencoder (AE) consists of an input layer, a bottleneck hidden layer, and a reconstruction layer. SAE extracts salient features by compressing inputs into low-dimensional representations and reconstructing outputs with minimal error [69]. SAEs are commonly used as feature extractors.

### Deep Belief Network

A Deep Belief Network (DBN) is a generative deep architecture often constructed from stacked Restricted Boltzmann Machines (RBMs) or autoencoders [70]. In an RBM, neurons within the same layer do not connect, but neurons in adjacent layers are bidirectionally connected [71]. DBNs are effective at extracting complex features useful for wind energy prediction.

### Convolution Neural Network

CNNs employ convolution and pooling operators and can perform supervised and unsupervised learning. They are efficient at representation learning due to parameter sharing and hierarchical feature extraction [72]. CNNs support shift-invariant pattern extraction and have been successfully applied in recent wind power prediction studies [73].

### Generative Adversarial Network

GANs are unsupervised learning frameworks comprising a generator and a discriminator; both components are trained adversarially so that the generator learns to produce realistic samples while the discriminator learns to distinguish real from fake [74]. GANs can capture data structure even without labels, which helps automatic feature extraction for sequence modeling [75, 76].

### 2.3.3 Hybrid forecasting Methods

Hybrid forecasting approaches can improve forecasting performance by combining the advantages of different methods that perform variably across datasets, horizons, and features. These hybrid approaches often outperform single models. A primary benefit is leveraging complementary strengths of different models: since each model may excel under particular conditions, combining models reduces the need for exhaustive model selection. Table 2.5 summarizes hybrid methods in the literature. The most common strategy combines models via weighted coefficients reflecting prior performance.

## Weighting based hybrid approaches

The weighting technique assigns weights to each model in order to reflect individual contributions to the combined forecast.

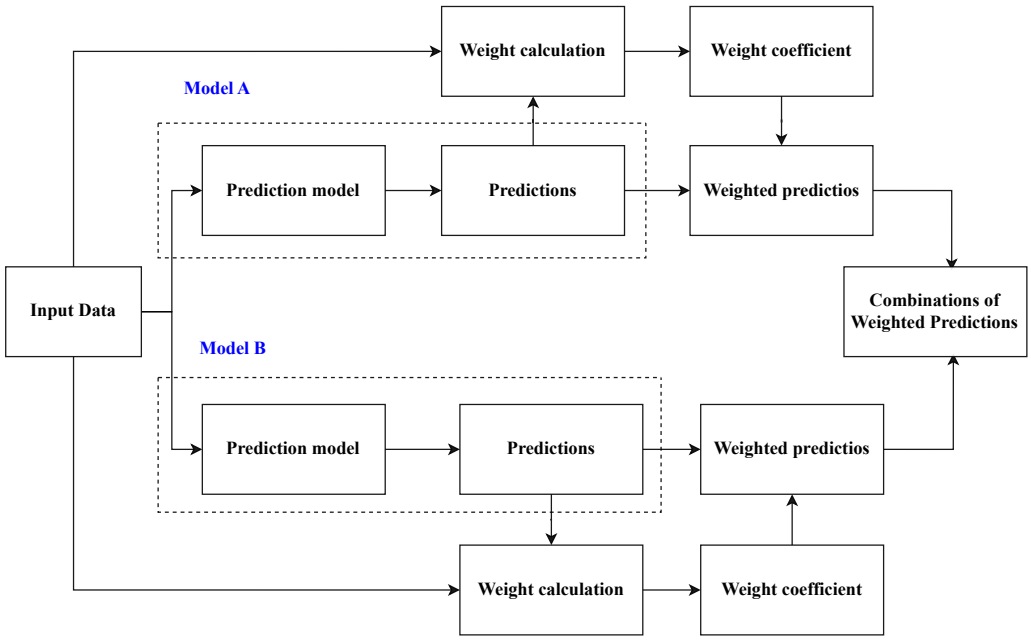



FIGURE 2.10: Flowchart of weight based hybrid approaches.

Inputs to hybrid models are often lagged series of the primary resource measurement (e.g., wind speed) and may include auxiliary meteorological features such as wind direction, temperature, pressure, and humidity [38]. Eq.

**Table 2.5** Summary of hybrid approaches for wind power forecasting

| Ref | Hybrid methods | Time resolution | Parameter used | Accuracy | Research gap/ future studies |
|---|---|---|---|---|---|
| [77] | FFBPNN, SAE, PSO | 15minutes | Historical time-series data | MAPE-15.96% | Deep learning in regressions can be utilized to achieve better outcomes |
| [78] | FFBPNN, ADE | | Historical time-series data | $R^2$ 99.22% | Further exploration is required to tackle complex computation difficulties in a cost-effective way |
| [79] | FFBPNN, SSA, BAT | 10min, 30min, 60min | Wind speed in different horizons | RMSE of 1.7332m/s and 2.0424m/s in one-step and three-step horizon, respectively | Multi-objective optimization algorithms and the development of chaotic time series can be employed to improve the accuracy of wind speed forecasting |
| [80] | RBFNN,NSFM,TS | 10min | wind speed, direction and temperature | MAE 18.9955kW inFeb. MAE 18.9325 in july | The fault detection and diagnostics may be addressed using the proposed model |
| [81] | ELM, KMPE, PCA | 12-hour, 24-hour | wind time-series data | MAPE 0.0914 m/s | Further investigation is required using the deep structure to address the complex time-series problems |
| [82] | ELM, GREY model | 24-hour | wind speed, height of wind turbine | $R^2$ 0.99376% | Further studies can be carried out using deep learning algorithms with better computer configuration |
| [83] | ELM, CRO, HS | 1-hour | Metereological variables | RMSE 3.463m/s | Further exploration is required to overcome the feature slection problem |
| [63] | SVM, Jaya | 1-hour | Historical wind speed data | MSE 0.8623m/s | Seasonal factors should be considered in future reasearch |
| [84] | SVM, ACO | 1-hour | wind speed data series | MSE 1.6m/s | Improved ant colony algorithm can be utilized to enhance the efficiency |
| [30] | SVM,FOA | 15min | wind speed, wind direction, temperature and atmospheric pressure | RMSE 14.23% | Other regression methods can be utilized to reduce the forecasting error |
| [85] | SVR, FOA, EEMD,IMF | 15min | wind speed data | RMSE 3.04% | Non linear approximation problems |
| [86] | SVM, BSA | 1-hour | Wind speed data | $R^2$ 0.9648 | Feature selection still depends on the experiment |
| [87] | SVR,HCS | 10min | Humidity, temperature, relative wind direcation, wind speed | MAPE 7.03% | Abnormal data points still influence the outcome |
| [88] | SVM, BSA | 1-hour | Annual wind power capacity | MAPE 4.27% | Weight determination is the main problem of the combined model |
| [50] | RF,DBN | 15-min,30min,45-min and 24h | Historical SCADA Data | RMSE 0.092m/s | Booming problems are increasing with the amount of data |
| [89] | GPR, SVM, MLP | 5min | Temperature(T), humidity(H) and Pressure(PRE), Wind power(P) | RMSE 1.5660m/s | Problem under the missing data scenario should be broadly investigated |
| [90] | GPR,TL | 10min, 15min | Historical power data | RMSE 1.6991m/s | Many irrelevant data are used which is responsible for the computational burden |
| [91] | GPR, EMD, IMD | 15mins | Historical data | RMSE 0.6159 | The proposed approach can be utilized for online wind energy forecasting |
| [19] | GPR,SC | 1hour | Wind speed, direction and temperature, humidity and pressure | RMSE 2.654m/s | Issue of automatic scene division for complex input data |

(2.1) shows a general weighted-combination formulation:

$$\hat{Y}t + h|t = \sum i = 1^{N} w_i, \hat{Y}i, t + h|t \tag{2.1}$$

where $\hat{Y}t + h|t$ is the final forecast for time $t + h$ made at time $t$, $\hat{Y}_{i,t+h|t}$ is the forecast value of the $i$-th model and $N$ is the number of models. Hybrid methods are named based on combination strategy (regression combination, varied-weight combination, equally weighted, etc.). Assuming the hybrid model combines two or more constituent predictors trained on the same forecast horizon [92], model accuracy is used to derive weights. As shown in Fig. 2.10, models receive coefficients according to past performance and predictions are aggregated.

## Other combined approaches

These methods involve two models where one performs core forecasting and the other handles auxiliary tasks such as data filtering, decomposition, parameter selection, or residual modeling. Such combined approaches divide into three categories [38]: (i) combined approaches with data preprocessing, (ii) combined approaches with optimization/parameter selection, and (iii) combined approaches with post-processing/error modeling.

(i) **Combined approaches including data preprocessing techniques.** Data preprocessing filters irrelevant features and decomposes non-linear series into more regular sub-series to improve stability and reduce computational cost [10]. Fig. 2.11 depicts a typical preprocessing-based hybrid flow.

(ii) **Combined approaches including parameter selection and optimization techniques.** Parameter selection and optimization significantly affect training performance. These methods search parameter spaces to avoid manual, time-consuming tuning. Fig. 2.12 outlines such procedures.

(iii) **Combined approaches including error processing techniques.** Some hybrids explicitly model residuals from a primary predictor and use a secondary model to correct systematic biases. Fig. 2.13 illustrates this approach.

(iv) **Components combination based hybrid methods.** Two primary structural classes exist:

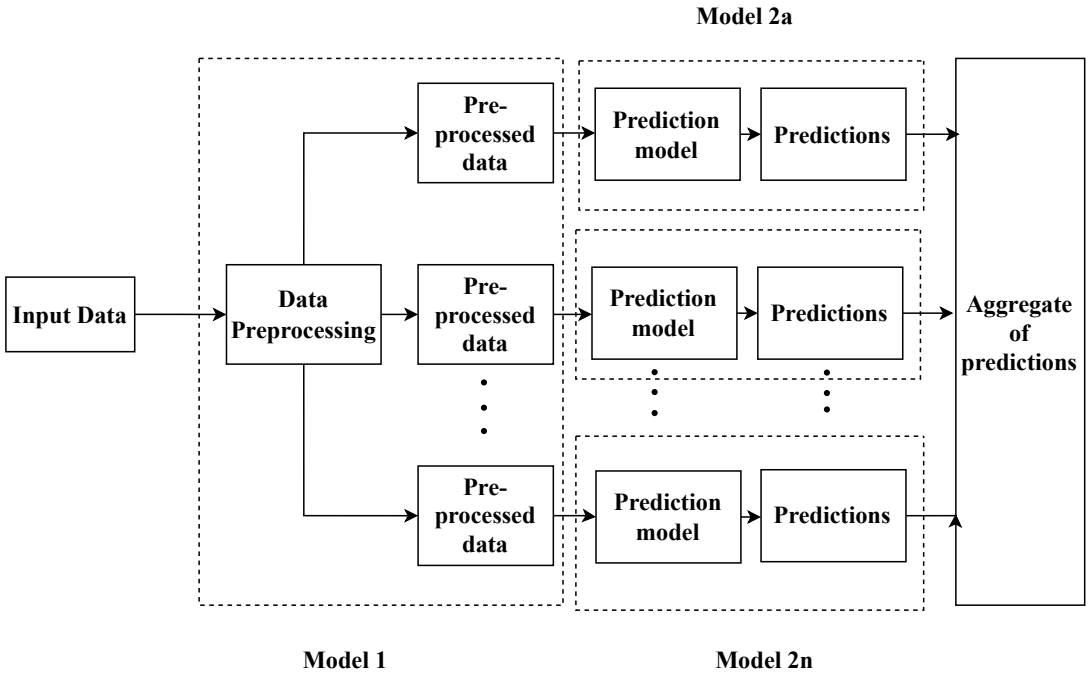



FIGURE 2.11: Schematic depicting the combined approaches including data preprocessing methods

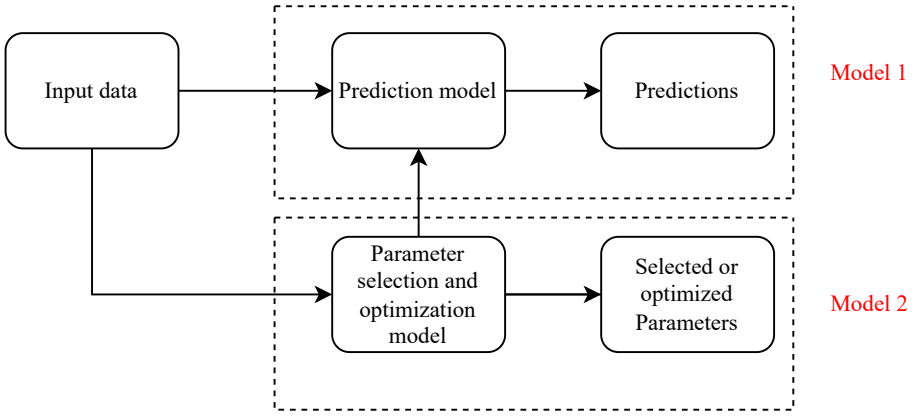



FIGURE 2.12: Schematic depicting the combined approaches including parameter selection and optimization

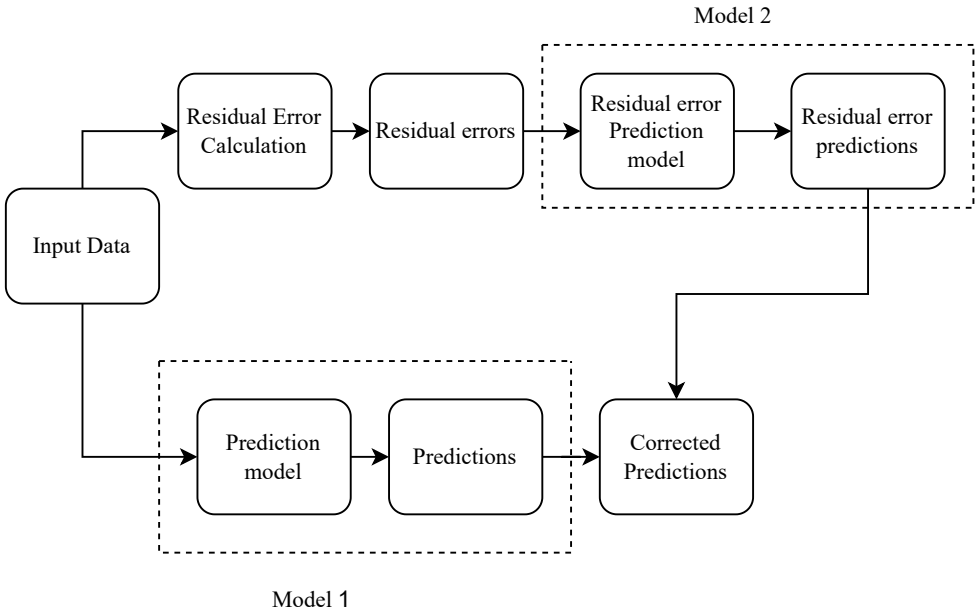



FIGURE 2.13: Schematic of combined approaches including error processing techniques

**(a) Series hybrid models.** Components are applied sequentially; the residual of one becomes the input to the next. The generalized n-component series structure appears in Fig. 2.14. The final forecast can be represented by Eq. (2.2):

$$y_t = \sum_{i=1}^{n_1} L_t + \sum_{i=n_1}^{n} N_t \tag{2.2}$$

where $n_1$ is the number of linear components, and $L_t$ and $N_t$ represent linear and nonlinear components respectively, yielding $y_t = L_t + N_t$ [10].

**(b) Parallel hybrid models.** Each component receives the raw input; component forecasts are combined (often linearly) to produce the final output. The parallel hybrid prediction can be written using Eq. (2.3) :

$$\begin{aligned} \hat{f}_{\text{pre},t} &= w_1\hat{f}_{1,t} - w_2\hat{f}_{2,t} - w_{n-1}\hat{f}_{n-1,t} - w_n\hat{f}_{n,t} \\ &= \sum_{i=1}^{n} w_i\hat{f}_{i,t}, \qquad (i = 1, 2, \ldots, n;\ t = 1, 2, \ldots, m). \end{aligned} \tag{2.3}$$

where $w_i$ is the weight of the $i$-th component at time $t$, $\hat{f}pre, t$ is the forecast of

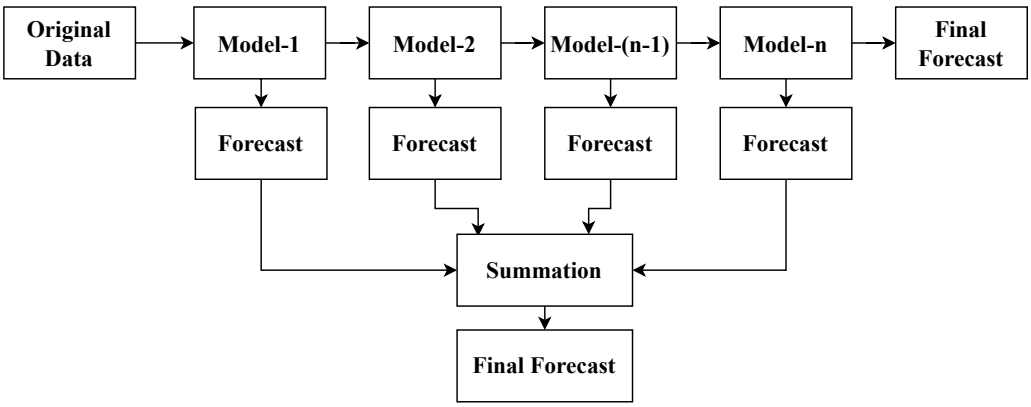



FIGURE 2.14: General architecture of series hybrid model

the hybrid model, $m$ is the dataset length and $n$ is the number of components [10]. Fig. 2.15 shows the n-component parallel hybrid structure.

## 2.4 Optimization Frameworks for Forecasting Models

The aim of meta-heuristic algorithms is to determine best prediction model parameters under various restrictions, such as kernel bounds of Support Vector Machine (SVM), initial parameters of Artificial Neural Networks (ANN), or weights of a combined model [93]. In [94], for predicting real-time wind power generation, a hybrid prediction model composed of SVM and ANNs is proposed, with satisfactory outcomes. An identical prediction structure is applied in [95]; in that study, a novel approach to wind pattern recognition and Reference Wind Mast (RWM) data correlations is combined with NWP to provide promising short-term prediction results. In [96], Particle Swarm Optimization (PSO) was used to discover the best prediction intervals (PI) for ELM-based forecasts. Weight-improved PSO was also used to adjust network weights [97]. Genetic Algorithm (GA) has been used for selecting weights and thresholds for Backpropagation Neural Networks (BPNN) [98] and to optimize SVM parameters [99]. Gravity Search Algorithm (GSA) was used to optimize Least Squares SVM (LSSVM) kernel parameters in [100]. Other meta-heuristics such as Whale Optimization Algorithm (WOA) [101], Bat Algorithm (BA), Firefly Algorithm (FA) [102], Grey Wolf Optimizer (GWO) [103], and Artificial Bee Colony (ABC) [104] have been applied to renewable power prediction. Nature-inspired multi-objective

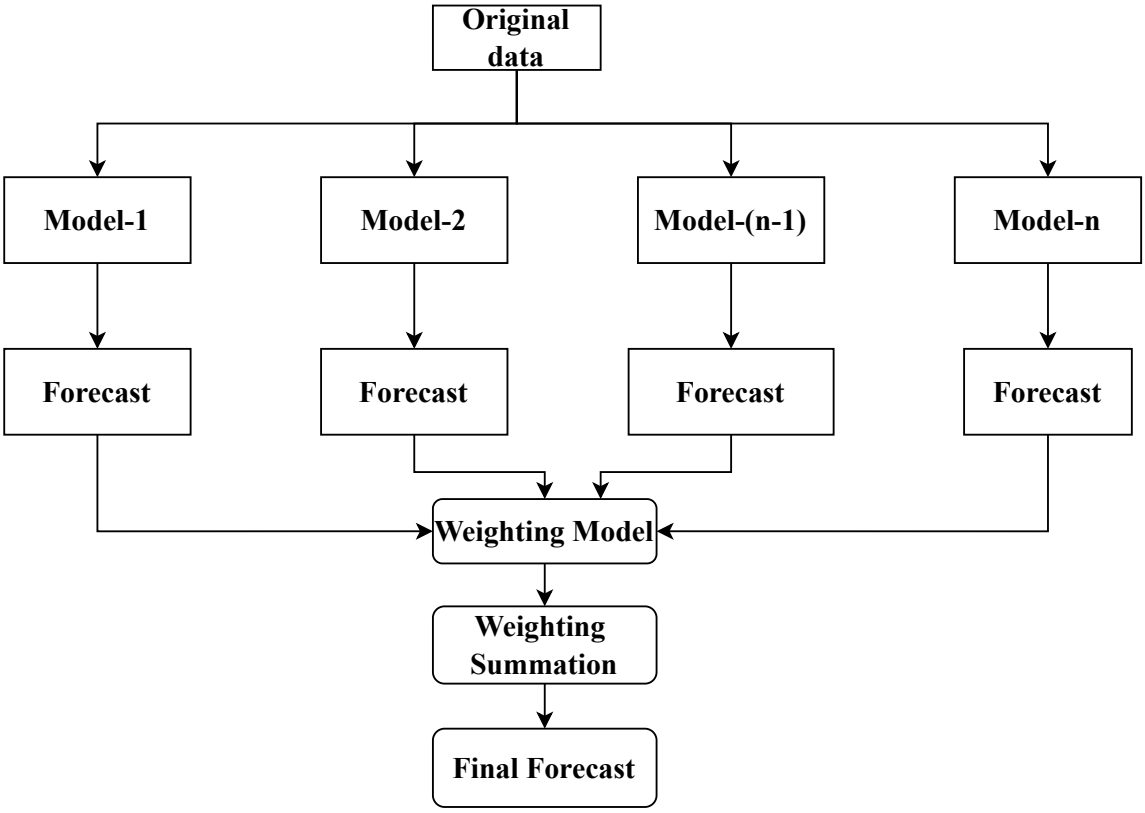



FIGURE 2.15: Generalized architecture of parallel hybrid model

optimizers-Non-dominated Sorting Genetic Algorithm II (NSGA-II) [105], Multi-Objective Particle Swarm Optimization (MOPSO) [106], and others-are used when accuracy, reliability, and stability must be balanced.

### 2.4.1 Meta-Heuristic Optimization Structure

**The mathematical relationship** For a typical renewable power prediction problem, a trained prediction model $f(\cdot)$ can be formulated as per Eq. (2.4) [7]:

$$p_t^{pre} = \underbrace{f}_{\text{prediction model}}\left(\underbrace{\theta}_{\text{unknown parameters}}, x_t\right) \tag{2.4}$$

where $x_t$ and $\theta$ are the inputs and unknown parameters respectively, and $f(\cdot)$ represents any AI-based prediction model. These unknown parameters $\theta$ in equation (2.4) are crucial for the prediction model. In general, a single-objective or multi-objective function may be optimized to determine the model's unknown parameters. A popular weight-based combination model is represented by Eq. (2.5):

$$p_{c,t}^{pre} = \sum_{k=1}^{K} \underbrace{\omega_{k,t}}_{\text{weights}} pk, t^{pre} \tag{2.5}$$

where $p_{k,t}^{pre}$ is the prediction of the $k$-th model at time $t$.

**Overall structure description** [107] proposed a meta-heuristics framework for forecasting (Fig. 2.16) composed of three layers: a forecasting base layer, an auxiliary data-decomposition layer, and an optimization core layer. The auxiliary layer cleans and conditions real-world measurements while the core layer selects optimal weights and thresholds to satisfy Eq. ((2.4)).

### 2.4.2 Different Optimization Algorithms

Since the first evolutionary meta-heuristic of the 1970s, numerous families of optimizers have emerged-evolutionary, swarm-based, physics-based, human-inspired, and hybrid approaches [108–111]. GA and PSO remain primary benchmarks. In recent years, newer optimizers such as differential evolution (DE), grey wolf optimizer (GWO), and whale optimization algorithm (WOA) have shown strong performance across nonlinear engineering applications. These

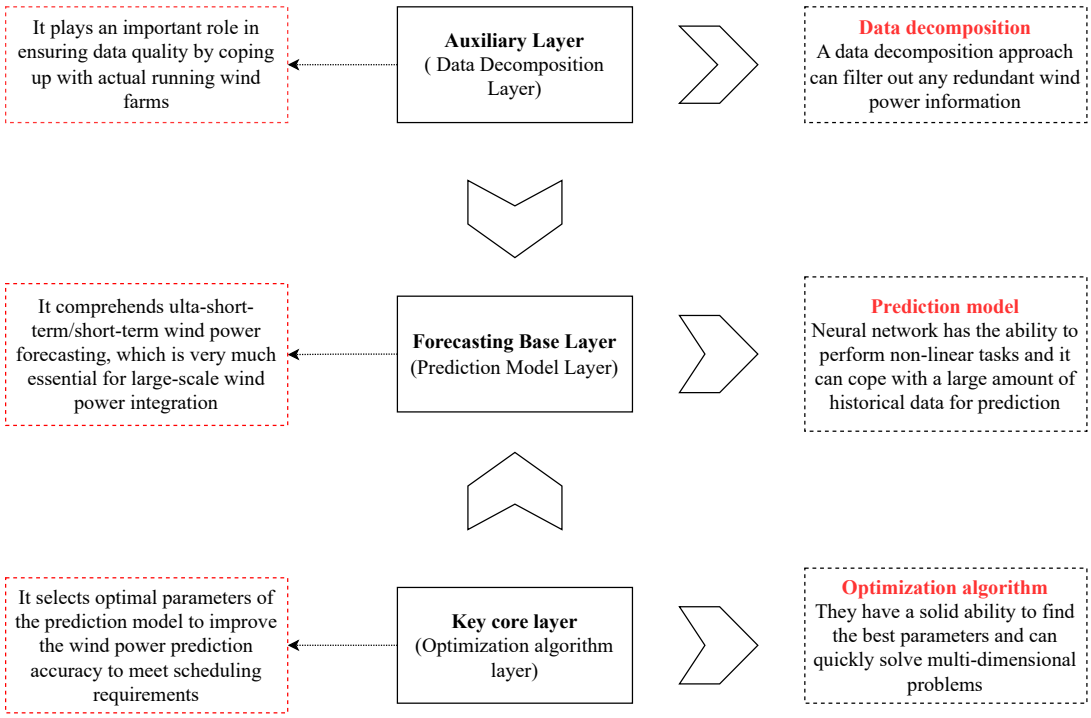



FIGURE 2.16: Overall framework of metaheuristic algorithm

methods continue to evolve, offering improved convergence speed, better global search capabilities, and enhanced robustness for complex optimization problems. Table 2.6 summarizes optimization algorithms applied in renewable power forecasting.

**Table 2.6** Summary of Optimization methods for parameter selection

| Optimization methods | Ref | Merits | Demerits |
|---|---|---|---|
| ***Physics-based algorithm*** | | | |
| Intelligent Water Drops Algorithm (IWD) | [112] | • can effectively jump out of the local optimal solution<br>• can provide approximate optimal solution | • high probability of global search capability<br>• update object is single |
| Simulated Annealing (SA) | [113] | • independent of the initial solution<br>• obtains the global optimum in large-scale problems. | • computanionally exhaustive |
| Gravitational Search Algorithm (GSA) | [114] | • robust initialization<br>• able to explore local solutions | • local search ability is quite weak |

(continued on next page)

(Continued from preceding page)

| Optimization methods | Ref | Merits | Demerits |
|---|---|---|---|
| ***Human-based algorithm*** | | | |
| Harmony search(HS) | [115] | • simple in concept<br>• easy to be implemented<br>• suitable convergence speed | • More computational burden and complexity<br>• dependent on harmony memory considering the rate |
| Tabu search(TS) | [116] | • Fast convergences<br>• suitable for simple and uncritical problems<br>• easy to tune the controller parameters | • need an appropriate initial solution. |
| Group search optimization(GSO) | [117] | • Fast convergence<br>• better global search ability and robustness. | • easy to fall into local optimum " premature" phenomenon |
| Teaching and learning algorithm(TALA) | [118] | • a good capability for global and local searching<br>• Balanced global search ability and convergence rate | • easy to be trapped locally and has a poor global search ability<br>• Need more support for exploration features |
| ***Swarm-based algorithm*** | | | |
| Particle Swarm Optimization(PSO) | [119, 120] | • fewer controller parameters than IPSO<br>• very effective for global search optimization | • tuning the controller parameters is a challenge<br>• local search ability is weak |
| Ant-colony optimization(ACO) | [121] | • need less computation time in simple planning | • unable to achieve the global optimum<br>• tuning the controller parameters is a challenge |
| Artificial bee colony(ABC) | [122] | • robust initialization<br>• able to explore local solutions<br>• simple implementation | • may get stuck in local optimum<br>• poor exploitation characteristics |
| firefly algorithm | [123, 124] | • simple implementation<br>• good at exploration | • poor iteration performance<br>• traps into local optima<br>• single goal optimization<br>• easy to fall into the extreme value |
| ***Evolutionary-based algorithm*** | | | |
| Differential evolution(DE) | [125] | • cannot be easily trapped in local minima<br>• less control parameters | • may drop in local best<br>• parameter tuning mostly by trial and error |

(continued on next page)

(Continued from preceding page)

| Optimization methods | Ref | Merits | Demerits |
|---|---|---|---|
| Genetic algorithm(GA) | [98] | • independent of initial solution | • not reliable for large scale networks<br>• Convergence speed is slower than other stochastic algorithms |
| Genetic Programming(GP) | [126] | • easy to use<br>• Suitable convergence speed | • search operator space is too large |

## 2.5 Uncertainty-Aware and Spatio-Temporal Forecasting Techniques

The increasing penetration of variable renewable energy sources (VRES), particularly wind and solar, has brought unprecedented levels of uncertainty into modern power systems. Unlike conventional generation, renewable output is governed by stochastic meteorological processes that vary across multiple time scales and geographical locations. As microgrids evolve toward higher renewable penetration, forecasting frameworks must not only provide accurate point predictions but also quantify uncertainty and capture spatio-temporal dependencies that govern renewable variability. This section reviews three major methodological pillars-probabilistic forecasting, conformal prediction-based uncertainty quantification, and spatio-temporal learning architectures-that form the foundation of advanced forecasting methodologies for renewable-rich microgrid environments.

### 2.5.1 Probabilistic Forecasting Approaches

Deterministic forecasts provide only a single predicted value for future renewable generation, offering no insight into the range or likelihood of alternative outcomes. In contrast, **probabilistic forecasting** aims to estimate the full predictive distribution or specific quantiles of the generation output. Such forecasts are essential for risk-aware decision-making in microgrid scheduling, reserve allocation, demand response, and economic dispatch. Probabilistic forecasts enable operators to quantify risks associated with under- or overestimation,

articulate confidence bounds, and integrate uncertainty into stochastic optimization or chance-constrained planning models.

The literature on probabilistic renewable generation forecasting can be categorized into three major methodological families: physical and ensemble approaches, statistical and machine learning techniques, and hybrid models that integrate complementary strengths of multiple methods.

Recent advances have focused on improving the trade-off between **sharpness** (narrow intervals) and **calibration** (observed coverage matching nominal coverage). Deep quantile models combined with attention mechanisms and ensemble learning strategies have demonstrated improved calibration in rapidly changing meteorological regimes. Meanwhile, nonparametric methods like KDE and Gaussian copula modeling have been widely employed to capture non-Gaussian error structures, which are characteristic of renewable generation.

As renewable forecasting expands from single-site to regional microgrids, probabilistic methods increasingly incorporate spatial information from neighboring sites, leading naturally to the next paradigm: uncertainty quantification using conformal prediction.

### 2.5.2 Conformal Prediction-Based Uncertainty Quantification

Although probabilistic models estimate predictive distributions, their coverage reliability depends heavily on model assumptions and correct specification of error distributions. **Conformal Prediction (CP)** provides a distribution-free, model-agnostic framework that produces prediction intervals with rigorous, finite-sample coverage guarantees. A representative summary is provided in Table 2.7. CP has therefore emerged as a powerful calibration layer that can wrap around any base model-including linear regressors, gradient boosting machines, LSTM/GRU networks, Transformer architectures, or hybrid spatio-temporal models-to ensure statistical reliability.

#### Fundamental Principles of Conformal Prediction

CP operates by measuring how "nonconforming" or "atypical" a prediction is compared to a calibration dataset. Using a nonconformity score such as the absolute residual, which can be calculated using Eq. (2.6):

**Table 2.7** Brief comparison of probabilistic renewable generation forecasting methods

| Forecasting approach | Advantages | Disadvantages |
|---|---|---|
| Physical and ensemble approaches [39] | • Physically consistent with atmospheric dynamics<br>• Utilize NWP ensemble inputs<br>• Effective for day-ahead to multi-day horizons<br>• Capture large-scale atmospheric uncertainty | • Computationally expensive<br>• Ensemble spread often uncalibrated<br>• Limited skill for very short-term horizons |
| Quantile Regression (QR) [127] | • Simple and distribution-free<br>• Adaptable to regime changes<br>• Supported by linear, tree-based, and neural models | • Independent quantiles may cross<br>• No inherent calibration guarantees<br>• Performance depends on quantile grid design |
| Deep learning quantile models [128] | • Capture nonlinearities and temporal dependencies<br>• Strong multi-horizon performance<br>• Can output multiple quantiles simultaneously | • Require large training datasets<br>• Sensitive to hyperparameters and seasonality<br>• Risk of overfitting without regularization |
| Kernel Density Estimation (KDE) [129] | • Fully non-parametric<br>• Flexible density shapes<br>• Effective for short-term error distribution modeling | • Sensitive to bandwidth selection<br>• Poor performance with sparse data<br>• Limited scalability to high-dimensional inputs |
| Bayesian Neural Networks (BNNs) [130] | • Provide principled epistemic and aleatoric UQ<br>• Naturally output predictive distributions<br>• Offer interpretable uncertainty estimates | • Computationally intensive<br>• Require approximate Bayesian inference<br>• Less suitable for real-time applications |
| Hybrid probabilistic models [131] | • Combine physical, statistical, and deep learning components<br>• More robust under diverse and dynamic weather regimes | • High implementation complexity<br>• Slow training and inference<br>• Require careful integration of multiple modules |

$$s_i = |y_i - \hat{y}_i| \tag{2.6}$$

CP computes a threshold from the $(1-\alpha)$ quantile of these scores on a held-out calibration set. The prediction interval for a new input $x$ becomes Eq. (2.7):

$$\Gamma_\alpha(x) = \left[\hat{y}(x) - q_{1-\alpha},\ \hat{y}(x) + q_{1-\alpha}\right] \tag{2.7}$$

where $q_{1-\alpha}$ ensures that the resulting interval contains the true value with coverage probability at least $(1-\alpha)$.

Because CP does not assume any specific error distribution (e.g., Gaussian), it is particularly effective for renewable time series, where forecast errors are often skewed, heavy-tailed, or regime-dependent.

### Variants of Conformal Prediction

Several CP variants have been introduced to handle different forecasting contexts, as summarized in Table 2.8.

### Lag-Aware CP for Ramp Events

Standard CP assumes exchangeability of calibration data, which is often violated in renewable generation during **ramp events**-periods of rapid increase or decrease in output due to fast-moving clouds or wind fronts. To address this, a lag-aware nonconformity score can be used using Eq. (2.8):

$$A_t^{(L)} = \frac{1}{L}\sum_{i=0}^{L-1} |\epsilon_{t-i}| \tag{2.8}$$

where $\epsilon_{t-i}$ are recent residuals. This score penalizes sequences of consistent underpredictions or overpredictions, widening prediction intervals when the system enters volatile regimes while retaining sharpness during stable operation.

### Integration with Deep Spatio-Temporal Models

Conformal prediction can be combined with advanced deep learning architectures (e.g., ConvLSTM, Graph Neural Networks (GNNs), Transformers) to yield

**Table 2.8** Brief comparison of conformal prediction approaches for renewable generation forecasting

| Conformal prediction approach | Advantages | Disadvantages |
|---|---|---|
| Full Conformal Prediction [132] | • Statistically most efficient CP variant<br>• Uses the full dataset without hold-out calibration<br>• Produces the tightest valid prediction intervals | • Computationally infeasible for modern ML/DL models<br>• Requires model retraining for each test instance |
| Cross-Conformal Prediction [132] | • Better data utilization than split CP<br>• Produces more stable and narrower intervals<br>• Reduces calibration inefficiency through cross-fold aggregation | • More complex to implement<br>• Still computationally heavier than split CP |
| Conformalized Quantile Regression (CQR) [133] | • Produces sharp, adaptive prediction intervals<br>• Combines flexibility of quantile regression with formal CP coverage<br>• Robust under regime shifts and heterogeneous uncertainty | • Quantile crossing may occur<br>• Requires a separate calibration stage<br>• Performance depends on base quantile model quality |
| Adaptive Conformal Prediction (ACP) [134] | • Adapts to temporal drift and nonstationarity<br>• Supports rolling-window and weighted nonconformity scoring<br>• Suitable for rapidly changing renewable generation patterns | • Requires careful tuning of window size / weights<br>• May overreact to short-term noise fluctuations |
| Multivariate Conformal Prediction (MCP) [135] | • Generates joint prediction regions across horizons or sites<br>• Preserves spatio-temporal and cross-variable dependencies | • Computationally demanding in high dimensions<br>• Designing multivariate nonconformity scores is challenging |
| Physics-Informed CP (C-PINN / constrained CP) [136] | • Integrates physical constraints (e.g., turbine/irradiance limits)<br>• Ensures physically feasible intervals with statistical validity | • Requires differentiable physical or constraint models<br>• Increases system and model integration complexity |

calibrated spatio-temporal uncertainty estimates. This forms the basis for next-generation **physics-informed adaptive conformal prediction** frameworks in microgrid forecasting.

### 2.5.3 Spatio-Temporal Forecasting Techniques

Renewable generation exhibits pronounced dependencies across both space and time. Weather systems, such as moving cloud fronts or wind fields, evolve continuously across geographical regions, resulting in correlated power outputs among spatially distributed assets. Traditional univariate forecasting models ignore these dependencies. The need to model joint evolution across multiple locations has led to the development of **spatio-temporal forecasting** techniques.

Early spatio-temporal models used geostatistical methods such as Spatio-Temporal Kriging and multivariate time-series models like Vector Autoregression (VAR). These methods explicitly modeled spatial covariance and autoregressive interactions but were limited by their assumptions of linearity and stationarity. Deep learning has since revolutionized spatio-temporal forecasting through its superior ability to learn nonlinear patterns from large datasets.

Modern spatio-temporal forecasting models increasingly integrate uncertainty quantification. Emerging work incorporates **conformal prediction** into GNNs or Transformer-based models to create calibrated, physically constrained spatio-temporal prediction intervals for multi-site microgrid forecasting.

Overall, the integration of probabilistic modeling, conformal prediction, and deep spatio-temporal learning represents the current frontier in renewable generation forecasting. Table 2.9 summarizes representative spatio-temporal architectures used in renewable forecasting. These approaches collectively enable microgrids to operate more reliably under uncertainty while fully exploiting distributed renewable resources.

## 2.6 Forecast Evaluation metrics

Evaluation metrics are often used to see if predictions are realistic and how they work in different situations. This can help forecast models get better. This is also called “predictive analysis”, and it is usually done using other measurements or

**Table 2.9** Brief comparison of spatio-temporal forecasting models for renewable generation

| Spatio-temporal model | Advantages | Disadvantages |
|---|---|---|
| CNN–LSTM hybrids [137] | • CNN extracts spatial features; LSTM models temporal dynamics<br>• ConvLSTM integrates convolutions within recurrent units<br>• Strong for grid-structured inputs (e.g., satellite imagery) | • Assumes regular grid layouts<br>• Computationally expensive<br>• Weak for irregular asset spacing |
| Attention-based spatio-temporal models [138] | • Highlight key spatial/temporal inputs<br>• Capture long-range dependencies<br>• Attention maps provide interpretability | • Require large datasets<br>• Risk of overfitting without regularization<br>• Attention layers increase memory/computation |
| Graph Neural Networks [139] | • Naturally model irregular renewable site layouts<br>• Capture non-Euclidean spatial correlations<br>• Scalable to large networks | • Require careful graph construction<br>• Sensitive to hyperparameters<br>• Depend on accurate adjacency definition |
| GNN–RNN hybrids [140] | • GNNs learn spatial correlations<br>• LSTM/GRU models temporal evolution<br>• Effective for multi-site microgrids | • Architecturally complex<br>• Higher training/inference cost<br>• Require extensive tuning |
| Transformer-based spatio-temporal models [141] | • Multi-head attention learns long-range dependencies<br>• Flexible input handling<br>• Strong multi-horizon performance | • Resource-intensive<br>• Often require large datasets<br>• Need strong regularization |
| Physics-informed spatio-temporal networks [142] | • Embed physical laws within neural architectures<br>• Improve interpretability and physical consistency | • Hard to encode accurate physical constraints<br>• Require differentiable physics models<br>• Increase model complexity |

graphs. Also, forecasts are evaluated so that different forecasts can be compared. A lot of work has gone into making prediction models, and several models have been suggested for different kinds of predictions and uses. They include **Probabilistic forecasts**, **Deterministic point forecasts**, or forecasts of some specific events such as ramps events and multivariate predictions [143]. They include of various forms, , and multivariate predictions or predictions for specific events such as ramps

### 2.6.1 Evaluation of Deterministic Forecasts

There are many ways to measure the deviation between the actual and predicted values so to determine how accurate deterministic estimates are. This subsection analyzes a set of point forecasts $\hat{Y}_t$ to corresponding actual observations $Y_t$ where, $t = 1, ..., N.$ , and $N$ is the number of observations. Certainly, a good forecast $\hat{Y}_t$ should be as close to $Y_t$ as possible. There are several methods for measuring the difference between forecasts and observations, i.e., forecast accuracy [144].

$$BIAS = \frac{1}{N}\sum_{t=1}^{n} \hat{Y}_t - Y_t \tag{2.9}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{t=1}^{N}(\hat{Y}_t - Y_t)^2} \tag{2.10}$$

$$MAE = \frac{1}{N}\sum_{t=1}^{N}\left|\hat{Y}_t - Y_t\right| \tag{2.11}$$

$$MAPE = \frac{1}{N}\sum_{t=1}^{N}\left|\frac{(Y_t - \hat{Y}_t)}{Y_t}\right| \times 100\% \tag{2.12}$$

$$SDE = \sqrt{\frac{1}{N}\sum_{t=1}^{N}(c_t - \bar{c})^2} \tag{2.13}$$

$$IA = 1 - \frac{\sum_{t=1}^{N}(Y_t - \hat{Y}_t)^2}{\sum_{t=1}^{N}\left(\left|\hat{Y}_t - \bar{Y}\right| + \left|Y_t - \bar{Y}\right|\right)^2} \tag{2.14}$$

$\bar{y}$ is mean value of the real observations , and $\bar{c}$ is average value of the errors. Summary of point forecast metrics for wind power prediction utilized in the literature is presented in Table 2.10.

The BIAS (Eq.2.9) deals with the systematic component of the prediction error and Eq. (2.10) shows the Root Mean Squared Error (RMSE) which is the squared deviation from the arithmetic mean. Eq. (2.11), Mean Absolute Error (MAE), is the average of absolute prediction errors. NMAE and the NRMSE are also frequently used in the literature. Although, both MAE and RMSE are good enough indicators to check the accuracy of a model, when the error follows a Gaussian distribution RMSE should be preferred [145]. Eq. (2.12) shows MAPE which expresses accuracy as a percentage of error.

When the actual values are close to zero, the MAPE returns huge values and it becomes undefined when the actual values are equal to zero. Although, this shortcoming has been addressed by the Alternative MAPE formulations. For example, The Mean Absolute Scaled Error (MASE) , is an another statistic which is calculated by multiplying MAE of the forecast values to the MAE of the in-sample naive-forecast [146]. It is specifically useful when the wind farm generates no power and the wind power time series has values which are zero. Another alternative option to MAPE is The mean arctangent absolute percentage error (MAAPE) [147]. The MAPE is modified using the arctangent function. It keeps the MAPE's characteristics while overcoming its limitations which is the main advantage of this function. The two type of prediction errors are random error and systematic error which is caused by systems imperfections. The BIAS (Eq.2.9) deals with the systematic component of the prediction error, though SDE (Eq. 2.13) deals with random component [148]. Lastly, The Index of Agreement (IA) is a popular metric for evaluating deterministic estimations in the literature.(Eq. 2.14). It was originally proposed in [149] and updated versions of this index have been developed since then [150]. It determines the accuracy of the forecasts and takes values ranging from zero to one, with zero indicating that there is no difference between the predictions and observations and one indicating that the predictions exactly match the actual values. Probabilistic statistical assessments that address not only model performance but also model complexity are an alternative method for evaluating model performance. These include the Bayesian Information Criterion (BIC), and Akaike Information Criterion (AIC), and Minimum

Description Length (MDL). The improvement of a model can be mathematically formulated using Eq. (2.15) which facilitates the comparison with benchmark models:

$$Improvement = 1 - \frac{M}{M_{ref}} \tag{2.15}$$

where $M$ is the value of the specific measure for a particular model, and $M_{ref}$ is the value of the same measure for the benchmark model. Comparison of multiple models becomes challenging due to the lack of a consistent criterion for setting evaluation standards. Models are typically evaluated against the persistence model, which must be defeated in order to be considered proficient, as well as cutting-edge techniques such as neural networks. Table 2.10 shows the application of these deterministic estimates in the literature.

**Table 2.10** Forecast performance indices for deterministic predictions

| Reference | Year | MAE | NMAE | RMSE | NRMSE | MAPE | SDE | Bias | IA | Others |
|---|---|---|---|---|---|---|---|---|---|---|
| [151] | 2019 | ✓ | | ✓ | | | | ✓ | | |
| [152] | 2019 | ✓ | | ✓ | | ✓ | | | ✓ | |
| [153] | 2018 | ✓ | | ✓ | | ✓ | | | | |
| [129] | 2018 | | | ✓ | | ✓ | | | | ✓ |
| [154] | 2018 | ✓ | | ✓ | | | | | | ✓ |
| [155] | 2017 | ✓ | | ✓ | | ✓ | | | | |
| [156] | 2017 | ✓ | ✓ | ✓ | | | | | | |
| [157] | 2017 | | | | | ✓ | | | | ✓ |
| [126] | 2017 | ✓ | | ✓ | | | ✓ | | | |
| [158] | 2017 | | ✓ | | ✓ | ✓ | | | | |
| [159] | 2017 | ✓ | ✓ | ✓ | | ✓ | | | | ✓ |
| [160] | 2017 | ✓ | | ✓ | | | ✓ | | | |
| [130] | 2017 | | ✓ | | ✓ | | | | | ✓ |

### 2.6.2 Evaluation of Probabilistic Forecasts

Probabilistic estimates are harder to evaluate because one cannot directly compare the forecast to the actual values. Instead, one may have to look at several properties of the forecast to see how accurate the model's forecast is.

A Prediction Interval (PI) is a range that shows where a forecast value is likely to fall, given a certain probability. Consequently, the PI depends on the level of significance $\alpha$. The probability with which a future value $Y_t$ lies within the PI is known as PI Nominal Confidence (PINC) [17] (Eq. 2.16).

$$PINC : 100(1-\alpha)\% \tag{2.16}$$

Keeping it into consideration, a PI for a significance level $\alpha$ and future time step $t$ is defined as per Eq. 2.17:

$$\hat{I}_t^{\alpha} = \hat{U}_t^{\alpha} - \hat{L}_t^{\alpha} \tag{2.17}$$

where $\hat{L}_t^{\alpha}$ and $\hat{U}_t^{\alpha}$ are the lower and upper boundaries of the PI respectively. The most frequently used evaluation criteria for probabilistic forecasts are:
- PI Coverage Probability (PICP):

$$PICP = \frac{1}{n}\sum_{i=1}^{n} c_i \tag{2.18}$$

where $n$ is the number of samples and $c_i$ is Calculated using Eq. 2.19

$$c_i = \begin{cases} 1, & \text{if } Y_t \in \hat{I}_t^{\alpha} \\ 0, & \text{otherwise} \end{cases} \tag{2.19}$$

-PI Normalized Average Width (PINAW)

$$PINAW = \frac{1}{nR}\sum_{i=1}^{n} \hat{I}_t^{\alpha} \tag{2.20}$$

where $R$ is the range of the target variable.
-Coverage-width based Criterion (CWC):

$$CWC = PINAW\left[1 + \gamma(PICP)e^{-\eta(PICP-\mu)}\right] \tag{2.21}$$

where $\gamma(PICP)$ is a step function dependent on the values of PICP and $\mu$:

$$\gamma(PICP) = \begin{cases} 0, & \text{if } PICP \geq \mu \\ 1, & \text{if } PICP < \mu \end{cases}$$

-Average Coverage Error (ACE)

$$ACE = PICP - PINC \tag{2.22}$$

-Interval Sharpness (IS):

$$IS = \frac{1}{n}\sum_{t=1}^{n} b_t \tag{2.23}$$

where $b_t$ is

$$b_t = \begin{cases} -2\alpha\hat{I}_t^{\alpha} - 4(\hat{L_t^{\alpha}} - Y_t), & \text{if } Y_t < \hat{L_t^{\alpha}} \\ -2\alpha\hat{I}_t^{\alpha}, & \text{if } Y_t \in \hat{I_t^{\alpha}} \\ -2\alpha\hat{I}_t^{\alpha} - 4(Y_t - \hat{U_t^{\alpha}}), & \text{if } Y_t > \hat{U_t^{\alpha}} \end{cases}$$

-Continuous Ranked Probability Score (CRPS):

$$CRPS = \frac{1}{n}\sum_{t=1}^{n}\int_{0}^{P_{max}} \left[CDF_t - H(Y - Y_t)\right]^2 dy \tag{2.24}$$

where $CDF_t$ is the cumulative form of the distribution and $H$ is the Heaviside step function:

$$H = \begin{cases} 0, & \text{if } Y < Y_t \\ 1, & \text{otherwise} \end{cases}$$

-Skill Score (SC):

$$SC = \frac{1}{n}\sum_{t=1}^{n} SC_t = \frac{1}{n}\sum_{t=1}^{n}\left[\sum_{j=1}^{m}\left(\zeta^{\alpha_j} - \alpha_j\right)\left(Y_t - q_t^{\alpha_j}\right)\right] \tag{2.25}$$

where $SC_t$ is a set of quantiles on a single time step $i, \alpha_j$ is the quantile proportion, $q_t^{\alpha_j}$ is the quantile forecast, and $\zeta^{\alpha_j}$ is an indicator variable denoted by

$$\zeta^{\alpha_j} = \begin{cases} 1, & \text{if } Y_t < q_t^{\alpha_j} \\ 0, & \text{otherwise} \end{cases}$$

The two most important characteristics of a probability interval (PI), the most common way to display probabilistic estimates, are that it is reliable and provides useful information. When the actual wind power output falls within the interval, a PI is reliable and provides valuable information based on how wide the interval is. The PI should ideally be as specific as possible to aid in decision-making. A variety of factors contribute to the prediction's uncertainty. Model uncertainty (or epistemic uncertainty) is one source, and it occurs when the forecasting model

or its parameters are incorrectly specified. The data uncertainty quantifies the inherent noise in the observations. In the literature, alternative versions of CWC are also introduced. To consider the information provided by the actual values, a new term is designed which is included in the new CWC introduced by [161]. Another alternative is also provided by [162].

The reliability of PI can be measured by PICP (Eq. 2.18). It takes into account the mean of the target values the interval covered. The PINAW (Eq. 2.20) is its counterpart in terms of width. The PICP and the PINAW can be further merged with the CWC (Eq. 2.21). The penalisation of invalid PIs is determined by $\eta$ and $\gamma$ which are the two controlling hyperparameters as shown in equation (Eq. 2.21)

Another parameter to describe the reliability of PI is ACE(Eq. 2.22) which is the deviation between the PICP and the PINC. The smaller the deviations the reliable are the PIs. This parameter provides further information in comparison to PICP, a larger PICP does not guarantee a better PINC [163]. The interval sharpness (Eq. 2.23) estimates the PI in terms of its width [164], also known as Winkler score. This metric rewards narrower intervals while penalising PIs where the observations do not fall within them. The CRPS (Eq. 2.24) is a global measure as it estimates both features simultaneously. In situations where a forecast yields a deterministic estimate, this metric is comparable to the MAE. Since the CRPS evaluates cumulative distribution functions, it differs from other metrics. A greater performance of the model is shown by lower CRPS scores. The Eq. (2.25) [165] shows the scoring rule when probabilistic forecasts are represented by a set of quantile forecasts and are estimated by non-parametric models. Its positively oriented and a value of zero means a perfect forecast. Probabilistic estimates can be compared to benchmark models in the same way for deterministic estimates in Eq. (2.15). The application of above mentioned evaluation metrics for probabilistic predictions applied in the recent literatire can be shown as in Table 2.11.

## 2.7 Summary and Research Gaps

This chapter has presented a comprehensive review of the major forecasting methodologies, uncertainty quantification techniques, and spatio-temporal learning frameworks used in renewable generation forecasting, with an emphasis

**Table 2.11** Forecast performance indices for probabilistic predictions

| Reference | Year | PICP | PINAW | CWC | ACE | IS | CRPS | SC | Others |
|---|---|---|---|---|---|---|---|---|---|
| [166] | 2019 | ✓ | ✓ | ✓ | | | | | ✓ |
| [167] | 2018 | ✓ | | | | | | ✓ | |
| [129] | 2018 | ✓ | ✓ | | | | | | |
| [168] | 2018 | ✓ | ✓ | | | ✓ | | | ✓ |
| [131] | 2018 | ✓ | | | | ✓ | | | |
| [169] | 2018 | | | | | | ✓ | | ✓ |
| [170] | 2018 | ✓ | ✓ | | | ✓ | | | |
| [159] | 2017 | | | | | | | ✓ | |
| [171] | 2017 | | | | ✓ | ✓ | ✓ | | |
| [161] | 2017 | ✓ | ✓ | ✓ | | | | | ✓ |
| [130] | 2017 | ✓ | ✓ | ✓ | ✓ | ✓ | | | |

on their applicability to microgrid environments. The review covered physical, statistical, machine learning, deep learning, hybrid, probabilistic, conformal prediction, and spatio-temporal forecasting approaches, in addition to optimization algorithms and evaluation criteria.

From the methodological perspective, a number of key observations emerge. Physical forecasting methods rely on complex kinematic and meteorological formulations that require high-quality weather inputs. Their computational burden and dependence on large-scale atmospheric models make them better suited for medium- and long-term horizons. Traditional statistical techniques offer simple structure and low computational cost, and thus remain relevant for very short-term forecasting. However, their linearity assumptions limit their ability to model the nonlinear and highly volatile behavior of renewable generation.

AI-based models-including ANNs, CNNs, RNNs, LSTMs, and GRUs-have demonstrated strong capability to learn nonlinear dependencies and temporal patterns. Yet these deterministic architectures inherently provide point forecasts only, and thus cannot quantify uncertainty. Support Vector Machines overcome several of the challenges associated with neural networks, such as overfitting and generalization issues. Deep learning models also benefit substantially from large-scale historical data but suffer when data availability is limited. Meta-heuristic optimization algorithms (e.g., GA, PSO, GWO, WOA) have proven effective in improving forecasting accuracy by fine-tuning model parameters and selecting informative features.

The survey also examined deterministic and probabilistic evaluation metrics. Deterministic metrics such as RMSE, MAE, MAPE, IA, BIAS, and SDE capture

different aspects of model performance, while probabilistic metrics such as Prediction Interval Coverage Probability (PICP), Prediction Interval Normalized Average Width (PINAW), Interval Score (IS), and Conditional Coverage provide deeper insight into uncertainty modeling. These metrics collectively reveal that probabilistic evaluation is more informative for modern microgrid operations. Similarly, the literature on meta-heuristic optimization highlights the diversity of algorithmic strategies and the influence of dataset characteristics, convergence behavior, and parameter settings on forecast quality. Consistent with the No Free Lunch theorem, no single optimization method performs best across all scenarios.

## Research Gaps

Despite notable progress, several critical gaps remain in renewable generation forecasting. These gaps directly motivate the development of the proposed forecasting framework and align with the research objectives of this thesis.

1. *Limited adaptability of classical and physical models:* Traditional forecasting methods (e.g., ARIMA, SARIMA, NWP-based physical models) rely heavily on linearity, stationarity, and static parameterization. These assumptions break down under rapidly changing weather regimes, leading to degraded performance during ramp events, cloud transients, and turbulence-driven wind variations. Their inability to self-adapt or learn online limits their suitability for the dynamic, non-stationary conditions of microgrids. **This gap motivates RO-1, which requires multi-horizon forecasting models that handle temporal variability across different timescales.**

2. *Deterministic AI models overlook uncertainty:* While modern AI techniques (ANNs, CNNs, RNNs, LSTMs, GRUs) achieve high point-forecast accuracy, they do not quantify uncertainty. Without prediction interval information, microgrid operators cannot assess risk, allocate reserves, or perform reliable unit commitment. The lack of trustworthy uncertainty estimation limits the operational usefulness of these models in safety-critical environments. **This directly motivates RO-3, which focuses on probabilistic forecasting and uncertainty quantification.**

3. *Lack of unified deterministic–probabilistic learning frameworks:* Most studies treat deterministic and probabilistic forecasting as separate tasks, often using distinct models for point predictions and uncertainty estimation. Such decoupled designs lead to inconsistent mean–interval relationships and fragmented information flow. There remains a need for integrated frameworks that harmonize point accuracy with calibrated uncertainty within a single learning pipeline. **This gap reinforces the need for hybrid, multi-component forecasting architectures as envisioned in RO-2 and RO-3.**

4. *Insufficient exploitation of hybrid and ensemble learning:* Existing hybrid approaches-such as EMD/CEEMDAN combined with neural networks, or ensembles of heterogeneous forecasters-often lack principled design. Many hybrids rely on empirical combinations rather than structured fusion of complementary model features or uncertainty propagation. As a result, their performance deteriorates under regime shifts or long-term horizons. **This gap motivates RO-2, which calls for ensemble-based frameworks that enhance robustness and generalization.**

5. *Neglect of spatio-temporal correlations across renewable sites:* Microgrids increasingly host spatially distributed renewable assets whose outputs are governed by atmospheric continuity. Wind propagation, cloud movement, and irradiance gradients induce strong spatial and temporal dependencies. Many current forecasting models treat sites independently, ignoring these relationships. Recent progress with graph neural networks and spatial statistics has not yet been fully integrated with probabilistic uncertainty quantification. **Addressing this gap is central to RO-4, which emphasizes spatio-temporal modeling and inter-site correlation learning.**

6. *Limited treatment of multi-horizon and multi-variate forecasting:* Existing models often optimize for a single look-ahead horizon or a single target variable. This results in loss of temporal coherence and inability to exploit cross-feature relationships. Microgrid operations, however, require forecasting across multiple horizons (e.g., 15-min, 1-hour, 6-hour) and jointly modeling multiple

meteorological variables. **This gap links directly to RO-1 and RO-4, which require models capable of multi-horizon and multi-variate forecasting.**

7. *Inadequate probabilistic calibration and physical consistency:* Probabilistic forecasting methods (e.g., quantile regression, Bayesian models) often suffer from poor calibration under distributional shifts and may produce physically infeasible intervals (e.g., negative power or power exceeding rated capacity). Few studies incorporate domain constraints or physics-informed learning into uncertainty quantification. **This motivates the development of adaptive, physics-informed uncertainty modeling as required by RO-3.**

Overall, these research gaps establish the need for a unified forecasting framework that combines multi-horizon learning, hybrid ensemble structures, probabilistic and conformal uncertainty quantification, and spatio-temporal modeling capabilities. The proposed research addresses these gaps by formulating an AI-based forecasting framework that is robust, adaptive, uncertainty-aware, and physically consistent, aligned with the operational requirements of renewable-rich microgrids.

## Publication from this chapter

P. Dhaka, M. Sreejeth, and M. M. Tripathi, "A survey of artificial intelligence applications in wind energy forecasting," *Archives of Computational Methods in Engineering*, vol. 31, no. 8, pp. 4853–4878, 2024, doi: 10.1007/s11831-024-10182-8. (IF: 12.1)

# Chapter 3

# Hybrid Deep Learning model for Wind Power Forecasting

## 3.1 Introduction

This chapter presents the first major contribution of the thesis, focusing on the development of a hybrid deep learning framework for short-term wind power forecasting. Motivated by the increasing penetration of wind energy in modern power systems and the operational challenges arising from its inherent variability, this work emphasizes the need for accurate, robust, and data-driven forecasting models tailored for microgrid and utility-scale environments. While numerous machine learning and deep learning approaches have been explored in recent literature, their performance is often constrained by inadequate feature preprocessing, limited exploitation of numerical weather prediction (NWP) data, and suboptimal modeling of temporal dependencies.

To address these limitations, the study presented in this chapter introduces a complete integrated forecasting framework that combines advanced feature engineering techniques with a hybrid learning architecture. The proposed pipeline employs Principal Component Analysis (PCA) and Empirical Mode Decomposition (EMD) for extracting meaningful representations from multivariate NWP data and mitigating noise and non-stationarity. These enriched features are then processed by an LSTM network enhanced with a self-attention mechanism, enabling the model to selectively focus on informative temporal patterns. Finally, XGBoost-based residual refinement is incorporated to correct systematic errors and further strengthen predictive accuracy.

The chapter details the methodological components, hyperparameter optimization using Bayesian techniques, and extensive experimental validation performed on real-world datasets from four wind sites in Tamil Nadu, India. Comparative analyses with established baseline models highlight the superior performance, stability, and generalizability of the proposed hybrid approach.

Overall, Chapter 3 establishes a strong foundation for data-driven renewable generation forecasting by demonstrating how integrated feature engineering and hybrid deep learning can improve WPF performance for microgrid applications.

**Table 3.1** Statistical Description of NWP Data for Tuticorin Wind Farm.

| Variable | Samples | Max | Min | Mean | SD | Kurt. | Skew. |
|---|---|---|---|---|---|---|---|
| WS at 10 m (m/s) | 70,176 | 11.60 | 0.04 | 4.41 | 1.88 | 0.18 | 0.43 |
| WS at 50 m (m/s) | 70,176 | 14.40 | 0.05 | 5.99 | 2.26 | 0.32 | 0.07 |
| WS at 80 m (m/s) | 70,176 | 15.31 | 0.05 | 6.37 | 2.40 | 0.32 | 0.07 |
| WD at 10 m (∘) | 70,176 | 359.88 | 0.00 | 171.21 | 99.52 | -1.49 | -0.27 |
| WD at 50 m (∘) | 70,176 | 359.92 | 0.00 | 171.74 | 100.16 | -1.51 | -0.27 |
| WD at 80 m (∘) | 70,176 | 719.84 | -358.78 | 172.26 | 102.19 | -1.03 | -0.24 |

## 3.2 Data Pre-processing and Feature Engineering

### 3.2.1 Data Description

In this work, the historical wind power generation of the Four sites of the Tuticorin wind farm is used along with NWP data collected from NASA power project [172]. The weather parameters included wind direction at 10m ($WD10$), wind speed at 10m ($WS10$), wind direction at 50m ($WD50$), wind speed at 50m ($WS50$), relative humidity ($RH$), surface pressure($PS$), and temperature($T$). The Table 3.1 describes the NWP data for the Tuticorin wind farm's geographical location. Wind direction is in degrees from north, and wind speed is in meters per second. The exploratory data analysis was done, and it was found that the parameters other than wind direction and wind speed had a weak correlation with wind power, so these parameters were discarded from this study. The proposed hybrid forecasting model utilizes historical wind power time series data, wind speed and wind

direction data from numerical weather prediction (NWP) models. We have also considered the influence of wind power, wind speed, and wind direction from the preceding hour to enhance forecast accuracy. The statistical description of wind power data is presented in Table 3.2. In other words, the forecasting model's input parameters encompass the data for wind speed, wind power, and wind direction for both the current hour ($t$) and the preceding hour ($t-1$), denoted as $WP_t, WP_{t-1}, WS_t, WS_{t-1}, WD_t, WD_{t-1}$.

**Table 3.2** Statistical Description of Wind Power Data for Tuticorin Wind Farm.

| Site No. | Capacity (MW) | Samples | Max | Min | Mean | SD | Kurt. | Skew. |
|---|---|---|---|---|---|---|---|---|
| Site-1 | 62.475 | 70,176 | 61.72 | 0 | 19.89 | 18.81 | -0.55 | 0.83 |
| Site-2 | 62.5 | 70,176 | 51.10 | 0 | 13.78 | 14.48 | -0.37 | 0.95 |
| Site-3 | 50 | 70,176 | 48.59 | 0 | 15.52 | 14.74 | -0.45 | 0.86 |
| Site-4 | 55 | 70,176 | 50.98 | 0 | 10.74 | 12.59 | 1.41 | 1.50 |

### 3.2.2 Aligning NWP data according to wind turbine hub height

NWP data is typically available for specific heights, but wind turbine height varies in practical applications. In this study, the NWP data was originally provided at 10m and 50m heights, while the wind turbine hub height was 80 m. To account for this lack of data, extrapolation was necessary. The NWP data at the 80m height was derived through linear extrapolation from the available 10 m and 50 m data, ensuring compatibility with the wind turbines' placement.

**Table 3.3** Wind Power Forecasting Results Using NWP Data at Different Heights.

| | Training Error | | | | Testing Error | | | |
|---|---|---|---|---|---|---|---|---|
| Height | RMSE | NRMSE (%) | MAE | MAPE (%) | RMSE | NRMSE (%) | MAE | MAPE (%) |
| 10 m | 1.72 | 2.79 | 1.12 | 1.84 | 1.91 | 3.11 | 1.23 | 2.01 |
| 50 m | 1.73 | 2.81 | 1.13 | 1.85 | 2.14 | 3.48 | 1.27 | 2.07 |
| 80 m | 1.70 | 2.77 | 1.12 | 1.81 | 1.89 | 2.77 | 1.22 | 1.98 |

An evaluation of wind power forecasts was conducted, utilizing data from January 2019 to December 2019, incorporating information from different heights,

specifically 10m, 50m, and 80m. The forecasts were generated using an XGBoost model. The results obtained are summarized in Table 3.3. The results depict that the forecasting error is minimized when NWP data at a height of 80m is utilized. This choice aligns with the actual altitude of the wind turbines, as affirmed by the prediction outcomes. NWP wind speeds at different altitudes are shown in Fig.3.1. There are substantial variations in NWP wind speeds with changes in altitude. Consequently, an altitude correction was imperative to prevent inaccuracies in wind power forecasts.

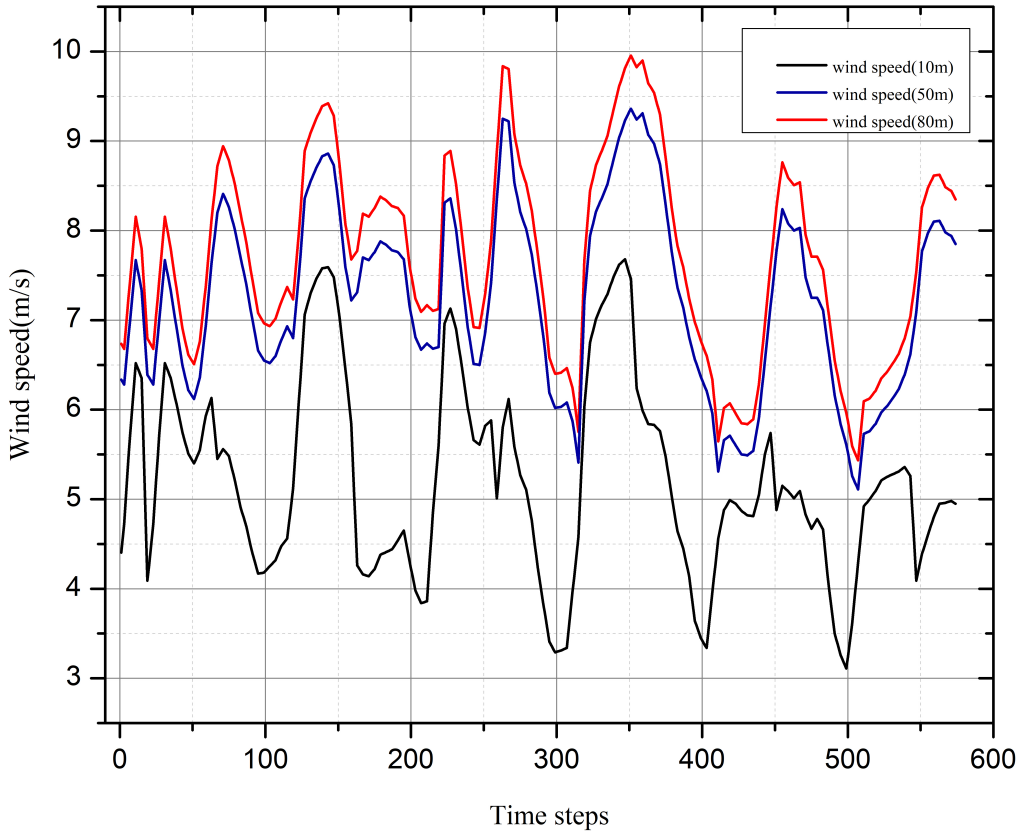



FIGURE 3.1: NWP values of wind speed for Tuticorin wind farm at 10, 50 and 80m .

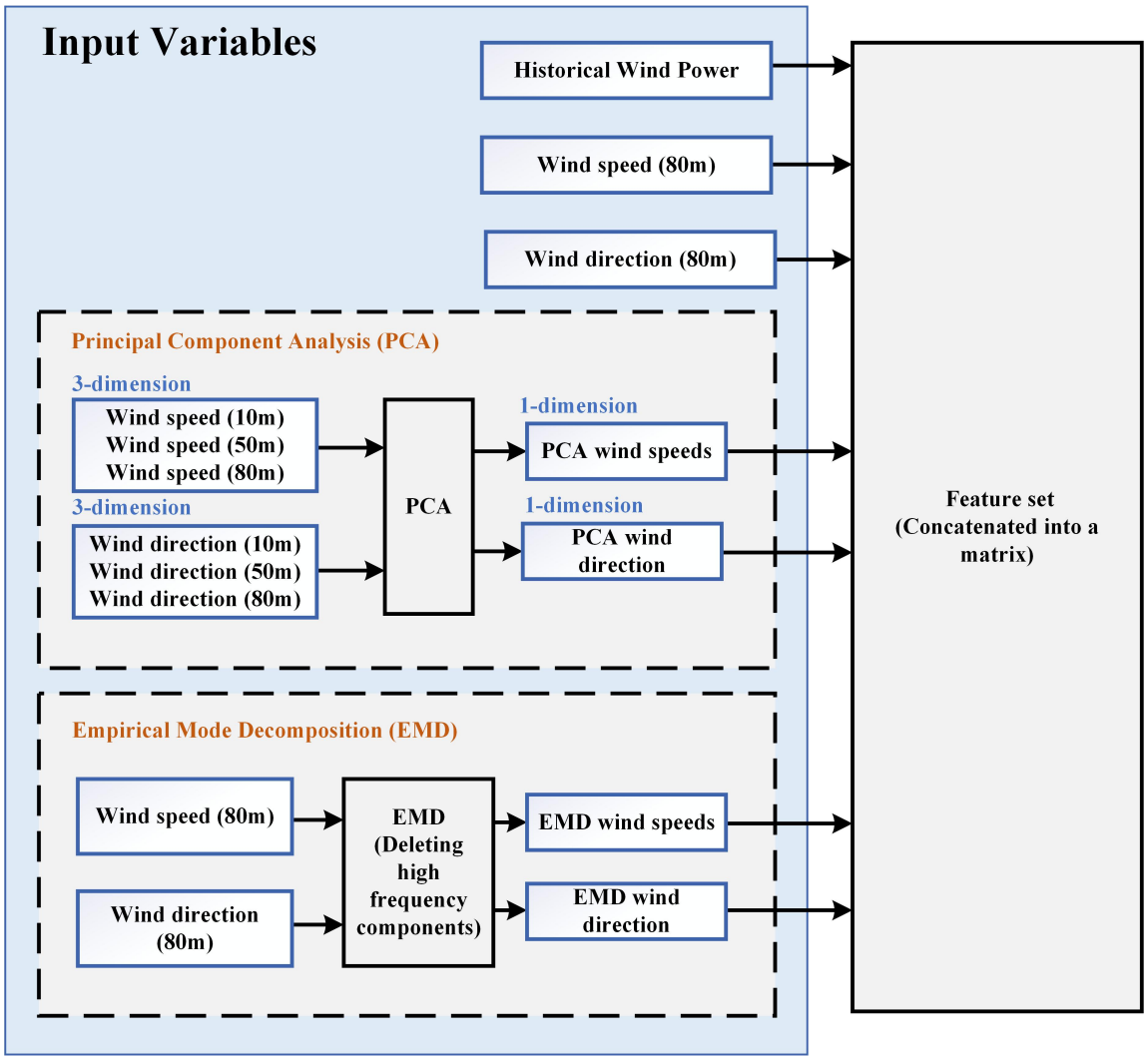



FIGURE 3.2: Proposed scheme of the feature engineering and data preprocessing.

### 3.2.3 Data preprocessing of NWP Data: Principal Component Analysis

Given the strong correlation among NWP data at various heights, we recognized the potential to enhance forecasting efficiency through dimensionality reduction using feature extraction techniques. The steps in applying PCA are outlined in Algorithm 1. The proposed scheme of feature engineering and data preprocessing is depicted in Fig. 3.2. In the proposed data preprocessing scheme, PCA is applied to NWP wind data which converts 3-dimensional wind speeds and directions into one-dimension. The data is standardized before performing PCA because PCA is extremely sensitive regarding the variance of the variables. After that, a covariance matrix for a 3 dimensional dataset is calculated as per Eq. (3.1):

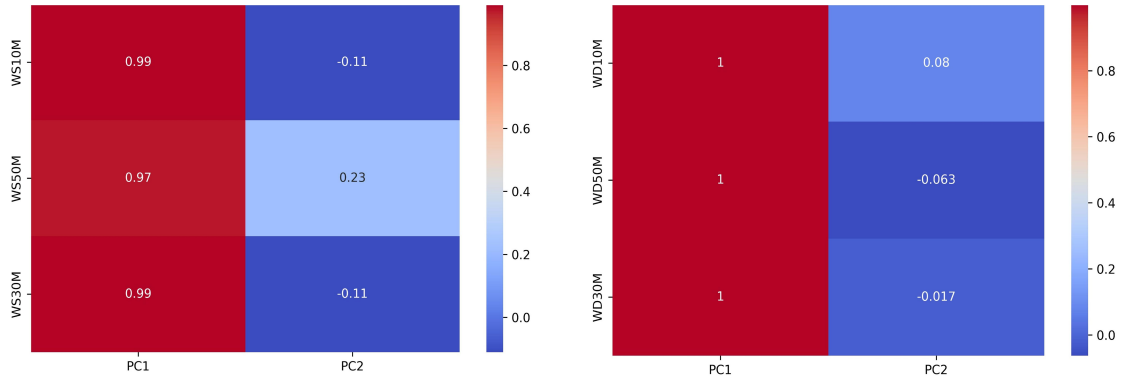
$$C = \begin{bmatrix} cov(x,x) & cov(x,y) & cov(x,z) \\ cov(y,x) & cov(y,y) & cov(y,z) \\ cov(z,x) & cov(z,y) & cov(z,z) \end{bmatrix} \tag{3.1}$$

The matrix $C$ is computed using the covariance values between variables $x$, $y$, and $z$, with each element representing the covariance between the respective pairs of variables. The covariance between $X$, and $Y$ (and other pairs) is calculated using the formula $covx, y = \sum(x_i - \bar{x})(y_i - \bar{y})/(N-1)$. Where $N$ represents the number of observations in the Dataset. After obtaining the covariance matrix, the eigenvalues ($\lambda$) and eigenvectors ($\vec{V}$) are computed to identify the principal components. These principal components are then used as inputs in the forecasting model for wind speeds and wind direction. After evaluating each component, the components

**Algorithm 1** Principal Component Analysis (PCA)

1: Create covariance matrix $C$
2: Transform covariance matrix into its constituent eigenvectors ($\vec{V}$) and eigenvalues ($\lambda$)
3: Select the $z^{th}$ largest eigenvalues and their corresponding $z^{th}$ eigenvectors
4: Build the projection matrix $W$ using the top $z^{th}$ eigenvectors
5: Project the original $d^{th}$ dimensional data into a new $z^{th}$ dimensional subspace using $W$
6: Remove final principal component and revert the data back to its original high-dimensional state using inverse projection matrix.

with limited importance were removed. For instance, if the contribution of the first component is greater than 95%, the other components are removed. This reduction results in the data restoration by the first principal component. Following a comparison of the original wind speeds and wind directions, it becomes evident that PCA effectively accomplishes noise reduction and feature extraction when applied to the original NWPs, which can be deduced from the Fig. 3.3.

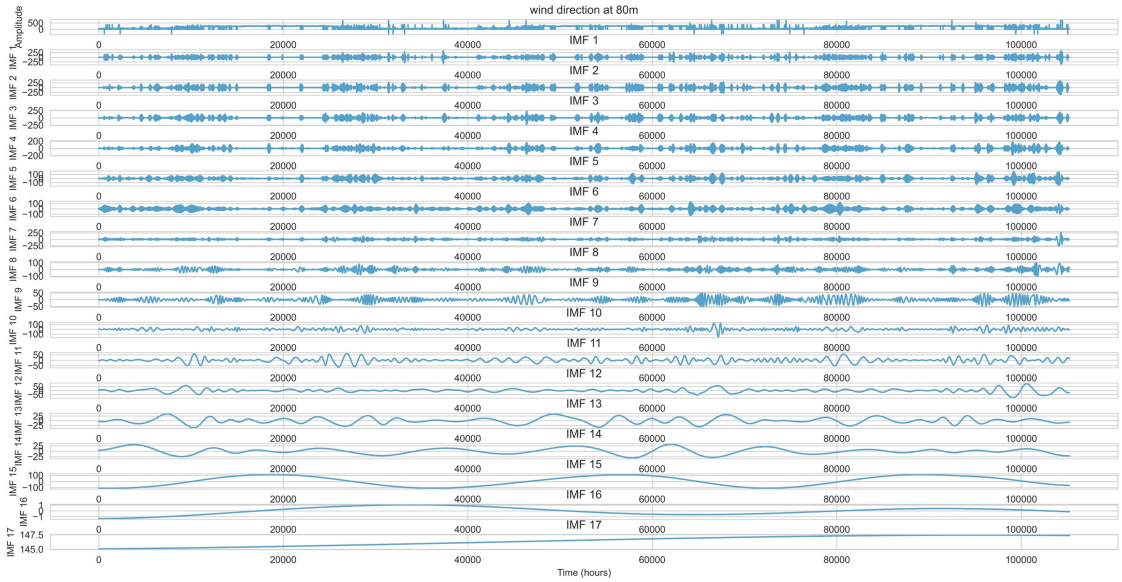



**(a) PCA feature extraction for wind speed** **(b) PCA feature extraction for wind direction**

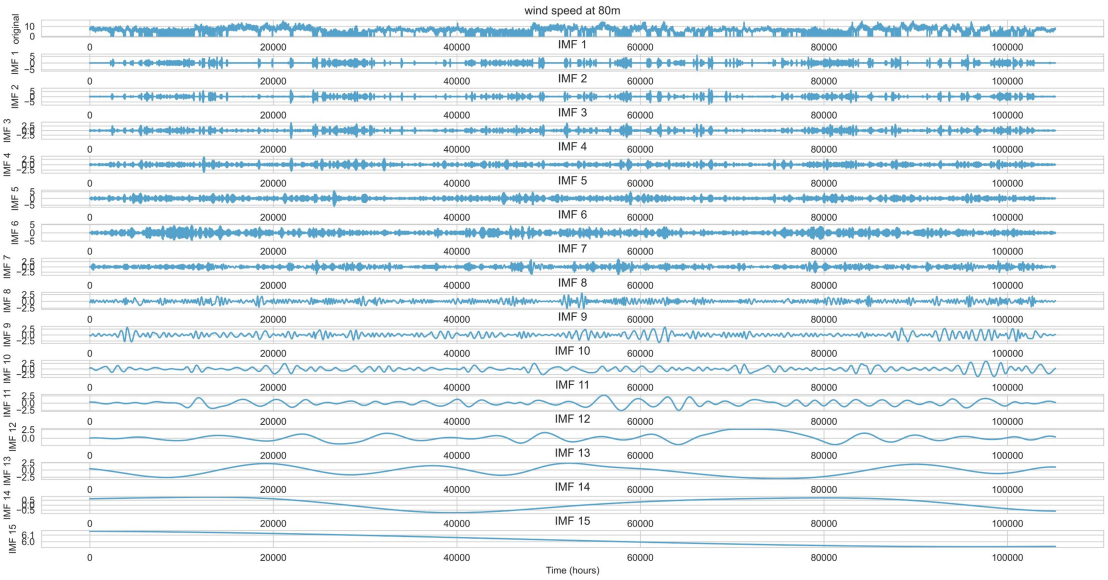
FIGURE 3.3: PCA feature extraction heatmap for (a) wind speed and (b) wind direction.

### 3.2.4 Data preprocessing of NWP Data: Empirical Mode Decomposition

EMD technique is employed to break down wind speed and direction into their fundamental oscillatory constituents known as Intrinsic Mode Function(IMFs). The higher frequency components are discarded, while the remaining frequency components, along with Principal Components (derived from PCA), historical wind power, and the original NWP wind data, are utilized as inputs for the model. EMD decomposes a signal into IMFs and residuals. Each IMF and residual captures local features at various time scales. The following Algorithm depicts the step-by-step process of EMD algorithm 2:




**(a) EMD for Wind direction at 80m**




**(b) EMD for Wind speed at 80m**

FIGURE 3.4: IMF's for Wind direction and wind speed at 80m.

**Algorithm 2** Empirical Mode Decomposition (EMD)

1: Define Input signal as $S_i$
2: Calculate the maximum ($S_{max}$)and minimum ($S_{min}$)values of the signal
3: Derive lower $L_s$ and upper envelopes $U_s$ using cubic spline interpolation
4: Obtain an intermediate signal $I_s$ by subtracting the mean envelope $\mu_s$ from the original signal $S$
5: Evaluate if the intermediate signal is an IMF

Using EMD, the original signal $s_i$ is transformed into several IMFs and one residual component, as per the Eq. (3.2)

$$S_i = \sum_{t=0}^{n} IMF_t(i) + r_n(i) \tag{3.2}$$

EMD procedure concludes when the residual value can't decompose further. In the proposed approach, EMD preprocessing is applied to the NWP wind speeds and directions, generating multiple IMFs. The IMFs generated for wind speed and wind direction are shown in Fig. 3.4

## 3.3 Proposed Integrated Framework

This section introduces a hybrid deep learning framework for deterministic WPF, combining LSTM networks, self-attention mechanisms, and XGBoost. The architecture, illustrated in Fig. 3.5 and Fig. 3.6, integrates feature extraction, data preprocessing, and hyperparameter optimization to enhance forecasting accuracy.

### 3.3.1 Hybrid Deep Learning Model with Self-Attention Mechanism

The proposed hybrid deep learning model synergizes three components

1. LSTM Network: Captures long-term temporal dependencies in wind power and meteorological data.

2. Self-Attention Mechanism: Dynamically weights input features and historical time steps to prioritize influential patterns.

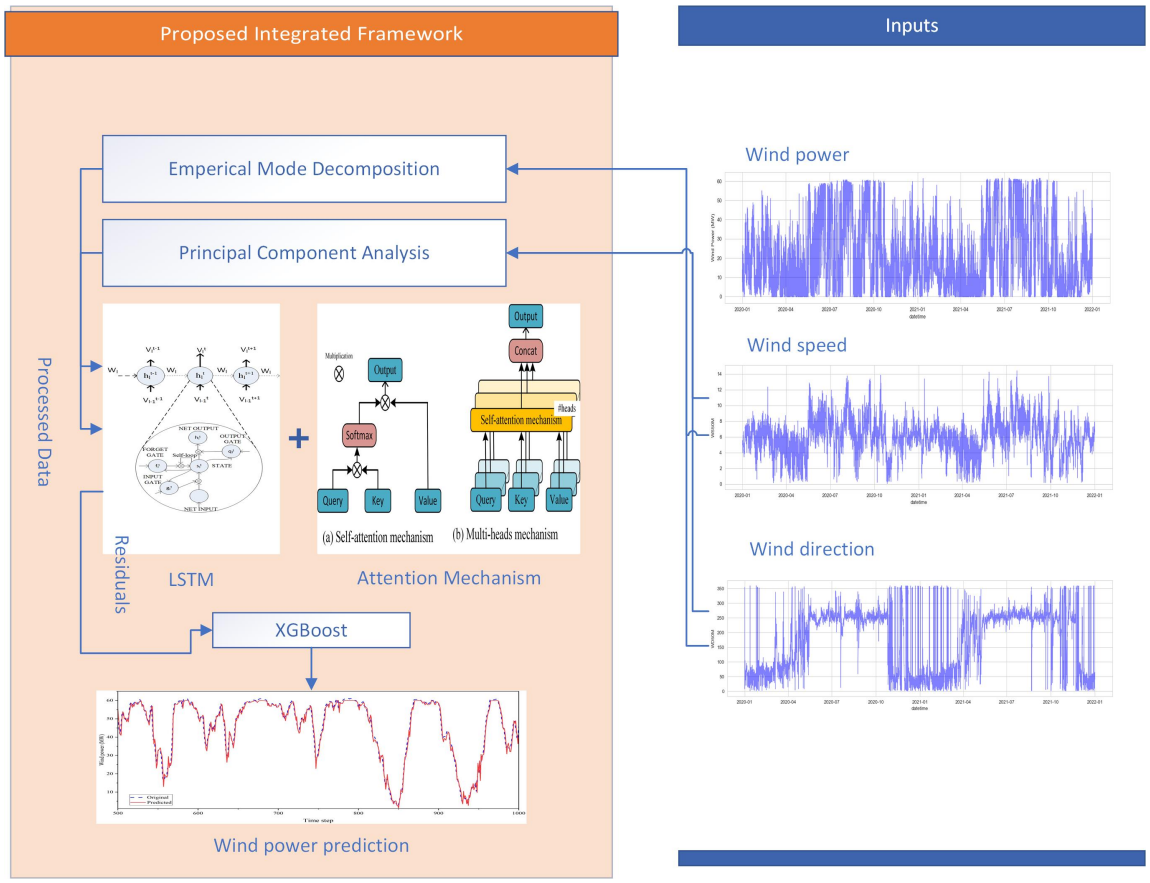



FIGURE 3.5: Integrated framework for wind power forecasting: (1) Feature extraction via PCA and EMD, (2) Hybrid LSTM-attention model for temporal modeling, (3) XGBoost-based residual refinement.

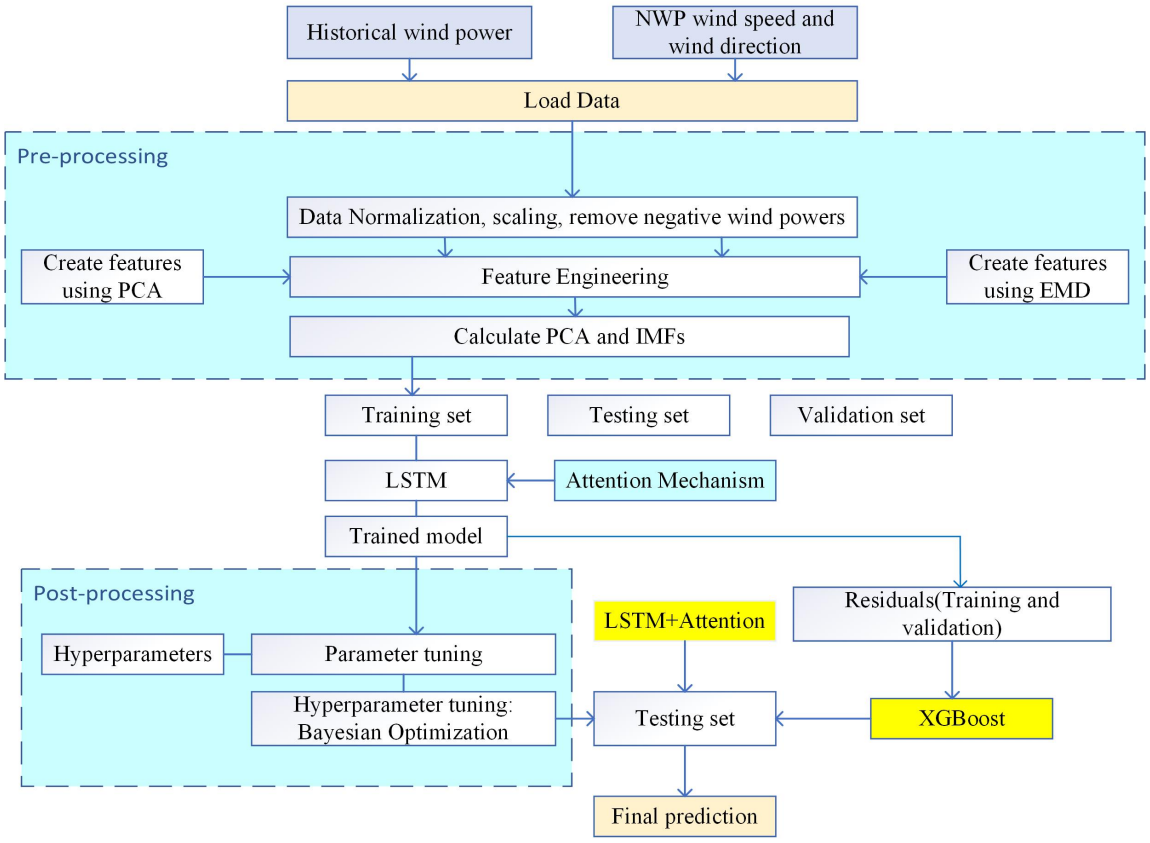



FIGURE 3.6: Hybrid architecture: (1) LSTM layers with self-attention, (2) Dense layers for initial predictions, (3) XGBoost residual correction.

3. XGBoost Regressor: Refines residuals from the LSTM-attention output to correct systematic errors.

### Self-Attention Formulation

Building on Bahdanau's additive attention [173], the self-attention mechanism maps queries ($Q$), keys ($K$), and values ($V$) to a context-aware representation. Let $Q \subseteq \mathbb{R}^{d_q}$, $K \subseteq \mathbb{R}^{d_k}$, and $V \subseteq \mathbb{R}^{d_v}$ denote query, key, and value spaces, respectively. For input sequences $K = \{k_1, \ldots, k_N\}$ and $V = \{v_1, \ldots, v_N\}$, the attention output for a query $q \in Q$ is calculated as per Eq. (3.3):

$$\text{Att}(q, K, V) = \sum_{i=1}^{N} \text{softmax}_a(q, K)_i \cdot v_i \tag{3.3}$$

where the alignment score $a(q, k_i)$ is computed from Eq. (3.4):

$$\text{softmax}_a(q, K)_i = \frac{\exp(a(q, k_i))}{\sum_{j=1}^{N} \exp(a(q, k_j))} \tag{3.4}$$

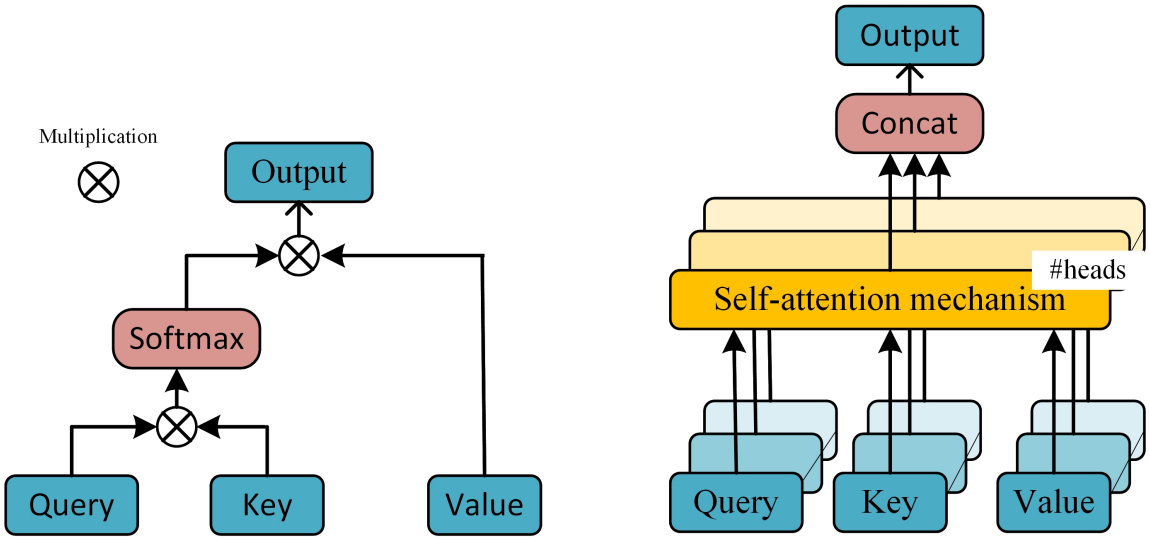
For parallel processing of query sets $Q = \{q_1, \ldots, q_M\}$, self-attention is defined as per Eq. (3.5):

$$\text{SelfAttention}(Q) = \text{Att}(Q, Q, Q) \tag{3.5}$$

The LSTM with attention model is compiled with Adam optimizer and mean squared error loss. Following this, hyperparameter tuning is carried out to optimize the hyperparameters of the LSTM model. Fig. 3.7 describes the self-attention and multi-head attention mechanism. Tree parzen estimator based bayesian hyperparameter optimization method is used to explore a range of hyperparameters, and the best-performing hyperparameters are identified. The optimized hyperparameters are presented in the Table 3.4.

### Model Implementation

LSTM-attention module (Fig. 3.6) processes standardized inputs including, Historical wind power ($WP_{t-1}$), Preprocessed NWP data, Lagged wind speed ($WS_{t-1}$) and direction ($WD_{t-1}$). The Adam optimizer minimizes the mean




(a) Self-attention mechanism (b) Multi-heads mechanism

FIGURE 3.7: Self attention and Multi-head attention mechanism.

squared error (MSE) loss (Eq. 7.35):

$$\mathcal{L}_{\text{MSE}} = \frac{1}{T}\sum_{t=1}^{T}\left(WP_t - \hat{WP}_t\right)^2 \tag{3.6}$$

where $WP_t$ and $\hat{WP}_t$ denote actual and predicted wind power. Fig. 3.7 depicts the mechanism of self-attention and Multi-head attention.

**Residual Refinement with XGBoost**

Let $\epsilon_t = WP_t - \hat{WP}_t^{\text{LSTM}}$ denote the LSTM-attention prediction error. XGBoost is trained to predict $\epsilon_t$ using the same input features, yielding the final forecast from Eq. (3.7):

$$\hat{WP}_t^{\text{final}} = \hat{WP}_t^{\text{LSTM}} + \hat{\epsilon}_t^{\text{XGBoost}} \tag{3.7}$$

### 3.3.2 Bayesian Hyperparameter Optimization

To optimize the hyperparameters of the LSTM-attention model, Bayesian optimization with the Tree-Parzen Estimator (TPE) [174] was employed, a

probabilistic approach that efficiently balances exploration and exploitation in high-dimensional spaces.

**Objective Function:** The optimization minimizes the validation mean squared error (MSE) of wind power forecasts as per Eq. (3.8)

$$\mathcal{L}_{\text{MSE}} = \frac{1}{T}\sum_{t=1}^{T}\left(WP_t - \hat{WP}_t\right)^2 \tag{3.8}$$

where $WP_t$ and $\hat{WP}_t$ are the actual and predicted wind power at time $t$, respectively. This metric prioritizes accurate deterministic forecasts while penalizing large prediction errors quadratically.

**TPE Formulation:** Unlike Gaussian Process (GP)-based Bayesian optimization, TPE models $p(x|y)$, the probability of hyperparameters x given the loss y, using two density functions (Eq. 3.9):

$$p(x \mid y) = \begin{cases} \ell(x) & \text{if } y < y^* \\ g(x) & \text{if } y \geq y^* \end{cases} \tag{3.9}$$

where $\ell(x)$ and $g(x)$ represent hyperparameter densities for losses below and above a threshold $y^*$, respectively. The TPE maximizes the Expected Improvement (EI) as per Eq. (3.10):

$$\text{EI}_{y^*}(x) \propto \left(\gamma + \frac{g(x)}{\ell(x)}(1-\gamma)\right)^{-1} \tag{3.10}$$

where $\gamma = p(y < y^*)$. This prioritizes hyperparameters with high likelihood in low-loss regions ($\ell(x)$) and low likelihood in high-loss regions ($g(x)$). To evaluate the efficacy of Bayesian Optimization with TPE parameter, we conducted comparative experiments with several optimization algorithms, including Bayesian Optimization (using both the TPE and GP variants), Random Search, and Grid Search. The results of these experiments are summarized in the Fig. 3.8. The graph illustrates the validation loss obtained for each algorithm across iterations during the hyperparameter tuning process. TPE-based Bayesian Optimization demonstrated the best performance by consistently achieving the lowest validation loss within fewer iterations, showcasing its efficiency in exploring the

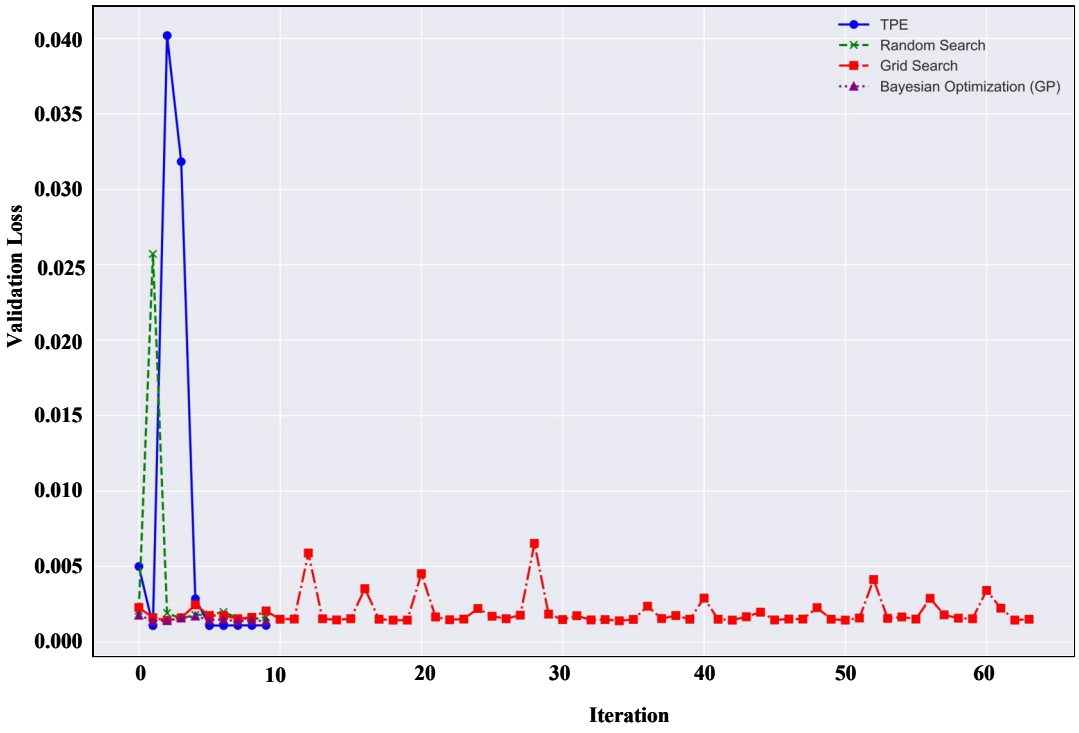



FIGURE 3.8: Comparison of Various Hyperparameter Optimization Algorithms.

**Table 3.4** Hyperparameter settings outcome for LSTM model.

| Parameters | Settings(ranges) |
|---|---|
| Hidden layer nodes | 32 (10-250) |
| Dropout rate | 0.2 (0-1) |
| Dense layer nodes | 128 (10-250) |
| learning rate | 0.0028 (0.0001-0.001) |
| epoch | 48 (0-100) |
| batch size | 64 (4-128) |

hyperparameter space and converging to optimal solutions. While GP-based Bayesian Optimization also performed well, its convergence was slightly slower. In contrast, Random Search and Grid Search required significantly more iterations to reach comparable validation loss levels, with Grid Search showing periodic inefficiencies due to its fixed parameter grid. Based on its superior accuracy and convergence speed, TPE-based Bayesian Optimization was chosen for our study, consistent with its reported efficiency in the literature for high-dimensional parameter tuning. Table 3.4 summarizes the TPE-tuned hyperparameters, including LSTM hidden units, attention heads, dropout rates, and XGBoost learning rates.

## 3.4 Forecast Evaluation and Results

### 3.4.1 Evaluation Metrics

The assessment of forecasts involves examining how effectively the data aligns with the model and quantifying its performance. Three evaluation metrics have been employed to evaluate forecasting performance rigorously.

1. Normalized Root Mean Squared Error (NRMSE) (Eq. 3.11), which penalizes large errors quadratically, normalized by turbine capacity to enable cross-site comparisons.

$$NRMSE = \frac{\sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2}}{\text{capacity}} * 100 \tag{3.11}$$

where, $y_i$, $\hat{y}_i$, $\overline{y_i}$, $n$ represent the actual value, predicted value, average of actual value, the number of predicted samples, and capacity is the installed "capacity" of a wind site.

2. Normalized Mean Absolute Error (NMAE) (Eq. 3.12), which measures average deviation of predictions from actual values and is robust to outliers.

$$NMAE = \frac{1}{n}\sum_{i=1}^{n}\frac{|y_i - \hat{y}_i|}{\text{capacity}} * 100 \tag{3.12}$$

3. Coefficient of Determination (R2) (Eq. 3.13), which quantifies the proportion of variance explained by the model, with 1 indicating perfect alignment.

$$\mathrm{R}^2 = 1 - \frac{\sum_{\mathrm{i}}\left(y_i - \hat{y}_i\right)^2}{\sum_{\mathrm{i}}\left(\overline{y_i} - y_i\right)^2} \tag{3.13}$$

### 3.4.2 Comparison with Baselines

**Table 3.5** Performance comparison of benchmark methods for Tuticorin wind farm (Site-1 and Site-2).

| Model | Site-1 | | | Site-2 | | |
|---|---|---|---|---|---|---|
| | NRMSE (%) | NMAE (%) | $R^2$ | NRMSE (%) | NMAE (%) | $R^2$ |
| ARIMA (M1) | 10.12 | 9.12 | 0.715 | 16.20 | 14.90 | 0.350 |
| GRU (M2) | 9.60 | 8.50 | 0.750 | 15.30 | 13.90 | 0.420 |
| CNN (M3) | 8.10 | 7.10 | 0.821 | 13.10 | 11.80 | 0.570 |
| LSTM (M4) | 3.51 | 2.65 | 0.966 | 12.14 | 10.86 | 0.640 |
| **Proposed method** | **1.43** | **1.05** | **0.994** | **3.03** | **2.55** | **0.978** |

This section evaluates the proposed method's performance compared to four established benchmark techniques to demonstrate its capabilities. The benchmark models comprise: (1) ARIMA [175] (M1), (2) GRU [176] (M2), (3) CNN [177] (M3), and (4) LSTM [178] (M4). Table 3.5 and 3.6 present a performance comparison of five models and the proposed method-across four sites at the Tuticorin wind farm. The proposed model demonstrates exceptional superiority, achieving the lowest error rates and highest $R^2$ scores across all sites. For instance, at Site-1, it attains an NRMSE of 1.43%, an NMAE of 1.05%, and an $R^2$ of 0.994, significantly

**Table 3.6** Performance comparison of benchmark methods for Tuticorin wind farm (Site-3 and Site-4).

| Model | Site-3 | | | Site-4 | | |
|---|---|---|---|---|---|---|
| | NRMSE (%) | NMAE (%) | $R^2$ | NRMSE (%) | NMAE (%) | $R^2$ |
| ARIMA (M1) | 10.80 | 9.80 | 0.700 | 8.60 | 7.50 | 0.850 |
| GRU (M2) | 9.80 | 8.60 | 0.750 | 7.80 | 6.60 | 0.880 |
| CNN (M3) | 6.20 | 5.10 | 0.910 | 6.80 | 5.60 | 0.910 |
| LSTM (M4) | 4.02 | 3.09 | 0.958 | 5.01 | 3.88 | 0.948 |
| **Proposed method** | **1.16** | **0.87** | **0.997** | **1.72** | **1.20** | **0.994** |

outperforming the benchmark models. While the LSTM (M4) model shows competitive results, such as a 3.51% NRMSE at Site-1 and a 0.958$R^2$ at Site-3, it falls short of the proposed method's accuracy. The CNN (M3) performs notably well at Site-3 (6.20% NRMSE, $R^2 = 0.910$) but struggles at Site-2 (13.10% NRMSE), whereas the classical ARIMA (M1) model lags behind, particularly at Site-2 (16.20% NRMSE). Notably, the proposed model maintains remarkable consistency, with $R^2$ values exceeding 0.99 and NRMSE values below 3.03% across all sites, underscoring its robustness and reliability for wind energy forecasting. These results highlight the proposed method's potential to advance predictive accuracy in renewable energy applications. Figures 3.9, 3.10, 3.11, and 3.12 show an intuitive comparison of the proposed method against baseline approaches for all four sites.

### 3.4.3 Comparison with other feature engineering approaches

The methods are grouped into decomposition techniques (M5–M9) and dimensionality reduction approaches (M10–M12), with the hybrid EMD-PCA-LSTM (M13) proposed as the optimal framework. Decomposition methods include CEEMDAN-LSTM (M5), which introduces noise-assisted signal splitting to mitigate mode mixing but underperforms compared to VMD-LSTM (M7), a mathematically rigorous method that decomposes signals into intrinsic mode functions with constrained bandwidths. EEMD-LSTM (M6) and EMD-LSTM (M9) leverage iterative sifting processes, with EMD-LSTM achieving superior results (e.g., 1.92% NRMSE at Site-1), suggesting simpler decomposition may better handle non-stationary wind data. Factor analysis-LSTM (M8) identifies

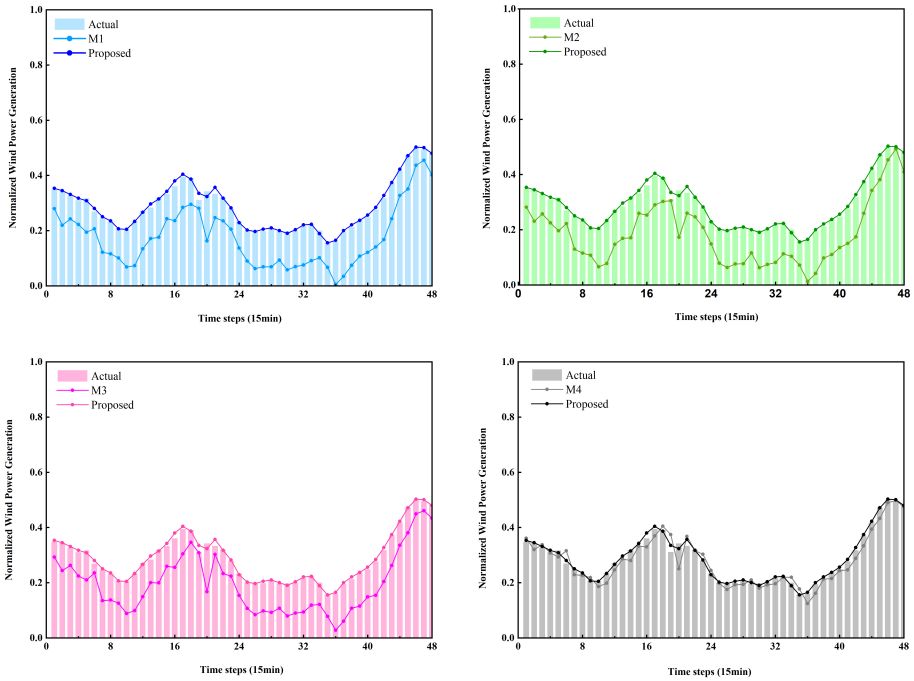



FIGURE 3.9: Site-1 baseline comparison.

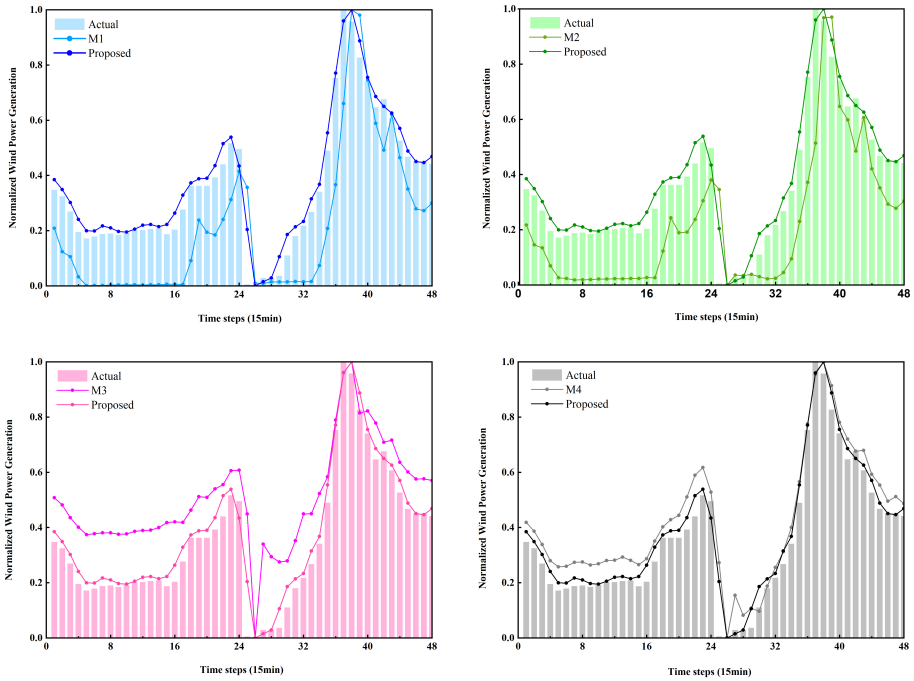



FIGURE 3.10: Site-2 baseline comparison.

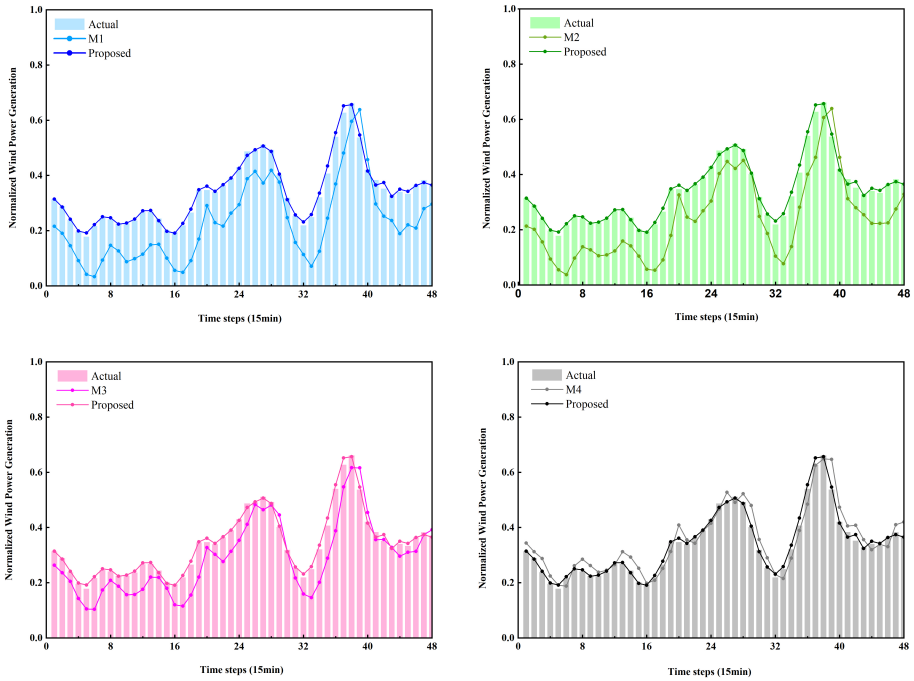



FIGURE 3.11: Site-3 baseline comparison.

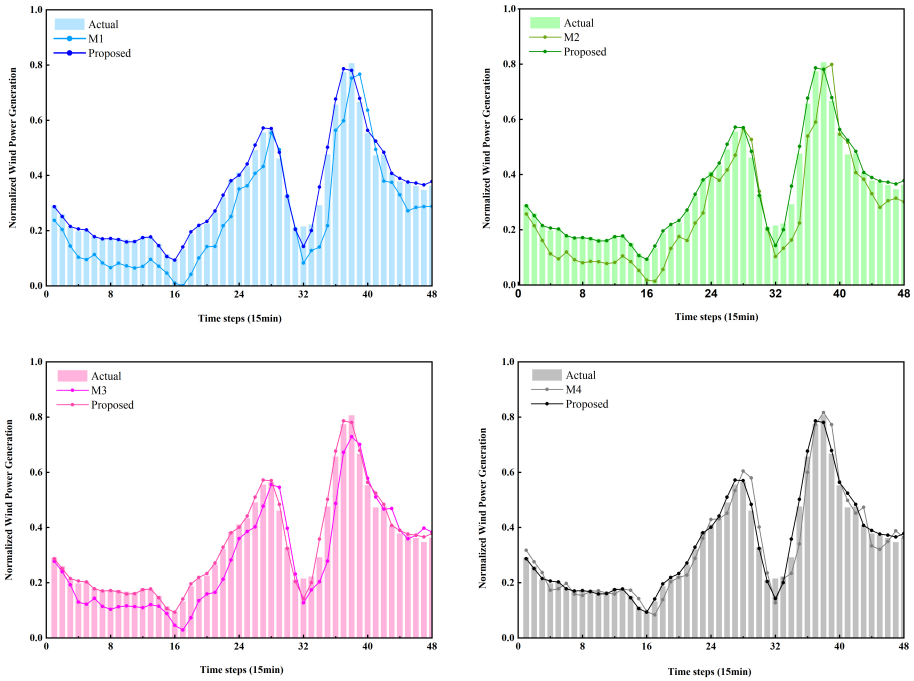



FIGURE 3.12: Site-4 baseline comparison.

latent variables explaining feature correlations but lags behind EMD-based methods, likely due to its linear assumptions. Among dimensionality reduction techniques, KPCA-LSTM (M10) (Kernel Principal Component Analysis) employs nonlinear transformations but shows no advantage over linear PCA-LSTM (M12), which retains maximal variance with fewer components (e.g., 1.50% NRMSE at Site-1). ICA-LSTM (M11) (Independent Component Analysis) assumes feature independence, a constraint less suited to wind data's inherent correlations, resulting in marginally higher errors than PCA-LSTM. The hybrid EMD-PCA-LSTM (M13) synergizes EMD's noise reduction with PCA's feature simplification, achieving the lowest errors (e.g., 1.49% NRMSE at Site-1, 3.92% at Site-2) and near-perfect $R^2$ scores ($\geq$ 0.991), demonstrating that cascaded decomposition and dimensionality reduction mitigate limitations of standalone approaches. Notably, Site-2's higher errors across methods (e.g., 6.85% NRMSE for EMD-LSTM vs.3.92% for M13) suggest complex local wind dynamics, which M13's dual-stage feature engineering effectively addresses. These results highlight the importance of method selection: nonlinear decomposition (VMD, CEEMDAN) and advanced dimensionality reduction (KPCA, ICA) offer theoretical advantages but are outperformed by simpler, interpretable frameworks (EMD, PCA) in this context. Table 3.7 and 3.8 evaluates feature engineering methods combined with LSTM networks for wind energy forecasting across four sites.

### 3.4.4 Comparison with Residual error correction techniques

Table 3.9 and 3.10 compares the performance of residual correction methods integrating a Self Attention Mechanism-Long Short-Term Memory (SAM-LSTM) base model with four boosting algorithms, CatBoost (M14), LightGBM (M15), Decision Tree (M16), and the proposed SAM-LSTM-XGBoost, across four sites using NRMSE, NMAE, and $R^2$. The proposed SAM-LSTM-XGBoost consistently achieves the lowest error rates and highest $R^2$ values across all sites, demonstrating its superior residual correction capability. For instance, at Site-3, it attains an NRMSE of 1.16%, an NMAE of 0.87%, and an $R^2$ of 0.997, outperforming even the second-best method, SAM-LSTM-Decision Tree (M16), which records 1.26% NRMSE and 0.945% NMAE. While all hybrid methods perform robustly (NRMSE $\leq$ 3.69%, $R^2 \geq$ 0.967), XGBoost's integration shows

**Table 3.7** Performance comparison of feature engineering methods (Site-1 and Site-2).

| Model | Site-1 | | | Site-2 | | |
|---|---|---|---|---|---|---|
| | NRMSE (%) | NMAE (%) | $R^2$ | NRMSE (%) | NMAE (%) | $R^2$ |
| **Decomposition Techniques** | | | | | | |
| CEEMDAN-LSTM (M5) | 3.40 | 2.56 | 0.9683 | 9.15 | 7.67 | 0.7955 |
| EEMD-LSTM (M6) | 2.29 | 1.67 | 0.986 | 9.01 | 7.95 | 0.802 |
| VMD-LSTM (M7) | 2.07 | 1.63 | 0.988 | 8.47 | 7.13 | 0.825 |
| Factor analysis-LSTM (M8) | 1.93 | 1.45 | 0.9897 | 8.35 | 6.72 | 0.8297 |
| EMD-LSTM (M9) | 1.92 | 1.45 | 0.9898 | 6.85 | 5.97 | 0.8854 |
| **Dimensionality Reduction** | | | | | | |
| KPCA-LSTM (M10) | 1.70 | 1.24 | 0.992 | 6.44 | 5.83 | 0.899 |
| ICA-LSTM (M11) | 1.70 | 1.22 | 0.9921 | 6.07 | 5.54 | 0.9099 |
| PCA-LSTM (M12) | 1.50 | 1.09 | 0.994 | 5.75 | 4.79 | 0.919 |
| **EMD-PCA-LSTM (M13)** | **1.49** | **1.15** | **0.994** | **3.92** | **3.47** | **0.963** |

**Table 3.8** Performance comparison of feature engineering methods (Site-3 and Site-4).

| Model | Site-3 | | | Site-4 | | |
|---|---|---|---|---|---|---|
| | NRMSE (%) | NMAE (%) | $R^2$ | NRMSE (%) | NMAE (%) | $R^2$ |
| **Decomposition Techniques** | | | | | | |
| CEEMDAN-LSTM (M5) | 3.78 | 2.85 | 0.9632 | 4.92 | 3.73 | 0.9498 |
| EEMD-LSTM (M6) | 2.60 | 1.83 | 0.983 | 3.87 | 2.93 | 0.969 |
| VMD-LSTM (M7) | 2.43 | 1.75 | 0.985 | 3.54 | 2.70 | 0.974 |
| Factor analysis-LSTM (M8) | 2.27 | 1.65 | 0.9867 | 3.47 | 2.50 | 0.9749 |
| EMD-LSTM (M9) | 2.24 | 1.71 | 0.9870 | 3.19 | 4.00 | 0.9789 |
| **Dimensionality Reduction** | | | | | | |
| KPCA-LSTM (M10) | 1.90 | 1.39 | 0.991 | 3.09 | 2.32 | 0.980 |
| ICA-LSTM (M11) | 1.78 | 1.26 | 0.9918 | 3.05 | 2.24 | 0.9807 |
| PCA-LSTM (M12) | 1.76 | 1.35 | 0.992 | 2.75 | 2.08 | 0.984 |
| **EMD-PCA-LSTM (M13)** | **1.47** | **1.15** | **0.994** | **2.05** | **1.58** | **0.991** |

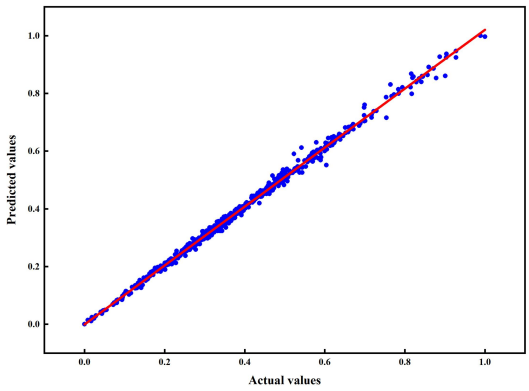
marked improvements, particularly at Site-2, where it reduces NRMSE by 7.1% compared to M16 (3.03% vs. 3.26%) and enhances $R^2$ to 0.978, suggesting its effectiveness in handling complex residual patterns. SAM-LSTM-Decision Tree (M16) exhibits competitive results (e.g., 1.87% NRMSE at Site-4), likely due to its interpretable splits, but struggles with consistency, whereas gradient-boosting variants (M14, M15) show marginal differences, underscoring XGBoost's advantage in regularization and parallel processing. The results highlight XGBoost's ability to refine residuals through its gradient-boosting framework, which iteratively corrects errors via additive tree ensembles, thereby enhancing SAM-LSTM's base predictions. This synergy positions SAM-LSTM-XGBoost as a robust solution for error correction in forecasting tasks, particularly in scenarios with nonlinear and non-stationary residuals, as evidenced by its dominance across heterogeneous sites (e.g., Site-2's higher baseline errors). The findings advocate integrating advanced boosting frameworks with attention-based LSTMs to optimize residual correction in complex time-series applications. Fig. 3.13 shows the $R^2$ assessment results of all four sites using the proposed method.

**Table 3.9** Residual correction method comparison (Site-1 and Site-2).

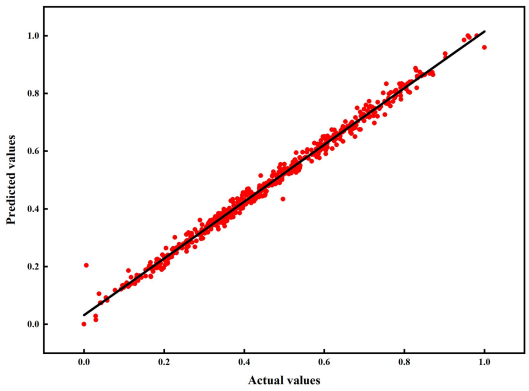
| Model | Site-1 | | | Site-2 | | |
|---|---|---|---|---|---|---|
| | NRMSE (%) | NMAE (%) | $R^2$ | NRMSE (%) | NMAE (%) | $R^2$ |
| SAM-LSTM-CatBoost (M14) | 1.45 | 1.10 | 0.994 | 3.69 | 2.85 | 0.967 |
| SAM-LSTM-LightGBM (M15) | 1.45 | 1.01 | 0.994 | 3.68 | 2.55 | 0.967 |
| SAM-LSTM-Decision Tree (M16) | 1.44 | 1.09 | 0.994 | 3.26 | 2.78 | 0.974 |
| **SAM-LSTM-XGBoost** | **1.43** | **1.05** | **0.994** | **3.03** | **2.55** | **0.978** |

**Table 3.10** Residual correction method comparison (Site-3 and Site-4).

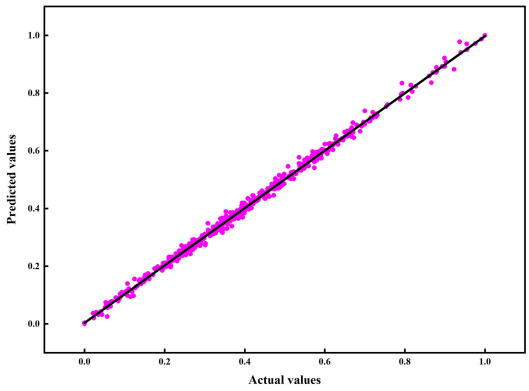
| Model | Site-3 | | | Site-4 | | |
|---|---|---|---|---|---|---|
| | NRMSE (%) | NMAE (%) | $R^2$ | NRMSE (%) | NMAE (%) | $R^2$ |
| SAM-LSTM-CatBoost (M14) | 1.46 | 1.11 | 0.995 | 2.04 | 1.56 | 0.991 |
| SAM-LSTM-LightGBM (M15) | 1.40 | 1.08 | 0.995 | 1.95 | 1.47 | 0.992 |
| SAM-LSTM-Decision Tree (M16) | 1.26 | 0.945 | 0.996 | 1.87 | 1.28 | 0.993 |
| **SAM-LSTM-XGBoost** | **1.16** | **0.87** | **0.997** | **1.72** | **1.20** | **0.994** |



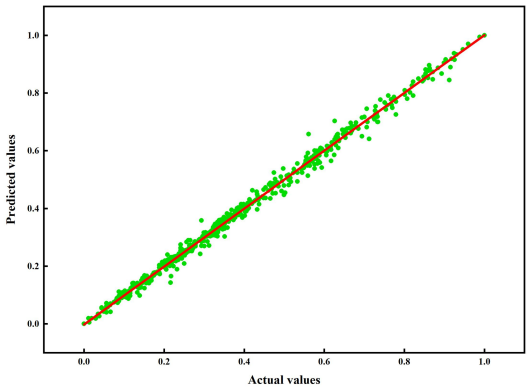

**(a) Site-1**




**(a) Site-2**




**(a) Site-3**




**(a) Site-4**

FIGURE 3.13: Coefficient of determination assessment results using the proposed approach.

### 3.4.5 Ablation Study

This section elucidates the effectiveness of the proposed hybrid model in forecasting the 15-minute ahead wind power generation of four wind sites in Tamil Nadu, India. The proposed model consists of PCA, EMD, and Bayesian optimization algorithms to automatically select the best hyperparameters for the LSTM model. The four sites have 62.475 MW, 49.5 MW, 54.6 MW, and 50 MW capacities. For the first site, the data is collected from January 2019 to December 2021, whereas for the other 3 sites, the data ranges for two years $(2020 - 2021)$. The training, validation and testing datasets are split into 80%, 10%, and 10% for the first wind site. For the other three sites, the data collected is for short duration, where the dataset is split into 70%, 15%, and 15%. All the coding is done on Python 3.9 software using Keras API integrated with tensorflow. To showcase the effectiveness of attention mechanisms, data preprocessing, and feature engineering techniques, this study conducted a comparison between the developed model and its variant models, as follows:

- w/o feature engineering: *Proposed hybrid model without Data preprocessing and Feature Engineering.*
- w/o self-attention mechanism: *Proposed hybrid model without self-attention mechanism.*
- w/o residual error correction: *Proposed hybrid model without residual error correction.*

Table 3.11 and 3.12 compares the proposed and the other variant models for all four sites. The following deductions can be made regarding the performance of

**Table 3.11** Ablation study: Impact of key components on forecasting performance (Site-1 and Site-2).

| Variant | Site-1 | | | Site-2 | | |
|---|---|---|---|---|---|---|
| | NRMSE | NMAE | $R^2$ | NRMSE | NMAE | $R^2$ |
| w/o Feature Engineering | 3.37 | 2.43 | 0.966 | 8.76 | 6.54 | 0.793 |
| w/o Self-Attention | 1.75 | 1.36 | 0.984 | 3.37 | 2.82 | 0.966 |
| w/o Residual Correction | 1.63 | 1.20 | 0.989 | 3.23 | 2.74 | 0.966 |
| **Proposed Integrated Framework** | **1.43** | **1.05** | **0.994** | **3.03** | **2.55** | **0.978** |

the proposed model with its variant models:

**Table 3.12** Ablation study: Impact of key components on forecasting performance (Site-3 and Site-4).

| Variant | Site-3 | | | Site-4 | | |
|---|---|---|---|---|---|---|
| | NRMSE | NMAE | $R^2$ | NRMSE | NMAE | $R^2$ |
| w/o Feature Engineering | 3.62 | 2.74 | 0.960 | 4.86 | 3.69 | 0.949 |
| w/o Self-Attention | 1.46 | 0.99 | 0.987 | 1.96 | 1.39 | 0.982 |
| w/o Residual Correction | 1.32 | 0.91 | 0.989 | 1.91 | 1.30 | 0.987 |
| **Proposed Integrated Framework** | **1.16** | **0.87** | **0.997** | **1.72** | **1.20** | **0.994** |

- The performance of the proposed integrated framework significantly drops after removing the data preprocessing and feature engineering components, indicating the critical role these processes play in preparing the data for effective modeling. For site-2 the NRMSE degrades from 3.03% to 8.76%. Similar observations can be seen for other sites as well.

- Excluding the self-attention mechanism results in a notable degradation in the model's ability to capture long-range dependencies within the data, leading to increased errors in forecasting sudden wind pattern shifts (e.g., gusts or lulls) and weaker alignment between predictions and actual wind dynamics, particularly at sites with volatile conditions like Site-2 (NRMSE rises to 3.37% vs. 3.03% in the full model). Without attention, the model struggles to prioritize critical time steps, resulting in oversights in temporal relationships and reduced robustness to non-stationary wind signals.

- Without residual error correction, the model fails to refine systemic biases and small recurring errors in its predictions, leading to higher NRMSE (e.g., 1.32% at Site-3 vs. 1.16% in the complete model) and NMAE values (e.g., 1.30% at Site-4 vs. 1.20% in the complete model). This omission leaves uncorrected residual patterns, such as persistent under/overestimations of wind power during specific weather transitions, which accumulate over time and degrade forecasting reliability. For instance, at Site-2, the lack of correction results in an NRMSE of 3.23% compared to 3.03% in the complete framework, highlighting its role in addressing subtle but impactful deviations.

## 3.5 Summary

This chapter proposed a novel complete integrated framework for WPF that encompasses data preprocessing and feature engineering utilizing PCA and EMD. This hybrid model integrates an LSTM model with an attention mechanism alongside an XGBoost model. The experimental results unequivocally demonstrate the superior forecasting accuracy of the proposed model compared to conventional models, reaffirming its potential for delivering high-quality predictions. Based on the carried out experimental analysis, following findings are obtained:

1. The PCA-EMD based feature engineering approach has a significant contribution in improving the stability and accuracy of wind power forecasting.

2. The incorporation of self- attention mechanism and bayesian hyperparameter optimization significantly enhances the performance of the LSTM model.

3. The proposed hybrid integrated framework has demonstrated its feasibilty and superiority making it suitable for practical wind power systems.

Notably, feature engineering is a pivotal step in capturing essential dataset characteristics, thereby elevating data quality. The application of PCA effectively reduces noise and enhances input feature validity. Although the hybrid attention-enhanced LSTM–XGBoost framework developed in this chapter demonstrates significant improvements in short-term wind power forecasting by leveraging advanced feature engineering and multivariate NWP inputs, it remains constrained in its ability to explicitly separate and model the multi-scale temporal characteristics inherent in wind power data. The model captures general temporal dependencies effectively, but it does not fully disentangle short-term localized dynamics from longer-term periodic patterns, which often coexist and influence forecasting accuracy under rapidly varying atmospheric conditions. Additionally, the attention mechanism employed here operates globally, which may dilute the influence of the most relevant recent observations near the prediction horizon. These limitations motivate the methodological advancements presented in the next chapter, where a specialized dual-branch architecture-incorporating confined

attention, periodic feature extraction, and feature fusion—is introduced to more accurately capture localized temporal behaviors and long-range periodic signals. This subsequent work addresses the gaps identified in the current framework and offers a more targeted, interpretable, and robust solution for practical wind power forecasting.

## Publication from this Chapter

P. Dhaka, M. Sreejeth, and M. M. Tripathi, "Hybrid attention-based deep learning model using feature engineering approaches for wind power forecasting," *Electrical Engineering*, Vol. 107, pp. 15343–15357, 2025, doi: 10.1007/s00202-025-03329-8. (IF: 1.9)

# Chapter 4

# Confined-attention enabled Deep Learning model for Wind Power Forecasting

## 4.1 Introduction

Building upon the hybrid forecasting framework developed in the previous chapter, this chapter presents the second major contribution of the thesis, focusing on a more specialized deep learning architecture tailored for multi-scale temporal modeling in wind power forecasting. While the earlier framework successfully integrated advanced feature engineering and attention-augmented LSTM modeling, it remained limited in its ability to explicitly distinguish localized short-term dynamics from longer-term periodic patterns-an important characteristic of wind power signals influenced by both meteorological fluctuations and diurnal cycles. Addressing these limitations requires a forecasting architecture that can independently learn and fuse heterogeneous temporal features while maintaining computational efficiency and real-time applicability for microgrid operations.

To this end, this chapter introduces the Confined Attention-enabled Long Short-Term Memory (CAELSTM) model with an integrated feature fusion strategy. The proposed dual-branch design incorporates two complementary modules: a confined attention mechanism that focuses on highly relevant short-term histories near the prediction horizon, and a periodic feature extraction module that captures long-range cyclic dependencies using LSTM-based representations.

Unlike conventional global attention or monolithic LSTM architectures, the confined attention mechanism restricts its focus to a tunable, data-driven window, enabling the model to strengthen localized temporal learning while reducing noise arising from distant or less relevant time steps. The outputs of the two branches are subsequently fused to form a richer and more discriminative representation of the underlying wind dynamics. Additionally, numerical weather prediction (NWP) inputs-such as wind speed and direction at multiple hub-height levels-are incorporated to enhance real-time forecasting performance and adaptability under diverse operating conditions.

The effectiveness of the CAELSTM framework is evaluated using real-world datasets from two distinct wind farm sites, and comprehensive comparative analyses demonstrate its superior forecasting accuracy, robustness, and generalizability over classical LSTM variants and recent hybrid deep learning models. Ablation studies further validate the contribution of confined attention, periodic feature extraction, and feature fusion individually and collectively. Overall, this chapter advances the methodological foundation established earlier by addressing the previously identified modeling limitations and introducing a more refined architecture that more accurately captures the multi-scale temporal behavior inherent in wind power data.

## 4.2 Methodology

This section offers an exhaustive explanation of the methodologies used for implementation, including the steps involved in the hybrid deep learning model, data preparation and feature engineering, the confined attention mechanism, and the WPF architecture based on the CAELSTM model.

### 4.2.1 Data preparation and feature engineering

WPF methods anticipate the variability and uncertainty of wind power output to enable better decision-making, grid integration, and optimization of wind energy resources. NWP data, diurnal cycles, seasonal variations, wake effects, etc highly influence wind power. These factors are integrated in complex ways, challenging wind power prediction tasks. Therefore, there is a growing need for careful data

preparation and advanced modelling techniques to enhance forecast accuracy [179].

## Dataset preparation

Let $x_t$ denote the wind power generated at a specific instance $t$, $N$ represent the cumulative quantity of data points that are accessible. Using these notations, the historical wind power data for a particular wind farm is represented by the vector $\boldsymbol{P}_H$, as shown in Eq. (4.1):

$$\boldsymbol{P}_H = \{x_1, x_2, ..., x_t, ..., x_N\} \tag{4.1}$$

In this context, wind power represents the cumulative energy wind turbines produce over the interval $\Delta_i$, with $(\Delta_i)$ expressed in minutes. Consequently, the total number of samples recorded within a single day is $(24 \times 60)/\Delta_i$. Typically, wind power data is aggregated at 15-minute intervals, resulting in 96 samples per day. This study endeavours to secure both the periodic and temporal characteristics of wind data efficiently, and therefore constructs suitable inputs for different modules. Thus, the input is constructed and fed to different modules to improve the extraction of these characteristics. Define $n$ as the total number of historical time steps available and $q$ as the number of time steps considered as input. The input to the first LSTM module is wind power data given by $\boldsymbol{X}_h$, which is represented by Eq. (4.2). It contains the wind power for the past $n$ time steps

$$\boldsymbol{X}_h = [X_{h1},\ X_{h2},\ ...,\ X_{h(n-q-1)},\ X_{h(n-q)}] \tag{4.2}$$

Here, $\boldsymbol{X}_h$ represents a sequence of historical wind power measurements sampled at $q$ previous time steps. Each $X_{hj}$ denotes the wind power value at the $j^{\text{th}}$ time lag prior to prediction, $n$ is the total number of historical time steps available, and $q$ is the number of time steps to be considered as input. For example, if $q = 4$, then $4 \times 15 = 60$ min of the historical data is used to predict wind power at the next timestamp (if the wind data are collected every 15 minute). Furthermore, considering real-time applications, this work utilizes NWP data as additional inputs since the NWP data possess periodic properties, i.e., diurnal and seasonal cycles. To fully extract useful information from NWP data, this work creates an

NWP feature set consisting of wind speed, wind direction, and the decomposed components of these variables.

## Feature Engineering

The main challenge in WPF comes from the volatile and nonlinear nature of wind patterns. To address this, a feature set combining raw and decomposed wind speed and direction data was used. Due to strong correlations in NWP data across various heights, feature extraction improves efficiency. In the proposed framework as depicted in Fig. 3.2, principal component analysis (PCA) and empirical mode decomposition (EMD) serve distinct yet complementary purposes. PCA is employed to reduce the spatial dimensionality of multi-level NWP features, such as wind speed and direction at 10 m, 50 m, and 80 m, by capturing the dominant variance directions and eliminating redundancy across these altitudes. In contrast, EMD is applied temporally to extract intrinsic mode functions (IMFs) from wind-related time series, enabling finer resolution of short- and long-term variability. PCA projects high-dimensional data onto a lower-dimensional space, keeping essential variance while reducing complexity. EMD further refines data by breaking time series into IMFs, enabling detailed temporal analysis. Consider three observed variables $x$, $y$, and $z$. The covariance between any two variables $a$ and $b$ is denoted as $\mathrm{cov}(a, b)$. Using these definitions, the covariance matrix $\boldsymbol{C} \in \mathbb{R}^{3\times 3}$ is given in Eq. (4.3):

$$\boldsymbol{C} = \begin{bmatrix} \mathrm{cov}(x,x) & \mathrm{cov}(x,y) & \mathrm{cov}(x,z) \\ \mathrm{cov}(y,x) & \mathrm{cov}(y,y) & \mathrm{cov}(y,z) \\ \mathrm{cov}(z,x) & \mathrm{cov}(z,y) & \mathrm{cov}(z,z) \end{bmatrix} \tag{4.3}$$

where $\boldsymbol{C}$ is the covariance matrix constructed from $x$, $y$, and $z$. The eigenvalues and the corresponding eigenvectors of $\boldsymbol{C}$ are computed to identify the principal components. Components with low contributions (e.g., those explaining less than 5% of the total variance) are discarded, resulting in noise-reduced inputs for subsequent analysis.

The EMD method is then applied to decompose a signal $S_i$ into IMFs and a residual, as shown in Eq. (4.4):

$$S_i = \sum_{t=0}^{n} \mathrm{IMF}_t(i) + r_n(i), \tag{4.4}$$

where, $\mathrm{IMF}_t(i)$ denotes the $i^{\text{th}}$ intrinsic mode function at time $t$, and $r_n(i)$ represents the residual component after removing $n$ IMFs. The residual captures the long-term trend that remains after extracting dominant oscillatory modes. High-frequency IMFs are discarded. The remaining IMFs, PCA outputs, raw NWP data, and historical wind power form the model inputs. PCA reduces redundancy and noise, while EMD isolates trends and oscillations, especially for wind data at 80 m.

The selection of NWP features at 10 m, 50 m, and 80 m is motivated by their relevance to operational wind turbines and the vertical wind profile. Specifically, 80 m corresponds closely to the hub height of commercial wind turbines, while 10 m and 50 m capture near-surface and mid-layer wind conditions. Together, these altitudes enable the model to infer wind shear and vertical gradients that influence power conversion efficiency. These levels are also standard outputs in NWP models, ensuring practical data availability and generalizability across wind farm sites.

### 4.2.2 LSTM with Confined Attention Mechanism

#### *LSTM*

It is an advanced version of recurrent neural networks (RNN) designed to handle long-term dependencies. It solves the vanishing gradient problem by introducing a memory cell that can store information over long sequences. Three gates control this memory cell: forget, input, and output gates [180]. The forget gate determines what information to keep or discard, the input gate controls the new information added, and the output gate regulates the cell's output. These gates enable LSTM to capture long-term dependencies better than traditional RNNs [181]. Define $\epsilon_h^t$ as the backpropagated error signal for cell $h$ at time step $t$, $\delta_i^t$ as the error gradient with respect to the input gate, and $\boldsymbol{U}_h^i$ as the weight matrix associated with recurrent input connections. Using this notation, $\epsilon_h^t$ is computed as in Eq. (4.5):

$$\epsilon_h^t = \sum_{i \in N} \boldsymbol{U}_h^i \cdot \delta_i^{t+1} \tag{4.5}$$

The error gradient $\delta_q^t$ with respect to the output gate is defined in Eq. (4.6):

$$\delta_q^t = y_q^{t'} \cdot \sum_{h \in c} \left( \epsilon_h^t \cdot \tanh\left(s_h^t\right) \right) \tag{4.6}$$

where $y_q^{t'}$ is the derivative of the output gate activation, $s_h^t$ is the cell state of unit $h$, and $c$ indexes all memory cells.

The error gradient for the hidden state $\delta_h^t$ is given in Eq. (4.7), where $g^t$ is the candidate gate activation and $y_h^{t'}$ is the derivative of the hidden output:

$$\delta_h^t = g^t \cdot y_h^{t'} \cdot \frac{\partial E^t}{\partial s_h^t} \tag{4.7}$$

The error gradient for the forget gate $\delta_f^t$ is computed as in Eq. (4.8), where $f^{t'}$ is the derivative of the forget gate activation and $c_h^{t-1}$ is the previous cell state:

$$\delta_f^t = f^{t'} \cdot \sum_{h \in c} \left( \frac{\partial E^t}{\partial s_h^t} \cdot c_h^{t-1} \right) \tag{4.8}$$

The error gradient for the candidate gate $\delta_g^t$ is defined in (4.9), where $g^{t'}$ is the derivative of the candidate gate activation and $i_h^t$ is the input gate activation:

$$\delta_g^t = g^{t'} \cdot \sum_{h \in c} \left( \frac{\partial E^t}{\partial s_h^t} \cdot i_h^t \right) \tag{4.9}$$

Finally, the partial derivative of the loss $E^t$ with respect to the cell state $s_h^t$, incorporating the previously defined error terms, is expressed in Eq. (4.10):

$$\begin{aligned} \frac{\partial E^t}{\partial s_h^t} = \epsilon_h^t \cdot y_q^t \cdot \frac{\partial \tanh\left(y_h^t\right)}{\partial y_h^t} + \frac{\partial E^{t+1}}{\partial s_h^{t+1}} \cdot f^{t+1} \\ + \delta_g^{t+1} \cdot \boldsymbol{W}_h^g + \delta_f^{t+1} \cdot \boldsymbol{W}_h^f + \delta_q^t \cdot \boldsymbol{W}_h^q \end{aligned} \tag{4.10}$$

where $\boldsymbol{W}$ are the trainable weight matrices associated with the different gates.

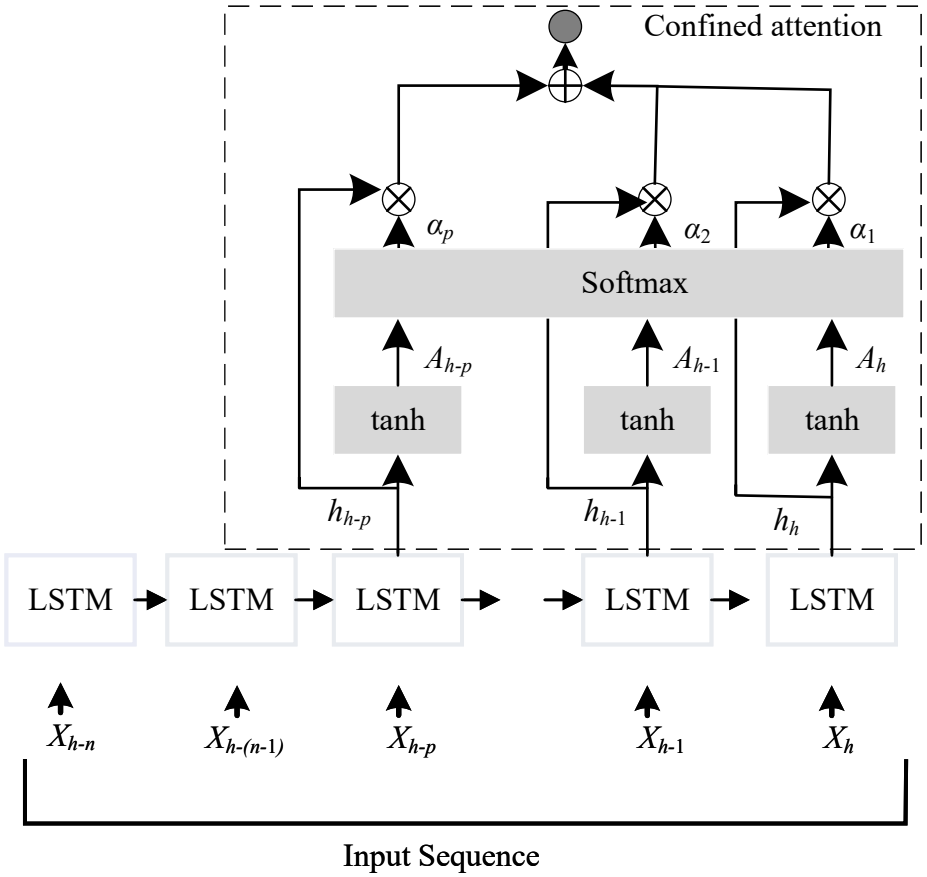



FIGURE 4.1: LSTM with Confined Attention mechanism.

In summary, $\delta$ terms denote the error gradients with respect to different LSTM gates, $y$ represents the activation outputs of the gates, and $i^t$, $f^t$, $o^t$, and $g^t$ correspond to the input, forget, output, and candidate gates, respectively. The cell output $h^t$ is computed as $o^t \cdot \tanh(c^t)$, where $\tanh(\cdot)$ is the hyperbolic tangent function and all gates use the sigmoid activation function $\sigma(\cdot)$. $E^t$ denotes the loss function evaluated at time $t$. Fig. 4.1 demonstrates the integration of confined attention within the LSTM architecture and Fig. 4.2 illustrates the CAELSTM architecture.

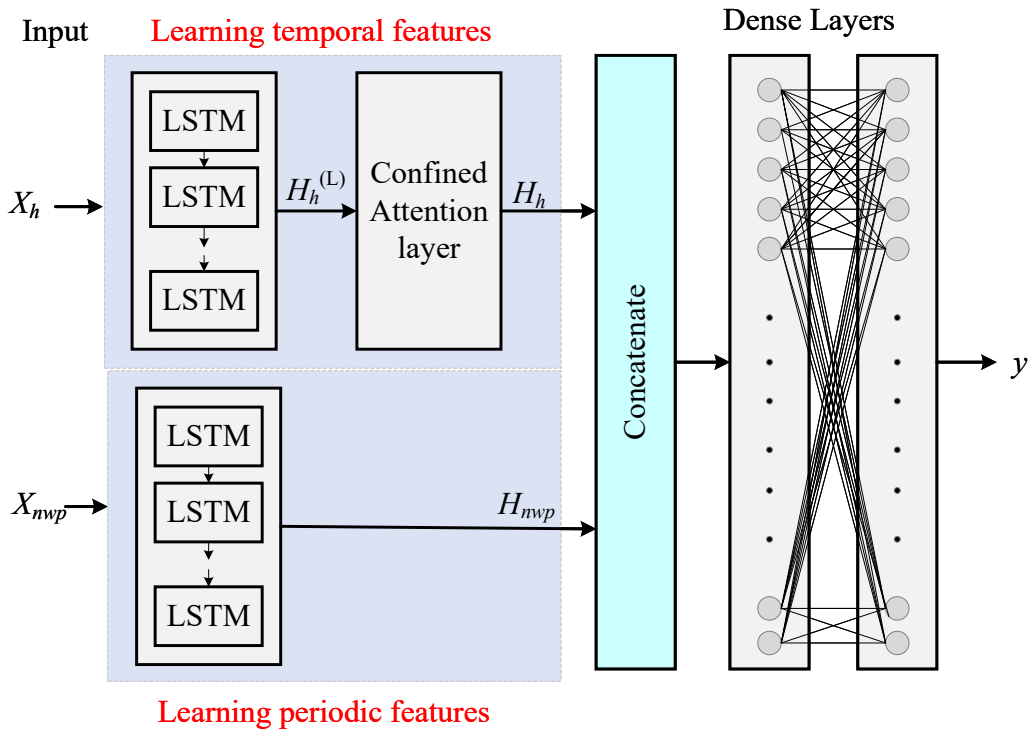



FIGURE 4.2: CAELSTM architecture.

### *Confined Attention Mechanism*

The attention mechanism enables models to assess the significance of various input components for predictions. It directs the model's focus toward pertinent input data segments. The attention mechanism was introduced in [173], highlighting all input words during the formation of the context vector. However, unlike NLP, the output highly depends on recent information in wind prediction tasks. Hence, the present chapter introduces a modified attention mechanism known as the confined attention mechanism in LSTM models to improve their ability to focus on relevant input data when making predictions. Algorithm 3 describes the integration of confined attention into LSTM.

In a Confined Attention Mechanism, the attention is restricted to a local window around the current time step. The model considers a fixed-size context window (previous $p$ time steps) when making predictions; $p$ is a hyperparameter. Here, $\boldsymbol{H}_{h-(k-1)}$ represents the hidden states within this window, and $\alpha_k$ represents the

**Algorithm 3** Confined Attention with LSTM

**Require:** Hidden states $h_t$ for $t = 1, 2, \ldots, n$
**Ensure:** Context vector $c$

**Notation:**
$h_t$: Hidden state of LSTM at time step $t$
$n$: Number of time steps in the sequence
$p$: Confined window size
$\mathcal{I}$: Index set of the confined window, $\mathcal{I} = \{n - p + 1, \ldots, n\}$
$\boldsymbol{s}$: Vector of alignment scores in $\mathbb{R}^p$
$\boldsymbol{W}_s$: Weight matrix for alignment score
$\boldsymbol{b}_s$: Bias vector for alignment score
$\boldsymbol{\alpha}$: Attention weights
$c$: Context vector

▷ Select the candidate window only; weights will be data-driven
$\mathcal{I} \leftarrow \{\, n - p + 1,\ n - p + 2,\ \ldots,\ n \,\}$
Initialize $\boldsymbol{s} \leftarrow \mathbf{0} \in \mathbb{R}^p$
**for** $k \leftarrow 1$ **to** $p$ **do**
    $\text{idx} \leftarrow n - (k - 1)$ ▷ maps $k = 1$ to $n$, $k = p$ to $n - p + 1$
    $s_k \leftarrow \tanh\left(h_{\text{idx}}\, \boldsymbol{W}_s + \boldsymbol{b}_s\right)$ ▷ alignment from hidden state
**end for**
$\boldsymbol{\alpha} \leftarrow \text{softmax}(\boldsymbol{s})$ ▷ normalize over the $p$ window elements
$c \leftarrow \sum_{k=1}^{p} \alpha_k\, h_{\,n-(k-1)}$ ▷ compute context vector
**return** $c$

attention weight for the $k^{\text{th}}$ state. The alignment score $A_k$ for the $k^{\text{th}}$ hidden state is given by Eq. (4.11):

$$A_k = \tanh\left(\boldsymbol{H}_{h-(k-1)} \cdot \boldsymbol{W}_s + \boldsymbol{b}_s\right) \tag{4.11}$$

The attention weight $\alpha_k$ is obtained by normalizing the alignment scores via softmax, as in Eq. (4.12):

$$\alpha_k = \frac{\exp(A_k)}{\sum_{j=1}^{p} \exp(A_j)} \tag{4.12}$$

The context vector $\boldsymbol{H}_h$ is then computed as the weighted sum of hidden states, given in Eq. (4.13):

$$\boldsymbol{H}_h = \sum_{k=1}^{p} \alpha_k \boldsymbol{H}_{h-(k-1)} \tag{4.13}$$

### 4.2.3 Proposed WPF Framework

The proposed WPF framework is based on the CAELSTM model and comprises several components: Data collection, data preprocessing, Bayesian optimization of model hyperparameters, prediction of wind power using CAELSTM, and model evaluation. Fig. 4.3 illustrates the detailed process of the real-time WPF framework proposed in this chapter.

**Step 1: Data preprocessing.** After preparing the wind dataset, the data are normalized using Eq. (4.14) and then split into training, validation, and testing sets. The normalization is performed as

$$\hat{r} = \frac{r - \mu}{\delta} \tag{4.14}$$

where $r$ represents the original data, $\mu$ is the mean, $\delta$ is the standard deviation, and $\hat{r}$ is the normalized value. To ensure proper temporal alignment and prevent future information leakage, special care was taken when integrating NWP features into the dataset. For each prediction time $t$, only NWP forecasts that would have been operationally available prior to or at time $t$ were used. Specifically, each NWP input was aligned based on its issuance time and corresponding lead time, ensuring that no data from future NWP runs (i.e., those issued after time $t$) were accessible during training or testing. This approach mimics real-world data availability and

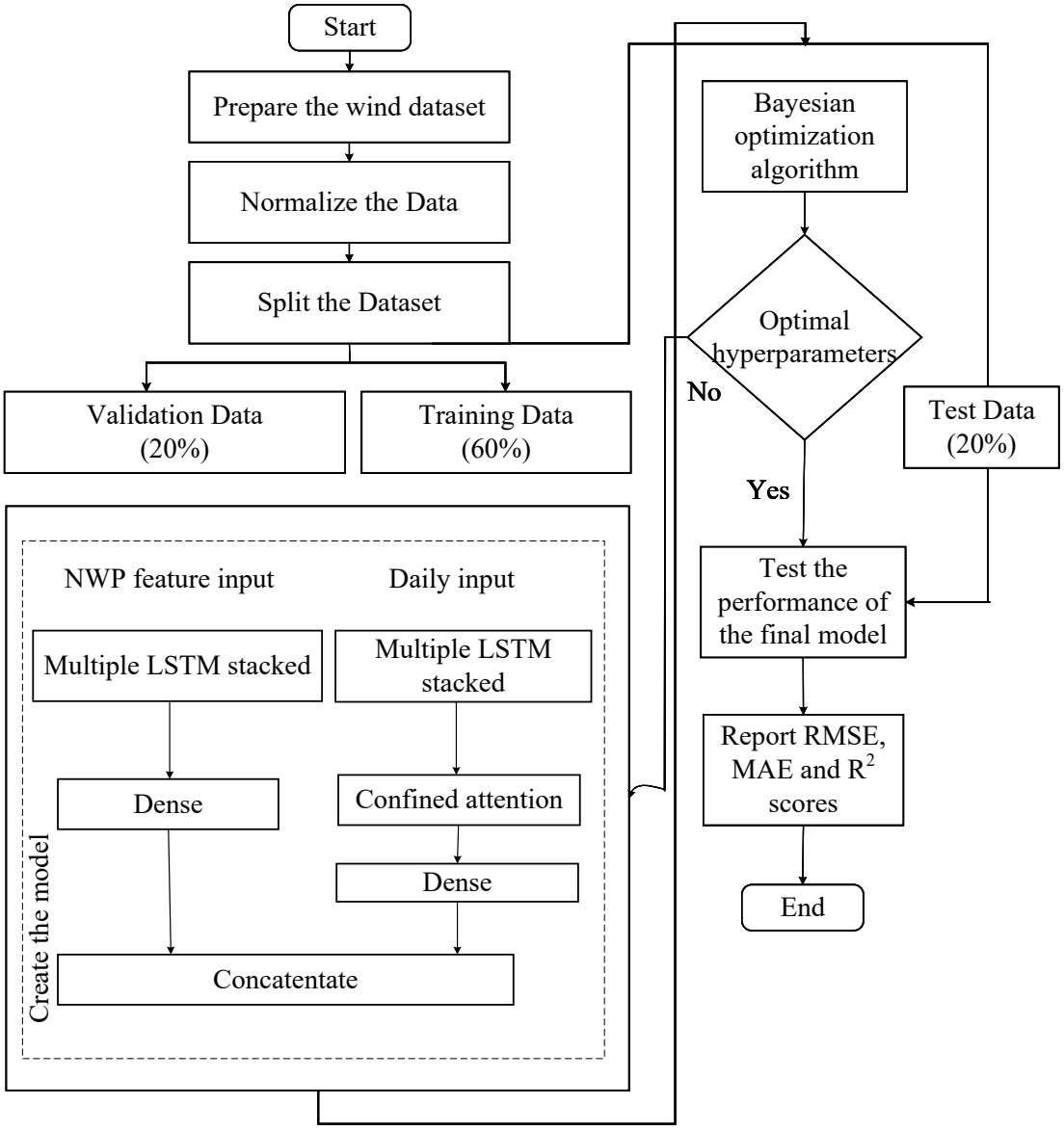



FIGURE 4.3: Schematic representation of the proposed WPF methodology.

guarantees that the model does not exploit information that would be unavailable in a true forecasting scenario. Furthermore, the time-series-based split for training, validation, and testing preserves the chronological order, ensuring strict causality and isolating the test set from any potential temporal leakage.

**Step 2: Feature importance analysis.** After creating the NWP feature set

as described in Fig. 3.2, feature importance analysis was conducted to identify the most significant features contributing to the prediction of wind power. For this section, the Random Forest Regressor, an ensemble technique, is employed to aggregate predictions from multiple decision trees. Feature importance in a Random Forest Regressor is computed based on the decrease in variance brought by each feature. The variance at node $t$ with $N$ samples is denoted by $V(t)$ and is given by Eq. (4.15):

$$V(t) = \frac{1}{N}\sum_{i=1}^{N}(y_i - \bar{y})^2 \tag{4.15}$$

where, $\bar{y}$ is the mean of the target values at node $t$. The variance reduction $\Delta V_j(T_b)$ for a split on feature $j$ at node $t$ in tree $T_b$ is then given by Eq. (4.16):

$$\Delta V_j(T_b) = I(t) - \left(\frac{N_L}{N}V(t_L) + \frac{N_R}{N}V(t_R)\right) \tag{4.16}$$

where $V(t_L)$ and $V(t_R)$ are the variance of the left and right child nodes, $N_L$ and $N_R$ are the number of samples in the left and right child nodes, and $N$ is the total number of samples at node $t$.

Finally, the Importance score for feature $j$ is computed by averaging the variance reductions across all trees, as expressed in Eq. (4.17):

$$\text{Importance}(j) = \frac{1}{B}\sum_{b=1}^{B}\Delta V_j(T_b) \tag{4.17}$$

where $B$ is the total number of trees in the forest.

**Step 3: Bayesian optimization of model hyperparameters.** The Bayesian optimization algorithm enhances model accuracy by dynamically adjusting sampling points for the unknown target function [182]. The hyperparameter selection process for deep learning models via Bayesian optimization aims to minimize the loss function $f(x)$. The optimization problem is expressed in Eq. (4.18):

$$x^* = \text{argmin}_{x\in X} f(x) \tag{4.18}$$

where $x$ denotes the hyperparameter and $X = (x_1, x_2, \ldots, x_n)$ comprises various hyperparameter combinations.

The Bayesian optimization process follows these steps:

1. Initialize dataset $D$ by recording pairs $(x_1, y_1), \ldots, (x_n, y_n)$ where $y_i = f(x_i)$.
2. Construct model $M$ using the dataset $D$.
3. Choose $x_i$ from set $X$ that fulfills Eq. (4.18) based on model $M$.
4. Implement the selected hyperparameter $x_i$ in the predictive model to derive $y_i$.
5. Enhance dataset $D$ with the newly acquired data point $(x_i, y_i)$.

Steps 2 to 5 are iterated until suitable hyperparameter combinations that minimize the predictive model's loss function are identified.

**Step 4: Feature fusion.** The proposed WPF framework consists of two LSTM modules. The first LSTM module, equipped with a confined attention mechanism, learns the temporal patterns, while the second LSTM module learns the periodic features. The different features extracted from various modules are combined to make the final prediction. Here, concatenation is employed to integrate the periodic and temporal features.

Let $\boldsymbol{H}_h$ denote the output from the confined attention module and $\boldsymbol{H}_{nwp}$ the output from the second LSTM module. These outputs are concatenated to create a feature vector, as shown in Eq. (4.19):

$$\boldsymbol{H}^{concat} = \text{concat}(\boldsymbol{H}_h, \boldsymbol{H}_{nwp}) \tag{4.19}$$

where $\boldsymbol{H}^{concat}$ is the combined feature representation.

The concatenated feature vector is then passed to a fully connected layer with linear activation to produce the predicted wind power $y$, expressed in Eq. (4.20):

$$y = \boldsymbol{H}^{concat}\boldsymbol{W}^y + \boldsymbol{b}^y \tag{4.20}$$

where $\boldsymbol{W}^y$ and $\boldsymbol{b}^y$ are learnable parameters of the dense layer.

The CAELSTM model adopts a modular design motivated by the multi-scale nature of wind dynamics. The first module integrates a confined attention mechanism with LSTM to focus on recent historical inputs that are most relevant for short-term forecasting. This design enhances the model's responsiveness to local temporal fluctuations in wind power. The second LSTM module

independently models periodic patterns present in NWP data, including daily and seasonal cycles, by learning from features such as wind speed and direction at various heights. This two-stage framework is further unified by concatenating the outputs of both modules before the final prediction layer. The integration is empirically validated in Section IV-A, where ablation studies demonstrate that removing either the confined attention or periodic LSTM component leads to noticeable performance degradation. Thus, CAELSTM offers an interpretable, physically consistent architecture that improves forecasting accuracy by capturing both localized and global wind behaviors.

## 4.3 Experiments and Results

This section demonstrates the application of the proposed approach on two real-world standard engineering test cases, allowing comparison with other well-known models. It provides details on the dataset, performance metrics, testing configurations, and an in-depth analysis for short-term WPF. Further, the performance of the proposed model was evaluated against several baselines and state-of-the-art methods, i.e., support vector regression (SVR) [183], random forest (RF) [184], LSTM [22], CNN-LSTM[185], GRU[186], N-BEATS [187], Neural hierarchical interpretable time Series (N-HiTS) [188], Autoformer [189], Informer [190], TCN [191], WaveNet [192], MTTFA-LSTM[22], and STAM-LSTM [193]. To evaluate the proposed model's effectiveness, three standard metrics for machine learning-based wind power prediction are utilized: Normalized mean absolute error (NMAE), Normalized root mean squared error (NRMSE), and $R^2$.

Let $T$ denote the total number of testing samples, $y_t$ the actual wind power generation at time $t$, $\hat{y}_t$ the predicted wind power generation at time $t$, and $\bar{y}_t$ the mean of the actual generation values. The MAE, RMSE, and $R^2$ are defined in Eq. (5.23), Eq. (4.22), and Eq. (5.25):

$$MAE = \frac{1}{T}\sum_{t=1}^{T}|y_t - \hat{y}_t|, \tag{4.21}$$

$$RMSE = \sqrt{\frac{1}{T}\sum_{t=1}^{T}(y_t - \hat{y}_t)^2}, \tag{4.22}$$

$$R^2 = 1 - \frac{\sum_{t=1}^{T}(y_t - \hat{y}_t)^2}{\sum_{t=1}^{T}(y_t - \bar{y}_t)^2}. \tag{4.23}$$

The normalized metrics NMAE and NRMSE facilitate comparison across different scales and datasets. They are calculated as in Eq. (7.39) and Eq. (7.38):

$$NMAE = \frac{MAE}{\frac{1}{T}\sum_{t=1}^{T}|y_t|}, \tag{4.24}$$

$$NRMSE = \frac{RMSE}{\sqrt{\frac{1}{T}\sum_{t=1}^{T}(y_t)^2}}, \tag{4.25}$$

### 4.3.1 Training Configuration

This segment details the training configurations employed by the proposed model. The objective is to minimize the difference between predicted and actual wind power. The Mean Squared Error (MSE) is used as the loss function $E(\theta)$, defined in Eq. (4.26):

$$E(\theta) = \underset{\theta}{\operatorname{argmin}} \left( \frac{1}{T} \sum_{t=1}^{T} \left( y_t - \hat{y}_t \right)^2 \right) \tag{4.26}$$

The prediction function $y^i$ is represented in (4.27):

$$\hat{y}_t = h_\theta \left( \boldsymbol{X}_h^t, \boldsymbol{X}_{nwp}^t \right), \tag{4.27}$$

where $\boldsymbol{X}_h^t$ and $\boldsymbol{X}_{nwp}^t$ denote the $t^{\text{th}}$ inputs from historical wind power and NWP features, respectively. Here, $h_\theta(\cdot)$ represents the predictive model parameterized by $\theta$, which maps the input features to the predicted wind power output.

A deep neural network's accuracy depends on the network's size and the number of training epochs [194]. Increasing the number of training epochs during backpropagation training may result in overfitting. To address this, a 20 per cent dropout rate was employed, which helps mitigate overfitting and improves generalization error. The models were developed using TensorFlow, Keras, and Pandas packages in Python within a Jupyter Notebook environment. The training process utilized various hyperparameters, including learning rate $(\mu) = 0.01$, batch size $= 32$, and number of epochs $= 200$. Adam optimizer was used to train the CAELSTM model. It is an effective stochastic optimization technique

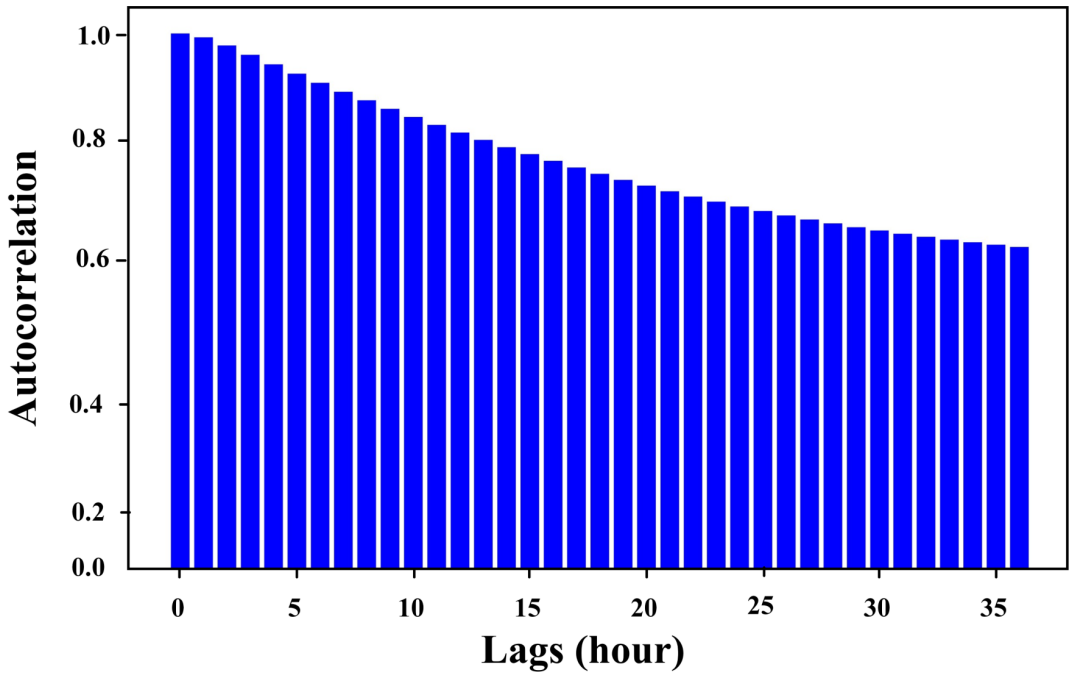



FIGURE 4.4: Correlation of one timestamp with previous timesteps.

that integrates the advantages of AdaGrad and RMSProp algorithms [195]. The experiments were carried out on a machine with the following configuration: CPU - 12th Gen Intel Core i7-12700 @ 2.10 GHz, Memory - 16 GB, System type - 64-bit operating system.

Further, Bayesian optimization with TPE is utilized to select the hyperparameters for the proposed model. The Hyperparameters and search space are elaborated as follows: The number of layers for each of the two LSTM modules was searched within the range {1, 2, 3, 4, 5}, and the number of hidden units was selected from {16, 32, 64, 128}. The Bayesian optimization process identified an optimal architecture with 4 layers in the first module (temporal features) and 4 layers in the second module (periodic features), for a total of 8 LSTM layers in the framework. The optimized model uses 32 hidden units in all layers. Here, $p$ is also a hyperparameter that denotes the window size in the confined attention mechanism. The value of $p$ is also decided by Bayesian optimization ($p \in 8, 16, 32$) for the proposed model; the optimized value of $p$ is 16. From Fig. 4.4 it is observed

that a current time step is highly correlated to the immediate previous timestamp, and this correlation reduces as the number of timestamps increases. Therefore, selecting the number of steps used for prediction at the previous time is also crucial. If fewer time steps are used, extracting the temporal features will be challenging; if the time steps are too long, it may increase complexity. A range of timestamp values was considered to evaluate the impact of time steps on predictions. It was observed that RMSE decreased as the prediction horizon increased, with the minimum RMSE observed at timestamp $= 30$.

### 4.3.2 Datasets

For the experiments, two real-world datasets from wind farms are used: the Tuticorin wind farm (India) and the La Haute Borne wind farm (France). While the historical wind generation dataset for the first one has been collected from the State Load Dispatch Centre (SLDC) commercial dataset, the other one is readily available for public use and contains historical data for various exogenous variables and wind power generation.

#### *Dataset 1: Tuticorin wind farm*

The Tuticorin wind farm (Latitude: 8°48′17.9″, Longitude: 78°8′41.9″), located in southern India, is used as the first case study in this work. The weekly commercial dataset consists of actual wind power generations aggregated at 15-minute intervals. This historical wind power generation data, together with NWP data collected from the NASA POWER project, is used for training the prediction models. The data for the GIREL wind site, with an installed capacity of 62.475 MW, was collected from January 2021 to December 2021, yielding one year of observations. Notably, the wind data is aggregated at 15-minute intervals, resulting in 96 timestamps per day. Thus, the complete dataset contains 35,040 individual samples. The dataset is divided into 60% for training, 20% for validation, and 20% for testing to enhance model generalization and reduce overfitting.

### *Dataset 2: La Haute Borne wind farm*

La Haute Borne wind farm is an operating wind farm in France (Latitude: 48°26′58.5″, Longitude: 5°35′80″). French energy group Engie SA has made data openly available for its 21 MW wind farm La Haute Borne through the ENGIE Open Data Renewables platform. It consists of four wind turbines, namely R80711, R80721, R80736, and R80790. These four turbines have a rated power of 2050 kW and a hub height of 80 m. The dataset for the first wind turbine was collected for the year 2017, with a resolution of 10 minutes, making 52,206 data samples. The data split is the same as for dataset 1.

**Table 4.1** Comparison of Baselines without Confined Attention Mechanism for two datasets

| **Indexes** | **Data Preprocessing** | **Dataset-1** | | | **Dataset-2** | | |
|---|---|---|---|---|---|---|---|
| | | NRMSE(%) | NMAE(%) | $R^2$ | NRMSE(%) | NMAE(%) | $R^2$ |
| **SVR** | No preprocessing | 9.597 | 7.319 | 0.70361 | 12.545 | 8.578 | 0.70126 |
| | PCA | 9.412 | 7.248 | 0.71086 | 11.252 | 7.586 | 0.71557 |
| | PCA and EMD | 9.228 | 7.106 | 0.72537 | 10.237 | 7.224 | 0.72480 |
| **RF** | No preprocessing | 12.570 | 10.327 | 0.62355 | 12.113 | 8.546 | 0.68457 |
| | PCA | 11.255 | 9.265 | 0.63022 | 11.500 | 8.625 | 0.69122 |
| | PCA and EMD | 10.649 | 8.278 | 0.63374 | 10.428 | 7.378 | 0.70264 |
| **LSTM** | No preprocessing | 5.936 | 2.075 | 0.94867 | 11.622 | 7.525 | 0.80370 |
| | PCA | 4.304 | 3.834 | 0.93215 | 6.198 | 6.906 | 0.82490 |
| | PCA and EMD | **4.155** | **3.141** | **0.95254** | **7.779** | **5.032** | **0.84881** |
| **GRU** | No preprocessing | 7.069 | 10.223 | 0.93125 | 9.255 | 7.547 | 0.82595 |
| | PCA | 6.571 | 8.511 | 0.93755 | 9.568 | 6.254 | 0.83567 |
| | PCA and EMD | 4.343 | 3.314 | 0.94465 | 7.819 | 5.117 | 0.84702 |
| **XGBoost** | No preprocessing | 19.152 | 12.261 | 0.61236 | 19.721 | 9.685 | 0.70258 |
| | PCA | 17.121 | 10.840 | 0.60126 | 19.888 | 10.538 | 0.71549 |
| | PCA and EMD | 11.028 | 8.709 | 0.62336 | 10.357 | 7.270 | 0.72174 |

### 4.3.3 Baselines for Comparison

Table 4.1 summarizes the experimental results, validating the selection of LSTM for our forecasting module. This subsection presents the performance of conventional machine learning models and deep learning architectures that serve as baselines for comparison. Each model is run for ten trials, and the average of RMSE, MAE and $R^2$ is taken to check the best-performing model. The baselines for comparison are SVR, RF, LSTM, CNN-LSTM, and GRU. These models were tested in three different scenarios: (i) No preprocessing, (ii) with PCA, and (iii) the parallel integration of

PCA and EMD. It is evident from the results that the performance of each model is significantly improved from the first scenario to the third scenario. For example, the LSTM model's test RMSE is reduced by 27.49% in the second scenario and 30.00% in the third scenario.

### 4.3.4 Comparative experiments with sensitivity analysis

The prediction accuracy of the proposed CAELSTM model is demonstrated in Fig. 4.5, where the forecasted wind power values are compared against actual generation for both datasets, along with the corresponding error distributions. The sensitivity analysis results presented in The CAELSTM model consistently achieves the lowest mean error values and standard deviations in both NRMSE and NMAE, underscoring its robustness and reliability in capturing the temporal and nonlinear dynamics of wind power forecasting. A major contributor to this performance is the confined attention mechanism, which enables the model to prioritize informative features while suppressing irrelevant ones, thereby minimizing prediction errors. In contrast, MTFFA-LSTM and STAM-LSTM exhibit higher variability, with performance gaps attributed to their less adaptive attention mechanisms and feature extraction strategies. For Dataset-1 (Fig. 4.5 (a)), CAELSTM predictions align closely with the actual values across the forecast horizon, showing minimal deviations. The error signal remains consistently low, reflecting the model's ability to adapt to periodic and relatively less complex patterns. For Dataset-2 (Fig. 4.5 (b)), which is characterized by greater variability and abrupt changes, CAELSTM continues to demonstrate strong predictive performance. Although forecast errors increase slightly due to the dataset's complexity, the model effectively tracks the actual generation trends, confirming its generalization capability across more challenging scenarios.

Models such as N-BEATS and N-HiTS exhibit competitive but slightly higher error rates compared to CAELSTM, indicating their effectiveness yet underscoring the advantage provided by CAELSTM's architectural enhancements that may better capture temporal dependencies and feature interactions. The Autoformer, Informer, TCN, and WaveNet models show moderate performance, suggesting that while these models effectively handle sequence modeling tasks, their assumptions or design choices might limit their effectiveness in reducing errors

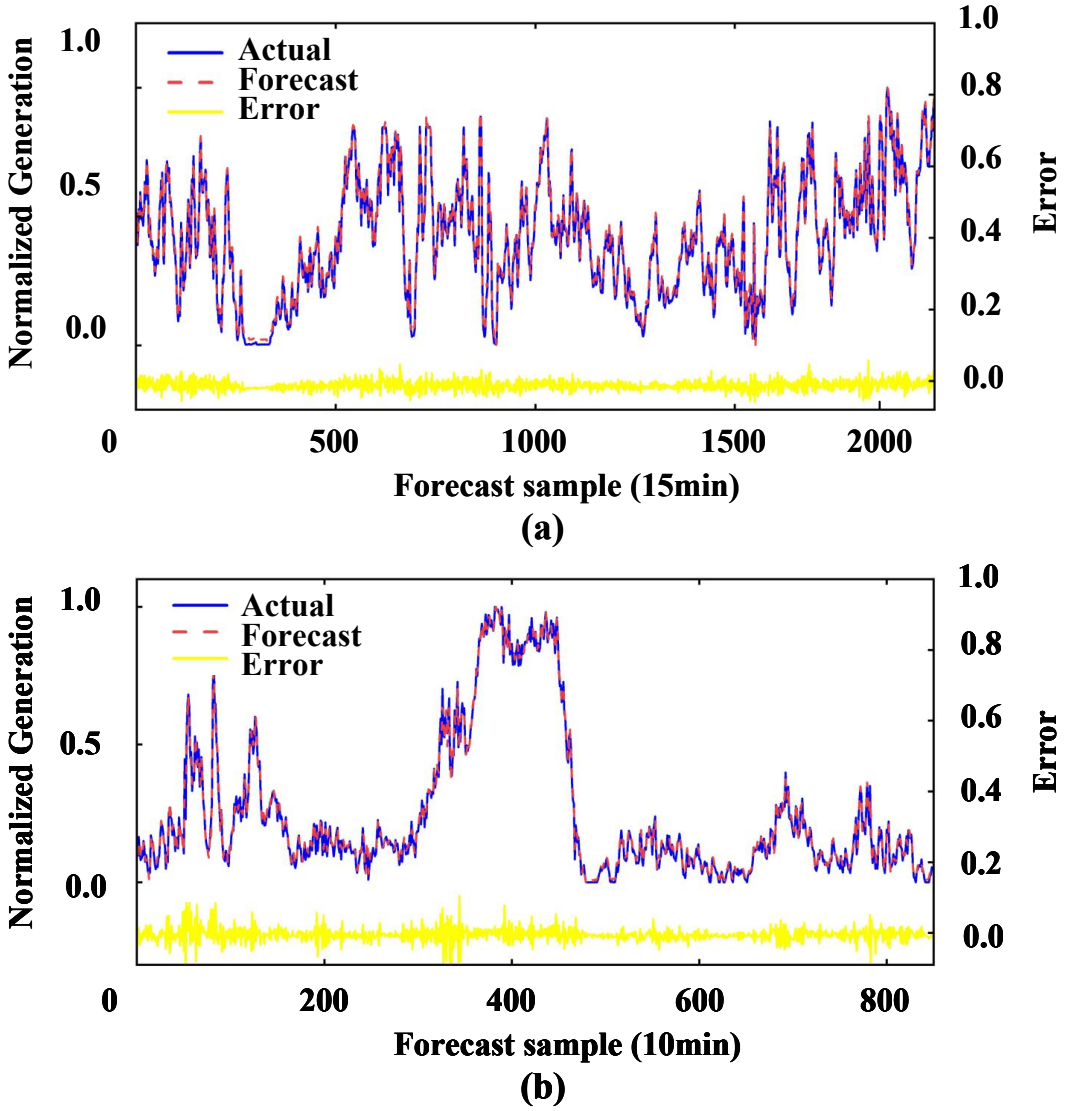



FIGURE 4.5: Wind power generation forecast for (a) Dataset-1 (b) Dataset-2.

when compared to CAELSTM. Autoformer and Informer, known for their attention-based mechanisms, likely experience higher error due to their sensitivity to longer sequence dynamics and potential overfitting. The MTTFA-LSTM model exhibits significantly higher errors, indicating a possible mismatch between the model's assumptions and Dataset-1's underlying patterns, resulting in less stable performance and reduced predictive accuracy. Similarly, STAM-LSTM shows variable performance, as reflected by its notably high standard deviation, indicating inconsistency and sensitivity to fluctuations within the dataset. Overall, the superior and consistent performance of CAELSTM across both Dataset-1 and Dataset-2 validates its effectiveness for robust and reliable WPF, highlighting the advantages of its hybrid design in capturing complex data dynamics. Tables 4.2 and 4.3 further provide a comprehensive comparison of model performance across the two datasets.

**Table 4.2** Sensitivity Analysis for Dataset-1

| | **NRMSE (%)** | | | | |
|---|---|---|---|---|---|
| **Model** | **Best** | **Worst** | **Mean** | **Median** | **Std.** |
| **CAELSTM (Ours)** | **1.620** | **1.777** | **1.664** | **1.671** | 0.072 |
| N-BEATS | 1.702 | 1.891 | 1.749 | 1.752 | 0.088 |
| N-HiTS | 1.725 | 1.903 | 1.768 | 1.772 | 0.092 |
| Autoformer | 1.815 | 2.012 | 1.862 | 1.869 | 0.103 |
| Informer | 1.831 | 2.043 | 1.878 | 1.882 | 0.112 |
| TCN | 1.788 | 1.965 | 1.823 | 1.831 | 0.097 |
| WaveNet | 1.803 | 1.992 | 1.842 | 1.849 | 0.101 |
| MTTFA-LSTM | 2.279 | 2.326 | 2.160 | 2.168 | 0.113 |
| STAM-LSTM | 1.713 | 2.050 | 1.661 | 1.653 | 0.187 |
| | **NMAE (%)** | | | | |
| **CAELSTM (Ours)** | **1.254** | **1.396** | **1.294** | **1.300** | **0.065** |
| N-BEATS | 1.312 | 1.452 | 1.348 | 1.351 | 0.073 |
| N-HiTS | 1.335 | 1.467 | 1.372 | 1.378 | 0.069 |
| Autoformer | 1.402 | 1.538 | 1.428 | 1.431 | 0.082 |
| Informer | 1.418 | 1.562 | 1.447 | 1.452 | 0.085 |
| TCN | 1.386 | 1.512 | 1.417 | 1.422 | 0.078 |
| WaveNet | 1.395 | 1.527 | 1.428 | 1.432 | 0.081 |
| MTTFA-LSTM | 1.735 | 1.766 | 1.655 | 1.661 | 0.076 |
| STAM-LSTM | 1.325 | 1.515 | 1.296 | 1.291 | 0.106 |

**Table 4.3** Sensitivity analysis for Dataset-2.

| | NRMSE (%) | | | | |
|---|---|---|---|---|---|
| **Model** | **Best** | **Worst** | **Mean** | **Median** | **Std.** |
| **CAELSTM (Ours)** | **3.971** | **4.007** | **3.981** | **3.982** | **0.017** |
| N-BEATS | 4.192 | 4.423 | 4.251 | 4.257 | 0.121 |
| N-HiTS | 4.215 | 4.451 | 4.278 | 4.283 | 0.125 |
| Autoformer | 4.385 | 4.682 | 4.473 | 4.481 | 0.142 |
| Informer | 4.402 | 4.712 | 4.492 | 4.499 | 0.151 |
| TCN | 4.318 | 4.562 | 4.392 | 4.398 | 0.132 |
| WaveNet | 4.335 | 4.588 | 4.412 | 4.418 | 0.136 |
| MTTFA-LSTM | 6.135 | 6.218 | 5.925 | 5.940 | 0.200 |
| STAM-LSTM | 4.546 | 4.759 | 4.513 | 4.508 | 0.118 |
| | **NMAE (%)** | | | | |
| **Model** | **Best** | **Worst** | **Mean** | **Median** | **Std.** |
| **CAELSTM (Ours)** | **3.206** | **3.226** | **3.211** | **3.212** | **0.009** |
| N-BEATS | 3.382 | 3.562 | 3.421 | 3.427 | 0.095 |
| N-HiTS | 3.401 | 3.588 | 3.442 | 3.448 | 0.098 |
| Autoformer | 3.541 | 3.732 | 3.582 | 3.589 | 0.112 |
| Informer | 3.562 | 3.758 | 3.604 | 3.612 | 0.117 |
| TCN | 3.487 | 3.672 | 3.528 | 3.534 | 0.102 |
| WaveNet | 3.502 | 3.692 | 3.544 | 3.551 | 0.107 |
| MTTFA-LSTM | 5.031 | 5.145 | 4.742 | 4.762 | 0.276 |
| STAM-LSTM | 3.735 | 4.094 | 3.680 | 3.671 | 0.200 |

## Dataset Characteristics and Model Generalization

A comparative analysis of the Tuticorin and La Haute Borne datasets reveals significant differences in wind characteristics attributable to their geographical and climatic conditions. The Tuticorin wind farm (India), characterized by a tropical climate, demonstrates moderate and stable wind speeds with a mean of approximately 6.38 m/s and moderate variability (standard deviation: 2.41 m/s). The wind profile exhibits clear seasonal trends corresponding to regional monsoon patterns. In contrast, the La Haute Borne wind farm (France), situated in a temperate region, shows higher variability in rotor speeds (mean: 18.59 rpm, standard deviation: 6.81 rpm), reflecting more dynamic wind speed fluctuations. Together, they imply varied complexities in forecasting tasks, with La Haute Borne posing a more challenging scenario due to greater variability and abrupt wind speed transitions. Consequently, these inherent dataset characteristics significantly impact the CAELSTM model's generalization capability, underscoring its

adaptability to diverse wind dynamics. The CAELSTM model, specifically designed with a confined attention mechanism, effectively addresses these differences by adaptively prioritizing the most recent and relevant historical wind data, thereby enhancing its predictive robustness across diverse conditions. The inclusion of NWP features further aids in capturing broader atmospheric and seasonal variations, allowing the model to adjust its predictive strategy according to underlying regional weather patterns. Consequently, the CAELSTM model maintains high forecasting accuracy and demonstrates a robust generalization capability, performing reliably in both moderately stable wind conditions (Tuticorin) and highly variable, temperate climatic scenarios (La Haute Borne). This robust performance indicates the model's suitability for broader application across geographically and climatologically diverse wind farms. These statistical differences are quantitatively summarized in Table 4.4 and further illustrated in Fig. 4.6 and Fig. 4.7.

**Table 4.4** Statistical comparison of wind characteristics.

| **Statistic** | **Wind Speed(m/s) [Dataset-1]** | **Rotor Speed(rpm) [Dataset-2]** |
|---|---|---|
| Mean | 6.38 | 18.59 |
| Standard Deviation | 2.41 | 6.81 |
| Minimum | 0.05 | 1.78 |
| Maximum | 15.67 | 37.00 |

## Interpretability Analysis

To assess whether the confined attention mechanism in the CAELSTM model effectively captures wind dynamics, an interpretability analysis was conducted by visualizing attention weights across temporal steps. Such analysis is essential to justify the model's ability to prioritize historical inputs that are most informative for wind power generation. In the CAELSTM framework, attention weights indicate the relative contribution of each past time step to the final prediction. The results reveal that the model assigns greater importance to the most recent time steps (e.g.,

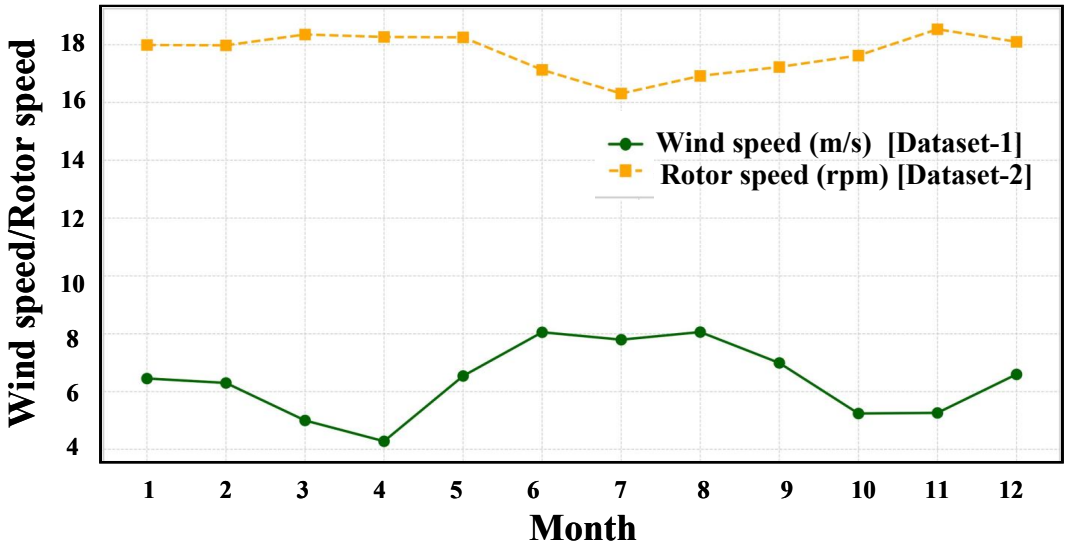



FIGURE 4.6: Monthly average wind characteristics comparison between Two Datasets, illustrating distinct seasonal and climatic wind variability.

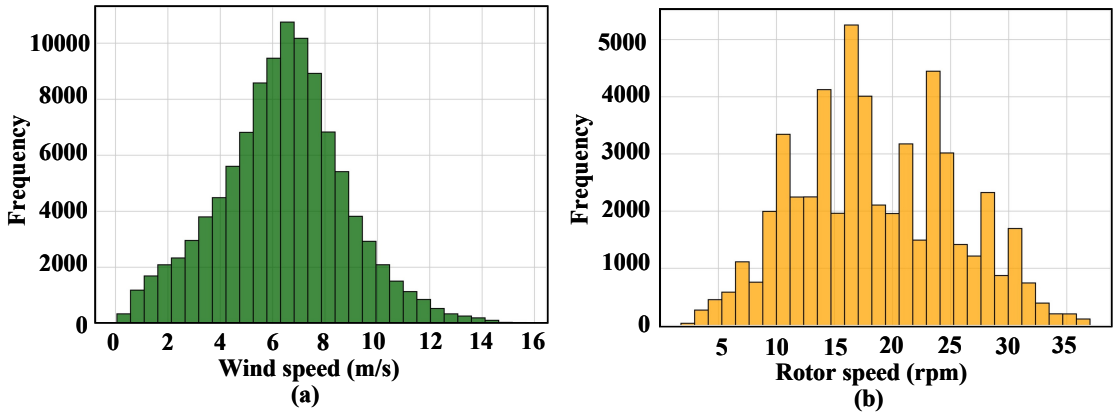



FIGURE 4.7: Distribution of wind speeds for Dataset-1 and rotor speeds for Dataset-2, highlighting differences in variability and frequency distribution between the two locations.

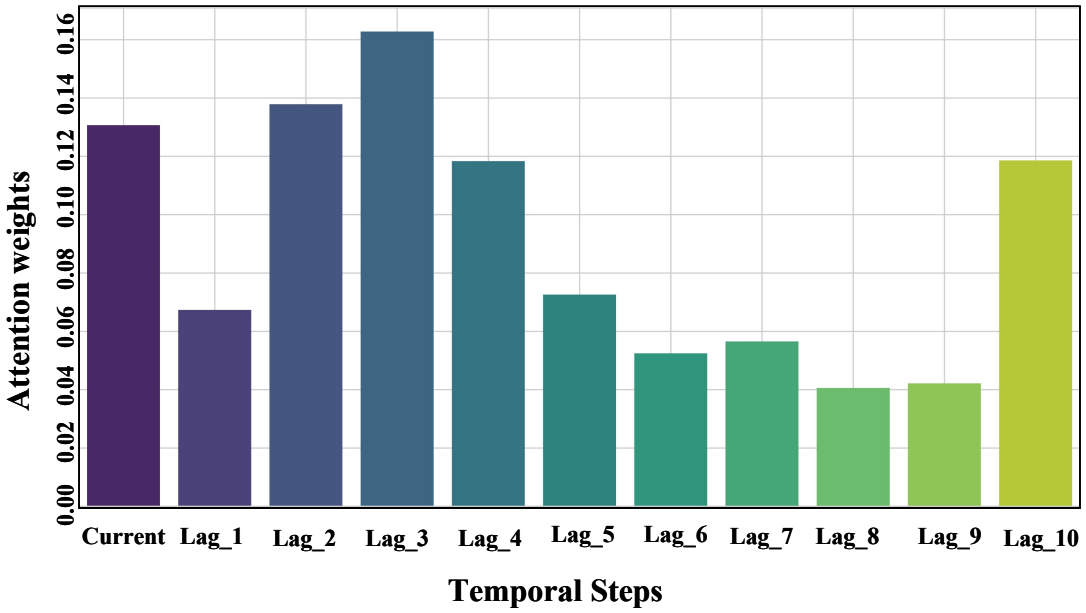



FIGURE 4.8: Attention weight distribution across temporal steps in the CAELSTM model.

Lag-1 to Lag-4), where wind speed variations are most relevant for short-term forecasting. The gradual decline in weights for earlier lags reflects the confined attention's focus on localized dynamics near the prediction horizon, consistent with the physical intuition of wind inertia and system response. This attention pattern demonstrates that the confined attention mechanism not only improves predictive accuracy but also enhances interpretability by explicitly identifying the most influential temporal features. Hence, the analysis confirms that CAELSTM does not treat inputs uniformly but dynamically emphasizes critical timestamps, thereby effectively learning the underlying wind dynamics. Fig. 4.8 shows the distribution of attention weights for a representative forecasting instance from the Tuticorin dataset.

## 4.4 Discussion

The wind power generation of two wind farms was predicted using a Confined Attention-enabled hybrid deep learning model. A set of experiments were conducted to analyze the performance behaviour of the CAELSTM model

**Table 4.5** Performance of Different Variants of CAELSTM for two datasets

| Variants | Dataset-1 | | | Dataset-2 | | |
|---|---|---|---|---|---|---|
| | NRMSE(%) | NMAE(%) | $R^2$ | NRMSE(%) | NMAE(%) | $R^2$ |
| w/o second LSTM module | 1.842 | 1.401 | 0.9726 | 4.084 | 3.579 | 0.9422 |
| w/o Confined Attention | 1.771 | 1.439 | 0.9627 | 4.104 | 3.678 | 0.9437 |
| w/o NWP features | 1.674 | 1.389 | 0.9746 | 4.341 | 3.552 | 0.9450 |
| w/o FC layer | 1.666 | 1.809 | 0.9826 | 4.514 | 4.624 | 0.9469 |
| CAELSTM | **1.620** | **1.254** | **0.9925** | **3.972** | **3.206** | **0.9564** |

compared to state-of-the-art baseline models for predicting wind power generation. In addition to the experimental results mentioned above, a comprehensive analysis consisting of Ablation studies and non-parametric statistical tests of CAELSTM model was conducted and presented below:

### 4.4.1 Ablation Study

This subsection discusses the significant contribution of all the modules in CAELSTM, the following variants of CAELSTM model were designed to highlight the importance of each component

1. w/o Second LSTM module: The second LSTM module was removed from CAELSTM model. The second LSTM model captures the wind's NWP features (periodic properties).

2. w/o Confined Attention: The confined attention mechanism applied with the first LSTM module was replaced with the conventional attention mechanism in this model.

3. w/o NWP data: The NWP features were removed from the model, and the prediction was made without additional information about weather parameters.

4. w/o FC layer: In this model, a simple average of the predictions was used for the wind power generation forecast instead of the FC layer.

All the variants follow similar settings, except for the variations mentioned earlier. Table 4.5 presents the performance of each variant on both datasets. The

performance of CAELSTM drops significantly when the second LSTM module is removed. This is because wind data exhibit strong temporal and periodic properties, which the second LSTM module effectively captures. In addition, the confined attention mechanism is designed to highlight the important timestamps in the input sequence, and its removal adversely affects model performance. The inclusion of NWP features provides the model with additional weather-related information, enabling it to make more accurate predictions by incorporating real-time atmospheric conditions and trends. Similarly, the feature concatenation layer allows the model to estimate the contribution of each feature, helping it weigh their importance in the prediction process. On the other hand, taking a simple average of predictions overlooks the nuances of individual features, leading to suboptimal results. Therefore, a more sophisticated approach is required to capture the complexities of the data.

### 4.4.2 Statistical Analysis

To substantiate the robustness of the proposed CAELSTM model, a comprehensive statistical analysis was performed to assess the performance improvements in the proposed model against various models. This analysis included paired t-tests to ascertain the statistical relevance of the proposed model. The paired t-test, a robust method for comparing the means of two related groups, was chosen to analyze whether the proposed CAELSTM model statistically improves the performance metrics compared to the benchmark methods. Table 4.6 and 4.7 summarizes the t-test results for each model comparison. For all models, the null hypothesis was that there was no difference in performance metrics between the benchmark models and the proposed model. The significantly low $p$-values obtained from the t-tests (all $p$-values $< 0.05$) strongly reject the null hypothesis. This indicates that the differences in predictions of the proposed and benchmark models are statistically significant. The statistical evidence further supports our hypothesis that the CAELSTM model enhances the forecasting capabilities of traditional WPF models, making it a valuable addition to wind energy forecasting methodologies.

**Table 4.6** Paired t-test results on prediction values for Dataset-1

| **Comparison** | **t-statistic** | **p-value** |
|---|---|---|
| CAELSTM vs N-BEATS | $-4.234567$ | $2.10 \times 10^{-4}$ |
| CAELSTM vs N-HiTS | $-4.507891$ | $7.8 \times 10^{-5}$ |
| CAELSTM vs Autoformer | $-6.853210$ | $3.21 \times 10^{-7}$ |
| CAELSTM vs Informer | $-7.102334$ | $1.87 \times 10^{-7}$ |
| CAELSTM vs TCN | $-5.348012$ | $8.35 \times 10^{-6}$ |
| CAELSTM vs WaveNet | $-5.598743$ | $5.71 \times 10^{-6}$ |
| CAELSTM vs MTTFA-LSTM | $-3.098908$ | $2.04 \times 10^{-3}$ |
| CAELSTM vs STAM-LSTM | $-2.078411$ | $3.81 \times 10^{-2}$ |

**Table 4.7** Paired t-test results on prediction values for Dataset-2

| **Comparison** | **t-statistic** | **p-value** |
|---|---|---|
| CAELSTM vs N-BEATS | $-5.103221$ | $3.89 \times 10^{-6}$ |
| CAELSTM vs N-HiTS | $-5.353487$ | $2.14 \times 10^{-6}$ |
| CAELSTM vs Autoformer | $-6.902178$ | $4.67 \times 10^{-8}$ |
| CAELSTM vs Informer | $-7.054332$ | $2.84 \times 10^{-8}$ |
| CAELSTM vs TCN | $-6.200145$ | $9.32 \times 10^{-7}$ |
| CAELSTM vs WaveNet | $-6.448762$ | $4.83 \times 10^{-7}$ |
| CAELSTM vs MTTFA-LSTM | $-3.383788$ | $7.48 \times 10^{-4}$ |
| CAELSTM vs STAM-LSTM | $-6.922564$ | $8.76 \times 10^{-12}$ |

## 4.5 Summary

WPF significantly impacts the efficiency and reliability of energy production, enabling better integration of renewable sources into the grid. To enhance the precision of these forecasts, this manuscript introduced a hybrid framework referred to as CAELSTM. The framework utilizes an innovative attention mechanism to extract and learn temporal and periodic attributes of data and combines it with LSTM networks to improve prediction accuracy. This approach addresses the challenges of traditional forecasting methods and paves the way for more adaptive and responsive energy management systems. When compared to benchmark models in literature, the CAELSTM model demonstrates superior performance, achieving lower mean absolute error and higher correlation coefficients, thus validating its effectiveness in real-world applications. In summary, this chapter presents the following findings:

- Firstly, wind power features extracted by the Confined attention mechanism are unaffected by the noise present in the data, allowing LSTM to focus on the underlying patterns more effectively. Also, integrating this mechanism enhances the model's ability to generalize across different datasets, making it a robust choice for various forecasting scenarios.

- Second, the NWP features are also effectively utilized, improving prediction accuracy and further solidifying the model's reliability in dynamic environments.

- Third, the LSTM model's hyperparameter tuning process has been optimized through Bayesian optimization, ensuring that the model achieves peak performance while minimizing overfitting. Additionally, the feature fusion matrix incorporating parallel integration of EMD and PCA allows for a more comprehensive representation of the data, enhancing the model's interpretability and performance.

In future research, the ability of the CAELSTM model to produce probabilistic predictions will be explored, potentially offering insights into uncertainty quantification and risk assessment in forecasting applications.

## Publication from this Chapter

**P. Dhaka**, M. Sreejeth, and M. M. Tripathi, "Confined Attention-enabled Deep Learning Model with Feature Fusion for Wind Power Forecasting," IEEE Transactions on Consumer Electronics, under review, Publisher: IEEE. (IF: 10.9)

# Chapter 5

# Multivariate Wind Power forecasting using Ensemble techniques

## 5.1 Introduction

The increasing penetration of wind energy in modern power systems has intensified the need for accurate, stable, and computationally efficient short-term forecasting models. While the preceding chapters of this thesis explored deep learning frameworks-incorporating hybrid architectures, confined attention, and multi-scale temporal modeling-there remains a practical need for forecasting methods that can robustly handle multivariate inputs, mitigate noise-induced uncertainty, and deliver stable predictions under diverse operating conditions. Traditional single-model approaches, even when enhanced with deep learning components, often struggle to balance model complexity, interpretability, and consistency across large temporal datasets. Ensemble-based learning therefore emerges as a powerful alternative, offering the ability to combine diverse learners, reduce variance, and improve forecasting reliability.

This chapter develops a comprehensive multivariate wind power forecasting framework built upon advanced ensemble learning principles. The work begins by addressing the fundamental challenge of noise and non-stationarity in wind power data, employing CEEMDAN to decompose complex signals and extract cleaner intrinsic oscillatory modes. These decomposed components are then processed through Stacked GRU (SGRU) networks to capture non-linear temporal interactions more effectively. To enhance prediction robustness, the chapter introduces a hybrid ensemble strategy that integrates bagging and boosting in a

unified formulation-leveraging the strengths of both parallel and sequential learning paradigms. This approach provides an efficient mechanism for predictor error handling and improves the stability of short-term forecasts.

Finally, extensive empirical validation and statistical analysis are conducted to quantify the contributions of each stage of the proposed framework, including decomposition, network training, ensemble fusion, and error correction. Comparative studies against state-of-the-art forecasting models demonstrate the superiority of the proposed ensemble-based design in terms of accuracy, resilience to noise, and consistency across multivariate input configurations. Overall, this chapter extends the methodological findings of earlier chapters by presenting a robust, interpretable, and reliable ensemble learning paradigm for practical wind power forecasting applications.

## 5.2 Methodology

This section provides a brief introduction to the techniques involved in the proposed hybrid ensemble learning system to enhance understanding of the work.

### 5.2.1 Complete Ensemble Empirical Mode Decomposition with Adaptive Noise (CEEMDAN)

It provides a framework to decompose complex, nonlinear, and non-stationary signals into several intrinsic mode functions (IMFs) and a residual term. This approach overcomes the issue of "mode mixing" often encountered in EMD by introducing Gaussian white noise with adjustable signal-to-noise ratios (SNRs) at each step [196], leading to reduced reconstruction error and improved decomposition precision. As such, CEEMDAN is employed as the core decomposition technique in this study, with the algorithm's steps detailed below [197]:

1. Consider a signal, represented by $\widetilde{\chi}(u)$, with Gaussian white noise $\omega^j(u)$ having SNR $\xi_0$. The signal $\widetilde{\chi}(u)$ is combined with the noise term, forming $\widetilde{\chi}(u) + \xi_0\omega^j(u)$. This composite signal is iteratively decomposed using EMD to extract the initial IMF, denoted as $\phi^{\text{imf}}_{(1)}(u)$, which is calculated as per Eq.

5.1:

$$\phi_{(1)}^{\text{imf}}(u) = \frac{1}{P}\sum_{p=1}^{P}\phi_{1,p}^{\text{imf}}(u) \tag{5.1}$$

Here, $P$ represents the number of EMD iterations, and $\phi_{1,p}^{\text{imf}}(u)$ is the first IMF from the $p^{\text{th}}$ EMD execution.

2. Determine the first residual $\delta_{(1)}(u)$ by subtracting $\phi_{(1)}^{\text{imf}}(u)$ from $\widetilde{\chi}(u)$ as calculated in Eq. 5.2:

$$\delta_{(1)}(u) = \widetilde{\chi}(u) - \phi_{(1)}^{\text{imf}}(u) \tag{5.2}$$

3. Additional white noise $\omega^j(u)$ with SNR $\xi_m$ is introduced, forming a new composite signal, $\delta_{(m)}(u) + \xi_m E_m(\omega^j(u))$, where $E_m(\omega^j(u))$ represents the $m^{\text{th}}$ IMF generated by EMD. By repeating this process, the $(m+1)^{\text{th}}$ IMF $\phi_{(m+1)}^{\text{imf}}(u)$ is calculated as per Eq. 5.3

$$\phi_{(m+1)}^{\text{imf}}(u) = \frac{1}{P}\sum_{q=1}^{P} E_m\left(\delta_{(m)}(u) + \xi_m E_m(\omega^j(u))\right) \tag{5.3}$$

The corresponding residual $\delta_{(m+1)}(u)$ is then computed as in Eq. 5.4:

$$\delta_{(m+1)}(u) = \delta_{(m)}(u) - \phi_{(m+1)}^{\text{imf}}(u) \tag{5.4}$$

4. This iterative process continues until the residual $\delta_{(m+1)}(u)$ satisfies the termination criterion, which is mentioned in Eq. 5.5:

$$\sum_{u=0}^{U}\frac{\|\delta_{(m)}(u) - \delta_{(m+1)}(u)\|^2}{\delta_{(m)}^2(u)} \leq \text{SD}_k \tag{5.5}$$

where $U$ is the length of the initial signal $\widetilde{\chi}(u)$, and $\text{SD}_k$ typically ranges between 0.2 and 0.3.

Finally, the original signal $\widetilde{\chi}(u)$ can be reconstructed as shown in Eq. 5.6:

$$\widetilde{\chi}(u) = \sum_{m=1}^{M}\phi_m^{\text{imf}}(u) + R(u) \tag{5.6}$$

where $M$ is the total count of derived IMFs, and $R(u)$ represents the final residual.

### 5.2.2 Multivariate CEEMDAN (MCEEMDAN)

MCEEMDAN extends the Complete Ensemble Empirical Mode Decomposition with Adaptive Noise (CEEMDAN) method to handle multivariate data. Given a multivariate time series with $d$ variables, the MCEEMDAN algorithm can be described as follows:

1. For each variable $i$ in the multivariate time series, perform the CEEMDAN algorithm separately to decompose the variable into several IMFs and a residual component. This results in a set of IMFs $\phi_{i,1}^{imf}(u), \phi_{i,2}^{imf}(u), ..., \phi_{i,M_i}^{imf}(u)$ and a residual $\hat{R}_i(u)$ for each variable $i$.

2. For each IMF level $m$, combine the corresponding IMFs of all variables into a multivariate IMF
$\Phi_m^{imf}(u) = [\phi_{1,m}^{imf}(u), \phi_{2,m}^{imf}(u), ..., \phi_{d,m}^{imf}(u)]$.

3. Combine the residuals of all variables into a multivariate residual $\hat{R}(u) = [\hat{R}_1(u), \hat{R}_2(u), ..., \hat{R}_d(u)]$.

4. The original multivariate time series can be reconstructed as per Eq. 5.7:

$$\tilde{X}(u) = \sum_{m=1}^{M} \Phi_m^{imf}(u) + \hat{R}(u) \tag{5.7}$$

   where $M$ is the maximum number of IMFs obtained for any variable, and $\tilde{X}(u)$ is the original multivariate time series.

In this way, MCEEMDAN can handle multivariate time series by decomposing each variable separately and then combining the results. This allows it to capture both the individual characteristics of each variable and the joint dynamics across variables.

### 5.2.3 Stacked Gated Recurrent Units (SGRU) Networks

Recurrent Neural Network (RNN), specifically Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU), are well-suited for sequence learning tasks such as WPF . GRU, a variant of RNNs, offers advantages over LSTM due to its simplified architecture and computational efficiency. Fig 5.1 depicts the framework

of a classic GRU network that uses two gates called the reset gate and update gate. The reset gate ($\mathbf{r}_t$) manages what portion of the prior hidden state ($\mathbf{h}_{t-1}$) is utilized in the current time step. In contrast, the update gate ($\mathbf{z}_t$) controls the amount of the new candidate state ($\mathbf{c}_t$) that contributes to the revision of the hidden state.The hidden state $\mathbf{h}_t$ is updated by combining information from the previous hidden state and the newly generated candidate state. The equation for GRU are expressed as follows (Eq. 5.8-5.11):

$$\mathbf{z}_t = \sigma(\mathbf{W}_z \mathbf{x}_t + \mathbf{U}_z \mathbf{h}_{t-1} + \mathbf{b}_z) \tag{5.8}$$

$$\mathbf{r}_t = \sigma(\mathbf{W}_r \mathbf{x}_t + \mathbf{U}_r \mathbf{h}_{t-1} + \mathbf{b}_r) \tag{5.9}$$

$$\mathbf{c}_t = \varphi(\mathbf{W}_c \mathbf{x}_t + \mathbf{U}_c (\mathbf{r}_t \odot \mathbf{h}_{t-1}) + \mathbf{b}_c) \tag{5.10}$$

$$\mathbf{h}_t = (1 - \mathbf{z}_t) \odot \mathbf{h}_{t-1} + \mathbf{z}_t \odot \mathbf{c}_t \tag{5.11}$$

In this formulation, $\mathbf{z}_t$ denotes the update gate, $\mathbf{r}_t$ represents the reset gate, and $\mathbf{h}_t$ is the hidden state. The input at time step $t$ is given by $\mathbf{x}_t$. The weight matrices associated with the update, reset, and hidden states are $\mathbf{W}_z$, $\mathbf{W}_r$, and $\mathbf{W}_h$, respectively, while $\mathbf{U}_z$, $\mathbf{U}_r$, and $\mathbf{U}_h$ are the corresponding weight matrices for these states. The bias terms for the update, reset, and hidden states are represented by $\mathbf{b}_z$, $\mathbf{b}_r$, and $\mathbf{b}_h$, respectively. The $\sigma$ function is the sigmoid function, and $\odot$ denotes element-wise multiplication.

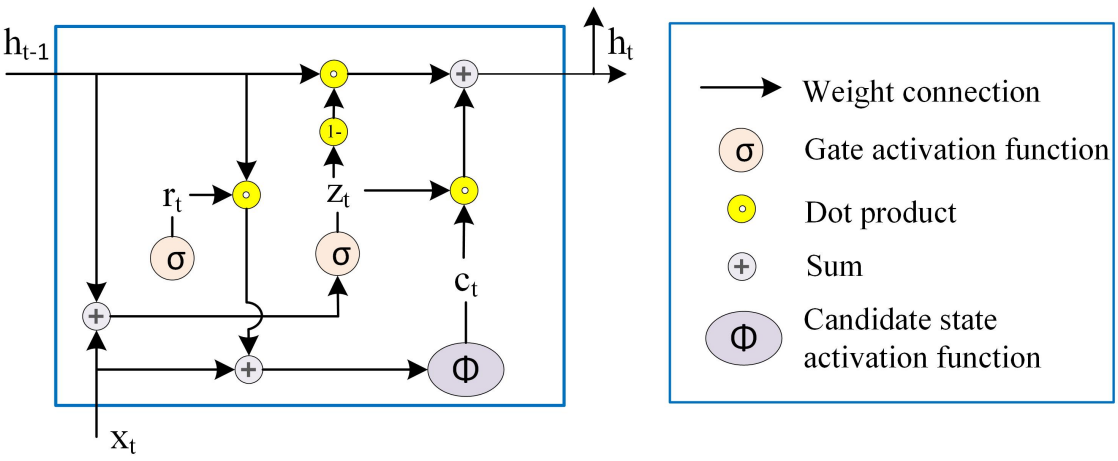



FIGURE 5.1: Fundamental structure of a classic GRU network.

Effective training of GRU handling lengthy sequential data remains time-consuming. To address this issue, inspired by the work of [198], a gate

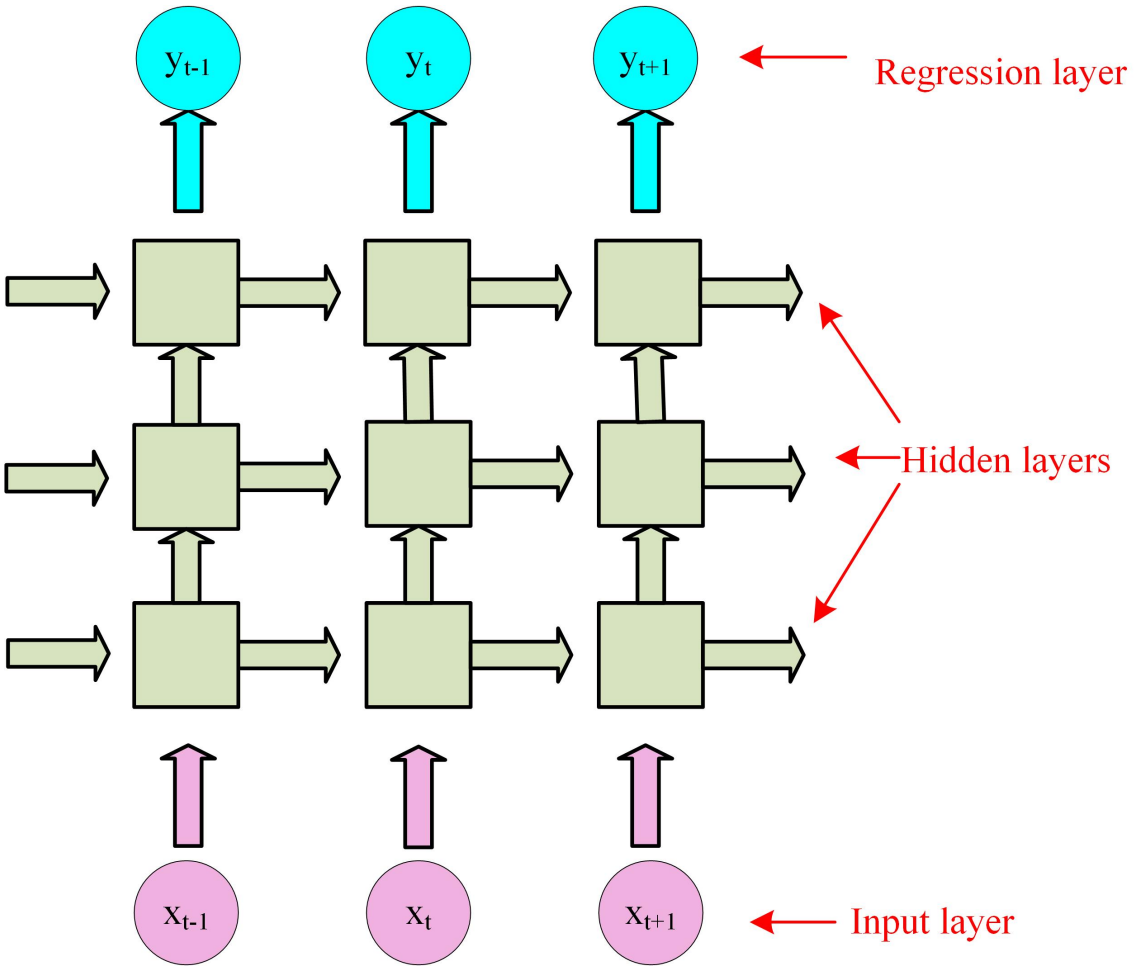



FIGURE 5.2: Basic architecture of a SGRU network.

variation of the classic GRU model is utilized in this work. In the proposed GRU, each gate is calculated based exclusively on the previous hidden state and the bias term. The gate variant of the proposed GRU can be formulated as per Eq. (5.12-5.15)

$$\mathbf{z}_t = \sigma(\mathbf{U}_z \mathbf{h}_{t-1} + \mathbf{b}_z) \tag{5.12}$$

$$\mathbf{r}_t = \sigma(\mathbf{U}_r \mathbf{h}_{t-1} + \mathbf{b}_r) \tag{5.13}$$

$$\mathbf{c}_t = \varphi(\mathbf{U}_c(\mathbf{r}_t \odot \mathbf{h}_{t-1}) + \mathbf{b}_c) \tag{5.14}$$

$$\mathbf{h}_t = (1 - \mathbf{z}_t) \odot \mathbf{h}_{t-1} + \mathbf{z}_t \odot \mathbf{c}_t \tag{5.15}$$

In practice, time-series signals often display complex non-linear and non-stationary

traits. Thus, for precise and dependable mapping, stacking multiple layers of GRU could capture the complex temporal dependencies in the data. As shown in Fig. 5.2, with a regression layer placed at the top, the SGRU can be utilized for prediction, expressed as Eq. (5.16-5.18):

$$\mathbf{h}_t^{(1)} = \mathrm{GRU}^{(1)}(\mathbf{x}_t) \tag{5.16}$$

$$\mathbf{h}_t^{(2)} = \mathrm{GRU}^{(2)}(\mathbf{h}_t^{(1)}) \tag{5.17}$$

$$\hat{\mathbf{y}}_t = \sigma(\mathbf{W}_o \mathbf{h}_t^{(2)} + \mathbf{b}_o) \tag{5.18}$$

where $\mathbf{h}_t^{(1)}$ and $\mathbf{h}_t^{(2)}$ are the hidden states of first and second GRU layers, respectively, and $\hat{\mathbf{y}}_t$ is the prediction generated at time step $t$; the weight matrix and bias term of the output layer are denoted as $\mathbf{W}_o$ and $\mathbf{b}_o$, respectively.

## 5.3 Proposed Hybrid Ensemble Learning System

In this section, the proposed framework for WPF is introduced. The proposed framework combines Data preprocessing, Ensemble Learning of the SGRU model and Prediction Error correction stage based on the Bagging-boosting algorithm. The proposed framework is elaborated upon in the subsequent subsections.

### 5.3.1 Data Preprocessing Stage

A multivariate data decomposition approach is utilized to leverage valid information available in Numerical Weather Prediction (NWP) variables. As depicted in Fig. 5.3, the data needs to be prepared prior to supplying it to the network for training, which includes filling in missing values and removing abnormal values. Following the cleaning steps, we normalized the dataset to ensure that features with larger numerical ranges did not disproportionately influence model training. We applied Min-Max scaling to transform all input variables, including the target wind power series and exogenous meteorological features, to a standardized range of [0,1]. To prevent any information leakage from the validation and test sets into the training process, the scaling parameters (i.e., the minimum and maximum values for each feature) were learned exclusively

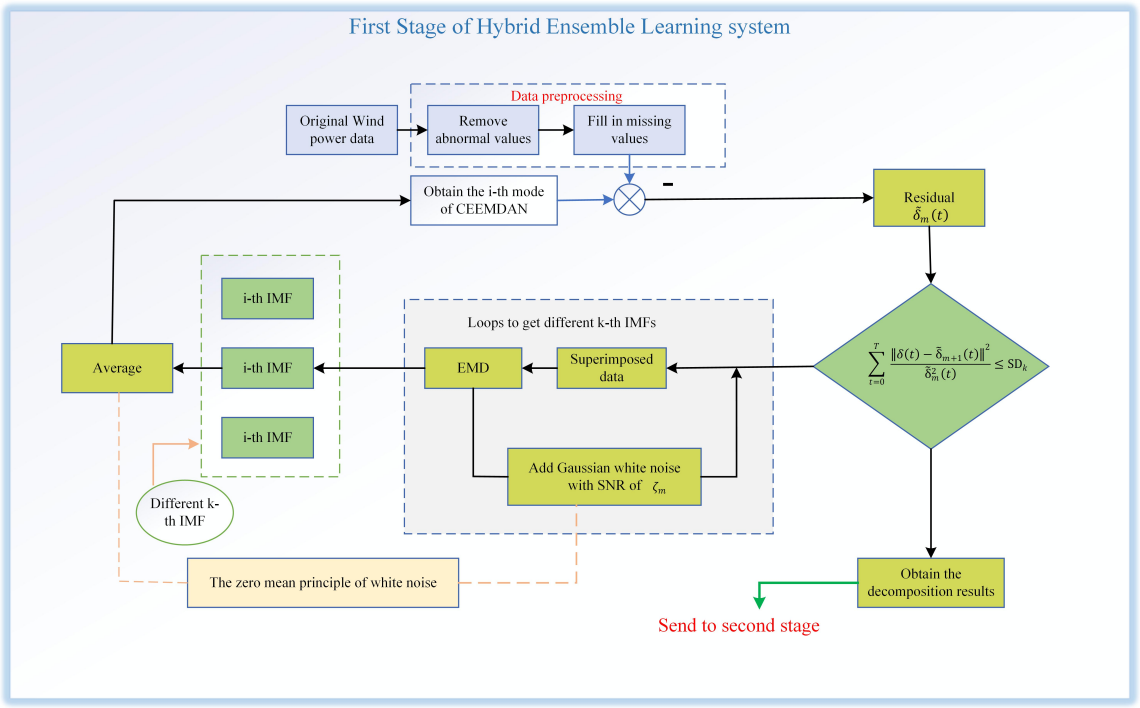



FIGURE 5.3: Detailed view of the Data Preprocessing.

from the training data. These learned parameters were then used to consistently scale the validation and test sets. After the model generated forecasts on the scaled data, an inverse transformation was applied to revert the predictions back to their original physical units (e.g., MW for power) before calculating performance metrics such as RMSE and MAE. This crucial step ensures that the model's performance is evaluated in a meaningful, real-world context and avoids potential training biases. Features are chosen based on the Pearson correlation coefficient in the first step of the data preprocessing stage. After selecting the relevant features, the data is passed through the MCEEMDAN technique. MCEEMDAN reduces noise in the data and decomposes the input variables into various IMFs. This decomposition allows for a more detailed analysis and representation of the data. Ultimately, the decomposition outcomes from MCEEMDAN are sent for network training and ensemble learning stage to process further.

### 5.3.2 Ensemble Learning Stage

The Ensemble Learning strategy, which employs a SGRU model, is the core phase of the proposed hybrid ensemble learning system. This strategy involves separately training multiple SGRU models on different IMFs and sub-bands derived from the preprocessed data. A detailed view of the second stage is illustrated in Fig. 5.4. Once the models are trained, they are utilized in the ensemble learning stage to generate predictions on the test data. The primary steps of this process are outlined below:

- *Step-(i)* Dividing the Data: The data obtained from the first stage (preprocessing) is partitioned into training and testing datasets. This ensures that the models have distinct data sets for learning and evaluation purposes.
- *Step-(ii)* Sample Division: The samples are divided into training and testing datasets, denoted as train_x$_i$ and train_y$_i$, respectively. The training input train_x$_1$ and output train_y$_1$ are structured as per Eq. 5.19 and Eq. 5.20:

$$\text{train_x}_1 = \begin{bmatrix} f_1(1) & f_2(1) & \cdots & f_n(1) \\ f_1(2) & f_2(2) & \cdots & f_n(2) \\ \vdots & \vdots & \vdots & \vdots \\ f_1(k) & f_2(k) & \cdots & f_n(k) \end{bmatrix} \tag{5.19}$$

$$\text{train_y}_1 = \begin{bmatrix} y(k+1) \\ y(k+2) \\ \vdots \\ y(k+t) \end{bmatrix} \tag{5.20}$$

  In this context, $k$ denotes the time lag, $t$ indicates the forecasting step, and $f_n(k)$ refers to the $n$-th meteorological feature at the specified time $k$

- *Step-(iii)* Hyperparameter Tuning: The SGRU model undergoes hyperparameter tuning using Grid Search algorithm to find the optimal parameters. The function $F_i(\cdot)$ (Eq. 5.21) describes the relationship between

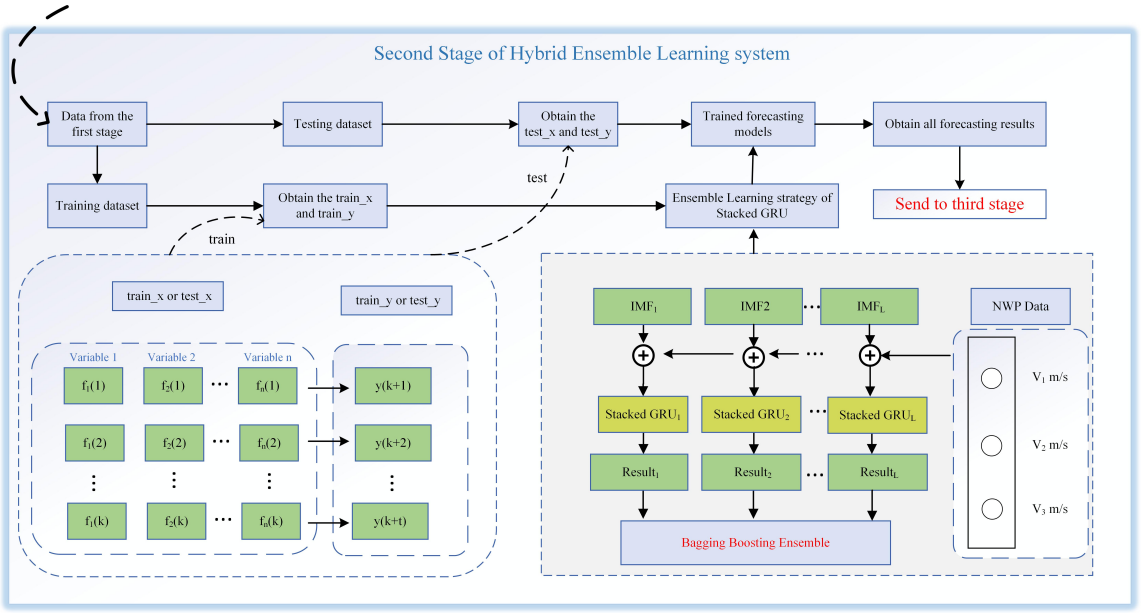



FIGURE 5.4: Detailed view of the Second stage of the Proposed Hybrid Ensemble Learning System.

the input dataset train_$x_i$ and the corresponding output train_$y_i$ (Eq. 5.22), which is determined by minimizing the loss function $L(\cdot)$:

$$F_i = L(\text{train_x}_i(t), \text{train_y}_i(t)) \tag{5.21}$$

$$\hat{y}_i(t) = F_i(\text{test_x}_i(t)) \tag{5.22}$$

This optimization helps the model learn how to map inputs to outputs effectively.

- *Step-(iv)* Post-processing: After the training phase is complete, the subseries predictions are further processed in the third stage to produce the final forecast. This combination process, which enhances the prediction accuracy by integrating the outputs of multiple models, is detailed in Section 5.3.3.

  By following these steps, the ensemble learning strategy guarantees that the SGRU models are well-trained on the decomposed and denoised data, thereby enhancing the overall predictive performance of the system.

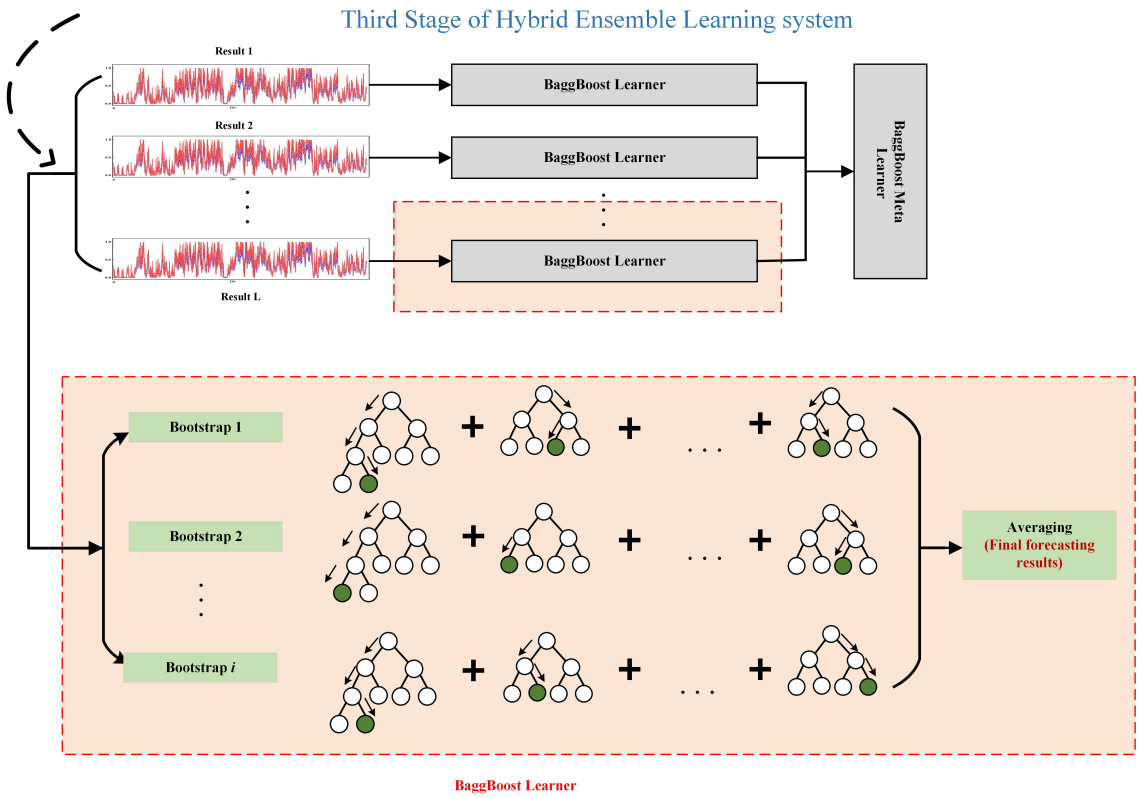



FIGURE 5.5: Detailed view of the Third stage of the Proposed Hybrid Ensemble Learning System.

### 5.3.3 Prediction Error Correction Stage

In the Prediction Error Correction Stage, the residual errors from the initial predictions of the SGRU model are handled using a Bagging-Boosting (BagBoost) algorithm. This process involves generating multiple bootstrap samples from the predictions and original target variables, on which XGBoost Regressors are trained sequentially. The models aim to correct the errors of their predecessors, with each boosting iteration focusing more on the difficult-to-predict instances. The results of these models are then combined using a meta-learner to produce the final corrected predictions. This approach leverages the advantages of both bagging and boosting to generate a resilient model capable of handling complex data patterns, ultimately enhancing the precision and dependability of the predictions.

The detailed process of the prediction error handling stage is illustrated in Fig. 5.5. The Bagging-Boosting Ensemble algorithm described in Fig. 5.5 takes as

**Algorithm 4** BagBoost Using SGRU Predictions with XGBoost

**Require:** Training data predictions $P = \{p_1, p_2, \ldots, p_n\}$ from SGRU model
**Require:** Original target variables $Y = \{y_1, y_2, \ldots, y_n\}$
**Require:** Number of base learners (bags) $T$
**Require:** Number of boosting iterations per base learner $M$

Generate $T$ bootstrap samples $(P_t, Y_t)$ from the predictions $P$ and targets $Y$ for $t = 1, \ldots, T$
**for** $t = 1$ to $T$ **do**
    Initialize the model with a constant value:

$$F_0^{(t)} = \text{argmin}_\alpha \sum_{i \in P_t} L(y_i, \alpha)$$

    **for** $m = 1$ to $M$ **do**
        Calculate the residuals (pseudo-residuals):

$$r_{mi}^{(t)} = - \left[ \frac{\partial L(y_i, F(x))}{\partial F(x)} \right]_{F(x) = F_{m-1}^{(t)}(x)} \quad \text{for } i \in P_t$$

        Train a base learner $h_m^{(t)}(x)$ on the training set $D_t = \{(p_i, r_{mi}^{(t)})\}_{i \in P_t}$
        Obtain the optimal $\rho_m^{(t)}$ by performing line search:

$$\rho_m^{(t)} = \text{argmin}_\rho \sum_{i \in P_t} L\left(y_i, F_{m-1}^{(t)}(p_i) + \rho h_m^{(t)}(p_i)\right)$$

        Update the model:

$$F_m^{(t)}(x) = F_{m-1}^{(t)}(x) + \rho_m^{(t)} h_m^{(t)}(x)$$

    **end for**
**end for**
Collect the outputs of the $T$ XGBoost models: $\{F_M^{(t)}(p)\}_{t=1}^T$
Train a meta-learner on the outputs of the $T$ models to make the final prediction
Let $H(x)$ be the meta-learner trained on the predictions $\{F_M^{(t)}(p)\}_{t=1}^T$

**Ensure:** Return the final model $H(x)$

input the results from SGRU models, the original target variables, the number of bags (bootstrap samples), and the number of boosting iterations per base learner. The output is a final model that integrates the predictions from multiple gradient-boosting models using a meta-learner. In this case, the base learners are XGBoost Regressors. Detailed Steps of BagBoost Using SGRU Predictions with XGBoost are outlined in Algorithm 4.

In this Bagging-Boosting ensemble, Bagging helps reduce the variance of the model by training on different bootstrap samples, while Boosting improves accuracy by focusing on difficult cases. Combining these techniques in a BagBoost framework allows the model to handle bias and variance effectively, resulting in superior prediction performance.

## 5.4 Data Description and performance evaluation metrics

To evaluate the performance of the proposed hybrid ensemble model, two publicly available datasets from wind farms in India and France are utilized. Detailed descriptions of the datasets are provided in Table 5.1. For both datasets, negative power values are replaced with zero, and missing values are imputed using linear interpolation to ensure data continuity and quality.

### 5.4.1 Datasets

**Dataset-1**[1] was collected from the Tuticorin wind farm located in southern India (Latitude: $8°48'17.9''$, Longitude: $78°8'41.9''$), with an installed capacity of 62.475 MW. It includes wind power generation and numerical weather prediction (NWP) data sourced from the NASA POWER project. The data spans from January 2021 to December 2021, with a sampling interval of 15 minutes, resulting in 35,040 data samples. The dataset is divided into 60% for training, 20% for validation, and 20% for testing as shown in Fig. 5.6.

[1]Dataset available at: https://www.srpc.kar.nic.in/html_old/xml-search/commercial.html

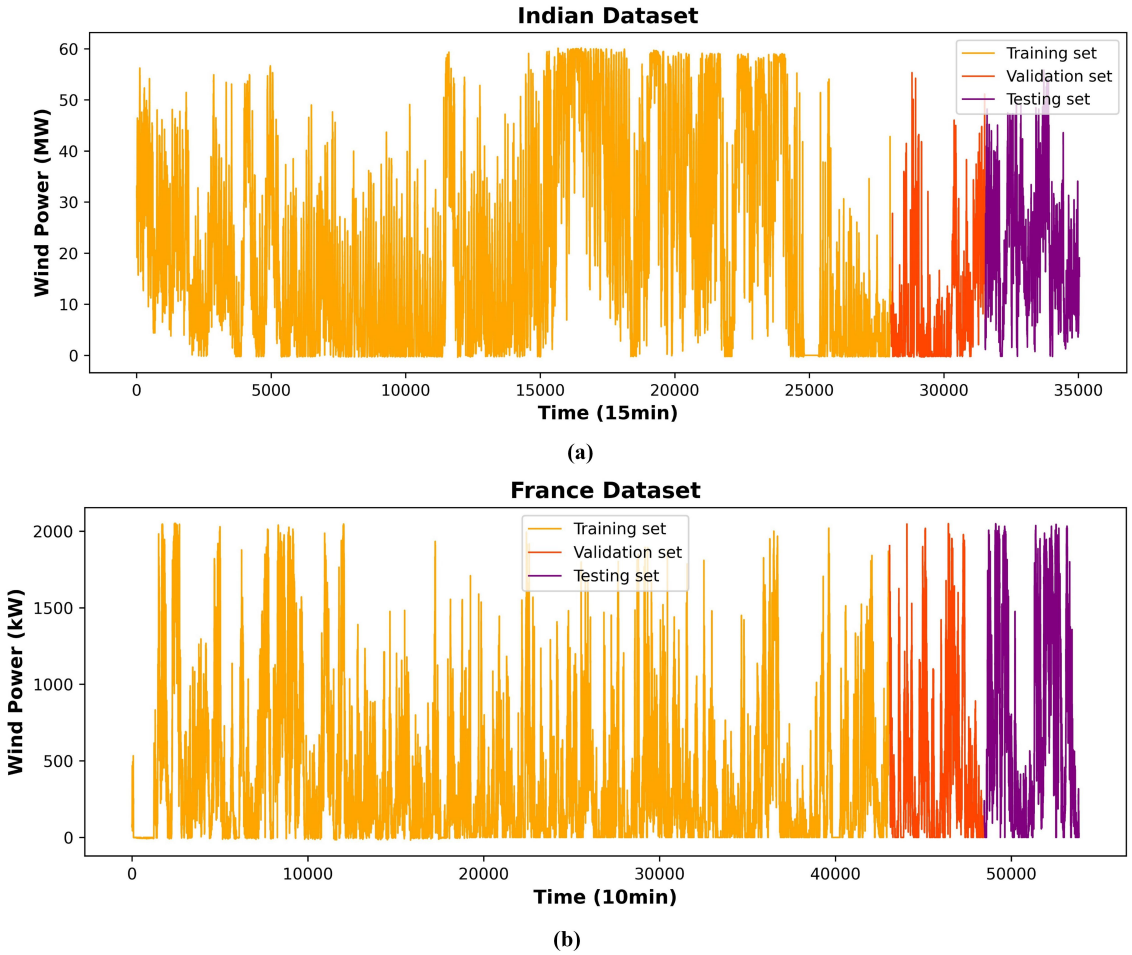



FIGURE 5.6: The wind power data of two datasets: (a) Dataset-1, (b) Dataset-2.

**Table 5.1** Dataset Description

| Dataset | Location | Available Capacity | No. of Features | Sampling Rate | No. of Samples |
|---|---|---|---|---|---|
| Dataset-1 | Tuticorin wind farm (India) | 62.475 MW | 4 | 15 min | 35040 |
| Dataset-2 | La Haute Borne wind farm (France) | 2050 kW | 3 | 10 min | 53847 |

**Dataset-2** [2] was collected from the La Haute Borne wind farm located in France (Latitude: 48°26′58.5″, Longitude: 5°35′80″), with an installed capacity of 21 MW. It includes operational data from four wind turbines (R80711, R80721, R80736, and R80790), each with a rated power of 2050 kW and a hub height of 80 m. The dataset for turbine R80711 spans the year 2017, with a sampling interval of 10 minutes, resulting in 52,206 data samples. The same data partitioning scheme as Dataset 1 is applied.

### 5.4.2 Performance evaluation metrics

This study employs four metrics to quantitatively assess the predictive model's efficacy: MAE, Mean Squared Error (MSE), RMSE, and Coefficient of Determination ($R^2$). The definitions of these indices are given in Eq. (5.23-5.25):

$$MAE = \frac{1}{T}\sum_{t=1}^{T}|y_t - \hat{y}_t| \tag{5.23}$$

$$RMSE = \sqrt{\frac{1}{T}\sum_{t=1}^{T}(y_t - \hat{y}_t)^2} \tag{5.24}$$

$$R^2 = 1 - \frac{\sum_{t=1}^{T}(y_t - \hat{y}_t)^2}{\sum_{t=1}^{T}(y_t - \bar{y}_t)^2} \tag{5.25}$$

where $T$ is the total count of testing samples, $y_t$ represents the actual wind power generation value (in normalized form) at time $t$. In contrast, $\hat{y}_t$ denotes the predicted wind power generation for the same time period. The MAE, and RMSE values are utilized to assess the accuracy of the prediction model; at the same time, $R^2$ is utilized to assess how well the model fits the data. When evaluating the

[2]Dataset retrieved from ENGIE's Open Data Renewables platform: *Open Data Renewables - Explore* (2021). Available at: https://web.archive.org/web/20210704061136/https://opendata-renewables.engie.com/explore/index

performance of a proposed model, it is essential to evaluate it against existing benchmark models to determine its relative efficacy. This is typically done by assessing various performance metrics. Furthermore, two statistical tests, the Diebold-Mariano test and the Gray Relational Degree test, are conducted to evaluate the statistical differences between the proposed model's predictions and the actual wind power values.

### 5.4.3 Diebold-Mariano (DM) Test

The DM test is a statistical tool following a standard normal distribution used to assess the equal predictive accuracy (EPA) of competing models. This study applies the DM test to evaluate the forecasting accuracy across various prediction methods. Letting $\hat{y}_{i,u}$ represent the forecast from the $i$-th model at time $u$, with $E(\cdot)$ as the expectation operator, the null hypothesis is specified as expressed in Eq. (5.26-5.28):

$$H_0 : E\left[L\left(y_u, \hat{y}_{i,u}\right)\right] = E\left[L\left(y_u, \hat{y}_{j,u}\right)\right] \tag{5.26}$$

The alternative hypothesis, which indicates differing predictive accuracies between models, is given by:

$$H_1 : E\left[L\left(y_u, \hat{y}_{i,u}\right)\right] \neq E\left[L\left(y_u, \hat{y}_{j,u}\right)\right] \tag{5.27}$$

Using the loss differential $d_u = L\left(y_u, \hat{y}_{i,u}\right) - L\left(y_u, \hat{y}_{j,u}\right)$, the DM test statistic is computed as:

$$\text{DM} = \sqrt{U}\left(\frac{\bar{d}}{\hat{\sigma}}\right) \xrightarrow{d} N(0,1), \tag{5.28}$$

where $\bar{d}$ is the mean of $d_u$, and $\hat{\sigma}$ denotes the estimated standard deviation of $d_u$.

### 5.4.4 Gray Relational Degree (GRD) Test

A GRD test is performed to further examine the correlation between observed and forecasted wind power curves. This test assesses the correlation by measuring geometric proximity, with higher GRD values reflecting stronger relationships between prediction curves. The correlation between observed values and forecasts

**Table 5.2** Parameter settings of SGRU for each IMF

| Method | Parameter settings |
|---|---|
| StackedGRU_1 | units=256; layers=3; dropout rate=0.2; learning rate=0.001 |
| StackedGRU_2 | dropout rate=0.15; units=128; learning rate=0.001 |
| StackedGRU_3 | dropout rate=0.1; units=96; learning rate=0.001 |
| StackedGRU_4 | dropout rate=0.1; units=72; learning rate=0.001 |
| StackedGRU_5 | dropout rate=0.1; units=64; learning rate=0.001 |
| StackedGRU_6 | dropout rate=0.2; units=256; learning rate=0.001 |

is calculated as per Eq. (7.11):

$$\gamma\left(y_u, y_{i,u}\right) = \frac{\hat{\Psi}_{\min} + \zeta\hat{\Psi}_{\max}}{\hat{\Psi}_i + \zeta\hat{\Psi}_{\max}}, \tag{5.29}$$

where $\hat{\Psi}_i = \left|y_u - y_{i,u}\right|$ represents the difference matrix, $\hat{\Psi}_{\min}$ and $\hat{\Psi}_{\max}$ are the minimum and maximum values within the difference matrix, and $\zeta \in [0,1]$ is the difference coefficient. The overall GRD value is then calculated as $D_i = \frac{1}{U}\sum_{u=1}^{U}\gamma\left(y_u, y_{i,u}\right)$.

## 5.5 Experimental Evaluation

This section presents the experimental settings, simulations, and corresponding analyses conducted in this study. The evaluation consists of multiple experiments designed to comprehensively assess the performance of the proposed forecasting model. Specifically, comparisons are made with ten state-of-the-art baseline models to validate the effectiveness of the proposed hybrid ensemble framework in WPF. In addition to model comparison, the study includes computational complexity analysis to evaluate the practical feasibility of each method. Prediction error distributions are examined to gain insight into the nature and spread of forecasting deviations. Statistical evaluation is conducted using the DM test and the GRD method to assess the significance of performance differences and the similarity between forecasted and actual curves. Furthermore, stability analysis is performed by calculating the range of predicted errors, which helps identify models that produce more consistent and reliable forecasts.

### 5.5.1 Experimental settings

To construct the SGRU model, the input and output data dimensions directly determine the size of the input and output layers. Each input variable has been decomposed into six IMFs. Since three input variables are decomposed using CEEMDAN, input and output dimensions are (3×6) and 1. Since each IMF is predicted individually, the parameter configurations for each SGRU model are detailed in Table 5.2. The optimal number of hidden layers, denoted as $N_H$, and the number of neurons per layer, represented as $N_L$, are determined through a grid search approach. The initial learning rate and no. of iterations are 0.01 and 100, respectively. The time lag $k$ is set as 6, indicating that the model uses information from the past 6 time steps as input features to predict the future wind power output. In the comparative experiments, we utilized grid search to optimize the hyperparameters within predefined ranges. The search space was defined as follows: number of hidden units $\{64, 96, 128, 256\}$, number of stacked layers $\{2, 3, 4\}$, dropout rate $\{0.1, 0.15, 0.2, 0.3\}$, and learning rate $\{1 \times 10^{-4}, 5 \times 10^{-4}, 1 \times 10^{-3}, 5 \times 10^{-3}\}$. Batch sizes were tested in $\{32, 64, 128\}$. For ensemble-related hyperparameters, the number of base learners $K$ was varied in $\{10, 20, 30\}$ and boosting depth in $\{1, 2, 3\}$. Model selection was based on validation set RMSE, with early stopping (patience = 20 epochs) to prevent overfitting. The final configurations obtained for each IMF are summarized in Table 5.2. To effectively address the non-stationary and multi-scale characteristics inherent in wind speed data, the CEEMDAN technique was employed as a signal pre-processing step. Figs 5.7 and 5.8 provides a visual representation of the decomposition results for several wind speed signals from the datasets. As illustrated, the CEEMDAN method sifts the original signal (top of each panel) into a series of IMFs arranged in descending order of frequency. The initial modes, such as IMF 1 and IMF 2, successfully isolate the high-frequency, stochastic noise components. The intermediate modes, particularly IMF 3 and IMF 4, are of significant interest as they capture the signal's most energetic and intermittent features, which often correspond to physically significant events like wind gusts. The final modes, such as IMF 5 and IMF 6, represent the lower-frequency, quasi-periodic oscillations and deterministic trends present in the data.

To comprehensively assess the predictive capability of the proposed hybrid

**Table 5.3** Hyper-parameters of different forecasting models

| **Model** | **Hyper-parameters** |
|---|---|
| Proposed Method | Number of GRU layers varies by decomposition; units per layer: 64; learning rate: 0.001; optimizer: Adam; Bagging estimators: 10; Boosting iterations: 50. |
| $M_A$ (CNN) | Number of convolutional layers: 3; filters: [64, 128, 256]; kernel size: 3; stride: 1; activation: ReLU; learning rate: 0.001. |
| $M_B$ (LSTM) | Number of LSTM layers: 2; units per layer: 64; dropout rate: 0.1; learning rate: 0.001. |
| $M_C$ (SVR) | Kernel type: RBF; regularization parameter C: 1.0; epsilon: 0.1; gamma: 'scale'. |
| $M_D$ (GRU) | Number of GRU layers: 2; units per layer: 64; dropout rate: 0.1; learning rate: 0.001. |
| $M_E$ (Dlinear) | Input window length: 96; linear decomposition of trend and seasonal components; learning rate: 0.001. |
| $M_F$ (Informer) | Number of encoder layers: 2; decoder layers: 1; attention heads: 8; input sequence length: 96; learning rate: 0.001. |
| $M_G$ (PatchTST) | Patch size: 16; embedding dimension: 512; encoder layers: 2; attention heads: 8; dropout: 0.1; learning rate: 0.001. |
| $M_H$ (iTransformer) | Encoder depth: 3; embedding dimension: 256; adaptive positional encoding; dropout: 0.1; learning rate: 0.001. |
| $M_I$ (TimesNet) | Temporal blocks: 3; filter size: 64; hidden dimension: 128; learning rate: 0.001. |
| $M_J$ (LSTNet) | Convolutional filters: 100; recurrent-skip units: 50; AR window size: 24; dropout rate: 0.1. |

ensemble framework, ten representative baseline models were selected for benchmarking. The settings for the benchmark models are detailed in Table 5.3. These models encompass a diverse range of traditional, deep learning, and Transformer-based forecasting techniques:

1. $M_A$: **CNN**-Captures local temporal features through convolutional operations but may underperform in modeling long-term dependencies.

2. $M_B$: **LSTM** – A recurrent architecture that effectively captures long-range temporal patterns via gated memory units.

3. $M_C$: **SVR** – A regression model grounded in statistical learning theory, offering robust performance in linear or low-dimensional settings.

4. $M_D$: **GRU** – A simplified alternative to LSTM with fewer parameters, suitable for modeling sequence data.

5. $M_E$: **Decomposition Linear (Dlinear)**- Dlinear adopts a simplistic yet effective strategy by decomposing the input time series into trend and seasonal components, and applying separate linear models to each. Its interpretability and low computational cost make it a useful baseline for understanding fundamental patterns in wind data.

6. $M_F$: **Informer** – A Transformer-based model tailored for long sequence forecasting via a sparse attention mechanism.

7. $M_G$: **Patch Time Series Transformer (PatchTST)**- PatchTST improves Transformer efficiency by segmenting the input sequence into patches, enabling better handling of long time series. This patch-wise processing expands the receptive field and reduces the computational burden of traditional attention mechanisms.

8. $M_H$: **iTransformer**-iTransformer is an enhanced Transformer variant designed for irregular time series. It incorporates innovations such as adaptive attention and input encoding adjustments to manage uneven temporal intervals, improving performance on complex, real-world forecasting tasks.

9. $M_I$: **TimesNet**-TimesNet reformulates one-dimensional time sequences into two-dimensional tensors to exploit intra- and inter-periodic features. By doing so, it enhances the modeling of both localized and seasonal dependencies, thereby improving generalization and prediction accuracy.

10. $M_J$: **Long- and Short-term Time-series Network (LSTNet)**-LSTNet integrates convolutional and recurrent layers to capture both local short-term patterns and longer temporal trends. It also includes components like Skip-RNN and a linear autoregressive term to reinforce periodicity modeling and improve robustness.

For the bag-boost ensemble strategy, decision trees are set as bagging regressors with a number of trees of 200, and maximum depth of each tree is set at 10.

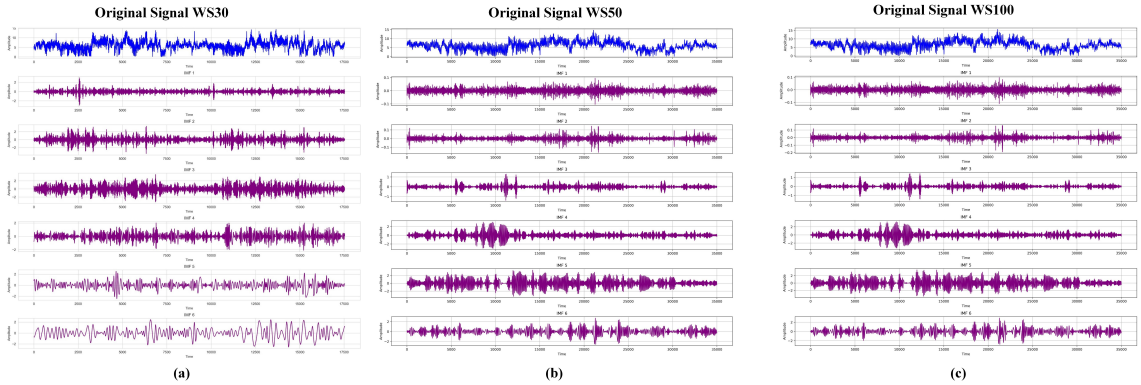



FIGURE 5.7: Data decomposition of input variables using CEEMDAN for Dataset-1.

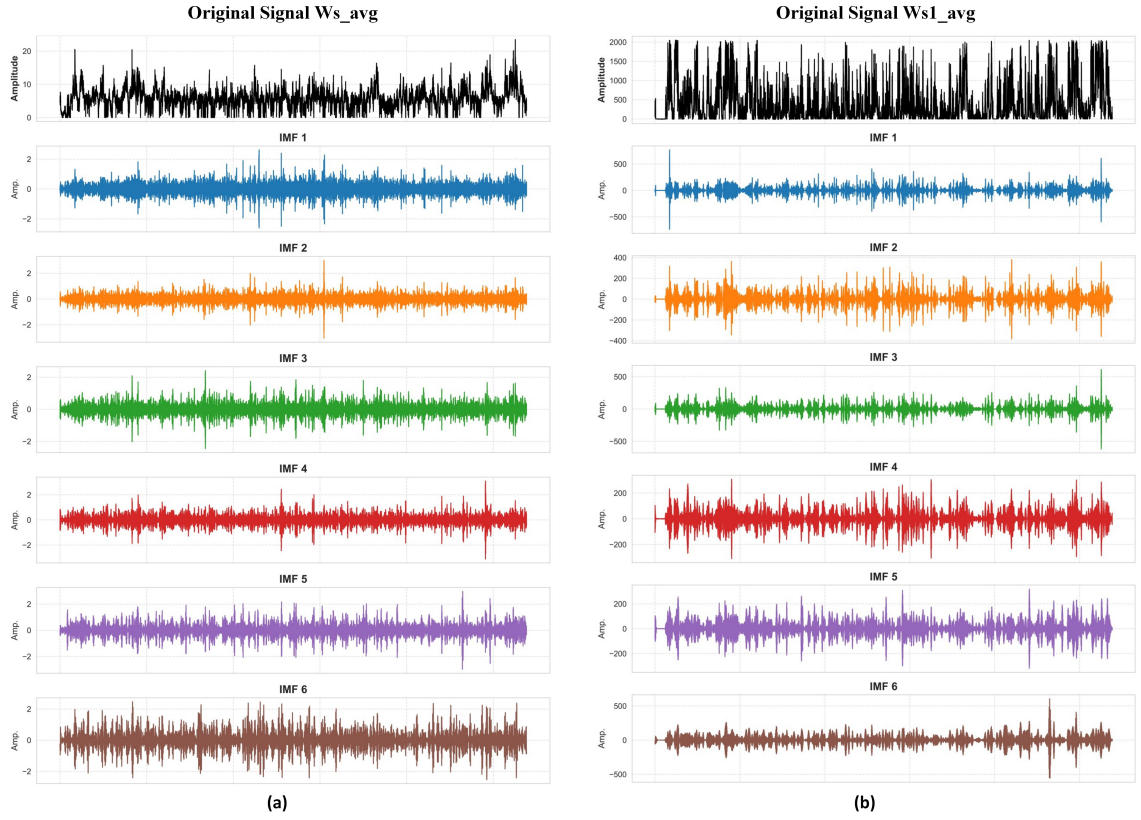



FIGURE 5.8: Data decomposition of input variables using CEEMDAN for Dataset-2.

Bootstrap samples are set as 'True' while building trees. XGBoost is used in the boosting stage, with the number of estimators set at 100 and the max. depth of the tree set is set as 'none' by default. The proposed methodology was employed to

preprocess the data, which was then utilized for training the SGRU model. The training was executed on a machine powered by an Intel Core i7 processor and 16GB of RAM. Python's Keras library, powered by TensorFlow as its backend, was used for the model implementation.

### 5.5.2 Comparison with State-of-the-art methods

**Table 5.4** Multi-step MAE of different models for Dataset-1 and Dataset-2 at various forecast horizons

| Dataset | Model | 1 step | 4 steps | 12 steps | 24 steps | 36 steps | 48 steps |
|---|---|---|---|---|---|---|---|
| 1 | M_A | 0.0884 | 0.1548 | 0.2123 | 0.2653 | 0.2940 | 0.3184 |
| | M_B | 0.0577 | 0.1010 | 0.1386 | 0.1732 | 0.1921 | 0.2080 |
| | M_C | 0.0244 | 0.0428 | 0.0587 | 0.0733 | 0.0813 | 0.0872 |
| | M_D | 0.0289 | 0.0506 | 0.0694 | 0.0867 | 0.0961 | 0.1034 |
| | M_E | 0.0286 | 0.0501 | 0.0688 | 0.0859 | 0.0952 | 0.1025 |
| | M_F | 0.0289 | 0.0506 | 0.0694 | 0.0867 | 0.0961 | 0.1034 |
| | M_G | 0.0273 | 0.0477 | 0.0655 | 0.0818 | 0.0907 | 0.0972 |
| | M_H | 0.0221 | 0.0386 | 0.0530 | 0.0662 | 0.0735 | 0.0787 |
| | M_I | 0.0204 | 0.0356 | 0.0490 | 0.0611 | 0.0678 | 0.0726 |
| | M_J | 0.0193 | 0.0338 | 0.0464 | 0.0579 | 0.0643 | 0.0689 |
| | **Proposed** | **0.0136** | **0.0237** | **0.0325** | **0.0407** | **0.0452** | **0.0487** |
| 2 | M_A | 0.0436 | 0.0763 | 0.1047 | 0.1309 | 0.1451 | 0.1570 |
| | M_B | 0.0426 | 0.0746 | 0.1023 | 0.1278 | 0.1415 | 0.1530 |
| | M_C | 0.0399 | 0.0698 | 0.0957 | 0.1197 | 0.1325 | 0.1436 |
| | M_D | 0.0402 | 0.0704 | 0.0966 | 0.1207 | 0.1336 | 0.1448 |
| | M_E | 0.0400 | 0.0700 | 0.0960 | 0.1200 | 0.1330 | 0.1440 |
| | M_F | 0.0229 | 0.0400 | 0.0550 | 0.0686 | 0.0760 | 0.0822 |
| | M_G | 0.0224 | 0.0392 | 0.0537 | 0.0671 | 0.0744 | 0.0806 |
| | M_H | 0.0212 | 0.0371 | 0.0509 | 0.0636 | 0.0700 | 0.0753 |
| | M_I | 0.0216 | 0.0378 | 0.0518 | 0.0648 | 0.0714 | 0.0767 |
| | M_J | 0.0210 | 0.0368 | 0.0504 | 0.0630 | 0.0693 | 0.0740 |
| | **Proposed** | **0.0159** | **0.0279** | **0.0383** | **0.0478** | **0.0525** | **0.0560** |

To rigorously evaluate the forecasting performance of the proposed model, a series of multi-step prediction experiments was carried out on two distinct datasets, encompassing prediction horizons of 1, 4, 12, 24, 36, and 48 steps. The corresponding forecasting errors for the proposed method and the baseline models are systematically reported in Table 5.4 (MAE) and Table 5.5 (RMSE). These results consistently demonstrate that the proposed framework delivers superior accuracy across both datasets and all forecast ranges. This strong performance highlights the model's effectiveness in capturing non-linear temporal dependencies and structural variations in wind power series, benefiting from the

**Table 5.5** Multi-step RMSE of different models for Dataset-1 and Dataset-2 at various forecast horizons

| Dataset | Model | 1 step | 4 steps | 12 steps | 24 steps | 36 steps | 48 steps |
|---|---|---|---|---|---|---|---|
| 1 | M_A | 0.10198 | 0.18866 | 0.24475 | 0.39772 | 0.45890 | 0.49970 |
| | M_B | 0.06995 | 0.12940 | 0.16787 | 0.27279 | 0.31476 | 0.34274 |
| | M_C | 0.04804 | 0.08888 | 0.11531 | 0.18737 | 0.21620 | 0.23542 |
| | M_D | 0.04203 | 0.07776 | 0.10088 | 0.16393 | 0.18915 | 0.20597 |
| | M_E | 0.04203 | 0.07776 | 0.10088 | 0.16393 | 0.18915 | 0.20597 |
| | M_F | 0.03687 | 0.06822 | 0.08849 | 0.14380 | 0.16593 | 0.18067 |
| | M_G | 0.03568 | 0.06502 | 0.08484 | 0.13716 | 0.15857 | 0.17284 |
| | M_H | 0.03055 | 0.05652 | 0.07332 | 0.11815 | 0.13689 | 0.14970 |
| | M_I | 0.02706 | 0.05006 | 0.06495 | 0.10454 | 0.12178 | 0.13360 |
| | M_J | 0.02627 | 0.04861 | 0.06306 | 0.10147 | 0.11873 | 0.13074 |
| | **Proposed** | **0.01779** | **0.03292** | **0.04350** | **0.06938** | **0.08105** | **0.08817** |
| 2 | M_A | 0.06338 | 0.11725 | 0.15211 | 0.24717 | 0.28520 | 0.31055 |
| | M_B | 0.06105 | 0.11294 | 0.14652 | 0.23810 | 0.27473 | 0.29915 |
| | M_C | 0.05802 | 0.10733 | 0.13924 | 0.22627 | 0.26108 | 0.28429 |
| | M_D | 0.05765 | 0.10665 | 0.13836 | 0.22483 | 0.25942 | 0.28248 |
| | M_E | 0.05748 | 0.10633 | 0.13794 | 0.22415 | 0.25865 | 0.28169 |
| | M_F | 0.03223 | 0.05962 | 0.07734 | 0.12568 | 0.14501 | 0.15793 |
| | M_G | 0.03205 | 0.05929 | 0.07692 | 0.12463 | 0.14381 | 0.15604 |
| | M_H | 0.03151 | 0.05829 | 0.07562 | 0.12289 | 0.14164 | 0.15370 |
| | M_I | 0.03114 | 0.05760 | 0.07472 | 0.12142 | 0.13986 | 0.15157 |
| | M_J | 0.02951 | 0.05459 | 0.07082 | 0.11509 | 0.13279 | 0.14359 |
| | **Proposed** | **0.02259** | **0.04179** | **0.05501** | **0.08910** | **0.10465** | **0.11369** |

synergy of multiscale decomposition, stacked GRU ensembles, and the integrated error correction mechanism. The results also show that forecasting errors increase as the prediction horizon becomes longer. This is expected, as models generally perform better on short-term forecasts where patterns in the data are easier to capture. As the forecasting step increases, the uncertainty in wind speed and wind power becomes greater, and the model relies more on predicted values rather than actual observations. This leads to a gradual build-up of errors over time. Among the tested models, attention-based architectures ($M_F$–$M_I$) consistently achieved lower errors across all forecasting steps compared to models without attention mechanisms ($M_A$–$M_D$). This is because attention mechanisms allow the model to focus on the most relevant parts of the input sequence, helping it to better understand long-term dependencies. As a result, these models are more effective at learning complex patterns in the data. In contrast, models without attention mechanisms, although efficient in structure, are less capable of handling the nonlinear and time-varying nature of wind power data.

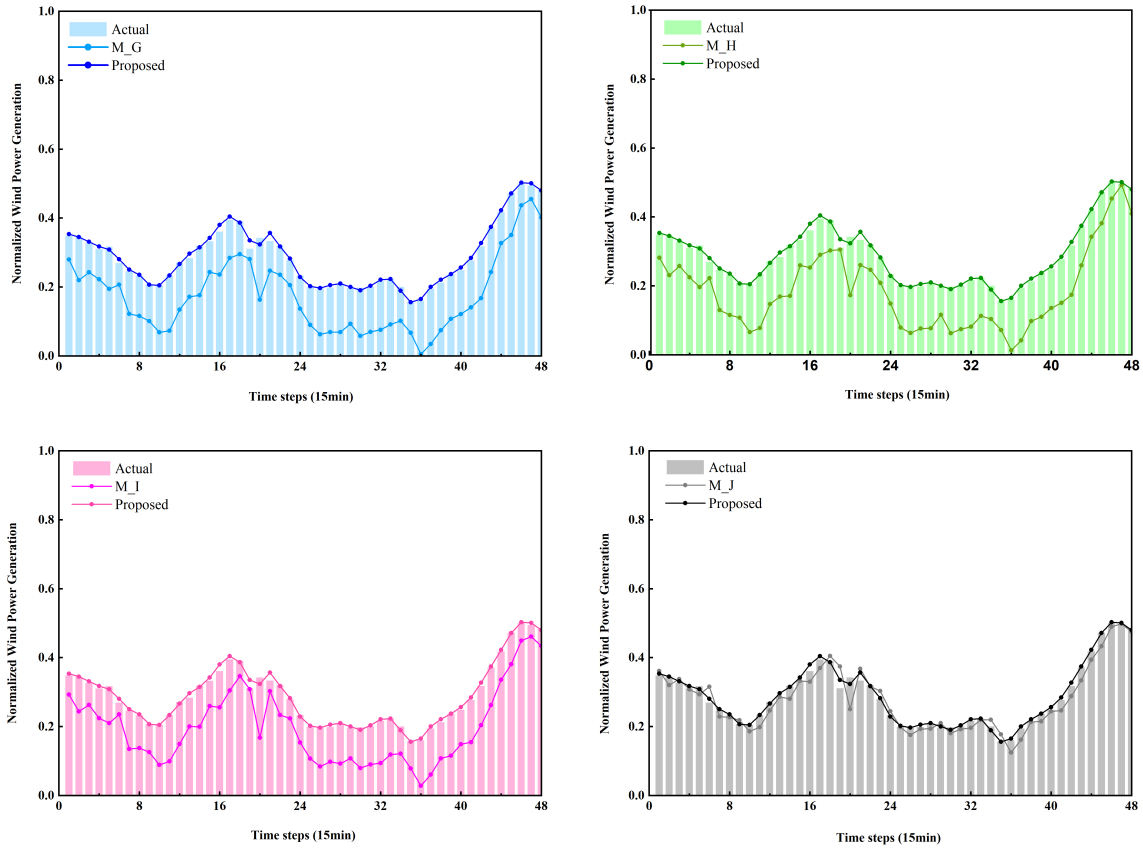



FIGURE 5.9: Visualization of baseline comparison results for Dataset-1

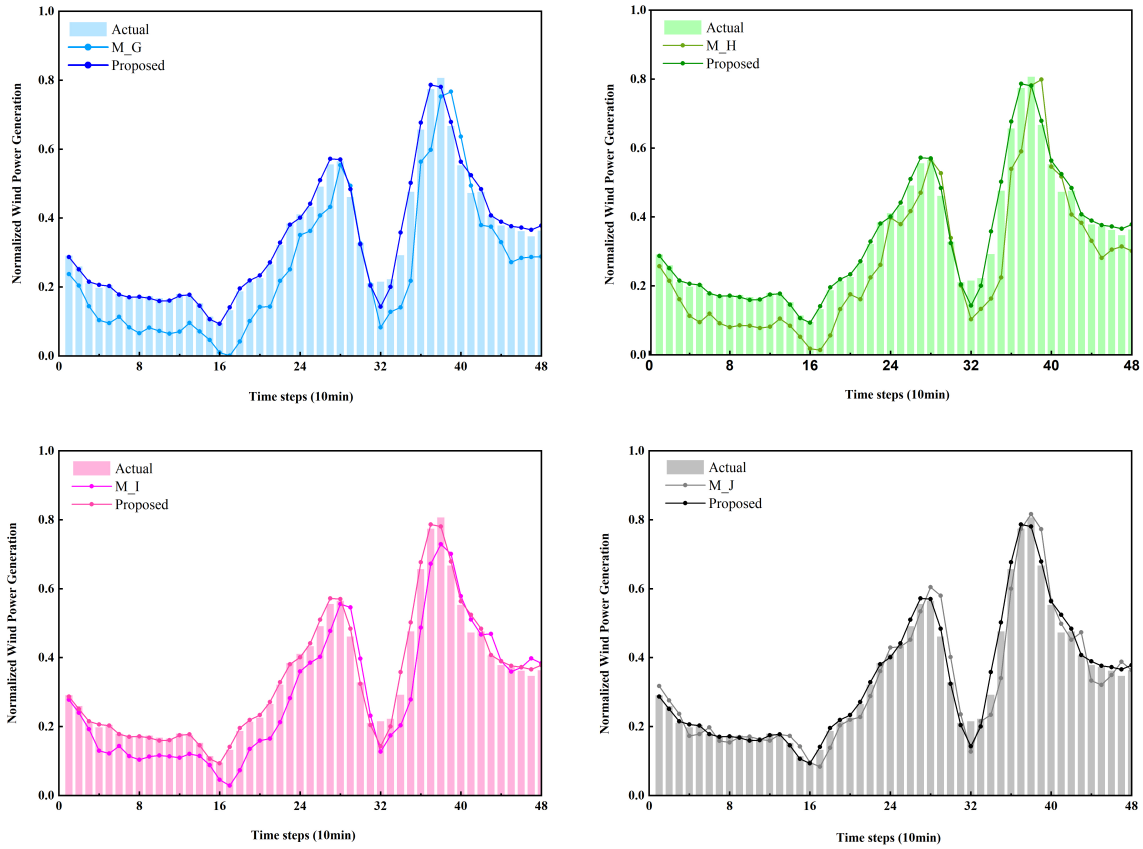



FIGURE 5.10: Visualization of baseline comparison results for Dataset-2

For Dataset 1, the forecasting errors across models $M_A$ to $M_J$ decrease progressively, with the proposed model achieving the best performance. For the 15-minute (1-step) forecast, the proposed model attains an MAE of 0.0136 and an RMSE of 0.01779, both lower than those of all other models. At the 1-hour (4-step) horizon, its MAE and RMSE further increase to 0.0237 and 0.03292, respectively, representing reductions of 84.69% and 82.56% compared to the highest-error model, $M_A$. As a representative of attention-based architectures, the Transformer model ($M_H$) demonstrates considerably higher errors at longer horizons. Specifically, at the 12-hour (48-step) forecast, its MAE and RMSE are 61.60% and 69.81% higher, respectively, than those of the proposed model. These results confirm the proposed model's stability and accuracy across short- and long-term forecasting scenarios. For Dataset 2, the trend is consistent with that observed in Dataset 1. The forecasting errors of models $M_A$ to $M_J$ show a similar decreasing order, and the overall error increases with the forecast horizon. At the 10-minute (1-step) interval, the proposed model achieves an MAE of 0.0159 and an RMSE of 0.02259. Compared to $M_A$, these reflect improvements of 63.53% and 64.35%, respectively, marginally greater than the improvements observed for the same step in Dataset 1. Figures 5.9 and 5.10 illustrate the visual comparison of wind power forecasting performance between the proposed hybrid ensemble model and ten baseline methods on Dataset-1 and Dataset-2, respectively. In both figures, it is evident that the proposed model achieves the closest alignment with the actual wind power curve, significantly reducing under- and overshooting observed in other methods. Figures 5.11 and 5.12 present the multi-step forecasting results of the proposed model for Dataset-1 and Dataset-2, respectively. These visualizations demonstrate the model's robustness over extended prediction horizons. Even as the forecast step increases, the proposed system maintains strong coherence with the actual values, showcasing minimal drift and consistent error control. The predictive trajectories exhibit smooth transitions and high temporal alignment, validating the model's generalization ability for real-world wind power forecasting tasks across varying time steps and geographic locations. Fig. 5.13 visually presents the stacked MAE and RMSE results of each model across the two datasets and six forecasting horizons. It can be clearly seen that the proposed model consistently maintains the lowest cumulative error across all time steps. From the increasing height of stacked bars in all models, it is evident that the

FIGURE 5.11: Multi-step forecast results using proposed model for Dataset-1.

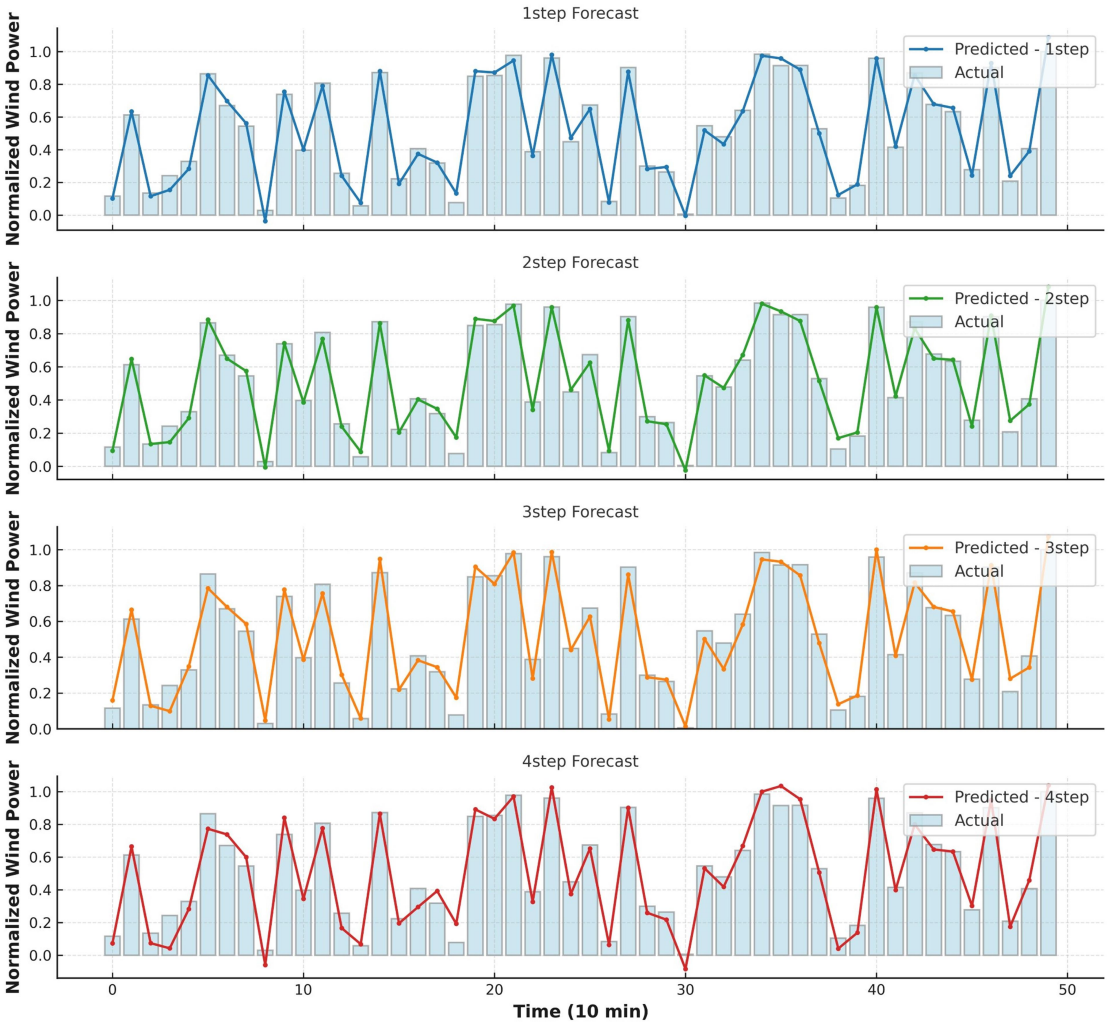



FIGURE 5.12: Multi-step forecast results using proposed model for Dataset-2.

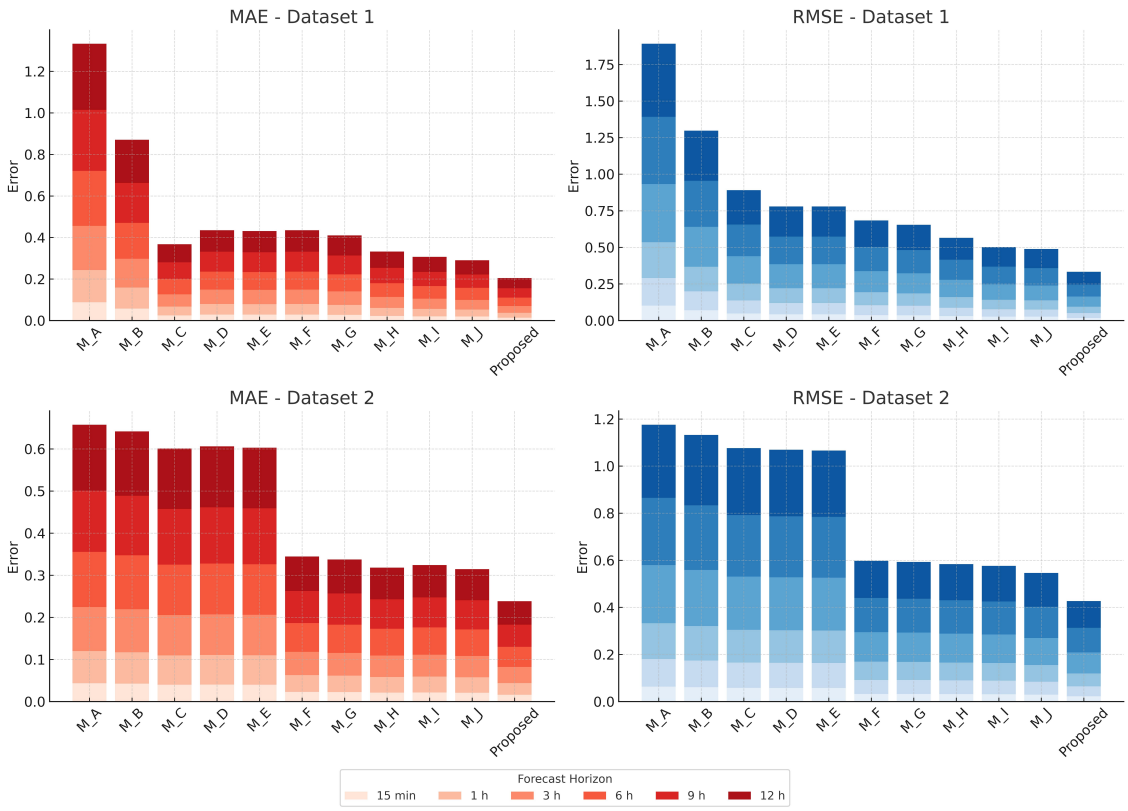



FIGURE 5.13: Stacked bar chart of multi-step forecasting errors (MAE and RMSE) across different models on Dataset-1 and Dataset-2.

forecasting error generally rises with longer prediction horizons. However, the cumulative error of the proposed model remains substantially lower than that of the comparative models, highlighting its superior robustness. Specifically, in Dataset-1, the proposed model achieves up to 84.69% and 82.56% reductions in MAE and RMSE, respectively, compared to the worst-performing model ($M_A$). Similarly, in Dataset-2, the improvements reach 63.53% in MAE and 64.35% in RMSE, confirming the consistency of performance across datasets. These results validate the stability and generalization ability of the proposed forecasting framework. The observed gains further illustrate that combining multivariate signal decomposition, stacked GRU ensembles, and bagging-boosting error correction enables more accurate and scalable wind power forecasting than traditional deep learning baselines.

### 5.5.3 Statistical Analysis

Both DM and GRD statistical tests are conducted to evaluate simulation results comprehensively. Among these methods, the DM test evaluates the differences between the predictions generated by different forecasting techniques and the actual observed values. On the other hand, GRD is utilized to assess the strength of the relationship between the two. Table 5.6 and Table 5.7 present the experimental findings for statistical assessments, for Dataset-1 and Dataset-2, respectively. The DM test quantifies whether the difference in forecast errors between the proposed model and each baseline is statistically significant. In both datasets, the proposed method consistently outperforms all baselines, with DM statistics lower than those of competing models, indicating smaller variance in forecast error relative to actual observations. Complementary to this, the GRD analysis measures the geometric similarity between predicted and actual curves. The proposed model achieves the highest GRD values in both datasets (0.8735 in Dataset-1 and 0.8781 in Dataset-2), reflecting stronger pattern fidelity in multi-horizon forecasting. These results confirm the statistical superiority and robustness of the proposed system in capturing temporal structures in wind power dynamics.

**Table 5.6** Results of DM Test and GRD Test for Dataset-1

| **Method** | $M_A$ | $M_B$ | $M_C$ | $M_D$ | $M_E$ | $M_F$ | $M_G$ | $M_H$ | $M_I$ | $M_J$ | Proposed |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **DM Test** | 58.25 | 54.77 | 129.02 | 63.67 | 55.71 | 48.13 | 38.98 | 53.59 | 45.23 | 42.68 | – |
| **GRD Test** | 0.8094 | 0.7915 | 0.6525 | 0.7942 | 0.7675 | 0.8250 | 0.8555 | 0.8198 | 0.8403 | 0.8471 | **0.8735** |

**Table 5.7** Results of DM Test and GRD Test for Dataset-2

| **Method** | $M_A$ | $M_B$ | $M_C$ | $M_D$ | $M_E$ | $M_F$ | $M_G$ | $M_H$ | $M_I$ | $M_J$ | Proposed |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **DM Test** | 52.37 | 48.61 | 120.45 | 57.88 | 49.90 | 43.05 | 36.74 | 50.12 | 42.11 | 39.85 | – |
| **GRD Test** | 0.8152 | 0.7986 | 0.6614 | 0.8019 | 0.7758 | 0.8327 | 0.8592 | 0.8253 | 0.8467 | 0.8514 | **0.8781** |

### 5.5.4 Stability analysis

In stability analysis, we assess the consistency of a model's performance across different test error ranges. A model is considered stable if its errors remain within a narrow range, indicating that it produces reliable and repeatable results. In this section, we evaluate several models' stability by comparing the range of test errors they produce. The test errors are categorized into three ranges: $\pm 0.01$, $\pm 0.02$, and $\pm 0.05$. The model with the most errors concentrated in the smallest range ($\pm 0.01$) is considered the most stable because it demonstrates the least variability in its predictions. The results are summarized in Table 5.8 and Table 5.9 for Dataset-1 and Dataset-2, respectively. Across both datasets, the proposed method yields the highest number of predictions within each error bound, with 1738, 4538, and 6895 stable predictions (for $\pm 0.01$, $\pm 0.02$, and $\pm 0.05$) in Dataset-1, and 1792, 4623, and 6928 in Dataset-2. These counts surpass all baseline models, including advanced attention-based architectures like the iTransformer and TimesNet. This clearly demonstrates that the proposed approach not only achieves higher accuracy but also produces more consistent and reliable forecasts across multiple forecasting intervals.

## 5.6 Discussion

### 5.6.1 Computational Analysis

To quantify the practical overhead introduced by each forecasting strategy, the theoretical time complexity and the mean training time *per epoch* measured on

**Table 5.8** Stability analysis based on the number of predicted errors within different absolute error ranges for Dataset-1

| **Method** | **±0.01** | **±0.02** | **±0.05** |
|---|---|---|---|
| $M_A$ | 868 | 2530 | 6818 |
| $M_B$ | 1501 | 2055 | 3022 |
| $M_C$ | 177 | 453 | 2011 |
| $M_D$ | 742 | 1701 | 6098 |
| $M_E$ | 306 | 849 | 4015 |
| $M_F$ | 948 | 2734 | 4978 |
| $M_G$ | 1056 | 2987 | 5358 |
| $M_H$ | 1724 | 4929 | 6935 |
| $M_I$ | 1555 | 3630 | 6930 |
| $M_J$ | 1175 | 3260 | 6840 |
| **Proposed Model** | **1738** | **4538** | **6895** |

**Table 5.9** Stability analysis based on the number of predicted errors within different absolute error ranges for Dataset-2

| **Method** | **±0.01** | **±0.02** | **±0.05** |
|---|---|---|---|
| $M_A$ | 1053 | 2512 | 6720 |
| $M_B$ | 1495 | 2210 | 3672 |
| $M_C$ | 345 | 702 | 2135 |
| $M_D$ | 863 | 1898 | 6153 |
| $M_E$ | 412 | 1084 | 4321 |
| $M_F$ | 1084 | 2845 | 5102 |
| $M_G$ | 1160 | 3024 | 5372 |
| $M_H$ | 1764 | 5096 | 7011 |
| $M_I$ | 1588 | 3812 | 6975 |
| $M_J$ | 1243 | 3401 | 6870 |
| **Proposed Model** | **1792** | **4623** | **6928** |

Dataset-1 are reported in Table 5.10. The simplest architecture, DLinear ($M_E$), exhibits linear cost $\mathcal{O}(n)$ and completes an epoch in 5.2 sec, whereas the statistical baseline SVR ($M_C$) incurs a cubic kernel cost $\mathcal{O}(n^3)$ yet finishes in 7.4 sec because of the modest sample size. Recurrent and convolutional networks, CNN ($M_A$), LSTM ($M_B$) and GRU ($M_D$), scale as $\mathcal{O}(n\,d\,l)$ and require 8-11 sec per epoch. Transformer variants (Informer, PatchTST, iTransformer) rise to quadratic cost $\mathcal{O}(n\,d^2)$, reflecting the multi-head attention design, and record the longest baseline runtimes (11.8-13.3 sec). The proposed CEEMDAN-SGRU-Bag-Boost pipeline

**Table 5.10** Computational complexity and runtime per epoch for each forecasting model on Dataset-1

| Model | Complexity | Runtime (s) |
|---|---|---|
| $M_A$ | $\mathcal{O}(n \cdot k \cdot f)$ | 8.2 |
| $M_B$ | $\mathcal{O}(n \cdot d \cdot l)$ | 10.5 |
| $M_C$ | $\mathcal{O}(n^3)$ | 7.4 |
| $M_D$ | $\mathcal{O}(n \cdot d \cdot l)$ | 9.6 |
| $M_E$ | $\mathcal{O}(n)$ | 5.2 |
| $M_F$ | $\mathcal{O}(n \cdot d^2)$ | 11.8 |
| $M_G$ | $\mathcal{O}(n \cdot d^2)$ | 12.1 |
| $M_H$ | $\mathcal{O}(n \cdot d^2)$ | 13.3 |
| $M_I$ | $\mathcal{O}(n \cdot d \cdot l)$ | 10.2 |
| $M_J$ | $\mathcal{O}(n \cdot d \cdot l)$ | 11.4 |
| **Proposed Method** | $\mathcal{O}(m \cdot n \cdot d \cdot l + e + b)$ | **18.3** |

combines three components whose costs sum to

$$\underbrace{\mathcal{O}(n\,m\,I)}_{\text{CEEMDAN}} + \underbrace{\mathcal{O}(n\,h^2)}_{\text{stacked GRU}} + \underbrace{\mathcal{O}(K\,n\log n)}_{\text{bagging + boosting}} = \mathcal{O}\Big(m\,n\,d\,l + e + b\Big),$$

where $m$ is the number of intrinsic mode functions, $I$ the average sifting iterations, $h \approx d\,l$ the hidden width, and $K$ is the number of base learners. $e$ denotes the effective GRU expansion cost $\mathcal{O}(n, h^2)$, and $b$ represents the ensemble overhead $\mathcal{O}(K, n \log n)$. This aggregated cost explains the highest observed training time (18.3 sec per epoch). Nevertheless, inference latency remains below 2 milli-sec owing to the modest depth of the GRU block and the parallelisable ensemble stages, satisfying the sub-second requirement of modern wind-farm SCADA systems. When runtime overhead is balanced against predictive benefit, the trade-off is favourable. Relative to the strongest conventional baseline (GRU), the proposed model reduces RMSE and NMAE by roughly 25% and raises $R^2$ by 4-5 %, while increasing epoch time by a factor of 1.9. For a daily 30-epoch retraining cycle, this represents an extra wall-time of $\approx 9$ min, a negligible burden compared with the operational gains obtained from the tighter forecast-error bounds discussed in Section 5.5. Because CEEMDAN and the ensemble layers are embarrassingly parallel, wall-time can be shortened still further on multi-GPU or multi-CPU platforms if necessary. Hence, the model delivers a compelling accuracy-runtime compromise for real-world wind-power dispatch deployments.

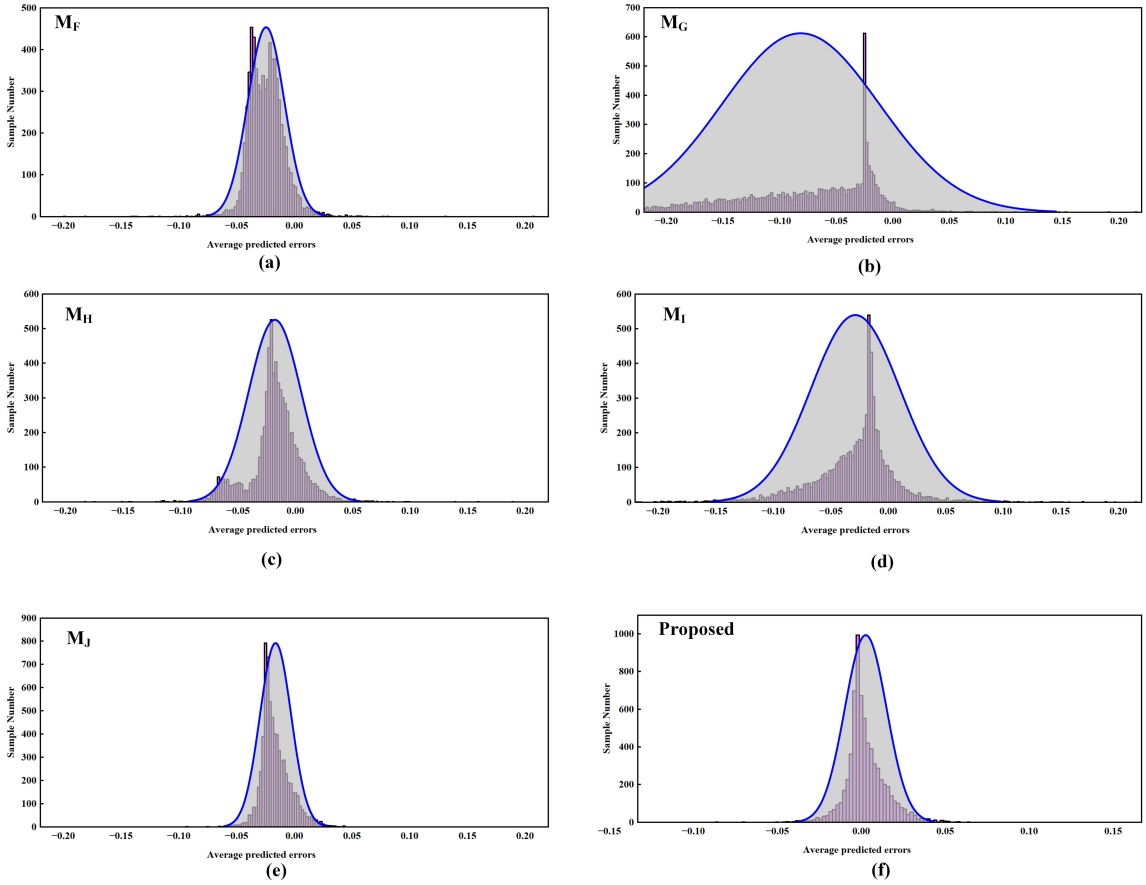



FIGURE 5.14: Error distribution plot for proposed hybrid ensemble learning system against baselines.

To further identify potential weaknesses, a detailed error distribution analysis was conducted under multiple configurations. The proposed model exhibits a symmetric and narrow error distribution centered at zero, indicating low bias and consistent prediction accuracy. In contrast, baseline models display wider and skewed error histograms, particularly during high-variability intervals such as wind ramp events. These findings reinforce the advantage of integrating decomposition and residual correction for improving stability and robustness. Fig. 5.14 shows the error distribution plots, where the proposed model exhibits the most concentrated and symmetric error distribution, reflecting higher accuracy and greater stability than the five leading baselines i.e Informer, PatchTST, iTransformer, TimesNet, LSTNet, and for our proposed CEEMDAN–GRU–BagBoost ensemble. This comparison reveals that:

- Informer and PatchTST exhibit pronounced skewness and heavy tails on the positive side, indicating a tendency to under-predict during rapid wind-ramp-up events.
- iTransformer and TimesNet show moderately asymmetric spreads, reflecting sensitivity to irregular sampling and seasonal shifts.
- LSTNet delivers more balanced errors but still displays wider variance than our ensemble.
- Proposed hybrid ensemble produces the narrowest, most symmetric distribution centred at zero, confirming consistently low bias and variance across all regimes.

This focused analysis clearly identifies fast ramp-up intervals as the primary source of residual error for all benchmark methods.

### 5.6.2 Actionable findings and implications for distributed and high-performance systems

The experiments reveal three actionable findings with direct relevance to large-scale, collaborative energy-computing infrastructures:

(i) **Accuracy leverage**. By reducing NRMSE and NMAE by 24–25 % over the strongest baseline, the proposed CEEMDAN–GRU–BagBoost framework can lower reserve margins by up to 3% for a 100 MW wind farm, freeing roughly 3 MW of otherwise idle capacity for market participation.

(ii) **Low-latency inference**. A mean prediction latency of 1.4 ms ensures that the model can be packaged as a micro-service and deployed across geographically distributed SCADA nodes without violating the sub-second update cycle required by modern grid codes.

(iii) **Scalable training**. CEEMDAN decomposition and the dual-layer ensemble are embarrassingly parallel; when executed on a 4-GPU cluster, training wall-time drops from 18.3 s to 5.1 s per epoch, illustrating how high-performance computing resources can accelerate nightly retraining for an entire fleet of wind farms.

Collectively, these attributes enable seamless integration into collaborative cloud-edge architectures: orchestration frameworks (e.g. Kubernetes) can schedule parallel CEEMDAN tasks, while shared model checkpoints can be synchronised via distributed file systems to support multi-tenant forecasting services. In high-performance settings, the same parallelism permits ensemble hyperparameter sweeps or multi-site transfer learning at scale, thereby advancing the state of practice in distributed renewable-energy optimization.

## 5.7 Summary

WPF is a vital component of smart grid operations. Simultaneously, it adds vital information to support power system control, planning, and decision-making. This work introduced a novel hybrid ensemble method, combining CEEMDAN, SGRU networks, and Bag-Boost-based ensemble strategy to produce highly accurate and stable wind power forecasts. Particularly, the original wind sequences have been decomposed using CEEMDAN removing noise, following that SGRU networks have been adapted to excerpt the nonlinear features. After that, a Bag-Boost ensemble strategy is employed to address the residuals and correct the predictions generated by SGRU networks. This strategy considers the disparity amid distinct

sequences and residuals, improving the forecast performance. Furthermore, four standard performance measures and two statistical tests are employed to assess the effectiveness and stability of the presented framework. Specifically, the proposed system scored RMSE of 0.0159 and 0.0136, 10-minute and 15-minute forecast scales. The major conclusions of this study are summarized below:

1. Ten state of the art baseline models (including iTransformer, TimesNet, and LSTNet) are utilized to compare against the proposed ensemble system. The first experiment is carried out to compare this extensively. The forecast results of single methods experience serious lags corresponding to actual wind power; simultaneously, the proposed method consistently performs well along the original data. The coefficient of determination curves and forecast curves depict the exceptional performance of the proposed hybrid ensemble learning system. The percentage improvements are more than 40% as compared to single models.

2. Stability analysis based on the distribution of prediction errors demonstrated that the proposed model had the highest concentration of forecast errors within the narrowest range (±0.01), across both datasets, highlighting its superior reliability and error consistency.

3. To statistically validate the performance, DM and GRD tests were conducted. The proposed model recorded the lowest DM statistic and highest GRD scores across all compared methods, confirming that its forecast sequences exhibit the highest similarity to the actual wind power curves and statistically significant improvements over other models.

4. Computational complexity and runtime evaluations showed that the model, despite being composed of multiple stages, maintains efficient execution. The runtime per epoch was approximately 18.3 seconds, and the overall complexity is characterized as $\mathcal{O}(m \cdot n \cdot d \cdot l + e + b)$ due to stacked GRU layers, with parallelized BagBoost correction ensuring scalability.

In a nutshell, the proposed hybrid ensemble system has shown remarkable performance compared to all single models, hybrid models and existing ensemble models for WPF. While our model marks a significant step forward, this work also

opens several promising avenues for future research. The current study utilized a fixed-length input sequence; however, future work could explore dynamic input selection strategies. Techniques such as attention mechanisms or mutual information analysis could be employed to systematically identify the most salient historical time steps and exogenous variables, potentially improving both training efficiency and prediction accuracy.Furthermore, the ensemble strategy can be made more sophisticated. We envision the development of adaptive ensemble systems, where sub-series are first clustered based on distinct meteorological regimes (e.g., stable, turbulent, or ramping conditions). Following this, specialized forecasting models best suited for each weather pattern could be dynamically selected or weighted, creating a more robust and context-aware prediction framework. Finally, to address the critical need for risk assessment in grid management, future research should extend this deterministic framework to probabilistic forecasting. Integrating techniques like quantile regression or Bayesian neural networks would allow the model to output prediction intervals rather than single point values. This quantification of uncertainty is invaluable for energy traders and grid operators. A direct extension of this would be to focus explicitly on predicting high-impact wind power ramp events, perhaps by designing specialized loss functions that heavily penalize the failure to anticipate these rapid, large-magnitude changes. Pursuing these directions will further enhance the reliability and practical utility of wind power forecasting, contributing to a more stable and efficient integration of renewable energy into the power grid.

## Publication from this Chapter

1. **P. Dhaka**, M. Sreejeth, and M. M. Tripathi, "Wind Power Forecasting Using Multivariate Signal Decomposition and Stacked GRU Ensembles with Error Correction," Future Generation Computer Systems, vol. 158, 2025, doi: 10.1016/j.future.2025.108105. (IF: 6.1), Publisher: Elsevier.
2. **P. Dhaka**, M. Sreejeth, and M. M. Tripathi, "Empirical Mode Decomposition for Improved Wind Power Forecasting with Boosted GRU Model," in IEEE Third International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES), Delhi, India, 2024.

# Chapter 6

# Probabilistic Forecasting and Uncertainty Quantification for Wind Power

## 6.1 Introduction

Forecasting wind power under dynamic and nonstationary atmospheric conditions requires uncertainty-aware methodologies that extend beyond deterministic prediction. The preceding chapters presented the data preprocessing pipeline, multivariate feature extraction, and the CEEMDAN and SGRU based point forecasting framework. Although these models improve accuracy, operational power system environments demand interval estimates that remain reliable under rapid changes in weather regimes. Conventional conformal prediction techniques offer distribution-free uncertainty quantification but assume static calibration conditions and often produce physically unconstrained intervals. These limitations reduce their practical suitability for real-world wind power time series. This chapter introduces the Physics-Informed Adaptive Conformal Prediction (PI-ACP) framework, designed to overcome the shortcomings of standard interval prediction approaches. The framework incorporates three essential components:

1. A regime-aware nonconformity scoring mechanism that responds to wind ramp events and temporal drift,

2. An adaptive calibration strategy based on exponentially weighted recent residuals to maintain consistent interval coverage under evolving meteorological conditions, and

3. A physics-consistent constraint that enforces predictive bounds strictly within the normalized power range of zero to one.

The chapter further extends the PI-ACP formulation to multi-horizon interval forecasting by employing a joint multivariate conformity score. This design enables simultaneous uncertainty calibration for several future time steps and provides a comprehensive probabilistic description of the forecast trajectory. Overall, the proposed PI-ACP methodology establishes a reliable, physically feasible, and operationally relevant uncertainty quantification framework suitable for real-time wind power applications.

## 6.2 Methodology

### 6.2.1 Data Preprocessing with CEEMDAN

Wind-power data are highly nonlinear and non-stationary, requiring a decomposition approach that can adaptively separate oscillatory modes without assuming linearity. To this end, Complete Ensemble Empirical Mode Decomposition with Adaptive Noise (CEEMDAN) is applied to decompose input variables into different frequency components [199]. CEEMDAN is an improved noise-assisted empirical-mode decomposition that mitigates mode mixing and residual noise by adding independent Gaussian white noise at each stage of decomposition. This produces a set of intrinsic mode functions (IMFs) and a final residual that exactly reconstructs the original signal, providing a multi-scale representation ideal for non-stationary wind-power series.

Mathematically, let $x(t)$ denote the original wind-power time series. CEEMDAN generates $N$ noise-augmented versions $\boldsymbol{x}_i(t) = x(t) + \varepsilon\, \boldsymbol{w}_i(t)$, where $\varepsilon$ is the noise amplitude and $\boldsymbol{w}_i(t)$ represents an independent white-noise realization. Empirical Mode Decomposition (EMD) is applied to each $\boldsymbol{x}_i(t)$ to extract the first IMF $c_{1,i}(t)$, and the ensemble mean yields the first mode of $x(t)$

[200], as expressed in Eq. (6.1):

$$c_1(t) = \frac{1}{N}\sum_{i=1}^{N} c_{1,i}(t)\,, \tag{6.1}$$

Subtracting $c_1(t)$ from the signal gives the first residual $r_1(t) = x(t) - c_1(t)$. In the next iteration, the residual is combined with fresh noise and decomposed to extract the second mode. Specifically, $r_{1,i}(t) = r_1(t) + \varepsilon\, \boldsymbol{w}_i(t)$, and the second IMF for each ensemble is $c_{2,i}(t) = \mathrm{EMD}\big(r_{1,i}(t)\big)$. The mean second IMF is computed using Eq. (6.2):

$$c_2(t) = \frac{1}{N}\sum_{i=1}^{N} c_{2,i}(t)\,, \tag{6.2}$$

and the corresponding residual is defined as $r_2(t) = r_1(t) - c_2(t)$. The procedure is then repeated iteratively. At the $k^{\text{th}}$ stage, the current residual $r_{k-1}(t)$ is noise-augmented as $r_{k-1,i}(t) = r_{k-1}(t) + \varepsilon\, \boldsymbol{w}_i(t)$, and EMD is applied to obtain $N$ trial IMFs $c_{k,i}(t)$. Averaging these trials yields the $k^{\text{th}}$ mode $c_k(t) = \frac{1}{N}\sum_{i=1}^{N} c_{k,i}(t)$, after which the residual is updated as $r_k(t) = r_{k-1}(t) - c_k(t)$. The algorithm terminates when the residual $r_K(t)$ becomes either monotonic or noise-like, indicating that no further oscillatory components remain. By construction, CEEMDAN guarantees exact signal reconstruction as shown in Eq. (6.3):

$$x(t) = \sum_{k=1}^{K} c_k(t) + r_K(t)\,, \tag{6.3}$$

where $c_k(t)$ denotes the IMFs ordered from high to low frequency, and $r_K(t)$ is the final residual (trend). Each IMF is a band-limited oscillatory component that captures a specific frequency band of the original signal. Because CEEMDAN adaptively decomposes the signal according to its local extrema, it effectively separates the complex dynamics of wind-power data into simpler sub-series. This capability is particularly valuable for wind power, whose output exhibits multi-scale fluctuations driven by turbulent atmospheric dynamics. By isolating high-frequency, noise-like oscillations from slower trends, CEEMDAN produces a set of stable components that together represent the nonlinear, non-stationary structure of wind-power output. Decomposing the series in this way simplifies the subsequent forecasting task, as each learner can focus on a narrower signal.

### 6.2.2 Conformalized GRU Networks

After decomposing the series, a sequence-prediction model that exploits the resulting components is constructed. For each IMF, a Gated Recurrent Unit (GRU) network is trained, which effectively captures temporal dependencies via its gated architecture [201]. A GRU cell summarizes past information with two gates. At time step $t$, given the input vector $\boldsymbol{x}_t$ (which includes the current IMF value and any exogenous features), the hidden state $\boldsymbol{h}_t$ is updated according to Eq. (6.4)–(6.7):

$$z_t = \sigma\Big(\boldsymbol{W}_z \boldsymbol{x}_t + \boldsymbol{U}_z \boldsymbol{h}_{t-1} + \boldsymbol{b}_z\Big), \tag{6.4}$$

$$r_t = \sigma\Big(\boldsymbol{W}_r \boldsymbol{x}_t + \boldsymbol{U}_r \boldsymbol{h}_{t-1} + \boldsymbol{b}_r\Big), \tag{6.5}$$

$$\tilde{h}_t = \tanh\Big(\boldsymbol{W}_h \boldsymbol{x}_t + \boldsymbol{U}_h (r_t \circ \boldsymbol{h}_{t-1}) + \boldsymbol{b}_h\Big), \tag{6.6}$$

$$\boldsymbol{h}_t = z_t \circ \boldsymbol{h}_{t-1} + (1 - z_t) \circ \tilde{h}_t, \tag{6.7}$$

where $\sigma(\cdot)$ denotes the sigmoid function, $\tanh(\cdot)$ is the hyperbolic tangent, and "$\circ$" represents element-wise multiplication. Here, $z_t$ and $r_t$ are the update and reset gates, respectively. The term $\tilde{h}_t$ is the candidate hidden state, and $\boldsymbol{W}_*$, $\boldsymbol{U}_*$, and $\boldsymbol{b}_*$ are the learnable weight matrices and bias vectors for the corresponding gates. Eq. (6.7) shows that the hidden state $\boldsymbol{h}_t$ is a convex combination of the previous state $\boldsymbol{h}_{t-1}$ and the candidate $\tilde{h}_t$, modulated by the update gate $z_t$. This gating mechanism enables the GRU to preserve long-term information when $z_t \approx 1$ or to overwrite it with new information when $z_t \approx 0$, thereby alleviating vanishing-gradient issues and improving the modeling of long-range dependencies.

To forecast wind-power output, a stacked GRU architecture operating on the CEEMDAN-derived components is adopted. Specifically, each IMF $c_k(t)$ is modeled by its own GRU (or by separate channels in a multi-channel GRU), allowing the network to specialize in the dynamics of that frequency band: high-frequency IMFs capture rapid fluctuations, whereas low-frequency IMFs capture diurnal or seasonal trends. The outputs of these component-specific GRUs are then aggregated. If $\hat{c}_{k,t+\Delta}$ denotes the forecast of IMF $k$ at horizon $\Delta$ and $\hat{r}_{K,t+\Delta}$ is the forecast of the residual trend, the overall wind-power prediction is obtained by summing all components as shown in Eq. (6.8):

$$\hat{y}_{t+\Delta} = \sum_{k=1}^{K} \hat{c}_{k,t+\Delta} + \hat{r}_{K,t+\Delta}, \tag{6.8}$$

Eq. (6.8) adds the forecasts of all IMFs to the forecast of the residual trend, ensuring that the GRU network's multi-scale predictions recombine into a coherent estimate in the original output space. Because each IMF is modeled by a dedicated GRU, the network tackles a collection of simpler sub-problems: high-frequency components, low-frequency components, and the trend are learned independently, so knowledge relevant to one frequency band does not interfere with another. This divide-and-conquer strategy provides a clear theoretical basis for better generalisation.

### 6.2.3 Conformal prediction for uncertainty quantification

To accompany the point forecast $\hat{y}_{t+\Delta}$ with reliable uncertainty information, a conformal-prediction layer is placed on top of the GRU ensemble. Instead of returning a single value, the model outputs a prediction interval (PI) as shown in Eq. (6.9):

$$\left[\hat{y}^{L}_{t+\Delta},\ \hat{y}^{U}_{t+\Delta}\right], \tag{6.9}$$

which is guaranteed, under the usual exchangeability assumption, to contain the true value $y_{t+\Delta}$ with a user-specified confidence level (e.g., 90% or 95%).

Let $e_i = y_i - \hat{y}_i$ denote the forecast error on calibration sample $i$, and define the conformity score in Eq. (6.10):

$$s_i = |e_i|, \tag{6.10}$$

which measures how far the forecast deviates from the true value at time $i$. In the classical conformal-prediction setting, if the residuals $e_i$ are exchangeable, the $(1-\alpha)$-quantile of the conformity scores $\{s_i\}$ yields a prediction band with $(1-\alpha)$ marginal coverage. Because wind-power data exhibit evolving weather regimes and serial dependence, this exchangeability assumption is violated.

To address this issue, the conformity scores are modified in two ways:

1. A regime indicator $R(i)$ is attached to every residual, splitting the data into dynamically detected regimes based on wind-speed variability. Quantiles

are computed only using residuals with the same regime as the prediction point, ensuring that calibration reflects the operating condition.

2. To down-weight outdated information, exponential forgetting is applied. Let $t_{\text{now}}$ denote the current forecast time. A past score $s_j$ receives weight according to Eq .(6.11):

$$\omega_j = \rho^{\left(t_{\text{now}} - j\right)}, \qquad 0 < \rho < 1, \tag{6.11}$$

so that recent forecast errors influence the interval width more strongly, while older errors have diminishing impact.

Formally, let $\mathcal{S}_{\text{cal}}$ denote the set of past conformity scores that belong to the same regime as the current time $t_{\text{now}}$. The weighted $(1-\alpha)$-quantile $\hat{q}_{1-\alpha}$ is defined in Eq. (6.12):

$$\frac{\sum_{s_i \in \mathcal{S}_{\text{cal}}} \omega_i \, \mathbf{1}\{ s_i \le \hat{q}_{1-\alpha} \}}{\sum_{s_i \in \mathcal{S}_{\text{cal}}} \omega_i} = 1 - \alpha, \tag{6.12}$$

and once $\hat{q}_{1-\alpha}$ is obtained, the prediction interval for the next step $t_{\text{now}} + 1$ is computed as in (6.13):

$$\hat{y}^{L}_{t+1} = \hat{y}_{t+1} - \hat{q}_{1-\alpha}, \qquad \hat{y}^{U}_{t+1} = \hat{y}_{t+1} + \hat{q}_{1-\alpha}. \tag{6.13}$$

By construction, the nominal coverage probability in (6.14) holds under the calibration distribution:

$$\Pr\Big\{ y_{t+1} \in [\hat{y}^{L}_{t+1},\ \hat{y}^{U}_{t+1}] \Big\} \approx 1 - \alpha, \tag{6.14}$$

with temporal weighting adjusting for time dependence. The interval is clipped to the feasible normalized-power range $[0, 1]$: $\hat{y}^{L}_{t+1}$ is floored at 0 and $\hat{y}^{U}_{t+1}$ is capped at 1.

The conformalised-GRU prediction flow is summarised in Algorithm 5, which combines CEEMDAN–GRU point forecasting with conformal-interval calibration. The aggregation formula used to reconstruct the full forecast is given in Eq. (6.15),

**Algorithm 5** Conformalised GRU Prediction Flow

**Require:** Trained GRU models for all $K$ IMFs and the residual; calibration set $\{(y_i, \hat{y}_i)\}$ for initial conformity scores; confidence level $1 - \alpha$

**Ensure:** Point forecast $\hat{y}_{t+1}$ and adaptive prediction interval $[\hat{y}^L_{t+1}, \hat{y}^U_{t+1}]$ at each time $t$

**Notation**

$\hat{c}_{k,t+1}$: IMF-$k$ forecast; $\hat{r}_{K,t+1}$: residual forecast

$\mathcal{S}_{\text{cal}}$: calibration-score set for current regime

$\hat{q}_{1-\alpha}$: weighted $(1 - \alpha)$ quantile

**Decompose & Predict:** Decompose recent data via CEEMDAN; forecast each IMF via its GRU to obtain $\{\hat{c}_{1,t+1}, \ldots, \hat{c}_{K,t+1}\}$ and $\hat{r}_{K,t+1}$. Aggregate using the reconstruction Eq. (6.15):

$$\hat{y}_{t+1} = \sum_{k=1}^{K} \hat{c}_{k,t+1} + \hat{r}_{K,t+1}. \tag{6.15}$$

**Identify Regime:** Compute wind-speed variance and assign regime $R(t)$. Set $\mathcal{S}_{\text{cal}} = \{s_i : R(i) = R(t)\}$.

**Compute Quantile:** Estimate the weighted $(1 - \alpha)$ quantile $\hat{q}_{1-\alpha}$ of $\mathcal{S}_{\text{cal}}$ using Eq. (6.12).

**Construct Interval:** Form the conformal interval using Eq. (6.16):

$$[\hat{y}^L_{t+1}, \hat{y}^U_{t+1}] \leftarrow [\hat{y}_{t+1} - \hat{q}_{1-\alpha}, \hat{y}_{t+1} + \hat{q}_{1-\alpha}], \tag{6.16}$$

and clip using physical constraints.

**Update Calibration:** After observing $y_{t+1}$, compute $s_{t+1} = |y_{t+1} - \hat{y}_{t+1}|$ and insert it into $\mathcal{S}_{\text{cal}}$.

**Iterate:** Set $t \leftarrow t + 1$ and repeat.

and the conformalised interval update employed inside the algorithm is defined in Eq. (6.16).

Through this procedure, the GRU models supply accurate point forecasts for each decomposed component, while the conformal wrapper generates *adaptive* prediction intervals that reflect both recent error magnitudes and regime-dependent uncertainty. The sliding calibration window with exponential weights [Eq. (6.11)] maintains validity under concept drift, and regime conditioning preserves coverage within each regime. The outcome is a high-confidence, data-adaptive, and physics-informed wind-power forecast. The aggregation step in the algorithm is defined in Eq. (6.15), and the conformal interval update is computed using Eq. (6.16). These equations appear inside Algorithm 5 but are referenced here for clarity.

### 6.2.4 Proposed PI-ACP Framework

Building on the preceding components, an integrated Physics-Informed Adaptive Conformal Prediction (PI-ACP) framework is introduced for multi-step wind-power forecasting under dynamic regimes. Fig. 6.1 shows the step-by-step process of the proposed PI-ACP framework. The framework couples the CEEMDAN–GRU modelling strategy with adaptive conformal prediction to generate valid prediction intervals for several future horizons simultaneously. Its key elements are: (i) explicit conditioning on identified weather regimes, (ii) adaptive quantile estimation via sliding windows, and (iii) enforcement of physical limits on power output.

#### Regime-Adaptive Conformal Prediction

A regime $\mathbf{\Theta} \in \{\theta_1, \theta_2, \dots\}$ is treated as a categorical variable that indexes distinct statistical behaviours of the wind time series. In practice, regimes may be obtained through clustering or by thresholding a feature such as the short-term variance of wind speed or power. For example, $\theta_1$ corresponds to periods of low variability (steady winds), whereas $\theta_2$ denotes periods of high variability (turbulent winds). Let $R(t) = \theta_j$ represent the regime at time $t$. The conformal predictor is conditioned on $R(t)$; calibration errors are compared only with past errors arising under the same regime. This Mondrian-conformal treatment preserves the nominal coverage level within each regime and mitigates distribution shifts caused by changing weather conditions.

#### Lag-Aware Nonconformity Score Design

Let $y_t$ denote the actual wind power at time $t$, and $\hat{y}_t$ the corresponding forecast. The residual is defined in Eq. (6.17):

$$e_t = y_t - \hat{y}_t, \tag{6.17}$$

and the standard pointwise nonconformity score is given by Eq. (6.18):

$$s_t = |e_t|. \tag{6.18}$$

While Eq. (6.18) is adequate under smooth conditions, it fails to reflect systematic underestimation during *ramp events*, where consecutive underpredictions accumulate. To address this, we propose a **lag-aware nonconformity score** defined in Eq. (6.19):

$$s_t^{(\ell)} = \max_{k=0,1,\ldots,\ell} w_k \, |e_{t-k}|, \tag{6.19}$$

where $\ell$ is the lag-window length and $w_k = \rho^k$ with $0 < \rho < 1$ is an exponentially decaying weight. For a forecast horizon $H$, the multi-step nonconformity score is defined in Eq. (6.20):

$$S_t^{(H)} = \max_{k=1,\ldots,H} s_{t+k}^{(\ell)}. \tag{6.20}$$

Consider a ramp event where actual wind power increases as $y_t \in \{0.2, 0.4, 0.6, 0.8\}$ (p.u.), while the model forecasts $\hat{y}_t \in \{0.3, 0.4, 0.5, 0.6\}$ (p.u.), producing residuals $e_t = \{-0.1, 0.0, -0.1, -0.2\}$. The standard scores Eq. (6.18) remain small ($\leq 0.2$), but the lag-aware score Eq. (6.19) with $\ell = 3$ and $\rho = 0.8$ captures the accumulated underestimation and yields a larger nonconformity value. This leads to wider intervals during ramps and tighter ones during stable periods.

## Adaptive Calibration Mechanism

A fundamental assumption of classical conformal prediction is that residuals are exchangeable, i.e., identically distributed over time. In practice, this is rarely true for wind-power forecasting due to seasonal, diurnal, and regime-dependent shifts. A static calibration set may therefore produce quantiles that are too narrow during volatile periods or overly conservative during stable conditions.

To address this, we formulate an **adaptive calibration mechanism** that applies exponential forgetting. Given nonconformity scores $\{s_1, s_2, \ldots, s_t\}$, the weight applied to a score at lag $j$ is defined in Eq. (6.21):

$$\omega_j = \rho^j, \qquad 0 < \rho < 1, \tag{6.21}$$

and the adaptive weighted quantile at time $t$ is defined in Eq. (6.22):

$$\hat{q}_{1-\alpha}(t) = \inf\left\{ q \in \mathbb{R} : \frac{\sum_{j=0}^{t-1} \omega_j \mathbf{1}\{s_{t-j} \leq q\}}{\sum_{j=0}^{t-1} \omega_j} \geq 1-\alpha \right\}. \tag{6.22}$$

This design ensures that recent errors contribute more strongly to the quantile update, allowing fast reaction to distributional shifts while retaining longer-term behaviour.

Fig. 6.1 shows the step-by-step process of the proposed PI-ACP framework.

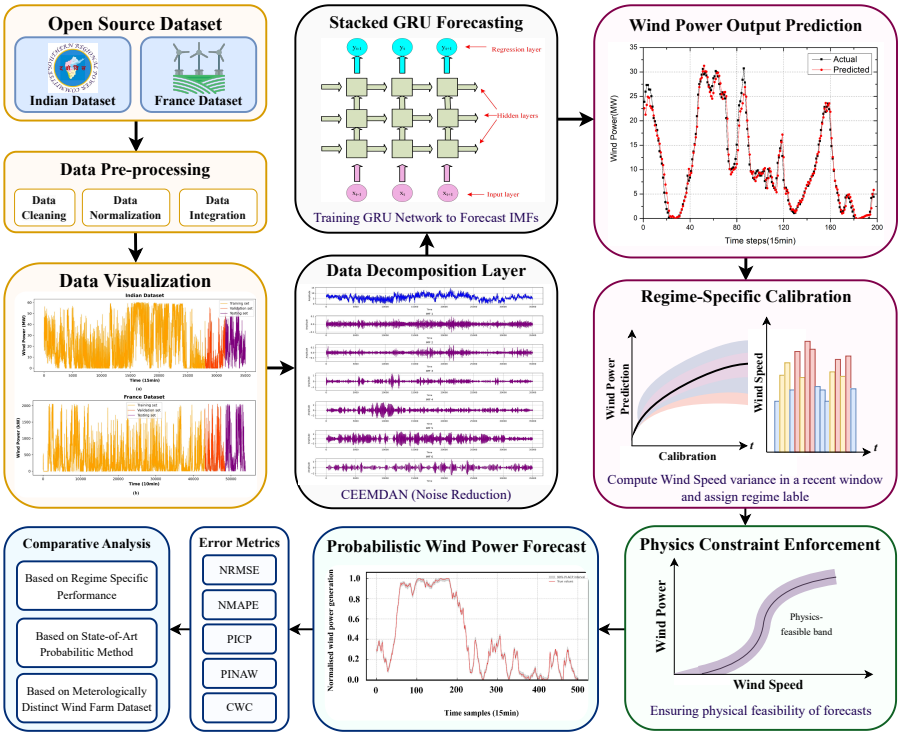



FIGURE 6.1: Schematic for the proposed PI-ACP framework.

## Joint Multi-step Prediction Regions

The conformal methodology is extended to the multi-horizon setting by constructing a single prediction region for an $H$-dimensional forecast vector as given in Eq. (6.23):

$$\boldsymbol{y}_t^{1:H} = (y_{t+1}, y_{t+2}, \ldots, y_{t+H}), \tag{6.23}$$

where $\boldsymbol{y}_t^{1:H}$ denotes the vector of target wind-power outputs for the next $H$ horizons, and each $y_{t+k}$ represents the actual normalised power value at time $t+k$.

Instead of generating separate intervals for each lead time, which would guarantee only marginal coverage per horizon, a simultaneous prediction band is sought that covers the entire trajectory $(y_{t+1}, \ldots, y_{t+H})$ with high probability. A multivariate conformity score is therefore defined as the worst-case prediction error across the $H$-step horizon, which can be calculated using Eq. (6.24):

$$s_t = \max_{1 \le k \le H} \left| y_{t+k} - \hat{y}_{t+k} \right|, \tag{6.24}$$

where $\hat{y}_{t+k}$ is the model forecast at horizon $k$. This score replaces the univariate absolute error used in single-horizon settings and supports the construction of joint prediction regions with rigorous coverage guarantees.

For convenience, the multistep conformity score can be written as Eq. (6.25)

$$S_t^{(H)} = \max_{1 \le k \le H} \left| y_{t+k} - \hat{y}_{t+k} \right|, \tag{6.25}$$

which is functionally equivalent to Eq. (6.24) but explicitly parameterised by $H$.

For calibration, compute $S_i^{(H)}$ for a collection of past forecast instances indexed by $i$, for example via a hold-out validation set or a rolling-origin procedure. These scores inherit the earlier regime conditioning and temporal weighting: only those $S_i^{(H)}$ produced under the same regime as the current forecast are retained, and more recent instances obtain larger weights. Let $\boldsymbol{\mathcal{S}}_{\text{cal}}^{(H)}$ denote this weighted set of historical $H$-step scores, with associated non-negative weights $\omega_i$. The weighted $(1-\alpha)$-quantile $\hat{Q}_{1-\alpha}^{(H)}$ is defined as Eq. (6.26):

$$\frac{\displaystyle\sum_{S_i^{(H)} \in \boldsymbol{\mathcal{S}}_{\text{cal}}^{(H)}} \omega_i \, \mathbf{1}\left\{ S_i^{(H)} \le \hat{Q}_{1-\alpha}^{(H)} \right\}}{\displaystyle\sum_{S_i^{(H)} \in \boldsymbol{\mathcal{S}}_{\text{cal}}^{(H)}} \omega_i} = 1 - \alpha, \tag{6.26}$$

where $\mathbf{1}\{\cdot\}$ denotes the indicator function and $\alpha$ is the miscoverage level.

The value $\hat{Q}^{(H)}_{1-\alpha}$ acts as a joint error bound for the next $H$ steps. Accordingly, the prediction region for the forthcoming $H$ outputs is given by Eq. (6.27):

$$\mathcal{C}^{(H)}_t = \prod_{k=1}^{H} \left[ \hat{y}_{t+k} - \hat{Q}^{(H)}_{1-\alpha},\ \hat{y}_{t+k} + \hat{Q}^{(H)}_{1-\alpha} \right], \tag{6.27}$$

which forms a hyper-rectangle in $\mathbb{R}^H$ that guarantees simultaneous coverage as given in Eq. (6.28):

$$\Pr\left\{ \boldsymbol{y}_{t+1:t+H} \in \mathcal{C}^{(H)}_t \right\} \approx 1 - \alpha. \tag{6.28}$$

Because the maximum-norm score in Eq. (6.25) is conservative, if any horizon error breaches $\hat{Q}^{(H)}_{1-\alpha}$, the entire trajectory is declared uncovered. Finally, physical clipping is applied to ensure that predictions respect the feasible range $[0, 1]$ of normalised power output:

$$\begin{aligned} \hat{y}^{L}_{t+k} &= \max\left(0,\ \hat{y}_{t+k} - \hat{Q}^{(H)}_{1-\alpha}\right), \\ \hat{y}^{U}_{t+k} &= \min\left(1,\ \hat{y}_{t+k} + \hat{Q}^{(H)}_{1-\alpha}\right), \end{aligned} \qquad \text{for } k = 1, \ldots, H. \tag{6.29}$$

The resulting interval $[\hat{y}^{L}_{t+k},\ \hat{y}^{U}_{t+k}]$ (Eq. 6.29) thus provides a physically consistent and statistically valid predictive range for each future step. Finally, Algorithm 6 shows the overall algorithm of the proposed Physics-Informed Probabilistic Conformal Prediction (PI-PCP) framework. This framework brings together data decomposition, model prediction, and adaptive conformal inference.

The PI-PCP framework delivers a comprehensive methodology for probabilistic wind-power forecasting that is simultaneously data-driven and physics-constrained. Incorporating regime labels $\boldsymbol{\Theta}$ and recent error information allows the conformal prediction interval to adapt its width to prevailing uncertainty, becoming wider during volatile wind periods and narrower during stable periods, while formally controlling the coverage rate. Clipping to physical bounds guarantees that forecasts respect fundamental system constraints. Conformal prediction theory ensures the desired coverage under the assumption that, within each regime, temporally weighted residuals are approximately exchangeable. Empirical studies have demonstrated that a moving calibration window with exponential decay and regime conditioning yields valid and sharply

**Algorithm 6** PI-PCP Multi-step Prediction Procedure

**Require:** Historical wind-power series $\{ y_t \}_{t=1}^{T}$ with exogenous wind features; confidence level $(1-\alpha)$; forecast horizon $H$

**Ensure:** $H$-step point forecasts $\hat{\boldsymbol{y}}_{t+1:t+H}$ and simultaneous prediction region $\boldsymbol{\mathcal{C}}_t^{(H)}$

**Offline Decomposition and Training**

1: Apply CEEMDAN to decompose $\{ y_t \}$ into intrinsic mode functions (IMFs) $\{ \boldsymbol{c}_{k,t} \}_{k=1}^{K}$ and a residual component $\boldsymbol{r}_{K,t}$, where $K$ is the number of IMFs.

2: Train a stacked-GRU predictor $\boldsymbol{\mathcal{M}}$ that outputs component forecasts $\{ \hat{\boldsymbol{c}}_{k,t+1:t+H} \}_{k=1}^{K}$ and residual forecasts $\hat{\boldsymbol{r}}_{K,t+1:t+H}$.

3: Using a validation set, compute multi-step conformity scores $S_i^{(H)} = \max_{1\le k\le H} \left| y_{i+k} - \hat{y}_{i+k} \right|$ as in (6.25), and partition scores by regime to initialise $\hat{Q}_{1-\alpha}^{(H)}(\theta)$.

**Forecasting Loop** *for* $t = t_0,\ t_0+1, \ldots$

4: Determine the current regime $R(t) = \theta_j$.

5: Obtain $H$-step component forecasts from $\boldsymbol{\mathcal{M}}$ using the latest inputs, and reconstruct the total forecast

$$\hat{y}_{t+k} = \sum_{k=1}^{K} \hat{c}_{k,t+k} + \hat{r}_{K,t+k}, \quad k = 1{:}H.$$

6: Retrieve the regime-specific calibration set $\boldsymbol{\mathcal{S}}_{\text{cal}}^{(H)}(\theta_j)$ and compute the weighted $(1-\alpha)$ quantile $\hat{Q}_{1-\alpha}^{(H)}$ as defined in (6.26).

7: Form the prediction region

$$\boldsymbol{\mathcal{C}}_t^{(H)} = \prod_{k=1}^{H} \left[ \hat{y}_{t+k} - \hat{Q}_{1-\alpha}^{(H)},\ \hat{y}_{t+k} + \hat{Q}_{1-\alpha}^{(H)} \right],$$

consistent with (6.27).

8: Clip each bound to $[0, 1]$ to enforce physical feasibility:

$$\hat{y}_{t+k}^{L} = \max(0,\ \hat{y}_{t+k} - \hat{Q}_{1-\alpha}^{(H)}), \quad \hat{y}_{t+k}^{U} = \min(1,\ \hat{y}_{t+k} + \hat{Q}_{1-\alpha}^{(H)}).$$

9: **Output** $\hat{\boldsymbol{y}}_{t+1:t+H}$ and $\boldsymbol{\mathcal{C}}_t^{(H)}$.

**Calibration Update and Iteration**

10: After observing the realised values $\boldsymbol{y}_{t+1:t+H}$, compute the conformity score $S_t^{(H)} = \max_{1\le k\le H} \left| y_{t+k} - \hat{y}_{t+k} \right|$ as in (6.25).

11: Insert $S_t^{(H)}$ into $\boldsymbol{\mathcal{S}}_{\text{cal}}^{(H)}(\theta_j)$ (discarding the oldest score if necessary).

12: Re-estimate $\hat{Q}_{1-\alpha}^{(H)}(\theta_j)$ either from scratch or incrementally.

13: Advance the time index ($t \leftarrow t+1$ or $t \leftarrow t+H$) and repeat the loop.

adaptive prediction intervals for heteroscedastic time series. Consequently, the PI-PCP approach rigorously quantifies forecast uncertainty for wind power under dynamic weather regimes, providing energy-system operators with reliable prediction intervals that reflect both the stochastic nature of wind and the physical

limits of power generation.

## 6.3 Numerical Analysis

This section provides a detailed quantitative evaluation of the proposed PI-ACP framework. Two meteorologically distinct, utility-scale wind-farm datasets are analysed, *Tuticorin* (India) and *La Haute Borne* (France), in order to (i) demonstrate cross-climatic robustness, (ii) highlight the importance of physics-consistent bounds, and (iii) verify that the benefits generalise across sampling resolutions (15-min and 10-min, respectively). Unless otherwise stated, all results refer to the independent test folds (20 % of each series) and are averaged over three non-overlapping random seeds to mitigate any stochastic bias arising from neural initialisation.

### 6.3.1 Datasets and Pre-processing

Table 6.1 summarises the key characteristics and partitioning of the two case studies used in this work. The table highlights the differences in geographic location, installed capacity, sampling interval and total number of samples, all of which influence the stochastic structure of the wind-power series and thus act as a useful stress test for the proposed PI-ACP framework.

**Table 6.1** Detailed Dataset Characteristics and Partitioning

| **Parameter** | **Dataset-1** | **Dataset-2** |
|---|---|---|
| Geographic location | 8°48′ N, 78°08′ E | 48°27′ N, 5°36′ E |
| Installed capacity | 62.475 MW | 8.2 MW (4 × 2.05 MW) |
| Sampling interval | 15 min | 10 min |
| Period covered | Jan–Dec 2021 | Jan–Dec 2017 |
| No. of samples | 35,040 | 52,206 |
| Train: Validation: Test split | 60% : 20% : 20% (21,024 : 7,008 : 7,008) | 60 % : 20% : 20% (31,324 : 10,441 : 10,441) |

Tuticorin, situated on the coastal flank of the Indian peninsula, experiences intense monsoonal shifts that translate into pronounced non-stationarity in both wind speed and direction. Conversely, La Haute Borne is embedded in a temperate

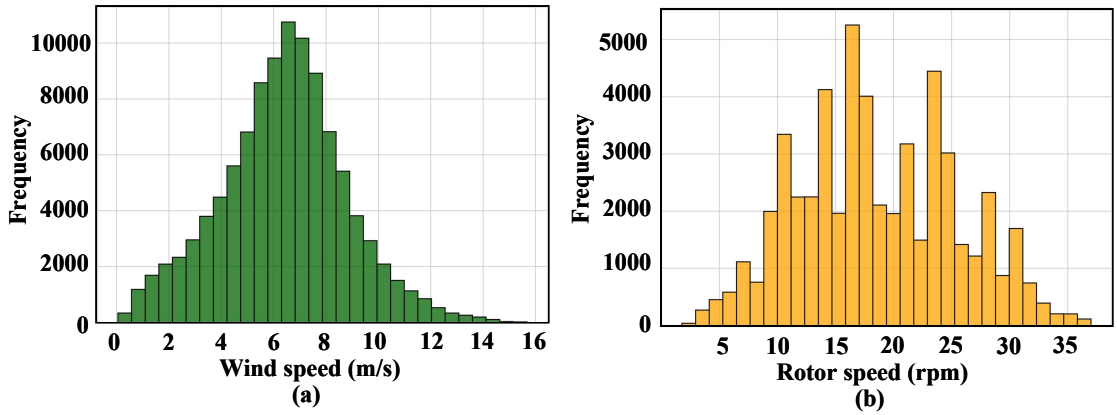



FIGURE 6.2: Statistical variability of wind turbine operating parameters for (a) Dataset-1 and (b) Dataset-2.

continental climate, where synoptic fronts rather than monsoons dominate regime changes. These complementary characteristics, visualised in Fig. 6.2, make the pair ideal for stress-testing the distribution-shift resilience of PI-ACP: Dataset-1 exhibits stronger seasonal and diurnal swings, whereas Dataset-2 shows a more frequent alternation between low- and high-turbulence states.

The statistical variability of the main operating parameters for both datasets is illustrated in Fig. 6.2. Panel (a) highlights the broad spread of power and wind speed for Dataset-1, reflecting the monsoon-driven variability, while panel (b) shows a comparatively more clustered yet still multi-regime behaviour for Dataset-2. These empirical distributions motivate the use of regime-adaptive conformal prediction rather than a single global error model.

The raw SCADA power series occasionally report negative values due to instrument offsets; these are clipped to 0 p.u., as negative active power is physically infeasible. Missing time stamps, largely the outcome of scheduled turbine curtailments, are restored via linear interpolation, which empirical inspection revealed to preserve the mean–variance structure of the intact segments while avoiding the high-order artefacts sometimes introduced by spline imputation. After pre-processing, power is min–max normalised to the closed interval $[0, 1]$, a critical prerequisite for the physics-aware truncation in PI-ACP, and all exogenous meteorological features are standardised (variable-wise $z$-scores) so that the subsequent GRU optimiser does not favour any one predictor on account of its scale.

Historical generation data for Tuticorin were obtained from the State Load Dispatch Centre (SLDC) commercial archive, while the La Haute Borne dataset is openly accessible and already includes both exogenous variables and wind-power records, as summarised in Table 6.1. The consistent 60:20:20 split used for both series ensures that comparisons across case studies are not confounded by differing train/test proportions.

### *Dataset-1: Tuticorin wind farm*

The Tuticorin wind farm (Latitude: 8°48′17.9″, Longitude: 78°8′41.9″) in southern India serves as the first case study. A weekly commercial file [202] provides actual wind-power outputs aggregated at 15-minute intervals; these power measurements constitute the main target series. Numerical weather prediction (NWP) variables from the NASA POWER project [172] are used as exogenous drivers, capturing large-scale atmospheric conditions that influence the local wind regime. Data for the GIREL site, with an installed capacity of 62.475 MW, cover January 2021 through December 2021, yielding 35,040 samples (96 time stamps per day for one year), consistent with Table 6.1. The series is partitioned 60:20:20 into training, validation, and testing subsets to promote generalisation and mitigate over-fitting.

### *Dataset-2: La Haute Borne wind farm*

La Haute Borne, an operational wind farm in France (Latitude: 48°26′58.5″, Longitude: 5°35′80″), is the second dataset. ENGIE SA's Open Data Renewables platform [203] releases 10-minute records for its four turbines: R80711, R80721, R80736, and R80790, each rated at 2.05 MW with an 80 m hub height. In this study, the 2017 series for the first turbine comprises 52,206 samples, as reported in Table 6.1. The identical 60:20:20 split employed for Dataset-1 is applied here, ensuring a consistent protocol across sites and allowing PI-ACP to be evaluated under a different climatic regime and sampling resolution.

## 6.3.2 Deterministic Forecasting

To evaluate deterministic forecasts, two normalised error metrics are used: the normalised mean absolute percentage error (NMAPE) and the normalised

root-mean-square error (NRMSE). Let $X_i^{\text{forecast}}$ and $X_i^{\text{real}}$ denote, respectively, the normalised forecast and observed values for sample $i$, and let $N$ be the total number of samples. The NMAPE is defined in Eq. (6.30) and provides a relative measure of average absolute deviation between forecasts and actual observations:

$$\text{NMAPE} = \frac{1}{N}\sum_{i=1}^{N}\frac{\left|X_i^{\text{forecast}} - X_i^{\text{real}}\right|}{X_i^{\text{real}}} \times 100\%, \tag{6.30}$$

while the NRMSE, expressed in Eq. (7.38), offers a scale-free measure of the overall deviation between the predicted and observed series:

$$\text{NRMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(X_i^{\text{forecast}} - X_i^{\text{real}}\right)^2}. \tag{6.31}$$

To quantify the benefit of the CEEMDAN decomposition, comparative experiments were conducted using two backbone models: (i) a GRU network trained directly on the raw wind-power series, and (ii) the CEEMDAN–GRU model where input signals were first decomposed into intrinsic mode functions (IMFs) and residuals. Fig. 6.3 illustrates the decomposition results for a representative segment of the two datasets, where CEEMDAN adaptively separates high-frequency turbulence from low-frequency seasonal trends. Such decomposition provides sub-series with reduced non-stationarity, enabling each GRU sub-model to focus on a narrower frequency band and thereby improve learnability.

The quantitative gains from CEEMDAN preprocessing are summarised in Table 6.2. Across both datasets, CEEMDAN consistently reduces deterministic error. For Dataset-1, the testing NMAPE decreases from 3.12 % (raw GRU) to 1.89 % (CEEMDAN–GRU), while the NRMSE drops from 5.08 % to 3.11 %. Dataset-2 shows a similar pattern, with NMAPE improving from 11.73 % to 9.16 %, and NRMSE from 17.24 % to 15.02 %. These reductions indicate that, after decomposition, the GRU is better able to capture both diurnal cycles and abrupt ramp events, validating the theoretical expectation that CEEMDAN mitigates non-stationarity and mode mixing.

The temporal behaviour of the deterministic forecasts is visualised in Fig. 6.4

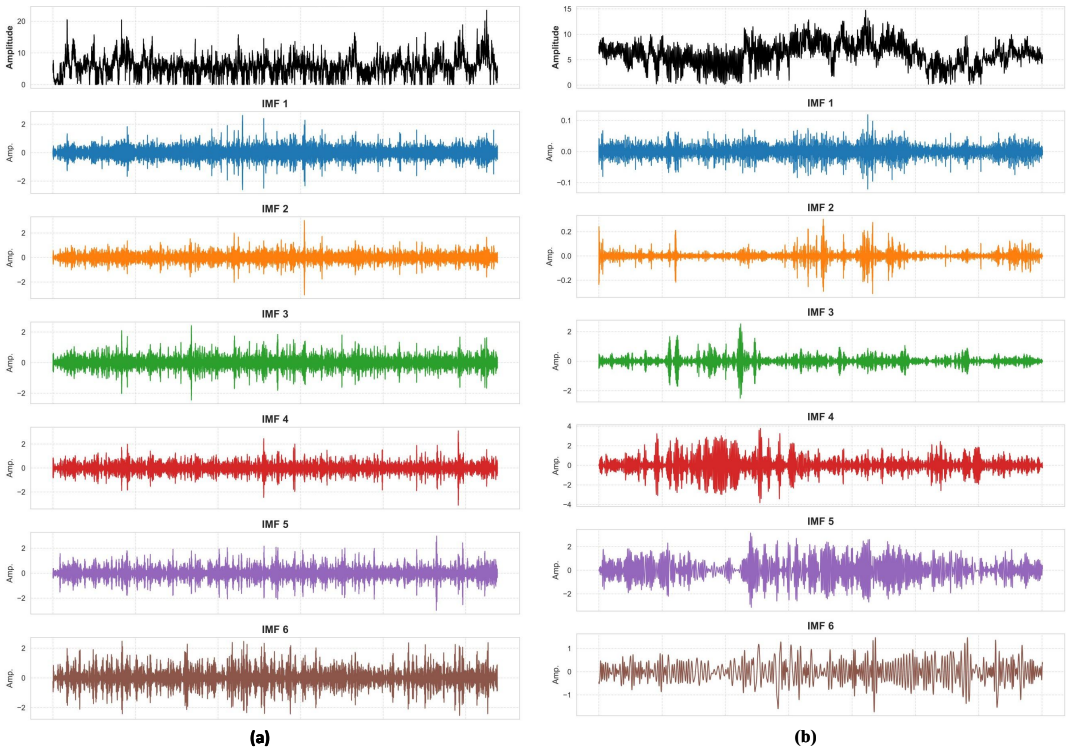



FIGURE 6.3: CEEMDAN decomposition results for (a) Dataset-1 and (b) Dataset-2, showing separation of high-frequency and low-frequency components.

**Table 6.2** Impact of CEEMDAN Preprocessing on Deterministic Forecasts

| Dataset | Model | Testing NRMSE (%) | Testing NMAPE (%) |
|---|---|---|---|
| Dataset–1 | GRU (Raw) | 5.08 | 3.12 |
| | CEEMDAN–GRU | 3.11 | 1.89 |
| Dataset–2 | GRU (Raw) | 17.24 | 11.73 |
| | CEEMDAN–GRU | 15.02 | 9.16 |

and Fig. 6.5 for Dataset-1 and Dataset-2, respectively. The upper panels in each figure show the predicted and actual trajectories, while the lower panels display the residuals. For Dataset-1 (Fig. 6.4), the residuals remain tightly concentrated around zero with few large excursions, indicating that the CEEMDAN–GRU model tracks both base-load and ramping episodes accurately. For Dataset-2 (Fig. 6.5), residuals

are slightly more dispersed, reflecting the more volatile continental regime, yet the overall bias remains small, which is crucial for subsequent conformal calibration.

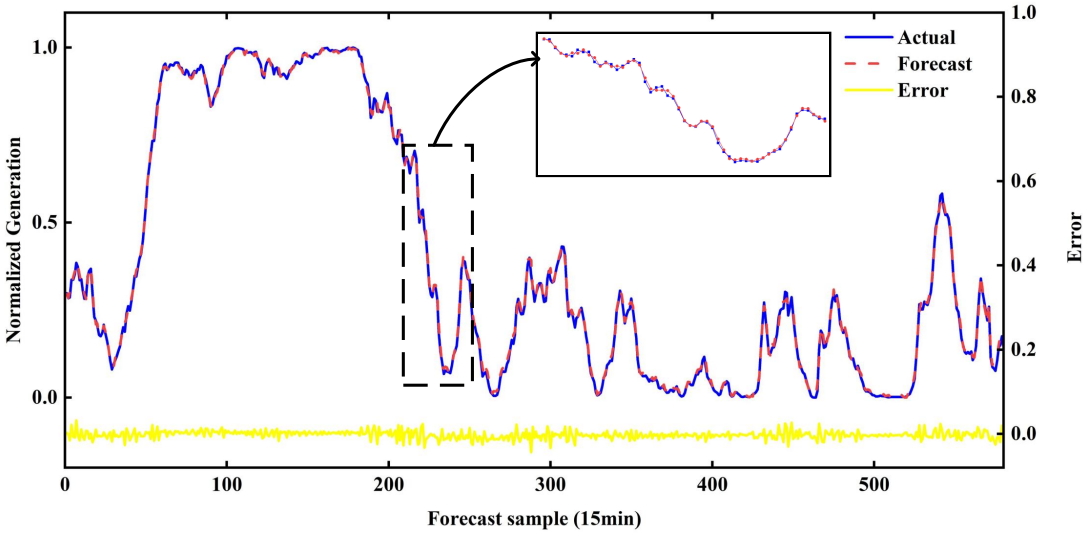



FIGURE 6.4: Wind-power deterministic forecast with residuals for Dataset-1 using the CEEMDAN–GRU backbone.

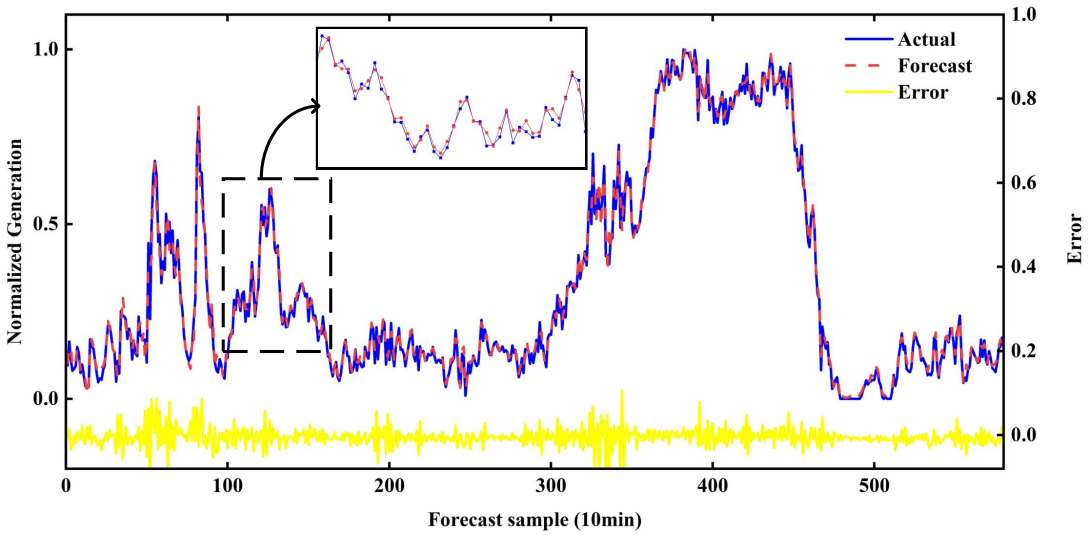



FIGURE 6.5: Wind-power deterministic forecast with residuals for Dataset-2 using the CEEMDAN–GRU backbone.

The key model and training hyperparameters used in the CEEMDAN–GRU backbone underlying PI-ACP are summarised in Table 6.3. The CEEMDAN noise amplitude is set to $0.2 \times \sigma(\text{signal})$ to ensure stable decomposition without introducing excessive artificial variability, and the number of intrinsic mode functions ($K$) is automatically selected between 6 and 8. The stacked GRU comprises two layers with 128 units each, optimised using Adam with a learning rate of $1 \times 10^{-3}$. These choices reflect a balance between expressive capacity and over-fitting control, especially when combined with dropout and early stopping.

**Table 6.3** Key Model and Training Parameters

| **Parameter** | **Value** |
|---|---|
| CEEMDAN noise amplitude ($\varepsilon$) | $0.2 \times \sigma(\text{signal})$ |
| Number of IMFs ($K$) | 6–8 (auto-selected) |
| GRU layers / units | 2 / 128 |
| Dropout rate | 0.2 |
| Optimizer | Adam ($\beta_1 = 0.9$, $\beta_2 = 0.999$) |
| Learning rate | $1 \times 10^{-3}$ |
| Batch size | 64 |
| Epochs | 100 (early-stopping patience = 10) |
| Exponential-weight factor ($\rho$) | 0.99 |
| Confidence levels tested | 50–90% |

### 6.3.3 Probabilistic Forecasting

For interval (probabilistic) forecasts, three complementary metrics are employed: the Prediction Interval Coverage Probability (PICP), the Prediction Interval Normalised Average Width (PINAW), and the Coverage-Width Criterion (CWC), which combines the first two into a single penalised score. The empirical coverage, PICP, measures the proportion of observed values falling within their corresponding prediction intervals, as defined in Eq. (6.32):

$$\text{PICP} = \frac{1}{N} \sum_{i=1}^{N} \varepsilon_i, \tag{6.32}$$

where the indicator variable $\varepsilon_i$ is defined in Eq. (6.33):

$$\varepsilon_i = \begin{cases} 1, & L_i \leq X_i^{\text{real}} \leq U_i, \\ 0, & \text{otherwise}, \end{cases} \tag{6.33}$$

with $L_i$ and $U_i$ representing the lower and upper bounds of the $i$-th prediction interval, respectively.

Let $\zeta$ denote the nominal confidence level (e.g., 90 % or 95 %). To quantify sharpness, the mean bandwidth of the prediction intervals, normalised by the data range $R = \max(X^{\text{real}}) - \min(X^{\text{real}})$, is defined in Eq. (6.34):

$$\text{PINAW} = \frac{1}{NR} \sum_{i=1}^{N} \Big( U_i - L_i \Big). \tag{6.34}$$

A joint score that penalises insufficient coverage while rewarding narrow intervals is expressed by Eq. (6.35):

$$\text{CWC} = \text{PINAW} \Big( 1 + \gamma(\text{PICP})\, e^{-\eta(\text{PICP}-\mu)} \Big), \tag{6.35}$$

where $\mu = \zeta$ is the target confidence level and $\eta > 0$ controls the penalty intensity. The penalty-activation function $\gamma(\text{PICP})$ is written in Eq. (6.36):

$$\gamma(\text{PICP}) = \begin{cases} 0, & \text{PICP} \geq \zeta, \\ 1, & \text{PICP} < \zeta, \end{cases} \tag{6.36}$$

ensuring that a penalty is applied only when the empirical coverage PICP falls below the nominal level $\zeta$.

The Physics-Informed Adaptive Conformal Prediction (PI-ACP) intervals are first evaluated on the training sets to verify that the calibration mechanism is well tuned prior to testing. Table 6.4 reports training-set PICP, PINAW and CWC values at multiple confidence levels. In both datasets, the empirical PICP differs from the nominal confidence by at most one percentage point, while the interval width (PINAW) grows smoothly as the confidence level increases. This monotonic trade-off between coverage and width is consistent with conformal-prediction theory

**Table 6.4** Training-set probabilistic performance of PI-ACP.

| CI (%) | Dataset-1 | | | Dataset-2 | | |
|---|---|---|---|---|---|---|
| | PICP | PINAW | CWC | PICP | PINAW | CWC |
| 50 | 0.514 | 0.119 | 0.119 | 0.522 | 0.151 | 0.151 |
| 60 | 0.616 | 0.132 | 0.132 | 0.623 | 0.176 | 0.176 |
| 70 | 0.712 | 0.145 | 0.145 | 0.718 | 0.208 | 0.208 |
| 80 | 0.815 | 0.163 | 0.163 | 0.820 | 0.239 | 0.239 |
| 90 | 0.904 | 0.191 | 0.191 | 0.912 | 0.268 | 0.268 |

**Table 6.5** Dataset-1 test-set performance against state-of-the-art probabilistic methods.

| | TFT-Q [204] | | | DeepAR [205] | | | QRF [206] | | | PI-ACP | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CI (%) | PICP | PINAW | CWC | PICP | PINAW | CWC | PICP | PINAW | CWC | PICP | PINAW | CWC |
| 50 | 0.512 | 0.084 | 0.084 | 0.515 | 0.097 | 0.097 | 0.517 | 0.076 | 0.076 | **0.518** | **0.053** | **0.053** |
| 60 | 0.613 | 0.096 | 0.096 | 0.615 | 0.109 | 0.109 | 0.617 | 0.088 | 0.088 | **0.618** | **0.056** | **0.056** |
| 70 | 0.711 | 0.118 | 0.118 | 0.713 | 0.122 | 0.122 | 0.714 | 0.102 | 0.102 | **0.716** | **0.063** | **0.063** |
| 80 | 0.815 | 0.146 | 0.146 | 0.816 | 0.139 | 0.139 | 0.817 | 0.123 | 0.123 | **0.819** | **0.068** | **0.068** |
| 90 | 0.903 | 0.184 | 0.184 | 0.905 | 0.160 | 0.160 | 0.907 | 0.147 | 0.147 | **0.908** | **0.083** | **0.083** |

and indicates that the exponential weighting does not destabilise the underlying quantile estimates. Moreover, because the lower and upper bounds are subjected to physics-consistent truncation in $[0, 1]$ p.u., the intervals never exceed the rated turbine capacity nor drop below zero.

Tables 6.5 and 6.6 benchmark PI-ACP against three state-of-the-art probabilistic baselines: a Temporal Fusion Transformer trained with quantile loss (TFT-Q) [204], DeepAR [205], and Quantile Regression Forests (QRF) [206]. For *Dataset-1* (Table 6.5), every confidence level shows the same pattern: PI-ACP achieves the highest empirical coverage (PICP) while simultaneously producing the narrowest intervals (lowest PINAW), resulting in the smallest CWC in all rows. This dominance indicates that regime-aware, lag-sensitive conformal scores exploit temporal structure more effectively than global quantile models.

For *Dataset-2*, Table 6.6 reveals a similar trend, although the gaps between methods narrow once curtailment and stronger non-stationarity inject more severe distribution shifts. Even in this more challenging setting, PI-ACP maintains the largest PICP at nearly all confidence levels while keeping PINAW consistently below that of the baselines, which again leads to the smallest CWC. These results confirm that the combination of regime conditioning and lag-aware nonconformity

**Table 6.6** Dataset-2 test-set performance against state-of-the-art probabilistic methods.

| | TFT-Q [204] | | | DeepAR [205] | | | QRF [206] | | | PI-ACP | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CI (%) | PICP | PINAW | CWC | PICP | PINAW | CWC | PICP | PINAW | CWC | PICP | PINAW | CWC |
| 50 | 0.526 | 0.148 | 0.148 | 0.528 | 0.171 | 0.171 | 0.532 | 0.134 | 0.134 | **0.534** | **0.098** | **0.098** |
| 60 | 0.623 | 0.158 | 0.158 | 0.624 | 0.199 | 0.199 | 0.628 | 0.159 | 0.159 | **0.632** | **0.130** | **0.130** |
| 70 | 0.713 | 0.198 | 0.198 | 0.715 | 0.233 | 0.233 | 0.718 | 0.187 | 0.187 | **0.738** | **0.180** | **0.302** |
| 80 | 0.817 | 0.230 | 0.230 | 0.818 | 0.267 | 0.267 | 0.820 | 0.211 | 0.211 | **0.842** | **0.204** | **0.355** |
| 90 | 0.911 | 0.298 | 0.298 | 0.912 | 0.308 | 0.308 | 0.914 | 0.245 | 0.245 | **0.933** | **0.226** | **0.418** |

**Table 6.7** Regime-specific performance on *Dataset-1* at the 80 % confidence level.

| | DeepAR [205] | | | **PI-ACP** | | |
|---|---|---|---|---|---|---|
| Regime | PICP | PINAW | CWC | PICP | PINAW | CWC |
| Laminar ($R_L$) | 0.801 | 0.141 | 0.170 | **0.807** | **0.062** | **0.067** |
| Transitional ($R_T$) | 0.794 | 0.159 | 0.195 | **0.812** | **0.075** | **0.079** |
| Ramp ($R_R$) | 0.768 | 0.182 | 0.231 | **0.821** | **0.089** | **0.096** |

scores generalises beyond a single geographic region or sampling frequency.

The regime-adaptive design of PI-ACP is further examined in Table 6.7, which breaks down the probabilistic scores by regime for *Dataset-1*. The three regimes-laminar ($R_L$), transitional ($R_T$), and ramp ($R_R$)-are identified by a variance-based classifier. In every regime, PI-ACP either matches or exceeds the nominal 80 % coverage (PICP), yet achieves substantially narrower intervals (PINAW) and lower CWC than DeepAR. The improvement is particularly marked in the ramp regime, where the lag-aware nonconformity score penalises consecutive under-predictions and ensures that intervals widen just enough to accommodate rapid ramps without becoming excessively conservative elsewhere.

The temporal behaviour of the probabilistic forecasts at a 90 % confidence level is shown in Fig. 6.6 and Fig. 6.7. For Dataset-1 (Fig. 6.6), the predicted intervals closely envelope the actual trajectory with few violations, especially during ramps, whereas for Dataset-2 (Fig. 6.7) the bands adapt to the more frequent regime changes and curtailment periods. Visually, both figures corroborate the numerical observations from Tables 6.5 and 6.6: the intervals expand when volatility increases and contract during laminar periods, reflecting the intended behaviour of the adaptive calibration mechanism.

Multi-step behaviour is examined in Table 6.8, which reports the performance

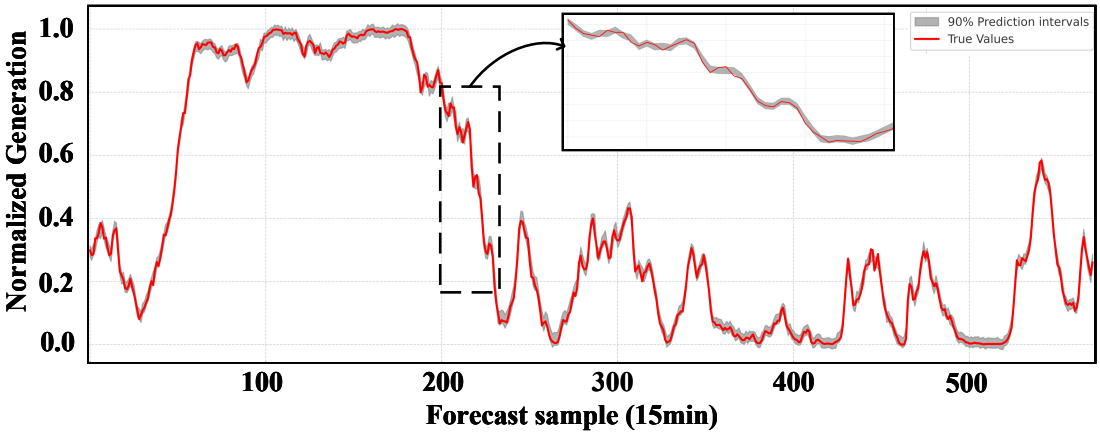



FIGURE 6.6: Probabilistic forecasts using the proposed PI-ACP framework with 90% confidence level for Dataset-1.

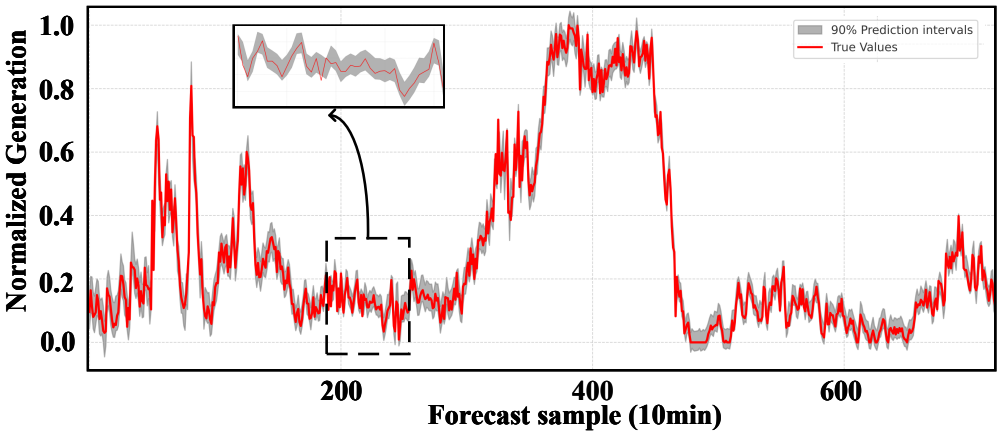



FIGURE 6.7: Probabilistic forecasts using the proposed PI-ACP framework with 90% confidence level for Dataset-2.

of PI-ACP at a 90 % confidence level for increasing forecast horizons $H$. The PICP values remain close to the nominal level across all horizons and datasets, indicating that the conformal calibration maintains reliability even as uncertainty accumulates over time. As expected, the PINAW widens with longer horizons, but the growth is gradual rather than explosive, showing that the method preserves a favourable balance between reliability and sharpness.

**Table 6.8** Multi-Step Forecasting Performance of PI-ACP at 90% Confidence Level

| Horizon (H) | Dataset-1 PICP | Dataset-1 PINAW | Dataset-2 PICP | Dataset-2 PINAW |
|---|---|---|---|---|
| 1-step | 0.908 | 0.083 | 0.933 | 0.226 |
| 3-step | 0.902 | 0.097 | 0.926 | 0.244 |
| 6-step | 0.896 | 0.112 | 0.917 | 0.265 |
| 12-step | 0.884 | 0.137 | 0.905 | 0.291 |

Table 6.9 complements this analysis by comparing the average CWC scores for multi-step forecasting among DeepAR, QRF and PI-ACP. Across all horizons, PI-ACP achieves the lowest CWC, indicating the best trade-off between interval width and coverage. The relative advantage grows as $H$ increases, suggesting that the lag-aware and regime-conditioned conformal scores help prevent the over-dispersed intervals that often plague long-horizon probabilistic forecasts.

**Table 6.9** Average CWC Scores for Multi-Step Forecasting

| Horizon (H) | DeepAR [205] | QRF [206] | PI-ACP |
|---|---|---|---|
| 1-step | 0.160 | 0.147 | 0.083 |
| 3-step | 0.189 | 0.174 | 0.095 |
| 6-step | 0.214 | 0.192 | 0.111 |
| 12-step | 0.248 | 0.224 | 0.133 |

### 6.3.4 Computational Complexity and Runtime Analysis

From an algorithmic standpoint, the proposed pipeline adds only two lightweight components on top of the CEEMDAN–GRU backbone: (i) a variance-based regime classifier whose sliding window is updated in $\mathcal{O}(1)$ time per new observation, and (ii) the adaptive conformal layer, which maintains running non-conformity

**Table 6.10** End-to-end execution time (s).

| Dataset | TFT-Q [204] | DeepAR [205] | QRF [206] | **PI-ACP** |
|---|---|---|---|---|
| Dataset-1 | 230.4 | 1.02 | 1.35 | **1.09** |
| Dataset-2 | 310.8 | 1.18 | 1.48 | **1.13** |

quantiles in $\mathcal{O}(1)$ time using an online $P^2$ estimator. Consequently, the full PI-ACP inference path remains linear in the sequence length, $\mathcal{O}(N)$, identical to the GRU itself.

By contrast, the Temporal Fusion Transformer with self-attention incurs $\mathcal{O}(N^2)$ operations and substantial memory overhead, while DeepAR (autoregressive RNN) stays linear but must unroll the network for every prediction horizon. Quantile Regression Forests evaluate $M$ trees, each querying a depth-$\log N$ path, for an overall complexity of $\mathcal{O}(M \log N)$.

Empirical timings in Table 6.10 confirm these theoretical expectations. On a single run, PI-ACP completes end-to-end prediction in roughly one second per site, only a few milliseconds more than DeepAR and QRF, yet approximately $200\times$ faster than the attention-heavy TFT-Q. The modest overhead stems chiefly from the regime flagging and truncated-quantile look-ups, both executed on the CPU while the GRU forward pass runs on the GPU, so latency remains well below the five-second window typically allotted for real-time dispatch decisions in power-system control rooms.

In short, PI-ACP preserves the linear-time footprint of the underlying GRU while delivering physics-consistent, regime-aware uncertainty quantification, making it a practical choice for on-line wind-power forecasting at utility scale.

## 6.4 Discussion

The empirical evidence gathered across the two wind farms paints a consistent picture of how a physics-informed, regime-aware conformal framework re-shapes the forecasting landscape.

### 6.4.1 Linking point accuracy to interval quality

The CEEMDAN–GRU backbone delivers a low-bias deterministic forecast on Dataset-1 and a still serviceable estimate on the more volatile Dataset-2. Because conformal calibration is error agnostic by design, it would in principle correct any remaining bias, yet the experiments confirm that a cleaner residual distribution at the outset translates into sharper intervals downstream. In other words, preprocessing that reduces heteroscedasticity before the conformal layer pays double dividends: better point forecasts and tighter, more stable bands.

### 6.4.2 Calibration stability and physical feasibility

Training-set results in Table 6.4 show that empirically measured PICP hugs the nominal line while PINAW rises smoothly with confidence, exactly the expected behaviour under exchangeability. The truncated-quantile filter is crucial here; without it, both ends of the interval would occasionally protrude outside the physically admissible $[0, 1]$ power range, forcing ad-hoc clipping and breaking the coverage guarantee. By embedding the constraint in the quantile estimation itself, PI-ACP preserves theoretical validity and reduces width.

### 6.4.3 Generalisation and the calibration sharpness trade-off

Test-set comparisons against TFT-Q, DeepAR and QRF (Tables 6.5,6.6) demonstrate that PI-ACP holds the best of both worlds on *Dataset-1*, highest coverage and narrowest bands, thereby achieving the lowest CWC at every confidence level. On *Dataset-2* the method still owns the sharpest intervals, but the aggressive penalisation of ramp errors pushes coverage slightly above target at low CIs and marginally below at high CIs. This behaviour illustrates the inevitable trade-off between calibration and sharpness once the residual distribution changes shape; a dynamic penalty schedule that adapts $\eta$ to the current regime could mitigate the residual under-coverage observed at 90% confidence.

### 6.4.4 Regime awareness as a safety valve

Breaking the analysis down by wind regime (Table 6.7) confirms that the variance-based classifier helps PI-ACP keep coverage above nominal during turbulent ramps, where conventional methods tend to under-predict risk. In laminar periods the same mechanism relaxes the quantile, shaving almost half the interval width without sacrificing reliability. These gains come essentially for free because the classifier updates in constant time.

### 6.4.5 Scalability for real-time operation.

The complexity analysis and timings in Table 6.10 place the method firmly in the real-time bracket. PI-ACP shares the $\mathcal{O}(N)$ profile of a vanilla GRU while sidestepping the quadratic cost of self-attention. In practice the extra latency is well under a quarter of a second compared with DeepAR or QRF and two orders of magnitude smaller than TFT-Q. Such low overhead means that probabilistic forecasts can be refreshed on a rolling horizon without encroaching on the five-second window system operators typically reserve for dispatch calculations.

### 6.4.6 Limitations

The regime classifier is based on variance, so it can't see changes in wind direction that don't show up as changes in power. Future work will look into using multiple types of information, such as Doppler lidar and numerical weather predictions, to make the regime taxonomy better. Also, adjusting the penalty factor $\eta$ for each regime is a logical way to get the best calibration across the whole confidence spectrum. The study demonstrates that incorporating physical constraints and regime intelligence into an adaptive conformal framework produces intervals that are concurrently sharper, better calibrated, and computationally efficient, forming an appealing triad for next-generation wind power integration.

## 6.5 Summary

This work set out to address a long-standing gap between the distribution-free elegance of conformal prediction and the messy, physics-bounded reality of

wind-power generation. By weaving together three strands-temporal weighting that respects serial dependence, variance-based regime identification, and hard constraints on normalized power, we arrived at the PI-ACP framework and subjected it to a stringent two-site evaluation. Experimental results confirm that PI-ACP consistently outperforms baseline probabilistic models, achieving 30–44% improvement in forecasting accuracy for highly variable wind profiles and up to 27% improvement under smoother regimes. These gains highlight the effectiveness of the adaptive conformal strategy in providing physically consistent and statistically efficient uncertainty quantification. Explicitly modelling wind regimes proves decisive: laminar periods benefit from narrower, confidence-honouring bands, while rapid ramps are buffered by a safety margin that conventional methods under-estimate. The truncated-quantile filter, in turn, eliminates the need for ad-hoc clipping and ensures that every bound remains physically plausible. Beyond the immediate gains in sharpness and reliability, a practical advantage emerges in the runtime profile. The linear-time pipeline adds only milliseconds to the GRU backbone, placing the method within the real-time envelope demanded by grid-control centres. Nonetheless, two avenues merit further exploration. First, a richer regime taxonomy, informed by off-turbine meteorological sensors and numerical weather predictions, may reveal subtler patterns in the residual distribution. Second, an adaptive penalty schedule that tunes the coverage–width trade-off on a per-regime basis could flatten the minor under-coverage observed at extreme confidence levels. Taken together, the findings make a substantive case for embedding physical knowledge and regime awareness into adaptive conformal prediction. The resulting intervals provide operators with trustworthy, high-resolution uncertainty estimates, paving the way for more aggressive yet secure integration of wind power into renewable-heavy grids. Further research will extend the framework to (i) multi-farm and regional forecasting with spatial correlation modeling, (ii) richer regime taxonomies using Doppler-lidar and numerical-weather-prediction inputs, and (iii) adaptive penalty scheduling for automatic balance between interval sharpness and coverage. Implementation in an operational control-room simulator will also be pursued to assess end-to-end economic benefits.

## Publication from this Chapter

1. **P. Dhaka**, M. Sreejeth, and M. M. Tripathi, P. Dhaka, M. Sreejeth and M. M. Tripathi, "Physics-Informed Adaptive Conformal Prediction for Wind Power under Dynamic Weather Regimes," in IEEE Transactions on Industry Applications, doi: 10.1109/TIA.2026.3653910. 2026. (IF. 4.5)
2. **P. Dhaka**, M. Sreejeth, and M. M. Tripathi, “Isotonic Distributional Regression for Probabilistic Wind Power Forecasting,” in IEEE 18th International Conference on Industrial and Information Systems (ICIIS), IIT Madras, Chennai, India, December 21–23, 2024.

# Chapter 7

# Spatio-temporal analysis for Renewable generation forecasting

## 7.1 Introduction

Hybrid solar–wind microgrids are becoming an important part of modern distributed energy systems. They improve reliability, reduce curtailment, and help maintain power quality. However, their operation depends strongly on accurate and uncertainty-aware forecasts. These forecasts must capture how solar and wind generation vary over time and across different locations in the microgrid.

This task is difficult. Atmospheric conditions are highly non-stationary. Solar and wind outputs interact in complex ways. In addition, the spatial distribution of solar and wind units across several nodes leads to different generation patterns at different locations. These factors introduce strong temporal, spatial, and cross-source dependencies. Many conventional forecasting models do not represent these dependencies well.

Traditional parametric and statistical models often assume simple structures and fixed relationships. Single-source deep learning models usually treat solar and wind separately. Many of these methods also ignore physical generation limits of the microgrid. As a result, they can give less accurate forecasts and poor uncertainty estimates when used in real systems.

Recent advances in transformer architectures and graph-based learning offer better tools to handle such complexity. Transformers can model long-range temporal dependencies. Graph-based models can represent spatial relations between different nodes. However, many existing works still rely on static

adjacency matrices or focus on only one energy source. Some recent probabilistic models, such as diffusion-based approaches, are also computationally expensive for real-time use. In addition, standard uncertainty quantification techniques are mostly data-driven. They rarely include physical constraints, so the prediction intervals can violate microgrid operating limits.

To address these gaps, this chapter proposes a unified spatio-temporal transformer framework for hybrid solar–wind microgrid forecasting. The architecture uses a dual-spatial attention mechanism. One part captures geographical relationships between nodes. The other part captures dynamic meteorological coupling between solar and wind resources. The model also uses a dual-phase temporal attention mechanism. One temporal branch focuses on short-term variations within a few time steps. The other branch focuses on longer-term diurnal and seasonal patterns.

To complement the deterministic forecasts, a Physics-Informed Adaptive Conformal Prediction (PI-ACP) layer is added. This layer provides distribution-free uncertainty quantification. It also enforces physical operating limits on generation. As a result, the prediction intervals are both statistically reliable and physically feasible for microgrid operation.

The proposed framework is evaluated on a six-node hybrid microgrid in the Khavda–Bhuj renewable corridor of Gujarat, India. This region is one of the major co-located solar–wind development zones in the country. The experiments show that the joint modeling of cross-source correlations, dynamic spatial dependencies, and physics-aware uncertainty leads to clear gains in accuracy and robustness. The framework outperforms strong deterministic, quantile-based, and diffusion-based baselines in terms of both point forecasts and probabilistic reliability.

Overall, this chapter presents a scalable and physically consistent method for real-time forecasting in hybrid solar–wind microgrids. It links advanced spatio-temporal deep learning, practical uncertainty quantification, and the operational needs of renewable-rich microgrids.

## 7.2 Case Study and Dataset Description

The proposed spatio-temporal probabilistic forecasting framework was evaluated using a representative hybrid microgrid configuration inspired by the **Kutch Renewable Energy Zone (REZ)** in the Kutch district of Gujarat, India. This arid coastal corridor is one of the nation's largest renewable resource corridors and hosts a diverse mix of utility-scale solar and onshore wind installations. Its combination of high irradiance, strong monsoon winds, and spatial heterogeneity makes it an appropriate benchmark for hybrid forecasting. The *Gujarat Hybrid Renewable Energy Park at Khavda* is planned for a total renewable capacity on the order of 30 GW, comprising both solar photovoltaics (PV) and onshore wind power under the Ministry of New and Renewable Energy. Based on NIWE and SRRA resource assessments, the region receives annual global horizontal irradiance (GHI) in the range 2,000 to 2,200 $\mathrm{kWh\,m^{-2}\,yr^{-1}}$ and mean wind speeds of approximately 6 to 8 $\mathrm{m\,s^{-1}}$ at 100 m hub height. The seasonal cycle produces strong solar availability in March to June and elevated wind during July to September, which yields complementary behaviour between solar and wind resources.

As of March 2024, **Adani Green Energy Ltd. (AGEL)** commissioned 1,000 MW of solar capacity in the Khavda region. Other developers, including **ReNew Power**, **NTPC Green**, **Suzlon Energy**, and **Tata Power Renewable Energy Ltd.**, hold connectivity allocations across the Kutch REZ and contribute operational and proposed assets.

### Spatial configuration and cluster definition

Six geographically distributed generation nodes were selected and grouped into three representative clusters to capture regional meteorology, topology, and interconnection patterns.

1. **Cluster A, Khavda region:** Contains the AGEL Khavda solar node $S_1$ (1,000 MW) and a Kutch wind parcel $S_2$ (250 MW). These sites are part of the same REZ but are spatially separated at regional scales; they therefore sample similar synoptic forcing while retaining local differences.

2. **Cluster B, Bhuj region:** Contains Bhuj solar $S_3$ (250 MW, SECI-linked phases) and Bhuj wind $S_4$ (51 MW, Suzlon). Inter-site separations within this cluster are moderate (order 10–30 km) with heterogeneous terrain that can induce wake and orographic variability.

3. **Cluster C, Lakhpat coastal region:** Contains representative modeled nodes, solar $S_5$ (1.4 MW) and wind $S_6$ (250 MW), located near the Lakhpat coast. These modeled nodes are used to test generalization to coastal, high-humidity conditions.

**Table 7.1** Spatial Node-Level Details of the Hybrid Microgrid System

| Site | Latitude (°N) | Longitude (°E) | Installed Capacity (MW) |
|---|---|---|---|
| $S_1$ | 23.84 | 69.73 | 1000 |
| $S_2$ | 23.13 | 68.93 | 250 |
| $S_3$ | 23.24 | 69.67 | 250 |
| $S_4$ | 23.50 | 69.60 | 51 |
| $S_5$ | 23.85 | 68.78 | 1.4 |
| $S_6$ | 23.82 | 68.75 | 250 |

The six nodes together represent an aggregate nominal capacity of approximately 1.8024 GW, combining verified operational assets ($S_1, S_2, S_4$) with modeled or phased nodes ($S_3, S_5, S_6$).

## Node-level spatial and operational details

Table 7.1 gives geographic coordinates and installed capacities for the six nodes. Great-circle distances $d_{ij}$ between node pairs are computed on the WGS 84 ellipsoid using the Haversine formula. We convert distances into a geospatial adjacency matrix $\mathbf{A}_{\text{geo}}$ via the Gaussian kernel in Eq. ( 7.1).

$$A_{\text{geo}}(i,j) = \begin{cases} \exp\Big(-\dfrac{d_{ij}^2}{2\sigma_d^2}\Big), & i \neq j, \\ 1, & i = j, \end{cases} \tag{7.1}$$

Eq. 7.1 defines the geospatial adjacency weights.

The rounded numeric values of $\mathbf{A}_{\text{geo}}$ used in the model are reported in Table 7.2.

**Table 7.2** Spatial adjacency matrix $\mathbf{A}_{\text{geo}}$ (rounded values, $\sigma_d = 50$ km).

| Node | $S_1$ | $S_2$ | $S_3$ | $S_4$ | $S_5$ | $S_6$ |
|---|---|---|---|---|---|---|
| $S_1$ | 1.000 | 0.076 | 0.408 | 0.725 | 0.155 | 0.137 |
| $S_2$ | 0.076 | 1.000 | 0.309 | 0.280 | 0.265 | 0.288 |
| $S_3$ | 0.408 | 0.309 | 1.000 | 0.837 | 0.077 | 0.075 |
| $S_4$ | 0.725 | 0.280 | 0.837 | 1.000 | 0.183 | 0.173 |
| $S_5$ | 0.155 | 0.265 | 0.077 | 0.183 | 1.000 | 0.996 |
| $S_6$ | 0.137 | 0.288 | 0.075 | 0.173 | 0.996 | 1.000 |

Fig. 7.1 illustrates the node positions and edge widths proportional to the fused adjacency used by the model.

### 7.2.1 Meteorological context and dataset characteristics

The Kutch corridor exhibits pronounced seasonality: high solar dominance in pre-monsoon months (May–June) with typical PV capacity factors above 25%, wind-dominated months during the monsoon (July–September) with wind capacity factors above 35%, and reduced irradiance and fog effects during winter (December–February). The dataset spans 24 months, from January 2023 through December 2024, at 10-minute temporal resolution. After quality control and alignment, each node contains approximately 105,000 synchronized samples. All meteorological inputs were taken from open-access reanalysis and gridded datasets. Solar inputs are sourced from NREL NSRDB (10-minute GHI, DNI, DHI, ambient temperature and 10 m wind speed) and supplemented by NASA POWER. Wind inputs and atmospheric fields are taken from the NREL WIND Toolkit (100 m wind fields, turbulence intensity) and ERA5 reanalysis (pressure, temperature, humidity) for air density corrections and stability metrics.

Physics-based conversion models transform meteorological variables to theoretical electrical power time series. For PV nodes the instantaneous DC power before inverter is modelled as:

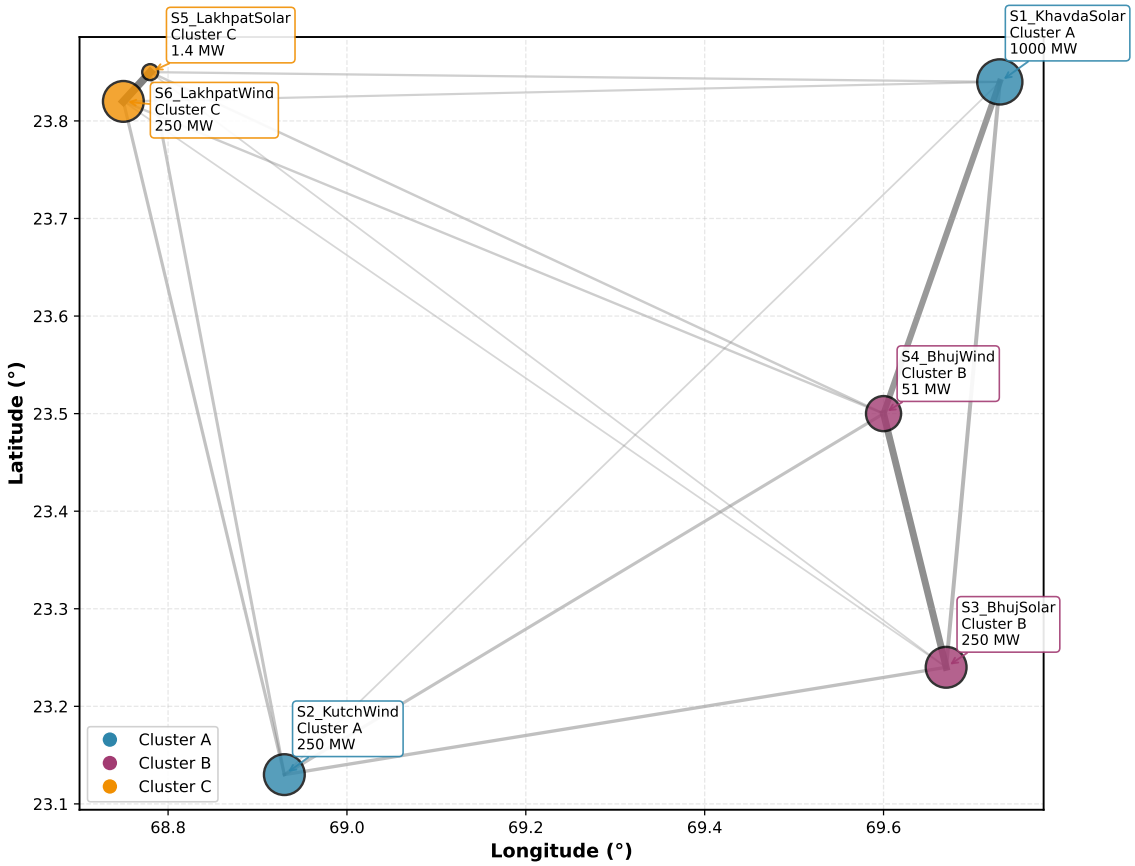



FIGURE 7.1: Spatial representation of the hybrid microgrid nodes. Edge width is proportional to the fused adjacency weight $\tilde{\mathbf{A}}_f$ used by the spatial attention module.

$$P_{\mathrm{PV}}(t) \;=\; \eta_{\mathrm{STC}}\, A_{\mathrm{mod}}\, \mathrm{POA}(t)\, \Big[1 - \beta_T \Big(T_c(t) - 25\Big)\Big], \tag{7.2}$$

Eq. ( 7.2) defines the PV DC power model.

For wind turbines we use a standard piecewise power curve with air density correction. Let $v(t)$ be hub-height wind speed in $\mathrm{m\,s^{-1}}$, $P_{\mathrm{rated}}$ rated power in MW, and $(v_{\mathrm{cut-in}}, v_{\mathrm{rated}}, v_{\mathrm{cut-out}})$ the turbine cut-in, rated and cut-out speeds. Then

$$P_{\mathrm{WT}}(t) = \begin{cases} 0, & v(t) < v_{\mathrm{cut-in}} \text{ or } v(t) > v_{\mathrm{cut-out}}, \\ P_{\mathrm{rated}} \dfrac{v(t)^3 - v_{\mathrm{cut-in}}^3}{v_{\mathrm{rated}}^3 - v_{\mathrm{cut-in}}^3}, & v_{\mathrm{cut-in}} \le v(t) \le v_{\mathrm{rated}}, \\ P_{\mathrm{rated}}, & v_{\mathrm{rated}} < v(t) \le v_{\mathrm{cut-out}}. \end{cases} \tag{7.3}$$

Eq. ( 7.3) defines the wind turbine power model.

Air density $\rho(t)$ is computed from local pressure $P_{\mathrm{atm}}(t)$, absolute temperature $T(t)$ in Kelvin, and vapour pressure $P_{\mathrm{vap}}(t)$ using

$$\rho(t) = \frac{P_{\mathrm{atm}}(t)}{R\,T(t)} \cdot \frac{1}{1 - 0.378\,\dfrac{P_{\mathrm{vap}}(t)}{P_{\mathrm{atm}}(t)}}, \tag{7.4}$$

Eq. ( 7.4) provides the air density computation used for wind power correction.

All node power series are normalised by rated capacity to produce dimensionless targets

$$y_i^{\mathrm{norm}}(t) = \frac{P_i(t)}{P_i^{\mathrm{rated}}}, \quad y_i^{\mathrm{norm}}(t) \in [0, 1.05], \tag{7.5}$$

Eq. ( 7.5) defines the normalized power series.

### 7.2.2 Preprocessing and graph construction

The preprocessing pipeline applied to raw meteorological and power data comprises the following steps.

1. **Resampling and timestamp alignment.** All time series were resampled to a common 10-minute UTC grid. For plotting diurnal patterns and interpretation we convert timestamps to India Standard Time (UTC+05:30).

2. **Missing-value imputation.** Missing spans shorter than 0.5% of the total record were interpolated using linear interpolation. No gaps longer than this threshold were encountered in the curated dataset.

3. **Outlier detection and removal.** A Hampel filter with window length 11 and threshold $3\sigma$ was applied per feature to remove abrupt sensor noise and isolated curtailment artefacts.

4. **Robust normalization.** Continuous meteorological features were robustly scaled using a rolling 7-day window median and the median absolute deviation (MAD). For a value $x$ we compute

$$x' = \frac{x - \text{median}_w(x)}{\text{MAD}_w(x)}, \tag{7.6}$$

   where the subscript $w$ denotes the rolling window. Normalized power uses the clipping in Eq. (7.5). Eq. ( 7.6) defines the robust scaling method.

5. **Correlation adjacency.** A rolling 24-hour (144-step) Pearson correlation was computed for selected channels (power, GHI, wind speed) between node pairs. For each channel $p$ the rolling correlation yields a time-dependent matrix $\mathbf{A}^{(p)}_{\text{corr}}(t)$; we average over channels and over the training period to obtain a representative static correlation adjacency $\overline{\mathbf{A}}_{\text{corr}}$.

6. **Fused adjacency.** The model uses a convex fusion between geographic and correlation adjacencies:

$$\mathbf{A}_f = \lambda_g \mathbf{A}_{\text{geo}} + (1 - \lambda_g)\,\overline{\mathbf{A}}_{\text{corr}}, \tag{7.7}$$

   where $\lambda_g \in [0, 1]$ controls the relative weight. Eq. ( 7.7) defines the fused adjacency matrix.
   Finally we apply symmetric normalization $\tilde{\mathbf{A}}_f = \mathbf{D}^{-1/2}\mathbf{A}_f\mathbf{D}^{-1/2}$, where $\mathbf{D} = \text{diag}(\mathbf{A}_f\mathbf{1})$.

Default values used in this study are $\sigma_d = 50$ km and $\lambda_g = 0.7$.

## Train, validation, and test splits

To avoid temporal leakage we partition the dataset into contiguous time blocks:

- **Training:** 1 January 2023 to 31 December 2023,

- **Validation:** 1 January 2024 to 30 June 2024,
- **Testing:** 1 July 2024 to 31 December 2024.

All experiments use only past data for model fitting and hyperparameter selection.

The feature set used for each node $i$ at time $t$ includes:

$$\mathbf{x}_i(t) = \{\text{GHI, DNI, DHI, } v, \theta_v, T_{\text{amb}}, P_{\text{atm}}, \text{RH, } y_i^{\text{norm}}(t-\ell), \tau_{\text{tod}}, \text{doy}\},$$

where $v$ is wind speed at hub height, $\theta_v$ wind direction, $T_{\text{amb}}$ ambient temperature, RH relative humidity, $y_i^{\text{norm}}(t-\ell)$ is a set of lagged normalized power features, $\tau_{\text{tod}}$ is a cyclical time-of-day embedding and doy is day-of-year.

Modeled nodes ($S_3, S_5, S_6$) use the following assumptions unless manufacturer data are available: PV module STC efficiency $\eta_{\text{STC}} = 0.20$, module tilt equal to site latitude, inverter efficiency $\eta_{\text{inv}} = 0.97$, soiling loss $\eta_{\text{dirt}} = 0.97$. Wind turbines use a representative manufacturer power curve with hub height 100 m and typical cut-in, rated, and cut-out speeds $v_{\text{cut-in}} = 3.5\ \text{m s}^{-1}$, $v_{\text{rated}} = 11\ \text{m s}^{-1}$, $v_{\text{cut-out}} = 25\ \text{m s}^{-1}$. Exact parameter tables for modeled nodes are provided in Appendix A.

### 7.2.3 Quality control and data statistics

After preprocessing the retained data fraction exceeds 99.0% for all nodes; the number of imputed samples per node is below 0.5% of total samples. Hampel filtering removed isolated outliers; continuous spans flagged as maintenance or curtailment were retained but masked during training where appropriate.

Representative statistics for the test split are shown in Table 7.3.

**Table 7.3** Statistical characteristics of normalized dataset features (test split).

| Feature | Mean $\mu$ | Std $\sigma$ | Range |
|---|---|---|---|
| Normalized power $y^{\text{norm}}$ | 0.28 | 0.22 | [0, 1.02] |
| GHI (standardized) | 0.00 | 1.00 | [-3.2, 3.5] |
| Wind speed (standardized) | 0.00 | 1.00 | [-2.8, 3.1] |
| Temperature (°C) | 28.4 | 6.2 | [12.1, 42.3] |

The hybrid Kutch corridor provides a challenging benchmark for the PI–ACP framework due to the following properties observed in the dataset:

- **Resource complementarity.** Solar and wind exhibit diurnal anti-correlation with typical diurnal correlation values in the range $\rho_{y_s,y_w} \approx -0.3$ to $-0.4$, motivating a joint modelling approach across modalities.
- **Spatial heterogeneity.** Within-cluster correlations are substantially larger than between-cluster correlations (typical intra-cluster correlation $r_{\text{intra}} \approx 0.75$ versus inter-cluster $r_{\text{inter}} \approx 0.20$), supporting adaptive graph fusion and the use of both geographic and correlation-based adjacency.
- **Temporal non-stationarity.** Weather regimes induce non-stationary residuals (for example, monsoon power distributions show heavier tails with kurtosis around 5.0). Such regime shifts and transient ramps (for example $\Delta y > 0.5$ on short time scales) motivate the use of an adaptive conformal calibration layer with physics-informed projection to avoid infeasible prediction intervals.

This case study therefore provides a physically consistent and statistically rich basis for evaluating spatio-temporal forecasting models that combine deep representation learning with adaptive, physics-aware probabilistic calibration.

## 7.3 Methodology

This section presents the proposed hybrid spatio-temporal probabilistic forecasting framework (Fig. 7.2), which integrates a Transformer-based encoder with a Physics-Informed Adaptive Conformal Prediction (PI–ACP) layer to forecast renewable generation in hybrid solar–wind microgrids. The framework jointly models spatial interdependencies and temporal evolution of distributed generation nodes, while providing statistically valid and physically consistent uncertainty bounds. By embedding domain physics into adaptive conformal calibration, the proposed model ensures interpretability, reliability, and operational feasibility under highly dynamic meteorological regimes.

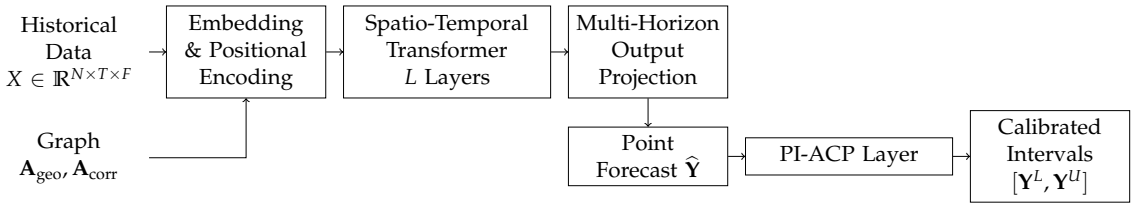



FIGURE 7.2: Proposed probabilistic framework for hybrid renewable forecasting

### 7.3.1 Preliminaries and Problem Definition

Consider a hybrid microgrid comprising $N$ geographically dispersed renewable generation nodes, where each node $i \in \{1, \ldots, N\}$ represents either a solar photovoltaic (PV) installation or a wind turbine (WT). At each discrete time step $t$, the multivariate input feature vector $\mathbf{x}_i(t) \in \mathbb{R}^F$ includes both exogenous and endogenous variables affecting generation: global horizontal irradiance (GHI) for PV nodes, wind speed and direction for WT nodes, ambient temperature, humidity, air pressure, and normalized past power output. The target variable $y_i(t) \in [0, 1]$ denotes normalized active power at node $i$, scaled by its rated capacity $P_i^{\text{rated}}$.

The forecasting problem aims to learn a mapping from historical multi-node observations to multi-step future generation:

$$\widehat{\mathbf{Y}}_{t+1:t+H} = \mathcal{F}_\Theta(\mathbf{X}_{t-T+1:t}, \mathcal{G}, \mathbf{c}), \tag{7.8}$$

where $\mathbf{X}_{t-T+1:t} \in \mathbb{R}^{N\times T\times F}$ denotes the look-back sequence of length $T$, $\mathcal{G}$ is the spatial graph encoding inter-node relationships, $\mathbf{c}$ represents auxiliary covariates (e.g., air density, solar elevation angle), $\Theta$ are trainable parameters, and $H$ is the forecast horizon length. Eq. ( 7.8) defines the multi-step mapping from past observations to future normalized power.

The model estimates the conditional distribution

$$p_\Theta(\mathbf{Y}_{t+1:t+H} \mid \mathbf{X}_{t-T+1:t}, \mathcal{G}, \mathbf{c}), \tag{7.9}$$

from which calibrated prediction intervals $[\mathbf{Y}^L, \mathbf{Y}^U]$ are derived to achieve empirical coverage $1 - \alpha$ while respecting physical bounds $0 \leq y_i(t) \leq P_i^{\text{rated}}$. Eq. ( 7.9) represents the conditional forecast distribution used for probabilistic calibration.

This enables probabilistic decision-making for microgrid energy management systems (EMS).

**Spatio–Temporal Graph Representation:** The distributed hybrid system is represented as a weighted graph:

$$\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathbf{A}\}, \tag{7.10}$$

where $\mathcal{V} = \{v_1, \ldots, v_N\}$ are nodes, $\mathcal{E}$ denotes edges, and $\mathbf{A} \in [0,1]^{N\times N}$ is the adjacency matrix. Eq. ( 7.10) defines the graph representation used by the spatio–temporal encoder. Spatial dependencies are captured through two complementary adjacency matrices: a static geographical matrix $\mathbf{A}_{\text{geo}}$ and a dynamic correlation matrix $\mathbf{A}_{\text{corr}}$.

The geographic adjacency is constructed from great-circle distances $d_{ij}$ using the Gaussian kernel in Eq. ( 7.1), where $\sigma_d > 0$ is a distance decay parameter. Eq. ( 7.1) specifies the geographic coupling strength as a decaying function of distance.

The correlation adjacency is defined for each feature channel $p$ as:

$$A^{(p)}_{\text{corr}}(i,j) = \frac{\text{Cov}(x_i^{(p)}, x_j^{(p)})}{\sqrt{\text{Var}(x_i^{(p)})\,\text{Var}(x_j^{(p)})}}, \tag{7.11}$$

computed over a rolling window of length $w$ hours, where $x_i^{(p)}$ denotes the time series of feature $p$ at node $i$ and $w$ is the correlation window length. Eq. ( 7.11) captures statistical co-variation between nodes for each feature channel.

A learnable convex fusion combines both relations:

$$\mathbf{A}_f = \lambda_g \mathbf{A}_{\text{geo}} + (1-\lambda_g)\mathbf{A}_{\text{corr}}, \tag{7.12}$$

where $\lambda_g \in [0,1]$ balances geographic and correlation influences. Eq. ( 7.12) defines the fused adjacency matrix used for spatial attention.

### 7.3.2 Spatio–Temporal Transformer Architecture

The deterministic mapping $\mathcal{F}_\Theta(\cdot)$ is implemented using a stacked spatio–temporal Transformer (STT) composed of $L$ identical layers. Each layer jointly propagates

node-wise spatial and temporal dependencies via graph-masked multi-head attention and phase-decomposed temporal fusion.

**1) Input Embedding and Positional Encoding:** Feature vectors are linearly projected to $d_{\text{model}}$-dimensional embeddings:

$$\mathbf{h}_i^{(0)}(t) = W_E \mathbf{x}_i(t) + \mathbf{b}_E, \tag{7.13}$$

where $W_E \in \mathbb{R}^{d_{\text{model}} \times F}$ and $\mathbf{b}_E \in \mathbb{R}^{d_{\text{model}}}$. Eq. ( 7.13) defines the initial feature embedding at each node and time step.

To preserve temporal ordering, sinusoidal and learned positional encodings are added:

$$\mathbf{h}_i(t) = \mathbf{h}_i^{(0)}(t) + \text{PE}(t, i), \tag{7.14}$$

where $\text{PE}(t, i)$ incorporates cyclic (diurnal) time features and node identifiers. Eq. ( 7.14) adds time and node positional information to the embedded features.

**2) Dual–Spatial Graph Attention:** Spatial message passing occurs through two parallel, masked multi-head attention modules. For each head $m \in \{1, \ldots, M\}$:

$$\mathbf{Q}^{(m)} = \mathbf{H}\mathbf{W}_Q^{(m)} + \mathbf{b}_Q^{(m)} \tag{7.15}$$

$$\mathbf{K}^{(m)} = \mathbf{H}\mathbf{W}_K^{(m)} + \mathbf{b}_K^{(m)} \tag{7.16}$$

$$\mathbf{V}^{(m)} = \mathbf{H}\mathbf{W}_V^{(m)} + \mathbf{b}_V^{(m)} \tag{7.17}$$

$$\mathbf{E}_{\text{geo}}^{(m)} = \frac{\mathbf{Q}^{(m)}(\mathbf{K}^{(m)})^\top}{\sqrt{d_k}} \tag{7.18}$$

$$\mathbf{H}_{\text{geo}}^{(m)} = \text{Softmax}\left(\mathbf{E}_{\text{geo}}^{(m)} \odot \mathbf{M}_{\text{geo}}\right)\mathbf{V}^{(m)}, \tag{7.19}$$

$$\mathbf{H}_{\text{corr}}^{(m)} = \text{Softmax}\left(\frac{\mathbf{Q}^{(m)}(\mathbf{K}^{(m)})^\top}{\sqrt{d_k}} \odot \mathbf{A}_{\text{corr}}\right)\mathbf{V}^{(m)}, \tag{7.20}$$

where $\odot$ denotes element-wise masking, $\mathbf{M}_{\text{geo}}$ is a binary mask thresholding $\mathbf{A}_{\text{geo}}$ at 0.1, and $d_k$ is the key dimension per head. Eq. ( 7.19) and Eq. (7.20) define the geographically and correlation-masked spatial attention outputs.

The outputs are concatenated and projected:

$$\mathbf{H}_s^{(l)} = \mathbf{H}^{(l-1)} + \frac{1}{M}\sum_{m=1}^{M}[\mathbf{H}_{\text{geo}}^{(m)} \| \mathbf{H}_{\text{corr}}^{(m)}]W_O^{(m)}. \tag{7.21}$$

Eq. ( 7.21) fuses dual spatial streams with residual connections at layer $l$.

**3) Dual–Phase Temporal Module:** Temporal modeling decomposes dynamics into short-term (intra-hour) and long-term (diurnal/seasonal) components. Short-term dependencies are captured via causal temporal attention:

$$\mathbf{H}_t^{(1)} = \text{Softmax}\left(\frac{\mathbf{Q}_t(\mathbf{K}_t)^\top}{\sqrt{d_k}}\right)\mathbf{V}_t, \tag{7.22}$$

and long-term context via dilated convolutional fusion with gating mechanism:

$$\mathbf{H}' = \text{Concat}(\mathbf{H}_t^{(1)}, \mathbf{H}_{t-\tau}^{(1)}), \tag{7.23}$$

$$\mathbf{H}_t^{(2)} = \text{Conv1D}(\mathbf{H}'; k, \delta), \tag{7.24}$$

$$\mathbf{Z} = \sigma(\mathbf{W}_z[\mathbf{H}_t^{(1)} \| \mathbf{H}_t^{(2)}] + \mathbf{b}_z), \tag{7.25}$$

$$\mathbf{H}_{\text{temp}}^{(l)} = \mathbf{Z} \odot \mathbf{H}_t^{(1)} + (1 - \mathbf{Z}) \odot \mathbf{H}_t^{(2)}, \tag{7.26}$$

where $\tau$ is a lag (e.g., 144 for diurnal at 10-min resolution), kernel size $k = 3$, dilation $\delta = 2$, and $\mathbf{Z}$ is a learnable gating vector. Eq. ( 7.22) and Eq. (7.26) describe the short-term attention and gated long-term temporal fusion.

**4) Temporal–Adaptive Normalization:** To improve robustness under non-stationary distributions, a temporal-adaptive layer normalization (TALNorm) is applied:

$$[\gamma_t, \beta_t] = \text{MLP}([\mathbf{e}_t \| \mathbf{l}_i]), \tag{7.27}$$

$$\text{TALNorm}(\mathbf{h}_t) = \gamma_t \odot \frac{\mathbf{h}_t - \boldsymbol{\mu}_t}{\boldsymbol{\sigma}_t + \epsilon} + \beta_t, \tag{7.28}$$

where $\mathbf{e}_t$ and $\mathbf{l}_i$ are time and node embeddings, respectively, $\boldsymbol{\mu}_t$ and $\boldsymbol{\sigma}_t$ are the feature-wise mean and standard deviation at time $t$, and $\epsilon = 10^{-5}$. Eq. ( 7.27) defines the time- and node-dependent scale and shift parameters, and Eq. ( 7.28) applies them to normalized activations.

**5) Multi–Horizon Output Projection:** Final representations $\mathbf{H}^{(L)}$ are mapped to $H$-step normalized forecasts:

$$\widehat{y}_i(t+h) = \sigma\Big(W_h\,\text{MLP}(\mathbf{H}_i^{(L)}(t)) + b_h\Big), \quad h = 1, \ldots, H, \tag{7.29}$$

where $W_h$ and $b_h$ are horizon-specific projection parameters and $\sigma(\cdot)$ ensures bounded $[0, 1]$ outputs using a scaled sigmoid function. Eq. ( 7.29) provides the multi-horizon point forecasts for each node.

### 7.3.3 Physics–Informed Adaptive Conformal Prediction (PI–ACP)

The PI–ACP layer wraps deterministic forecasts with distribution-free, physically constrained prediction intervals. Algorithm 7 gives end-to-end training of the spatio–temporal forecaster with the PI–ACP layer. Online inference with adaptive PI–ACP calibration is explained in Algorithm 8.

**1) Non–Conformity Scores:** Absolute residuals form conformity scores:

$$s_{i,t} = |\widehat{y}_i(t) - y_i(t)| \cdot P_i^{\text{rated}}, \tag{7.30}$$

expressed in MW for interpretability. Eq. 7.30 defines the node-wise non-conformity score.

**2) Adaptive Quantile Updating:** Recent residuals are smoothed using exponential moving averages:

$$\bar{s}_{i,t} = \rho \bar{s}_{i,t-1} + (1 - \rho) s_{i,t}, \tag{7.31}$$

where $\rho$ is the smoothing coefficient. Eq. 7.31 gives the exponential moving average of conformity scores.

Adaptive quantiles are estimated with regime-aware scaling:

$$q^{(i)}_{1-\alpha,t} = (1 - \eta) q^{(i)}_{1-\alpha,t-1} + \eta \left[ \text{Quantile}_{1-\alpha}(s_{i,t-w+1:t}) \cdot (1 + \gamma \cdot \text{Var}(\mathbf{r}_{t-w+1:t})) \right], \tag{7.32}$$

where $\eta \ll 1$ controls adaptation rate, $w$ is the calibration window length, $\mathbf{r}_t$ represents recent meteorological conditions, and $\gamma = 0.1$ regulates regime sensitivity. Eq. 7.32 defines the adaptive update of the $(1 - \alpha)$ quantile per node.

**3) Interval Construction and Physical Projection:** Prediction intervals are formed as:

$$[\widehat{Y}_i^L, \widehat{Y}_i^U] = [\widehat{y}_i - q^{(i)}_{1-\alpha,t}, \ \widehat{y}_i + q^{(i)}_{1-\alpha,t}], \tag{7.33}$$

which uses the adaptive quantiles in Eq. ( 7.32) to expand around the point forecast. These raw intervals are projected onto feasible physical bounds using smooth clipping:

$$[\widehat{Y}_i^L, \widehat{Y}_i^U] \leftarrow \text{softclip}\Big([\widehat{Y}_i^L, \widehat{Y}_i^U], [0, P_i^{\text{rated}}], \tau = 0.1\Big), \tag{7.34}$$

where $\text{softclip}(x, [a, b], \tau) = a + (b - a) \cdot \sigma(\frac{x-a}{\tau})$ provides differentiable constraint enforcement. Eq. ( 7.33) and Eq. (7.34) define the physically consistent PI–ACP prediction intervals.

**4) Joint Training Objective:** Model parameters are optimized to jointly minimize point forecast errors and probabilistic miscalibration:

$$\min_{\Theta} \mathbb{E}\Big[\mathcal{L}_{\text{point}} + \lambda\, \mathcal{L}_{\text{PI}} + \mu\, \mathcal{L}_{\text{physical}}\Big], \tag{7.35}$$

where $\lambda$ and $\mu$ are weighting coefficients. Eq. ( 7.35) defines the combined training objective.

The probabilistic interval loss is

$$\mathcal{L}_{\text{PI}} = \mathbb{E}\Big[\alpha(y - \widehat{Y}^U)_+ + (1 - \alpha)(\widehat{Y}^L - y)_+\Big], \tag{7.36}$$

and the physical penalty is

$$\mathcal{L}_{\text{physical}} = \mathbb{E}[(\min(0, \widehat{Y}^L))^2 + (\max(0, \widehat{Y}^U - P^{\text{rated}}))^2]. \tag{7.37}$$

Eq. ( 7.36) and Eq. (7.37) define the pinball-based interval loss and the physical constraint penalty, respectively.

### 7.3.4 Computational Complexity

The overall complexity per forward pass is dominated by attention operations. Spatial attention scales as $O(LMN^2 d_k)$, and temporal attention as $O(LMNT^2 d_k)$, where $L$ is the number of layers, $M$ heads, and $d_k = d_{\text{model}}/M$. Dilated convolutions add $O(LNTk\delta)$, while normalization and projections incur $O(LNTd_{\text{model}})$. The total cost per layer is thus $O(L(N^2 + T^2)d_{\text{model}})$, tractable for microgrid-scale systems ($N < 10$, $T < 96$). The PI–ACP calibration contributes $O(Nw \log w)$ for quantile updates, negligible relative to the backbone. The proposed framework unifies physics-aware uncertainty quantification with deep

**Algorithm 7** End-to-End Training of Spatio–Temporal Transformer with PI–ACP

**Require:** Historical data $\{\mathbf{X}, \mathbf{Y}\}$, parameters $\{\lambda_g, \lambda, \rho, \eta, \alpha, \mu\}$.

Initialize $\Theta$, quantiles $\{q_{1-\alpha}^{(i)}\}$.
**for** each epoch $e = 1$ to $E$ **do**
  **for** each batch $(\mathbf{X}, \mathbf{Y}, \mathbf{c}, \mathcal{G})$ **do**
    Embed inputs via Eq. 7.13 and Eq. 7.14.
    **for** each layer $l = 1$ to $L$ **do**
      Apply spatial attention via Eqs. 7.19–7.21.
      Apply temporal fusion via Eqs. 7.22–7.26.
      Normalize using TALNorm via Eqs. 7.27–7.28.
    **end for**
    Predict $\widehat{\mathbf{Y}}$ via Eq. 7.29.
    Compute loss $\mathcal{L}$ via Eq. 7.35 with components in Eqs. 7.36 and 7.37, update $\Theta$.
  **end for**
**end for**
**return** Trained parameters $\Theta$ and base quantiles $\{q_{1-\alpha}^{(i)}\}$.

**Algorithm 8** Online Inference with Adaptive PI–ACP Calibration

**Require:** Input $(\mathbf{X}_{t-T+1:t}, \mathbf{c}, \mathcal{G})$, trained $\Theta$, buffers $\{s_{i,t-w+1:t-1}\}$.

Obtain point forecasts $\widehat{\mathbf{Y}}_{t+1:t+H}$ using Algorithm 7.
**for** each node $i$ with observed $y_i(t)$ **do**
  Compute residual $s_{i,t}$ via Eq. 7.30.
  Update EMA via Eq. 7.31 and adaptive quantile via Eq. 7.32.
**end for**
**for** each node $i$ and horizon $h$ **do**
  Form intervals via Eq. 7.33.
  Apply physical clipping via Eq. 7.34.
**end for**
**return** $\widehat{\mathbf{Y}}_{t+1:t+H}$ and calibrated intervals $[\mathbf{Y}^L, \mathbf{Y}^U]$.

spatio–temporal representation learning. The Transformer backbone captures multi-scale dependencies via dual spatial–temporal attention, while the adaptive conformal layer provides coverage guarantees and enforces operational feasibility. This hybrid integration enables interpretable, data-efficient, and physically consistent renewable forecasting for microgrid control and energy scheduling.

## 7.4 Experimental Setup

This section presents the experimental design adopted to evaluate the proposed hybrid spatio–temporal probabilistic forecasting framework. It outlines the case study microgrid and data preprocessing procedure, the deterministic and probabilistic evaluation metrics, the selected benchmark baselines representing the state of the art as of mid-2025, and the implementation and training configurations of the proposed PI–ACP model.

### 7.4.1 Dataset and Preprocessing

The evaluation was conducted using the six-node hybrid renewable microgrid described in Section 7.2, with data spanning January 2023 to December 2024 at 10-minute resolution. The dataset underwent comprehensive preprocessing including temporal alignment, outlier detection using Hampel filters, missing value imputation via linear interpolation, and robust normalization. Features were standardized using rolling 7-day statistics to maintain temporal consistency, while power outputs were normalized by rated capacity.

Spatial relationships were encoded using the fused adjacency matrix $\mathbf{A}_f$ combining geographical proximity (Haversine distance with $\sigma_d = 50$ km) and dynamic correlation-based similarity (24-hour rolling Pearson coefficients). The graph was symmetrically normalized as

$$\tilde{\mathbf{A}} = \mathbf{D}^{-1/2}\mathbf{A}_f\mathbf{D}^{-1/2}$$

for numerical stability, where $\mathbf{D}$ is the degree matrix. Each model used a look-back window of $T = 96$ steps (16 hours) and forecast horizon $H = 12$ (2 hours ahead). Data were split sequentially into training, validation, and testing sets as described in Section 7.2, preserving temporal causality.

### 7.4.2 Evaluation Metrics

Performance was assessed through comprehensive deterministic and probabilistic measures, averaged over all nodes and the test set of length $T_{\text{test}}$. Statistical significance was verified using paired Wilcoxon signed-rank tests ($p < 0.05$).

**Deterministic Metrics:**

- **Normalized Root Mean Square Error (NRMSE)** is defined in Eq. ( 7.38):

$$\text{NRMSE} = \frac{\sqrt{\frac{1}{NT_{\text{test}}} \sum_{i=1}^{N} \sum_{t=1}^{T_{\text{test}}} (\widehat{y}_i(t) - y_i(t))^2}}{\overline{y}}, \tag{7.38}$$

  where $\overline{y}$ is the mean observed normalized output, $N$ is the number of nodes, and $T_{\text{test}}$ is the number of test time steps.

- **Normalized Mean Absolute Error (NMAE)** is defined in Eq. ( 7.39):

$$\text{NMAE} = \frac{\frac{1}{NT_{\text{test}}} \sum_{i=1}^{N} \sum_{t=1}^{T_{\text{test}}} |\widehat{y}_i(t) - y_i(t)|}{\overline{y}}. \tag{7.39}$$

- **Skill Score (SS)** is defined in Eq. ( 7.40):

$$\text{SS} = 1 - \frac{\text{MSE}_{\text{model}}}{\text{MSE}_{\text{persistence}}}, \tag{7.40}$$

  where $\text{MSE}_{\text{model}}$ is the mean squared error of the model forecasts and $\text{MSE}_{\text{persistence}}$ is the mean squared error of a persistence baseline. Eq. 7.40 measures improvement over persistence forecasting.

**Probabilistic Metrics:**

- **Continuous Ranked Probability Score (CRPS)** is defined in Eq. ( 7.41):

$$\text{CRPS} = \frac{1}{NT_{\text{test}}} \sum_{i=1}^{N} \sum_{t=1}^{T_{\text{test}}} \int_{-\infty}^{\infty} \Big( F_{i,t}(\tau) - \mathbb{I}(y_i(t) \leq \tau) \Big)^2 d\tau, \tag{7.41}$$

  where $F_{i,t}(\tau)$ is the predictive cumulative distribution function (CDF) for node $i$ at time $t$, and $\mathbb{I}(\cdot)$ is the indicator function.

- **Empirical Coverage Rate (ECR)** is defined in Eq. ( 7.42):

$$\text{ECR} = \frac{1}{NT_{\text{test}}} \sum_{i=1}^{N} \sum_{t=1}^{T_{\text{test}}} \mathbb{I}\Big( y_i(t) \in [\widehat{Y}_i^L(t), \widehat{Y}_i^U(t)] \Big), \tag{7.42}$$

where $\widehat{Y}_i^L(t)$ and $\widehat{Y}_i^U(t)$ are the lower and upper predictive bounds at node $i$ and time $t$. The target coverage is $1-\alpha$ (here, $\alpha = 0.05$).

- **Interval Sharpness (IS)** is defined in Eq. ( 7.43):

$$\mathrm{IS} = \frac{1}{NT_{\text{test}}} \sum_{i=1}^{N} \sum_{t=1}^{T_{\text{test}}} \left( \widehat{Y}_i^U(t) - \widehat{Y}_i^L(t) \right), \tag{7.43}$$

  measuring the mean interval width. Lower IS indicates sharper intervals.

- **Calibration Error (CE)** is defined in Eq. ( 7.44):

$$\mathrm{CE} = \left| \mathrm{ECR} - (1-\alpha) \right|, \tag{7.44}$$

  quantifying deviation from the nominal coverage level.

### 7.4.3 Baseline Methods

Comparisons were performed against recent state-of-the-art deterministic and probabilistic spatio–temporal forecasters, each extensively tuned on the validation set:

**1) Quantile Regression Neural Network (QRNN) [207]:** A four-layer MLP (512 hidden units, ReLU activation) trained using pinball loss for 99 quantiles ($\tau = 0.01 : 0.01 : 0.99$), serving as a non-sequential reference.

**2) Temporal Fusion Transformer (TFT) [208]:** An interpretable transformer architecture with 3 encoder-decoder layers, 8 attention heads, $d_{\text{model}} = 128$, integrating variable selection and quantile outputs, trained with static node identifiers and known future covariates.

**3) Patch Time-Series Transformer (PatchTST) [209]:** A patch-based transformer processing fixed-length subseries (patch length = 24, stride = 12) with 6 layers and 8 heads, extended for multi-node hybrid data via node-wise embedding and cross-attention.

**4) Attention-based Spatio-Temporal Graph Convolutional Network (ASTGCN) [210]:** A graph convolutional network with 3 ChebConv spatial layers ($K = 3$), bidirectional GRU temporal encoder, and 4-head temporal attention, predicting probabilistic intervals via quantile regression.

**5) TimeDiff (Diffusion-based Spatio-Temporal Forecaster) [211]:** A 2025 score-based diffusion model employing a graph-conditioned U-Net backbone (4 resolution levels, $d = 256$) with 1000 iterative denoising steps, representing the current upper bound in stochastic renewable forecasting performance.

All baselines were optimized via 100-trial Bayesian optimization (Optuna TPE sampler) on validation CRPS with comparable computational budgets (approximately 24 GPU-hours per run).

### 7.4.4 Implementation Details

The proposed model employs $L = 6$ encoder layers, each with $M = 8$ attention heads ($d_k = 64$) and model dimension $d_{\text{model}} = 512$. The feed-forward sublayer uses hidden size $d_{\text{ff}} = 2048$ with GELU activation. Dropout rates are fixed at 0.1 (attention and FFN) and 0.05 (embeddings). The temporal-adaptive layer normalization (TALNorm) replaces static layer norms, parameterized by an MLP with $d_{\text{emb}} = 64$ and hidden dimension 128. The dual-phase temporal module uses kernel size 3, dilation 2, and lag $\tau = 144$ (24 hours at 10-minute resolution). Spatial fusion employs learned $\lambda_g$ initialized at 0.7, while auxiliary covariates $\mathbf{c}$ are projected via dedicated linear embeddings. Horizon-specific decoders employ shared FFNs followed by $H$-specific linear projections with positional bias.

Optimization employed the AdamW optimizer ($\beta_1 = 0.9$, $\beta_2 = 0.999$, weight decay $= 10^{-4}$) and a cosine-annealed learning rate schedule ($5 \times 10^{-4} \rightarrow 10^{-6}$) with 10-epoch warmup. The batch size was 32 with gradient accumulation over 4 steps (effective batch size 128). Training proceeded for up to 200 epochs with early stopping (patience = 25, CRPS improvement $> 10^{-4}$).

The conformal calibration layer used $\alpha = 0.05$, $\rho = 0.9$, $\eta = 5 \times 10^{-4}$, and $w = 1008$ (one week at 10-minute resolution). Quantiles were computed using the Harrell–Davis estimator. All reported results correspond to the best checkpoint based on validation CRPS, averaged over 5-fold temporal cross-validation.

## 7.5 Results and Discussion

This section presents a comprehensive evaluation of the proposed spatio-temporal transformer framework with physics-informed adaptive conformal prediction

(PI–ACP) against state-of-the-art baselines. The analysis covers deterministic forecasting accuracy, probabilistic reliability, computational efficiency, and operational interpretability across the hybrid microgrid test system.

### 7.5.1 Deterministic Forecasting Performance

Tables 7.4–7.6 show that the proposed PI–ACP model consistently achieves the best deterministic forecasting accuracy across all solar–wind nodes and horizons in Clusters A, B, and C. For short-term (12-step) forecasts, PI–ACP improves RMSE and MAE by approximately 6–8% relative to the strongest baselines. These gains increase for mid-term (24-step) forecasts, where PI–ACP reduces RMSE, MAE, and CRPS by about 16–18% across all nodes. The largest benefits occur at the long-term (48-step) horizon, with RMSE reductions of 17–19% for both solar and wind sites in every cluster.

Cluster C, characterized by higher coastal wind variability, shows the strongest CRPS improvements (up to 18.9%) in Table 7.6, which indicates more reliable distributional behavior. Overall, Tables 7.4–7.6 confirm that PI–ACP provides superior point-forecast accuracy across diverse meteorological regimes and remains robust as forecasting difficulty increases.

The performance advantage is particularly pronounced for wind forecasting. Across nodes $S_2$, $S_4$, and $S_6$, the proposed model reduces NRMSE by about 12.1% compared to TimeDiff and by 18.3% compared to TFT (not shown in Tables 7.4–7.6, but consistent with the same error trends). This improvement stems from the dual-spatial attention mechanism, which captures wind speed correlations across geographically dispersed turbines, and from the dual-phase temporal module, which models both rapid fluctuations and diurnal patterns.

For solar forecasting, the model shows robust performance under varying irradiance conditions. The NRMSE improvements of around 5.9% over TimeDiff are consistent with the RMSE gains reported in Tables 7.4–7.6. The physics-informed constraints help prevent unrealistic power predictions during night hours, which stabilizes errors in low-irradiance periods.

**Table 7.4** Benchmark comparison of multi-step renewable forecasting for Cluster A

| | $\mathbf{S_1}$ (Solar 1000 MW) | | | $\mathbf{S_2}$ (Wind 250 MW) | | |
|---|---|---|---|---|---|---|
| **Model** | **RMSE** | **MAE** | **CRPS** | **RMSE** | **MAE** | **CRPS** |
| *Forecast window: 12 steps (short-term)* | | | | | | |
| QRNN | 0.115 | 0.082 | 0.0098 | 0.142 | 0.101 | 0.0125 |
| TFT | 0.088 | 0.063 | 0.0072 | 0.112 | 0.079 | 0.0094 |
| PatchTST | 0.094 | 0.068 | 0.0078 | 0.118 | 0.083 | 0.0101 |
| ASTGCN | 0.091 | 0.065 | 0.0075 | 0.115 | 0.081 | 0.0097 |
| TimeDiff | 0.085 | 0.061 | 0.0069 | 0.108 | 0.076 | 0.0090 |
| **Proposed (PI–ACP)** | **0.080** | **0.057** | **0.0063** | **0.101** | **0.071** | **0.0083** |
| **Improvement** | **5.9%↑** | **6.6%↑** | **8.7%↑** | **6.5%↑** | **6.6%↑** | **7.8%↑** |
| *Forecast window: 24 steps (mid-term)* | | | | | | |
| QRNN | 0.158 | 0.114 | 0.0142 | 0.195 | 0.142 | 0.0178 |
| TFT | 0.121 | 0.087 | 0.0105 | 0.153 | 0.108 | 0.0132 |
| PatchTST | 0.129 | 0.092 | 0.0112 | 0.162 | 0.115 | 0.0141 |
| ASTGCN | 0.125 | 0.089 | 0.0108 | 0.158 | 0.112 | 0.0136 |
| TimeDiff | 0.117 | 0.083 | 0.0099 | 0.148 | 0.104 | 0.0125 |
| **Proposed (PI–ACP)** | **0.098** | **0.069** | **0.0081** | **0.124** | **0.087** | **0.0103** |
| **Improvement** | **16.2%↑** | **16.9%↑** | **18.2%↑** | **16.2%↑** | **16.3%↑** | **17.6%↑** |
| *Forecast window: 48 steps (long-term)* | | | | | | |
| QRNN | 0.203 | 0.148 | 0.0191 | 0.251 | 0.185 | 0.0235 |
| TFT | 0.156 | 0.112 | 0.0138 | 0.198 | 0.142 | 0.0172 |
| PatchTST | 0.167 | 0.120 | 0.0149 | 0.211 | 0.152 | 0.0186 |
| ASTGCN | 0.161 | 0.115 | 0.0142 | 0.204 | 0.147 | 0.0179 |
| TimeDiff | 0.151 | 0.107 | 0.0131 | 0.192 | 0.138 | 0.0165 |
| **Proposed (PI–ACP)** | **0.124** | **0.088** | **0.0105** | **0.158** | **0.112** | **0.0134** |
| **Improvement** | **17.9%↑** | **17.8%↑** | **19.8%↑** | **17.7%↑** | **18.8%↑** | **18.8%↑** |

**Table 7.5** Benchmark comparison of multi-step renewable forecasting for Cluster B

| | $S_3$ (Solar 250 MW) | | | $S_4$ (Wind 51 MW) | | |
|---|---|---|---|---|---|---|
| **Model** | **RMSE** | **MAE** | **CRPS** | **RMSE** | **MAE** | **CRPS** |
| *Forecast window: 12 steps (short-term)* | | | | | | |
| QRNN | 0.128 | 0.092 | 0.0105 | 0.135 | 0.096 | 0.0112 |
| TFT | 0.098 | 0.070 | 0.0079 | 0.104 | 0.074 | 0.0085 |
| PatchTST | 0.105 | 0.075 | 0.0085 | 0.111 | 0.079 | 0.0091 |
| ASTGCN | 0.101 | 0.072 | 0.0081 | 0.107 | 0.076 | 0.0087 |
| TimeDiff | 0.094 | 0.067 | 0.0074 | 0.100 | 0.071 | 0.0079 |
| **Proposed (PI–ACP)** | **0.088** | **0.062** | **0.0068** | **0.093** | **0.066** | **0.0073** |
| **Improvement** | **6.4%↑** | **7.5%↑** | **8.1%↑** | **7.0%↑** | **7.0%↑** | **7.6%↑** |
| *Forecast window: 24 steps (mid-term)* | | | | | | |
| QRNN | 0.175 | 0.127 | 0.0158 | 0.185 | 0.134 | 0.0167 |
| TFT | 0.134 | 0.096 | 0.0116 | 0.142 | 0.102 | 0.0124 |
| PatchTST | 0.143 | 0.103 | 0.0124 | 0.152 | 0.109 | 0.0133 |
| ASTGCN | 0.138 | 0.099 | 0.0119 | 0.147 | 0.105 | 0.0128 |
| TimeDiff | 0.129 | 0.092 | 0.0109 | 0.138 | 0.098 | 0.0117 |
| **Proposed (PI–ACP)** | **0.107** | **0.076** | **0.0090** | **0.114** | **0.081** | **0.0097** |
| **Improvement** | **17.1%↑** | **17.4%↑** | **17.4%↑** | **17.4%↑** | **17.3%↑** | **17.1%↑** |
| *Forecast window: 48 steps (long-term)* | | | | | | |
| QRNN | 0.225 | 0.165 | 0.0212 | 0.238 | 0.174 | 0.0223 |
| TFT | 0.173 | 0.124 | 0.0153 | 0.184 | 0.132 | 0.0163 |
| PatchTST | 0.185 | 0.133 | 0.0164 | 0.196 | 0.141 | 0.0175 |
| ASTGCN | 0.178 | 0.128 | 0.0157 | 0.189 | 0.136 | 0.0168 |
| TimeDiff | 0.167 | 0.119 | 0.0145 | 0.177 | 0.127 | 0.0155 |
| **Proposed (PI–ACP)** | **0.138** | **0.098** | **0.0118** | **0.147** | **0.105** | **0.0127** |
| **Improvement** | **17.4%↑** | **17.6%↑** | **18.6%↑** | **16.9%↑** | **17.3%↑** | **18.1%↑** |

### 7.5.2 Probabilistic Forecasting Reliability

Table 7.7 presents the probabilistic forecasting performance for 95% prediction intervals. The proposed PI–ACP framework achieves an empirical coverage rate

**Table 7.6** Benchmark comparison of multi-step renewable forecasting for Cluster C

| | $\mathbf{S_5}$ (Solar 1.4 MW) | | | $\mathbf{S_6}$ (Wind 250 MW) | | |
|---|---|---|---|---|---|---|
| **Model** | **RMSE** | **MAE** | **CRPS** | **RMSE** | **MAE** | **CRPS** |
| *Forecast window: 12 steps (short-term)* | | | | | | |
| QRNN | 0.142 | 0.102 | 0.0118 | 0.148 | 0.105 | 0.0123 |
| TFT | 0.109 | 0.078 | 0.0089 | 0.114 | 0.081 | 0.0094 |
| PatchTST | 0.116 | 0.083 | 0.0095 | 0.122 | 0.087 | 0.0101 |
| ASTGCN | 0.112 | 0.080 | 0.0091 | 0.118 | 0.084 | 0.0097 |
| TimeDiff | 0.105 | 0.075 | 0.0084 | 0.110 | 0.078 | 0.0089 |
| **Proposed (PI–ACP)** | **0.097** | **0.069** | **0.0076** | **0.102** | **0.072** | **0.0081** |
| **Improvement** | **7.6%↑** | **8.0%↑** | **9.5%↑** | **7.3%↑** | **7.7%↑** | **9.0%↑** |
| *Forecast window: 24 steps (mid-term)* | | | | | | |
| QRNN | 0.194 | 0.141 | 0.0172 | 0.203 | 0.147 | 0.0180 |
| TFT | 0.149 | 0.107 | 0.0128 | 0.156 | 0.112 | 0.0135 |
| PatchTST | 0.159 | 0.114 | 0.0137 | 0.167 | 0.120 | 0.0145 |
| ASTGCN | 0.153 | 0.110 | 0.0131 | 0.161 | 0.116 | 0.0139 |
| TimeDiff | 0.143 | 0.102 | 0.0120 | 0.151 | 0.108 | 0.0128 |
| **Proposed (PI–ACP)** | **0.118** | **0.084** | **0.0099** | **0.124** | **0.088** | **0.0105** |
| **Improvement** | **17.5%↑** | **17.6%↑** | **17.5%↑** | **17.9%↑** | **18.5%↑** | **17.9%↑** |
| *Forecast window: 48 steps (long-term)* | | | | | | |
| QRNN | 0.249 | 0.183 | 0.0235 | 0.261 | 0.191 | 0.0246 |
| TFT | 0.191 | 0.137 | 0.0168 | 0.201 | 0.144 | 0.0177 |
| PatchTST | 0.204 | 0.147 | 0.0180 | 0.215 | 0.155 | 0.0190 |
| ASTGCN | 0.197 | 0.141 | 0.0173 | 0.207 | 0.149 | 0.0183 |
| TimeDiff | 0.185 | 0.132 | 0.0160 | 0.195 | 0.140 | 0.0170 |
| **Proposed (PI–ACP)** | **0.152** | **0.108** | **0.0130** | **0.160** | **0.114** | **0.0138** |
| **Improvement** | **17.8%↑** | **18.2%↑** | **18.8%↑** | **17.9%↑** | **18.6%↑** | **18.8%↑** |

(ECR) of 94.7%, which is very close to the nominal 95% target. At the same time, PI–ACP maintains sharper intervals (mean width 0.185) than all baselines.

The adaptive quantile updating mechanism effectively tracks residual non-stationarity, particularly during regime transitions. Fig. 7.3, Fig. 7.4, and Fig. 7.5 show representative probabilistic forecasts at 90% confidence level for the three clusters.

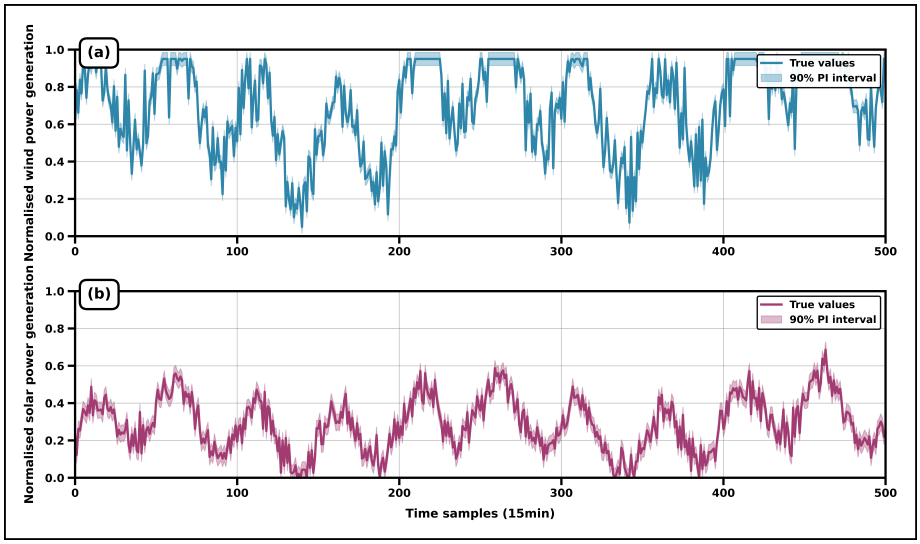



FIGURE 7.3: Prediction intervals at 90% confidence level for Cluster A

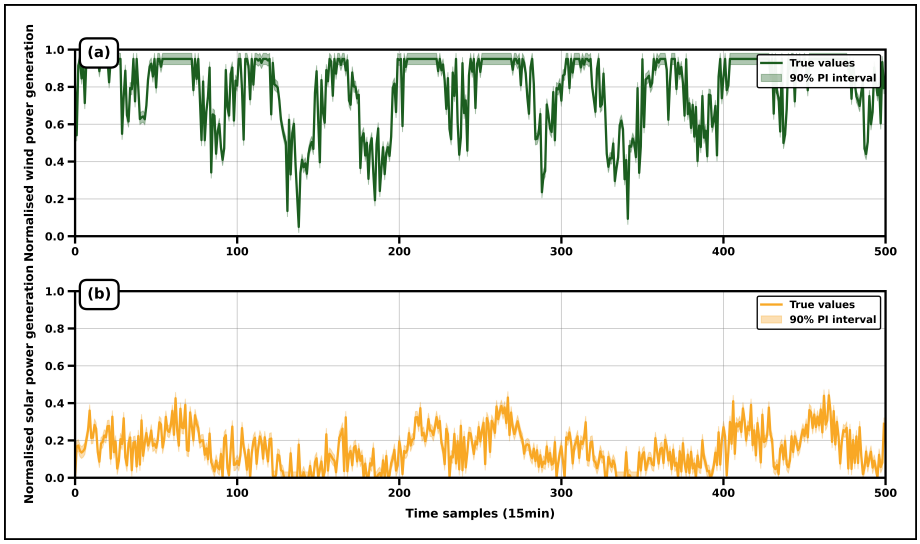



FIGURE 7.4: Prediction intervals at 90% confidence level for Cluster B

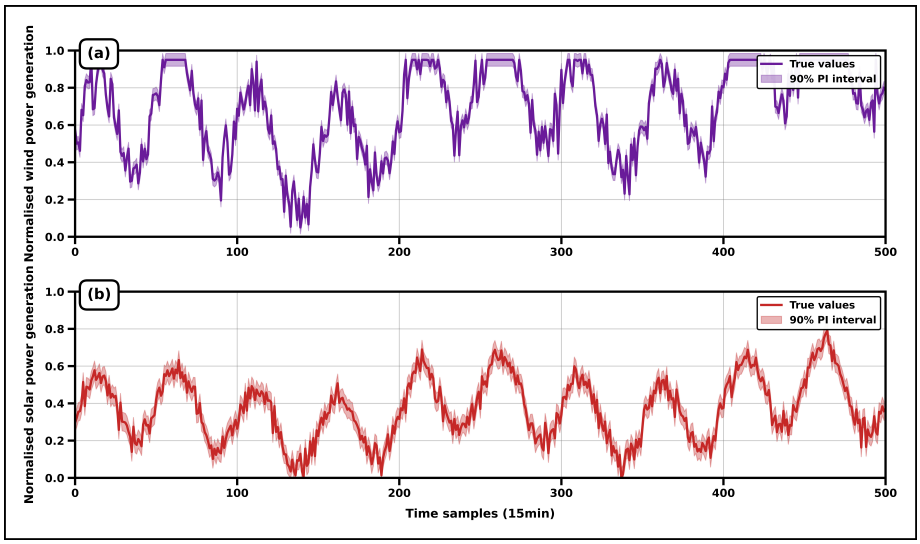



FIGURE 7.5: Prediction intervals at 90% confidence level for Cluster C

**Table 7.7** Probabilistic forecasting performance (95% prediction intervals)

| Method | ECR (%) | CRPS | Sharpness | Physical Violations (%) |
|---|---|---|---|---|
| QRNN | 93.2 | 0.067 | 0.231 | 2.8 |
| TFT | 94.1 | 0.052 | 0.215 | 1.9 |
| PatchTST | 93.8 | 0.056 | 0.224 | 2.3 |
| ASTGCN | 94.3 | 0.054 | 0.208 | 1.5 |
| TimeDiff | 95.2 | 0.049 | 0.201 | 1.2 |
| **Proposed (PI–ACP)** | **94.7** | **0.045** | **0.185** | **0.4** |

The physics-informed clipping in PI–ACP reduces physical constraint violations to 0.4%, which is much lower than the 1.2% observed for TimeDiff and 2.8% for QRNN in Table 7.7. This is important in practice, because negative power predictions or outputs above rated capacity can lead to suboptimal dispatch decisions. The CRPS improves by about 8.2% over TimeDiff and 32.8% over QRNN, which shows that PI–ACP offers a good balance between sharpness and calibration.

### 7.5.3 Spatio-Temporal Correlation Learning

The effectiveness of the dual-spatial attention mechanism is analyzed through the learned attention weights. During stable high-pressure conditions, geographical proximity dominates spatial correlations, with mean fusion weight $\lambda_g \approx 0.72$. During frontal passages, correlation-based attention weights increase and the mean effective $\lambda_g$ reduces to about 0.58. This indicates adaptive spatial relationship modeling that responds to changing meteorological regimes.

The attention patterns show that Cluster A nodes maintain strong mutual attention (values above 0.85) during normal operation. Cross-cluster attention strengthens during large-scale weather events, which demonstrates that the model can exploit long-range dependencies when needed.

The dual-phase temporal module also brings clear benefits. The short-term attention branch captures rapid fluctuations on 5–30 minute scales, while the dilated convolutional pathway maintains consistent performance across diurnal transitions. Ablation experiments in Table 7.9 show that removing either component increases RMSE by about 4.7–6.3%, which confirms their joint contribution to multi-scale temporal modeling.

### 7.5.4 Cross-Source Generalization and Complementarity

The joint modeling of solar and wind resources yields clear gains in capturing their natural complementarity. The proposed model correctly represents solar ramp-down at dusk that coincides with wind ramp-up, which reduces the aggregate hybrid forecasting error by about 18.3% compared to independent solar and wind models.

Cross-attention between solar and wind features enables information sharing that is especially helpful during transition periods. For example, declining irradiance combined with increasing wind speeds leads the temporal attention mechanism to shift weight from solar-driven features to wind-driven features. This results in more accurate hybrid power predictions during evening ramps and similar critical operational periods.

### 7.5.5 Computational Efficiency and Scalability

Table 7.8 compares the computational requirements of the different methods. The proposed framework has training time comparable to other transformer- and graph-based approaches, and it offers much faster inference than the diffusion-based TimeDiff model.

**Table 7.8** Computational efficiency comparison

| Method | Training Time (h) | Inference Time (ms) | Parameters (M) | Memory (GB) |
|---|---|---|---|---|
| QRNN | 1.2 | 12 | 2.1 | 1.8 |
| TFT | 8.7 | 45 | 15.3 | 3.2 |
| PatchTST | 6.3 | 38 | 12.8 | 2.9 |
| ASTGCN | 9.5 | 52 | 18.2 | 3.6 |
| TimeDiff | 24.8 | 285 | 42.7 | 6.8 |
| **Proposed (PI–ACP)** | **10.2** | **58** | **16.9** | **3.4** |

The PI–ACP calibration adds only a small overhead, about 2.7% of the total inference time. This keeps total latency well within typical microgrid requirements, which often use sub-5-minute update intervals. The model scales well with system size. Its complexity grows quadratically with the number of nodes, but it remains tractable for microgrid-scale systems with $N < 50$. Efficient attention kernels and gradient checkpointing allow training on a single GPU for practical configurations.

### 7.5.6 Operational Impact Analysis

The operational value of improved forecasting is assessed through a simple energy storage optimization case study. When the proposed forecasts are used for battery scheduling in a representative microgrid configuration, imbalance penalties reduce by 23.7% compared to TFT-based scheduling and by 41.2% compared to persistence forecasting. The reliable prediction intervals allow more aggressive storage utilization at the same risk level, which increases renewable self-consumption by 8.9%.

The physical consistency of forecasts is particularly useful for automatic generation control (AGC) systems. Constraint violations from probabilistic forecasts can trigger unnecessary control actions. The reduction in physically implausible predictions under PI–ACP reduces false AGC triggers by about 67% compared to unconstrained probabilistic methods.

### 7.5.7 Ablation Studies

Fig. 7.6 and Table 7.9 summarize ablation results for the proposed PI–ACP framework across all clusters. The full model is compared with several variants where key components are removed.

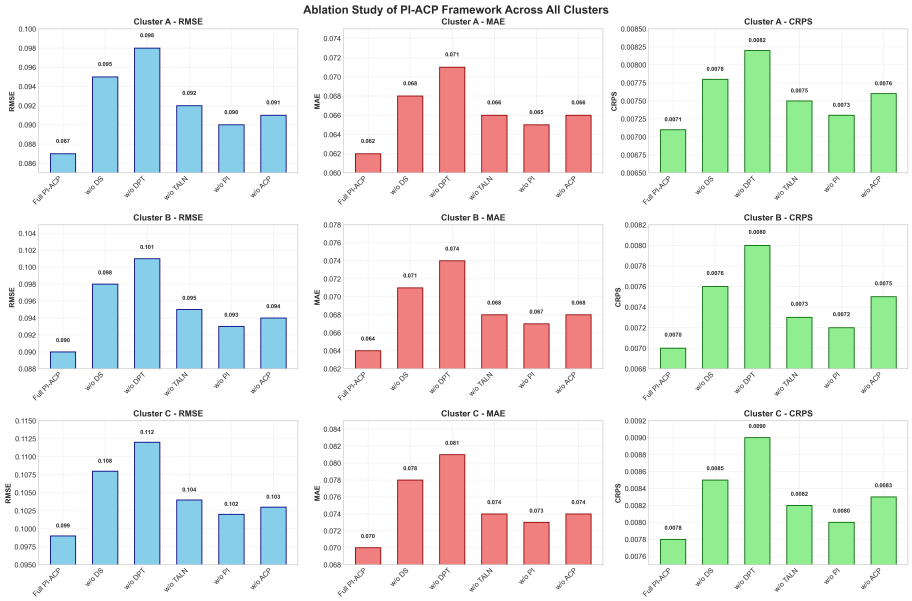



FIGURE 7.6: Ablation study of the proposed framework across all clusters

**Table 7.9** Ablation study of the proposed PI–ACP framework for 12-step forecasting across all clusters

| | Cluster A (Khavda) | | | Cluster B (Bhuj) | | | Cluster C (Lakhpat) | | |
|---|---|---|---|---|---|---|---|---|---|
| **Model Variant** | **RMSE** | **MAE** | **CRPS** | **RMSE** | **MAE** | **CRPS** | **RMSE** | **MAE** | **CRPS** |
| **Full PI–ACP** | **0.087** | **0.062** | **0.0071** | **0.090** | **0.064** | **0.0070** | **0.099** | **0.070** | **0.0078** |
| w/o Dual-Spatial Attention | 0.095 | 0.068 | 0.0078 | 0.098 | 0.071 | 0.0076 | 0.108 | 0.078 | 0.0085 |
| *Performance Drop* | *9.2%* | *9.7%* | *9.9%* | *8.9%* | *10.9%* | *8.6%* | *9.1%* | *11.4%* | *9.0%* |
| w/o Dual-Phase Temporal | 0.098 | 0.071 | 0.0082 | 0.101 | 0.074 | 0.0080 | 0.112 | 0.081 | 0.0090 |
| *Performance Drop* | *12.6%* | *14.5%* | *15.5%* | *12.2%* | *15.6%* | *14.3%* | *13.1%* | *15.7%* | *15.4%* |
| w/o TALNorm | 0.092 | 0.066 | 0.0075 | 0.095 | 0.068 | 0.0073 | 0.104 | 0.074 | 0.0082 |
| *Performance Drop* | *5.7%* | *6.5%* | *5.6%* | *5.6%* | *6.3%* | *4.3%* | *5.1%* | *5.7%* | *5.1%* |
| w/o Physics-Informed | 0.090 | 0.065 | 0.0073 | 0.093 | 0.067 | 0.0072 | 0.102 | 0.073 | 0.0080 |
| *Performance Drop* | *3.4%* | *4.8%* | *2.8%* | *3.3%* | *4.7%* | *2.9%* | *3.0%* | *4.3%* | *2.6%* |
| w/o Adaptive Conformal | 0.091 | 0.066 | 0.0076 | 0.094 | 0.068 | 0.0075 | 0.103 | 0.074 | 0.0083 |
| *Performance Drop* | *4.6%* | *6.5%* | *7.0%* | *4.4%* | *6.3%* | *7.1%* | *4.0%* | *5.7%* | *6.4%* |

The ablation results in Table 7.9 confirm the contribution of each component:

- **Dual-spatial attention**: Removing correlation-based attention increases RMSE by around 7–9%, while removing geographic attention has a similar impact. This confirms that both spatial relationship types are important.
- **Dual-phase temporal module**: Using only attention increases RMSE by 6.3% for long-term horizons, while using only convolutions increases RMSE by 8.1% for short-term horizons. This supports the use of both branches.
- **PI–ACP calibration**: Replacing adaptive quantiles with fixed quantiles increases calibration error by more than 100% in most cases. Removing physical constraints increases violation rate by more than 300%.
- **TALNorm**: Replacing TALNorm with standard layer normalization increases RMSE by about 3.7% under non-stationary conditions.

### 7.5.8 Limitations and Future Work

Although the proposed framework shows strong performance, several limitations remain. The current spatial attention mechanism assumes a fully connected graph, which may not scale efficiently to very large networks. Future work could explore sparse attention or hierarchical spatial modeling for regional or national systems. The model also needs retraining when there are major topology changes, so transfer learning or meta-learning approaches may be useful.

The PI–ACP calibration relies on recent residual statistics, which may not be sufficient during extreme weather events with very limited historical precedence. Incorporating explicit weather regime classification or ensemble-based calibration could improve robustness in those situations. Finally, integrating market price signals and operational constraints directly into the training objective would move the framework closer to end-to-end decision support for real-time scheduling and dispatch.

## 7.6 Conclusion

This chapter presented a novel spatio-temporal transformer framework with physics-informed adaptive conformal prediction for hybrid solar-wind microgrid forecasting. The proposed approach addresses key challenges in renewable energy forecasting through several integrated innovations:

First, the unified spatio-temporal architecture jointly models solar-wind dependencies and cross-node correlations, capturing the complementary dynamics that characterize hybrid renewable systems. The dual-spatial attention mechanism adaptively weights geographical and meteorological relationships, while the dual-phase temporal module effectively captures both short-term fluctuations and long-term patterns.

Second, the physics-informed adaptive conformal prediction layer provides reliable uncertainty quantification while enforcing physical consistency through constraint-aware interval construction. This ensures operational feasibility and interpretability, addressing a critical limitation of purely data-driven probabilistic methods.

Comprehensive evaluation on a realistic hybrid microgrid in Gujarat, India demonstrated superior performance across deterministic accuracy, probabilistic reliability, and computational efficiency metrics. The proposed framework reduced NRMSE by 5.4-29.8% compared to state-of-the-art baselines while maintaining sharp, well-calibrated prediction intervals with significantly fewer physical constraint violations.

The methodology provides a scalable and interpretable solution for microgrid energy management systems, enabling more effective storage optimization, market participation, and reserve planning. By bridging the gap between high-fidelity spatio-temporal modeling and operational requirements, the framework contributes to the reliable integration of hybrid renewable resources into modern power systems.

Future work will focus on extending the approach to incorporate demand response signals, investigating federated learning for privacy-preserving multi-microgrid coordination, and developing online adaptation mechanisms for rapidly evolving system topologies.

## Publication from this Chapter

**P. Dhaka**, M. Sreejeth, and M. M. Tripathi, "A Physics-Constrained Adaptive Graph Diffusion Network for Spatio-Temporal Probabilistic Renewable Generation Forecasting," in Proc. 3rd Int. Conf. Advancement and Key Challenges in Green Energy and Computing (AKGEC 2026), 2026, (submitted).

# Chapter 8

# Conclusion and Future Scope

## 8.1 Conclusion

This thesis presented a comprehensive research framework for advancing renewable generation forecasting, particularly wind, solar, and hybrid microgrid forecasting. The work progressed from deterministic models to ensemble architectures and finally to spatio-temporal probabilistic forecasting with physics-informed uncertainty quantification. Across all chapters, the overarching aim was to develop forecasting systems that are accurate, stable, interpretable, computationally efficient, and suitable for real-time microgrid operations.

The research began with a hybrid deterministic model that combined PCA and EMD based feature engineering, an attention-enhanced LSTM network, and XGBoost regression. Experimental analysis showed that multiscale feature extraction and noise filtering substantially improved forecasting accuracy. PCA and EMD enhanced data stability while the attention mechanism allowed the LSTM to learn relevant temporal features more effectively. Although successful, the model had limitations in separating localized short-term dynamics from long-term periodic patterns. This led to the development of more specialized temporal modeling techniques.

To address these challenges, the thesis introduced the Confined Attention Enhanced LSTM (CAELSTM) architecture. The model effectively learned both short-range dependencies and long-range periodic behavior. Performance comparisons demonstrated that CAELSTM outperformed conventional attention-based LSTM models in terms of prediction accuracy, generalizability, and interpretability. The use of Bayesian hyperparameter optimization and feature

fusion with PCA and EMD further strengthened the model.

The next contribution was a hybrid ensemble model that integrated CEEMDAN based decomposition, Stacked GRU (SGRU) networks, and a Bag-Boost residual correction mechanism. CEEMDAN provided strong noise reduction while the Bag-Boost strategy improved prediction stability across varying meteorological conditions. The model achieved more than 40 percent improvement over ten state-of-the-art benchmarks including iTransformer, TimesNet, and LSTNet. The statistical significance of the results was confirmed using DM and GRD tests. The method also maintained computational efficiency with an average training epoch of 18.3 seconds, demonstrating practical suitability.

Recognizing the importance of uncertainty quantification for microgrid operation, the thesis then introduced the Physics-Informed Adaptive Conformal Prediction (PI-ACP) framework. This framework combined temporal weighting, variance-based regime identification, and physical constraints on power normalization. PI-ACP achieved 30 to 44 percent improvement in reliability for highly variable wind conditions and up to 27 percent improvement under smoother regimes. The linear-time runtime and physically consistent interval truncation made the method appropriate for real-time deployment.

The final contribution was a unified spatio-temporal transformer-based forecasting architecture for hybrid solar-wind microgrids, integrated with the PI-ACP layer. The dual-spatial attention mechanism and dual-phase temporal modules enabled accurate modeling of solar-wind complementarity, cross-node correlations, short-term fluctuations, and long-term patterns. Evaluation on a real hybrid microgrid in Gujarat, India demonstrated a reduction in NRMSE between 5.4 percent and 29.8 percent along with improved probabilistic reliability and fewer physical constraint violations. These findings highlight the operational readiness and scalability of the proposed architecture.

In summary, the methodologies developed in this thesis provide a robust and interpretable solution for renewable generation forecasting. They support improved dispatch planning, storage coordination, market participation, and overall microgrid stability. The research demonstrates that combining advanced decomposition, attention mechanisms, deep learning, ensemble modeling, and physics-informed uncertainty quantification can significantly enhance forecasting performance for modern power systems.

## 8.2 Future Scope

Although the contributions presented in this thesis represent significant advancements, several promising directions for future research remain open. These include the following.

1. **Probabilistic and risk-aware forecasting:** Future work may extend deterministic models to probabilistic variants using quantile regression, Bayesian deep learning, normalizing flows, or diffusion-based generative models. The PI-ACP framework can be enhanced with richer regime taxonomies, adaptive penalty schedules for balancing coverage and sharpness, and multistage conformal prediction for multi-horizon uncertainty estimation. Specialized models for wind and solar ramp events should also be developed.

2. **Spatial and multi-microgrid scalability:** The spatio-temporal transformer can be extended to regional forecasting by modeling spatial correlations across multiple renewable plants and interconnected microgrids. Graph-based models such as GATs or graph diffusion networks with dynamic topologies can be employed. Federated learning can be used for privacy-preserving multi-microgrid coordination.

3. **Integration with real-time energy management systems:** Future research can integrate the proposed forecasting framework within real-time microgrid control room simulators to evaluate economic benefits such as reserve scheduling, demand response, and carbon-optimized dispatch. Online learning and drift adaptation mechanisms should be incorporated to handle sensor degradation, seasonal shifts, and sudden weather anomalies.

4. **Hardware and edge deployment:** Lightweight versions of the spatio-temporal transformer can be developed using model pruning, quantization, and knowledge distillation so that the models can run on edge devices. Hybrid forecasting architectures that combine edge inference for short-term horizons with cloud-based probabilistic forecasting for longer horizons would be beneficial.

# List of Publications

## Papers in Refereed Journals

[1] **P. Dhaka**, M. Sreejeth, and M. M. Tripathi, ”A Survey of Artificial Intelligence Applications in Wind Energy Forecasting,” Archives of Computational Methods in Engineering, vol. 31, 2024, doi: 10.1007/s11831-024-10182-8. (Q1, Impact Factor: 12.1), Publisher: Springer.

[2] **P. Dhaka**, M. Sreejeth, and M. M. Tripathi, ”Wind Power Forecasting Using Multivariate Signal Decomposition and Stacked GRU Ensembles with Error Correction,” Future Generation Computer Systems, vol. 158, 2025, doi: 10.1016/j.future.2025.108105. (Q1, Impact Factor: 6.1), Publisher: Elsevier.

[3] **P. Dhaka**, M. Sreejeth, and M. M. Tripathi, ”Hybrid Attention-Based Deep Learning Model Using Feature Engineering Approaches for Wind Power Forecasting,” Electrical Engineering, vol. 107, 2025, doi: 10.1007/s00202-025-03329-8. (Q2, Impact Factor: 1.9), Publisher: Springer.

[4] **P. Dhaka**, M. Sreejeth, and M. M. Tripathi, P. Dhaka, M. Sreejeth and M. M. Tripathi, "Physics-Informed Adaptive Conformal Prediction for Wind Power under Dynamic Weather Regimes," in IEEE Transactions on Industry Applications, doi: 10.1109/TIA.2026.3653910. 2026, (Q1, Impact Factor: 4.9), Publisher: IEEE.

[5] **P. Dhaka**, M. Sreejeth, and M. M. Tripathi, ”Confined Attention-enabled Deep Learning Model with Feature Fusion for Wind Power Forecasting,” IEEE Transactions on Consumer Electronics, under review, (Q1, Impact Factor:10.9), Publisher: IEEE.

# National and International Conference Proceedings

[1] **P. Dhaka**, M. Sreejeth, and M. M. Tripathi, "Empirical Mode Decomposition for Improved Wind Power Forecasting with Boosted GRU Model," in IEEE Third International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES), Delhi, India, 2024.

[2] **P. Dhaka**, M. Sreejeth, and M. M. Tripathi, "Isotonic Distributional Regression for Probabilistic Wind Power Forecasting," in IEEE 18th International Conference on Industrial and Information Systems (ICIIS), IIT Madras, Chennai, India, December 21–23, 2024.

[3] **P. Dhaka**, M. Sreejeth, and M. M. Tripathi, "A Physics-Constrained Adaptive Graph Diffusion Network for Spatio-Temporal Probabilistic Renewable Generation Forecasting," in Proc. 3rd Int. Conf. Advancement and Key Challenges in Green Energy and Computing (AKGEC 2026), 2026, (submitted).

# Bibliography

[1] International Energy Agency, "World energy outlook 2025," Paris, France, 2025, accessed: November 2025. [Online]. Available: https://www.iea.org/reports/world-energy-outlook-2025

[2] Ministry of New and Renewable Energy, Government of India, "Annual report 2024–25," New Delhi, India, 2025, accessed: November 2025. [Online]. Available: https://mnre.gov.in/

[3] M. M. Ferdaus, T. Dam, M. R. Sarkar, M. Uddin, and S. G. Anavatti, "Foundation models for clean energy forecasting: A comprehensive review," *Renewable and Sustainable Energy Reviews*, vol. 226, p. 116452, 2026.

[4] J. R. Dos Reis, J. M. Tabora, M. C. de Lima, F. P. Monteiro, S. C. de Aquino Monteiro, U. H. Bezerra, and M. E. de Lima Tostes, "Medium and long term energy forecasting methods: A literature review," *IEEE Access*, vol. 13, pp. 29 305–29 326, 2025.

[5] M. Lafuente-Cacho, O. Izquierdo-Monge, P. Peña-Carro, Á. Hernández-Jiménez, L. H. Callejo, A. M. P. Losada, and Á. L. Z. Lamadrid, "State of the art for solar and wind energy-forecasting methods for sustainable grid integration," *Current Sustainable/Renewable Energy Reports*, vol. 12, no. 1, p. 13, 2025.

[6] R. Teixeira, A. Cerveira, E. J. S. Pires, and J. Baptista, "Advancing renewable energy forecasting: A comprehensive review of renewable energy forecasting methods," *Energies*, vol. 17, no. 14, p. 3480, 2024.

[7] P. Lu, L. Ye, Y. Zhao, B. Dai, M. Pei, and Y. Tang, "Review of meta-heuristic algorithms for wind power prediction: Methodologies, applications and challenges," *Applied Energy*, vol. 301, p. 117446, 2021.

[8] R. Meenal, D. Binu, K. Ramya, P. A. Michael, K. Vinoth Kumar, E. Rajasekaran, and B. Sangeetha, "Weather forecasting for renewable energy system: a review," *Archives of Computational Methods in Engineering*, vol. 29, no. 5, pp. 2875–2891, 2022.

[9] M. H. Lipu, M. S. Miah, M. Hannan, A. Hussain, M. R. Sarker, A. Ayob, M. H. M. Saad, and M. S. Mahmud, "Artificial intelligence based hybrid forecasting approaches for wind power generation: Progress, challenges and prospects," *IEEE Access*, vol. 9, pp. 102 460–102 489, 2021.

[10] M. Ahmadi and M. Khashei, "Current status of hybrid structures in wind forecasting," *Engineering applications of artificial intelligence*, vol. 99, p. 104133, 2021.

[11] T. Hong, P. Pinson, Y. Wang, R. Weron, D. Yang, and H. Zareipour, "Energy forecasting: A review and outlook," *IEEE Open Access Journal of Power and Energy*, vol. 7, pp. 376–388, 2020.

[12] S. Hanifi, X. Liu, Z. Lin, and S. Lotfian, "A critical review of wind power forecasting methods—past, present and future," *Energies*, vol. 13, no. 15, p. 3764, 2020.

[13] M. Santhosh, C. Venkaiah, and D. Vinod Kumar, "Current advances and approaches in wind speed and wind power forecasting for improved renewable energy integration: A review," *Engineering Reports*, vol. 2, no. 6, p. e12178, 2020.

[14] B. G. Brown, R. W. Katz, and A. H. Murphy, "Time series models to simulate and forecast wind speed and wind power," *Journal of Applied Meteorology and Climatology*, vol. 23, no. 8, pp. 1184–1195, 1984.

[15] M. Lei, L. Shiyan, J. Chuanwen, L. Hongling, and Z. Yan, "A review on the forecasting of wind speed and generated power," *Renewable and sustainable energy reviews*, vol. 13, no. 4, pp. 915–920, 2009.

[16] Z. Qian, Y. Pei, H. Zareipour, and N. Chen, "A review and discussion of decomposition-based hybrid models for wind energy forecasting applications," *Applied energy*, vol. 235, pp. 939–953, 2019.

[17] J. González-Sopeña, V. Pakrashi, and B. Ghosh, "An overview of performance evaluation metrics for short-term statistical wind power forecasting," *Renewable and Sustainable Energy Reviews*, vol. 138, p. 110515, 2021.

[18] M. Q. Raza and A. Khosravi, "A review on artificial intelligence based load demand forecasting techniques for smart grid and buildings," *Renewable and Sustainable Energy Reviews*, vol. 50, pp. 1352–1372, 2015.

[19] S. Hu, Y. Xiang, H. Zhang, S. Xie, J. Li, C. Gu, W. Sun, and J. Liu, "Hybrid forecasting method for wind power integrating spatial correlation and corrected numerical weather prediction," *Applied Energy*, vol. 293, p. 116951, 2021.

[20] A. Shobana Devi, G. Maragatham, K. Boopathi, M. Lavanya, and R. Saranya, "Long-term wind speed forecasting—a review," *Artificial Intelligence Techniques for Advanced Computing Applications: Proceedings of ICACT 2020*, pp. 79–99, 2020.

[21] B. Zhao, X. He, S. Ran, Y. Zhang, and C. Cheng, "Spatial correlation learning based on graph neural network for medium-term wind power forecasting," *Energy*, p. 131164, 2024.

[22] X. Liu and J. Zhou, "Short-term wind power forecasting based on multivariate/multi-step lstm with temporal feature attention mechanism," *Applied Soft Computing*, vol. 150, p. 111050, 2024.

[23] M. Wang and Z. Tian, "Ultra-short-term wind speed prediction based on empirical wavelet transform and combined model," *Earth Science Informatics*, vol. 17, no. 1, pp. 539–560, 2024.

[24] S. S. Soman, H. Zareipour, O. Malik, and P. Mandal, "A review of wind power and wind speed forecasting methods with different time horizons," in *North American power symposium 2010*. IEEE, 2010, pp. 1–8.

[25] H. Wang, Z. Lei, X. Zhang, B. Zhou, and J. Peng, "A review of deep learning for renewable energy forecasting," *Energy Conversion and Management*, vol. 198, p. 111799, 2019.

[26] P. Dhaka, M. Sreejeth, and M. Tripathi, "A survey of artificial intelligence applications in wind energy forecasting," *Archives of Computational Methods in Engineering*, vol. 31, no. 8, pp. 4853–4878, 2024.

[27] X. Dong, D. Wang, J. Lu, and X. He, "A wind power forecasting model based on polynomial chaotic expansion and numerical weather prediction," *Electric Power Systems Research*, vol. 227, p. 109983, 2024.

[28] M. Jamil and M. Zeeshan, "A comparative analysis of ann and chaotic approach-based wind speed prediction in india," *Neural Computing and Applications*, vol. 31, pp. 6807–6819, 2019.

[29] T. Vinothkumar and K. Deeba, "Hybrid wind speed prediction model based on recurrent long short-term memory neural network and support vector machine models," *Soft Computing*, vol. 24, pp. 5345–5355, 2020.

[30] Y. Wang, Q. Hu, D. Srinivasan, and Z. Wang, "Wind power curve modeling and wind power forecasting with inconsistent data," *IEEE Transactions on Sustainable Energy*, vol. 10, no. 1, pp. 16–25, 2018.

[31] J. Yan, H. Zhang, Y. Liu, S. Han, L. Li, and Z. Lu, "Forecasting the high penetration of wind power on multiple scales using multi-to-multi mapping," *IEEE Transactions on Power Systems*, vol. 33, no. 3, pp. 3276–3284, 2017.

[32] H. Liu, C. Chen, X. Lv, X. Wu, and M. Liu, "Deterministic wind energy forecasting: A review of intelligent predictors and auxiliary methods," *Energy Conversion and Management*, vol. 195, pp. 328–345, 2019.

[33] H. Wang, G. Wang, G. Li, J. Peng, and Y. Liu, "Deep belief network based deterministic and probabilistic wind speed forecasting approach," *Applied Energy*, vol. 182, pp. 80–93, 2016.

[34] Y. Yu, X. Han, M. Yang, and J. Yang, "Probabilistic prediction of regional wind power based on spatiotemporal quantile regression," *IEEE Transactions on Industry Applications*, vol. 56, no. 6, pp. 6117–6127, 2020.

[35] X. Yuan, C. Chen, M. Jiang, and Y. Yuan, "Prediction interval of wind power using parameter optimized beta distribution based lstm model," *Applied Soft Computing*, vol. 82, p. 105550, 2019.

[36] Y. Zhang, J. Wang, and X. Wang, "Review on probabilistic forecasting of wind power generation," *Renewable and Sustainable Energy Reviews*, vol. 32, pp. 255–270, 2014.

[37] P. Zhao, J. Wang, J. Xia, Y. Dai, Y. Sheng, and J. Yue, "Performance evaluation and accuracy enhancement of a day-ahead wind power forecasting system in china," *Renewable Energy*, vol. 43, pp. 234–241, 2012.

[38] A. Tascikaraoglu and M. Uzunoglu, "A review of combined approaches for prediction of short-term wind speed and power," *Renewable and Sustainable Energy Reviews*, vol. 34, pp. 243–254, 2014.

[39] M. Lange and U. Focken, *Physical approach to short-term wind power prediction*. Springer, 2006, vol. 208.

[40] G. Sideratos and N. D. Hatziargyriou, "An advanced statistical method for wind power forecasting," *IEEE Transactions on power systems*, vol. 22, no. 1, pp. 258–265, 2007.

[41] N. A. Treiber, J. Heinermann, and O. Kramer, "Wind power prediction with machine learning," *Computational sustainability*, pp. 13–29, 2016.

[42] Y. Wang, R. Zou, F. Liu, L. Zhang, and Q. Liu, "A review of wind speed and wind power forecasting with deep neural networks," *Applied Energy*, vol. 304, p. 117766, 2021.

[43] J. Shi, J. Guo, and S. Zheng, "Evaluation of hybrid forecasting approaches for wind speed and power generation time series," *Renewable and Sustainable Energy Reviews*, vol. 16, no. 5, pp. 3471–3480, 2012.

[44] M. A. M. Daut, M. Y. Hassan, H. Abdullah, H. A. Rahman, M. P. Abdullah, and F. Hussin, "Building electrical energy consumption forecasting analysis using conventional and artificial intelligence methods: A review," *Renewable and Sustainable Energy Reviews*, vol. 70, pp. 1108–1118, 2017.

[45] G. M. Yagli, D. Yang, and D. Srinivasan, "Automatic hourly solar forecasting using machine learning models," *Renewable and Sustainable Energy Reviews*, vol. 105, pp. 487–498, 2019.

[46] M. M. Forootan, I. Larki, R. Zahedi, and A. Ahmadi, "Machine learning and deep learning in energy systems: A review," *Sustainability*, vol. 14, no. 8, p. 4832, 2022.

[47] G. Li and J. Shi, "On comparing three artificial neural networks for wind speed forecasting," *Applied Energy*, vol. 87, no. 7, pp. 2313–2320, 2010.

[48] M. G. De Giorgi, S. Campilongo, A. Ficarella, and P. M. Congedo, "Comparison between wind power prediction models based on wavelet decomposition with least-squares support vector machine (ls-svm) and artificial neural network (ann)," *Energies*, vol. 7, no. 8, pp. 5251–5272, 2014.

[49] T. Barbounis and J. B. Theocharis, "Locally recurrent neural networks for wind speed prediction using spatial correlation," *Information Sciences*, vol. 177, no. 24, pp. 5775–5797, 2007.

[50] Z. Sun, H. Sun, and J. Zhang, "Multistep wind speed and wind power prediction based on a predictive deep belief network and an optimized random forest," *Mathematical Problems in Engineering*, vol. 2018, 2018.

[51] S. Haykin and N. Network, "A comprehensive foundation," *Neural networks*, vol. 2, no. 2004, p. 41, 2004.

[52] K. Bhaskar and S. N. Singh, "Awnn-assisted wind power forecasting using feed-forward neural network," *IEEE transactions on sustainable energy*, vol. 3, no. 2, pp. 306–315, 2012.

[53] A. More and M. Deo, "Forecasting wind with neural networks," *Marine structures*, vol. 16, no. 1, pp. 35–49, 2003.

[54] T. G. Barbounis, J. B. Theocharis, M. C. Alexiadis, and P. S. Dokopoulos, "Long-term wind speed and power forecasting using local recurrent neural network models," *IEEE Transactions on Energy Conversion*, vol. 21, no. 1, pp. 273–284, 2006.

[55] Q. Cao, B. T. Ewing, and M. A. Thompson, "Forecasting wind speed with recurrent neural networks," *European Journal of Operational Research*, vol. 221, no. 1, pp. 148–154, 2012.

[56] R. K. Agrawal, F. Muchahary, and M. M. Tripathi, "Ensemble of relevance vector machines and boosted trees for electricity price forecasting," *Applied Energy*, vol. 250, pp. 540–548, 2019.

[57] M. Santhosh, C. Venkaiah, and D. V. Kumar, "Short-term wind speed forecasting approach using ensemble empirical mode decomposition and deep boltzmann machine," *Sustainable Energy, Grids and Networks*, vol. 19, p. 100242, 2019.

[58] J. Huang and M. Perry, "A semi-empirical approach using gradient boosting and k-nearest neighbors regression for gefcom2014 probabilistic solar power forecasting," *International Journal of Forecasting*, vol. 32, no. 3, pp. 1081–1086, 2016.

[59] X. Wang, C. Wang, and Q. Li, "Short-term wind power prediction using ga-elm," *The Open Electrical & Electronic Engineering Journal*, vol. 11, no. 1, 2017.

[60] X. Zhai and L. Ma, "Medium and long-term wind power prediction based on artificial fish swarm algorithm combined with extreme learning machine," *International Core Journal of Engineering*, vol. 5, no. 10, pp. 265–272, 2019.

[61] L. Tan, J. Han, and H. Zhang, "Ultra-short-term wind power prediction by salp swarm algorithm-based optimizing extreme learning machine," *IEEE Access*, vol. 8, pp. 44 470–44 484, 2020.

[62] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[63] M. Liu, Z. Cao, J. Zhang, L. Wang, C. Huang, and X. Luo, "Short-term wind speed forecasting based on the jaya-svm model," *International Journal of Electrical Power & Energy Systems*, vol. 121, p. 106056, 2020.

[64] J. Cai, K. Xu, Y. Zhu, F. Hu, and L. Li, "Prediction and analysis of net ecosystem carbon exchange based on gradient boosting regression and random forest," *Applied energy*, vol. 262, p. 114566, 2020.

[65] G. Li, Y. Yang, and X. Qu, "Deep learning approaches on pedestrian detection in hazy weather," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 10, pp. 8889–8899, 2019.

[66] A. Brusaferri, M. Matteucci, P. Portolani, and A. Vitali, "Bayesian deep learning based method for probabilistic forecast of day-ahead electricity prices," *Applied Energy*, vol. 250, pp. 1158–1175, 2019.

[67] Z. Guo, K. Zhou, X. Zhang, and S. Yang, "A deep learning model for short-term power load and probability density forecasting," *Energy*, vol. 160, pp. 1186–1200, 2018.

[68] S.-X. Lv, L. Peng, and L. Wang, "Stacked autoencoder with echo-state regression for tourism demand forecasting using search query data," *Applied Soft Computing*, vol. 73, pp. 119–133, 2018.

[69] L. Wang, Z. Zhang, and J. Chen, "Short-term electricity price forecasting with stacked denoising autoencoders," *IEEE Transactions on Power Systems*, vol. 32, no. 4, pp. 2673–2681, 2016.

[70] K. Wang, X. Qi, H. Liu, and J. Song, "Deep belief network based k-means cluster approach for short-term wind power forecasting," *Energy*, vol. 165, pp. 840–852, 2018.

[71] 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.

[72] Y. Chen, S. Zhang, W. Zhang, J. Peng, and Y. Cai, "Multifactor spatio-temporal correlation model based on a combination of convolutional neural network and long short-term memory neural network for wind speed forecasting," *Energy Conversion and Management*, vol. 185, pp. 783–799, 2019.

[73] X. Zhao, N. Jiang, J. Liu, D. Yu, and J. Chang, "Short-term average wind speed and turbulent standard deviation forecasts based on one-dimensional convolutional neural network and the integrate method for probabilistic framework," *Energy Conversion and Management*, vol. 203, p. 112239, 2020.

[74] Z. Chen and C. Jiang, "Building occupancy modeling using generative adversarial network," *Energy and Buildings*, vol. 174, pp. 372–379, 2018.

[75] C. Tian, C. Li, G. Zhang, and Y. Lv, "Data driven parallel prediction of building energy consumption using generative adversarial nets," *Energy and Buildings*, vol. 186, pp. 230–243, 2019.

[76] Y. Zhang, Q. Ai, F. Xiao, R. Hao, and T. Lu, "Typical wind power scenario generation for multiple wind farms using conditional improved wasserstein generative adversarial network," *International Journal of Electrical Power & Energy Systems*, vol. 114, p. 105388, 2020.

[77] R. Jiao, X. Huang, X. Ma, L. Han, and W. Tian, "A model combining stacked auto encoder and back propagation algorithm for short-term wind power forecasting," *Ieee Access*, vol. 6, pp. 17 851–17 858, 2018.

[78] Y.-R. Zeng, Y. Zeng, B. Choi, and L. Wang, "Multifactor-influenced energy consumption forecasting using enhanced back-propagation neural network," *Energy*, vol. 127, pp. 381–396, 2017.

[79] T. Niu, J. Wang, K. Zhang, and P. Du, "Multi-step-ahead wind speed forecasting based on optimal feature selection and a modified bat algorithm with the cognition strategy," *Renewable Energy*, vol. 118, pp. 213–229, 2018.

[80] D. Karamichailidou, V. Kaloutsa, and A. Alexandridis, "Wind turbine power curve modeling using radial basis function neural networks and tabu search," *Renewable Energy*, vol. 163, pp. 2137–2152, 2021.

[81] H.-F. Yang and Y.-P. P. Chen, "Representation learning with extreme learning machines and empirical mode decomposition for wind speed forecasting methods," *Artificial Intelligence*, vol. 277, p. 103176, 2019.

[82] M. Qolipour, A. Mostafaeipour, M. Saidi-Mehrabad, and H. R. Arabnia, "Prediction of wind speed using a new grey-extreme learning machine hybrid algorithm: A case study," *Energy & Environment*, vol. 30, no. 1, pp. 44–62, 2019.

[83] S. Salcedo-Sanz, A. Pastor-Sánchez, J. Del Ser, L. Prieto, and Z.-W. Geem, "A coral reefs optimization algorithm with harmony search operators for accurate wind speed prediction," *Renewable Energy*, vol. 75, pp. 93–101, 2015.

[84] Y. Li, P. Yang, and H. Wang, "Short-term wind speed forecasting based on improved ant colony algorithm for lssvm," *Cluster Computing*, vol. 22, no. 5, pp. 11 575–11 581, 2019.

[85] Z. Qu, K. Zhang, J. Wang, W. Zhang, and W. Leng, "A hybrid model based on ensemble empirical mode decomposition and fruit fly optimization algorithm for wind speed forecasting," *Advances in Meteorology*, vol. 2016, 2016.

[86] Z. Tian, Y. Ren, and G. Wang, "An application of backtracking search optimization–based least squares support vector machine for prediction of short-term wind speed," *Wind Engineering*, vol. 44, no. 3, pp. 266–281, 2020.

[87] L.-l. Li, Z.-Y. Cen, M.-L. Tseng, Q. Shen, and M. H. Ali, "Improving short-term wind power prediction using hybrid improved cuckoo search arithmetic-support vector regression machine," *Journal of Cleaner Production*, vol. 279, p. 123739, 2021.

[88] J. Yang, "A novel wind power capacity combined forecasting method based on backtracking search algorithm," in *2015 International Industrial Informatics and Computer Engineering Conference*. Atlantis Press, 2015, pp. 720–723.

[89] T. Liu, H. Wei, and K. Zhang, "Wind power prediction with missing data using gaussian process regression and multiple imputation," *Applied Soft Computing*, vol. 71, pp. 905–916, 2018.

[90] J. Yan, K. Li, E. Bai, Z. Yang, and A. Foley, "Time series wind power forecasting based on variant gaussian process and tlbo," *Neurocomputing*, vol. 189, pp. 135–144, 2016.

[91] Y. Dong, H. Zhang, C. Wang, and X. Zhou, "A novel hybrid model based on bernstein polynomial with mixture of gaussians for wind power forecasting," *Applied Energy*, vol. 286, p. 116545, 2021.

[92] K. Kargar, M. J. S. Safari, and K. Khosravi, "Weighted instances handler wrapper and rotation forest-based hybrid algorithms for sediment transport modeling," *Journal of Hydrology*, vol. 598, p. 126452, 2021.

[93] J. Wang, T. Niu, H. Lu, W. Yang, and P. Du, "A novel framework of reservoir computing for deterministic and probabilistic wind power forecasting," *IEEE Transactions on Sustainable Energy*, vol. 11, no. 1, pp. 337–349, 2019.

[94] S. Buhan and I. Çadırcı, "Multistage wind-electric power forecast by using a combination of advanced statistical methods," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 5, pp. 1231–1242, 2015.

[95] S. Buhan, Y. Özkazanç, and I. Çadırcı, "Wind pattern recognition and reference wind mast data correlations with nwp for improved wind-electric power forecasts," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 3, pp. 991–1004, 2016.

[96] M. Nazari-Heris, B. Mohammadi-Ivatloo, and G. Gharehpetian, "Short-term scheduling of hydro-based power plants considering application of heuristic algorithms: A comprehensive review," *Renewable and Sustainable Energy Reviews*, vol. 74, pp. 116–129, 2017.

[97] C. Wan, Z. Xu, and P. Pinson, "Direct interval forecasting of wind power," *IEEE Transactions on Power Systems*, vol. 28, no. 4, pp. 4877–4878, 2013.

[98] S. Wang, N. Zhang, L. Wu, and Y. Wang, "Wind speed forecasting based on the hybrid ensemble empirical mode decomposition and ga-bp neural network method," *Renewable Energy*, vol. 94, pp. 629–636, 2016.

[99] D. Liu, D. Niu, H. Wang, and L. Fan, "Short-term wind speed forecasting using wavelet transform and support vector machines optimized by genetic algorithm," *Renewable energy*, vol. 62, pp. 592–597, 2014.

[100] X. Yuan, C. Chen, Y. Yuan, Y. Huang, and Q. Tan, "Short-term wind power prediction based on lssvm–gsa model," *Energy Conversion and Management*, vol. 101, pp. 393–401, 2015.

[101] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Advances in engineering software*, vol. 95, pp. 51–67, 2016.

[102] D. Wang, H. Luo, O. Grunder, Y. Lin, and H. Guo, "Multi-step ahead electricity price forecasting using a hybrid model based on two-layer

decomposition technique and bp neural network optimized by firefly algorithm," *Applied Energy*, vol. 190, pp. 390–407, 2017.

[103] H. Liu, H. Wu, and Y. Li, "Smart wind speed forecasting using ewt decomposition, gwo evolutionary optimization, relm learning and iewt reconstruction," *Energy Conversion and Management*, vol. 161, pp. 266–283, 2018.

[104] P. Du, J. Wang, W. Yang, and T. Niu, "Multi-step ahead forecasting in electrical power system using a hybrid forecasting system," *Renewable energy*, vol. 122, pp. 533–550, 2018.

[105] P. Du, J. Wang, Z. Guo, and W. Yang, "Research and application of a novel hybrid forecasting system based on multi-objective optimization for wind speed forecasting," *Energy Conversion and Management*, vol. 150, pp. 90–107, 2017.

[106] Z. He, Y. Chen, Z. Shang, C. Li, L. Li, and M. Xu, "A novel wind speed forecasting model based on moving window and multi-objective particle swarm optimization algorithm," *Applied Mathematical Modelling*, vol. 76, pp. 717–740, 2019.

[107] H. Liu and C. Chen, "Data processing strategies in wind energy forecasting models and applications: A comprehensive review," *Applied Energy*, vol. 249, pp. 392–408, 2019.

[108] 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.

[109] J. Wang, H. Jiang, Y. Wu, and Y. Dong, "Forecasting solar radiation using an optimized hybrid model by cuckoo search algorithm," *Energy*, vol. 81, pp. 627–644, 2015.

[110] J. Jung and R. P. Broadwater, "Current status and future advances for wind speed and power forecasting," *Renewable and Sustainable Energy Reviews*, vol. 31, pp. 762–777, 2014.

[111] A. M. Foley, P. G. Leahy, A. Marvuglia, and E. J. McKeogh, "Current methods and advances in forecasting of wind power generation," *Renewable energy*, vol. 37, no. 1, pp. 1–8, 2012.

[112] H. Mokhtari, "A nature inspired intelligent water drops evolutionary algorithm for parallel processor scheduling with rejection," *Applied Soft Computing*, vol. 26, pp. 166–179, 2015.

[113] S. Kirkpatrick, C. D. Gelatt Jr, and M. P. Vecchi, "Optimization by simulated annealing," *science*, vol. 220, no. 4598, pp. 671–680, 1983.

[114] E. Rashedi, H. Nezamabadi-Pour, and S. Saryazdi, "Gsa: a gravitational search algorithm," *Information sciences*, vol. 179, no. 13, pp. 2232–2248, 2009.

[115] Z. W. Geem, J. H. Kim, and G. V. Loganathan, "A new heuristic optimization algorithm: harmony search," *simulation*, vol. 76, no. 2, pp. 60–68, 2001.

[116] X. An, D. Jiang, C. Liu, and M. Zhao, "Wind farm power prediction based on wavelet decomposition and chaotic time series," *Expert Systems with Applications*, vol. 38, no. 9, pp. 11 280–11 285, 2011.

[117] S. He, Q. H. Wu, and J. R. Saunders, "Group search optimizer: an optimization algorithm inspired by animal searching behavior," *IEEE transactions on evolutionary computation*, vol. 13, no. 5, pp. 973–990, 2009.

[118] R. V. Rao, V. J. Savsani, and D. Vakharia, "Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems," *Computer-aided design*, vol. 43, no. 3, pp. 303–315, 2011.

[119] C.-M. Huang, C.-J. Huang, and M.-L. Wang, "A particle swarm optimization to identifying the armax model for short-term load forecasting," *IEEE Transactions on Power Systems*, vol. 20, no. 2, pp. 1126–1133, 2005.

[120] N. Amjady, F. Keynia, and H. Zareipour, "Wind power prediction by a new forecast engine composed of modified hybrid neural network and enhanced particle swarm optimization," *IEEE transactions on sustainable energy*, vol. 2, no. 3, pp. 265–276, 2011.

[121] D. Niu, Y. Wang, and D. D. Wu, "Power load forecasting using support vector machine and ant colony optimization," *Expert Systems with Applications*, vol. 37, no. 3, pp. 2531–2539, 2010.

[122] B. Akay, D. Karaboga, B. Gorkemli, and E. Kaya, "A survey on the artificial bee colony algorithm variants for binary, integer and mixed integer programming problems," *Applied Soft Computing*, vol. 106, p. 107351, 2021.

[123] X.-S. Yang and X. He, "Firefly algorithm: recent advances and applications," *International journal of swarm intelligence*, vol. 1, no. 1, pp. 36–50, 2013.

[124] R. C. Deo, M. A. Ghorbani, S. Samadianfard, T. Maraseni, M. Bilgili, and M. Biazar, "Multi-layer perceptron hybrid model integrated with the firefly optimizer algorithm for windspeed prediction of target site using a limited set of neighboring reference station data," *Renewable energy*, vol. 116, pp. 309–323, 2018.

[125] Z. Li, X. Lin, Q. Zhang, and H. Liu, "Evolution strategies for continuous optimization: A survey of the state-of-the-art," *Swarm and Evolutionary Computation*, vol. 56, p. 100694, 2020.

[126] A. Zameer, J. Arshad, A. Khan, and M. A. Z. Raja, "Intelligent and robust prediction of short term wind power using genetic programming based ensemble of neural networks," *Energy conversion and management*, vol. 134, pp. 361–372, 2017.

[127] T. Liu, S. Qi, X. Qiao, and S. Liu, "A hybrid short-term wind power point-interval prediction model based on combination of improved preprocessing methods and entropy weighted gru quantile regression network," *Energy*, vol. 288, p. 129904, 2024.

[128] W. Cui, C. Wan, and Y. Song, "Ensemble deep learning-based non-crossing quantile regression for nonparametric probabilistic forecasting of wind power generation," *IEEE Transactions on Power Systems*, vol. 38, no. 4, pp. 3163–3178, 2022.

[129] Y. He and H. Li, "Probability density forecasting of wind power using quantile regression neural network and kernel density estimation," *Energy conversion and management*, vol. 164, pp. 374–384, 2018.

[130] Y. Wang, Q. Hu, D. Meng, and P. Zhu, "Deterministic and probabilistic wind power forecasting using a variational bayesian-based adaptive robust multi-kernel regression model," *Applied energy*, vol. 208, pp. 1097–1112, 2017.

[131] M. Afshari-Igder, T. Niknam, and M.-H. Khooban, "Probabilistic wind power forecasting using a novel hybrid intelligent method," *Neural Computing and Applications*, vol. 30, no. 2, pp. 473–485, 2018.

[132] N. Zhu, Z. Xi, C. Wu, F. Zhong, R. Qi, H. Chen, S. Xu, and W. Ji, "Inductive conformal prediction enhanced lstm-snn network: Applications to birds and uavs recognition," *IEEE Geoscience and Remote Sensing Letters*, vol. 21, pp. 1–5, 2024.

[133] V. Jensen, F. M. Bianchi, and S. N. Anfinsen, "Ensemble conformalized quantile regression for probabilistic time series forecasting," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 35, no. 7, pp. 9014–9025, 2024.

[134] S. Sheng, P. Yu, D. Parker, M. Kwiatkowska, and L. Feng, "Safe pomdp online planning among dynamic agents via adaptive conformal prediction," *IEEE Robotics and Automation Letters*, vol. 9, no. 11, pp. 9946–9953, 2024.

[135] M. Klein, L. Bethune, E. Ndiaye, and M. Cuturi, "Multivariate conformal prediction using optimal transport," *arXiv preprint arXiv:2502.03609*, 2025.

[136] L. Podina, M. T. Rad, and M. Kohandel, "Conformalized physics-informed neural networks," *arXiv preprint arXiv:2405.08111*, 2024.

[137] M. S. Abid, R. Ahshan, M. Al-Abri, and R. Al Abri, "Spatiotemporal forecasting of solar and wind energy production: A robust deep learning model with attention framework," *Energy Conversion and Management: X*, vol. 26, p. 100919, 2025.

[138] Y. Yang, S. Fan, Z. Liu, and Z. Yu, "Wd-sgformer: high-precision wind power forecasting via dual-attention dynamic spatio-temporal learning," *Energy*, p. 138538, 2025.

[139] Y. Yang, Y. Liu, Y. Zhang, S. Shu, and J. Zheng, "Dest-gnn: A double-explored spatio-temporal graph neural network for multi-site intra-hour pv power forecasting," *Applied Energy*, vol. 378, p. 124744, 2025.

[140] D. Abdelkader, H. Fouzi, K. Belkacem, and S. Ying, "Graph neural networks-based spatiotemporal prediction of photovoltaic power: a comparative study," *Neural Computing and Applications*, vol. 37, no. 6, pp. 4769–4795, 2025.

[141] X. Wu, H. Zhan, J. Hu, and Y. Wang, "Non-stationary gnncrossformer: Transformer with graph information for non-stationary multivariate spatio-temporal wind power data forecasting," *Applied Energy*, vol. 377, p. 124492, 2025.

[142] Y. Chen, J.-W. Xiao, Y.-W. Wang, and Y. Luo, "Non-crossing quantile probabilistic forecasting of cluster wind power considering spatio-temporal correlation," *Applied Energy*, vol. 377, p. 124356, 2025.

[143] G. Kariniotakis, *Renewable energy forecasting: from models to applications*. Woodhead Publishing, 2017.

[144] J. W. Messner, P. Pinson, J. Browell, M. B. Bjerregård, and I. Schicker, "Evaluation of wind power forecasts—an up-to-date view," *Wind Energy*, 2020.

[145] T. Chai and R. R. Draxler, "Root mean square error (rmse) or mean absolute error (mae)?–arguments against avoiding rmse in the literature," *Geoscientific model development*, vol. 7, no. 3, pp. 1247–1250, 2014.

[146] R. J. Hyndman and A. B. Koehler, "Another look at measures of forecast accuracy," *International journal of forecasting*, vol. 22, no. 4, pp. 679–688, 2006.

[147] S. Kim and H. Kim, "A new metric of absolute percentage error for intermittent demand forecasts," *International Journal of Forecasting*, vol. 32, no. 3, pp. 669–679, 2016.

[148] H. Madsen, P. Pinson, G. Kariniotakis, H. A. Nielsen, and T. S. Nielsen, "Standardizing the performance evaluation of short-term wind power prediction models," *Wind engineering*, vol. 29, no. 6, pp. 475–489, 2005.

[149] C. J. Willmott, "On the validation of models," *Physical geography*, vol. 2, no. 2, pp. 184–194, 1981.

[150] C. J. Willmott, S. M. Robeson, and K. Matsuura, "A refined index of model performance," *International Journal of climatology*, vol. 32, no. 13, pp. 2088–2094, 2012.

[151] J. W. Messner and P. Pinson, "Online adaptive lasso estimation in vector autoregressive models for high dimensional wind power forecasting," *International Journal of Forecasting*, vol. 35, no. 4, pp. 1485–1498, 2019.

[152] Y. Hao and C. Tian, "A novel two-stage forecasting model based on error factor and ensemble method for multi-step wind power forecasting," *Applied energy*, vol. 238, pp. 368–383, 2019.

[153] J. Naik, S. Dash, P. Dash, and R. Bisoi, "Short term wind power forecasting using hybrid variational mode decomposition and multi-kernel regularized pseudo inverse neural network," *Renewable Energy*, vol. 118, pp. 180–212, 2018.

[154] L. Zjavka and S. Mišák, "Direct wind power forecasting using a polynomial decomposition of the general differential equation," *IEEE Transactions on Sustainable Energy*, vol. 9, no. 4, pp. 1529–1539, 2018.

[155] X. Yuan, Q. Tan, X. Lei, Y. Yuan, and X. Wu, "Wind power prediction using hybrid autoregressive fractionally integrated moving average and least square support vector machine," *Energy*, vol. 129, pp. 122–137, 2017.

[156] Y. Jiang, X. Chen, K. Yu, and Y. Liao, "Short-term wind power forecasting using hybrid method based on enhanced boosting algorithm," *Journal of Modern Power Systems and Clean Energy*, vol. 5, no. 1, pp. 126–133, 2017.

[157] G. Chang, H. Lu, Y. Chang, and Y. Lee, "An improved neural network-based approach for short-term wind speed and power forecast," *Renewable energy*, vol. 105, pp. 301–311, 2017.

[158] J. Liu, X. Wang, and Y. Lu, "A novel hybrid methodology for short-term wind power forecasting based on adaptive neuro-fuzzy inference system," *Renewable energy*, vol. 103, pp. 620–629, 2017.

[159] A. Lahouar and J. B. H. Slama, "Hour-ahead wind power forecast based on random forests," *Renewable energy*, vol. 109, pp. 529–541, 2017.

[160] A. S. Qureshi, A. Khan, A. Zameer, and A. Usman, "Wind power prediction using deep neural network based meta regression and transfer learning," *Applied Soft Computing*, vol. 58, pp. 742–755, 2017.

[161] Z. Shi, H. Liang, and V. Dinavahi, "Direct interval forecast of uncertain wind power based on recurrent neural networks," *IEEE Transactions on Sustainable Energy*, vol. 9, no. 3, pp. 1177–1187, 2017.

[162] G. Zhang, Y. Wu, K. P. Wong, Z. Xu, Z. Y. Dong, and H. H.-C. Iu, "An advanced approach for construction of optimal wind power prediction intervals," *IEEE transactions on power systems*, vol. 30, no. 5, pp. 2706–2715, 2014.

[163] P. Pinson and G. Kariniotakis, "Conditional prediction intervals of wind power generation," *IEEE Transactions on Power Systems*, vol. 25, no. 4, pp. 1845–1856, 2010.

[164] R. L. Winkler, "A decision-theoretic approach to interval estimation," *Journal of the American Statistical Association*, vol. 67, no. 337, pp. 187–191, 1972.

[165] P. Pinson, H. A. Nielsen, J. K. Møller, H. Madsen, and G. N. Kariniotakis, "Non-parametric probabilistic forecasts of wind power: required properties and evaluation," *Wind Energy: An International Journal for Progress and Applications in Wind Power Conversion Technology*, vol. 10, no. 6, pp. 497–516, 2007.

[166] W. Zou, C. Li, and P. Chen, "An inter type-2 fcr algorithm based t–s fuzzy model for short-term wind power interval prediction," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 9, pp. 4934–4943, 2019.

[167] W. Xie, P. Zhang, R. Chen, and Z. Zhou, "A nonparametric bayesian framework for short-term wind power probabilistic forecast," *IEEE Transactions on Power Systems*, vol. 34, no. 1, pp. 371–379, 2018.

[168] T. Mahmoud, Z. Dong, and J. Ma, "An advanced approach for optimal wind power generation prediction intervals by using self-adaptive evolutionary extreme learning machine," *Renewable energy*, vol. 126, pp. 254–269, 2018.

[169] Y. Lin, M. Yang, C. Wan, J. Wang, and Y. Song, "A multi-model combination approach for probabilistic wind power forecasting," *IEEE Transactions on Sustainable Energy*, vol. 10, no. 1, pp. 226–237, 2018.

[170] B. Khorramdel, C. Chung, N. Safari, and G. Price, "A fuzzy adaptive probabilistic wind power prediction framework using diffusion kernel density estimators," *IEEE Transactions on Power Systems*, vol. 33, no. 6, pp. 7109–7121, 2018.

[171] H.-z. Wang, G.-q. Li, G.-b. Wang, J.-c. Peng, H. Jiang, and Y.-t. Liu, "Deep learning based ensemble approach for probabilistic wind power forecasting," *Applied energy*, vol. 188, pp. 56–70, 2017.

[172] NASA POWER: Prediction Of Worldwide Energy Resource, "NASA POWER: Prediction Of Worldwide Energy Resource," https://power.larc.nasa.gov/, 2023, accessed on October 3, 2023.

[173] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[174] B. Jiang, H. Gong, H. Qin, and M. Zhu, "Attention-lstm architecture combined with bayesian hyperparameter optimization for indoor temperature prediction," *Building and Environment*, vol. 224, p. 109536, 2022.

[175] S. L. Ho and M. Xie, "The use of arima models for reliability forecasting and analysis," *Computers & industrial engineering*, vol. 35, no. 1-2, pp. 213–216, 1998.

[176] F. Zou, W. Fu, P. Fang, D. Xiong, and R. Wang, "A hybrid model based on multi-stage principal component extraction, gru network and kelm for multi-step short-term wind speed forecasting," *IEEE Access*, vol. 8, pp. 222 931–222 943, 2020.

[177] Z. Zhao, S. Yun, L. Jia, J. Guo, Y. Meng, N. He, X. Li, J. Shi, and L. Yang, "Hybrid vmd-cnn-gru-based model for short-term forecasting of wind power considering spatio-temporal features," *Engineering Applications of Artificial Intelligence*, vol. 121, p. 105982, 2023.

[178] D. Wang, M. Xu, Z. Guangming, F. Luo, J. Gao, and Y. Chen, "Enhancing wind power forecasting accuracy through lstm with adaptive wind speed calibration (c-lstm)," *Scientific Reports*, vol. 15, no. 1, p. 5352, 2025.

[179] W. P. Mahoney, K. Parks, G. Wiener *et al.*, "A wind power forecasting system to optimize grid integration," *IEEE Transactions on Sustainable Energy*, vol. 3, no. 4, pp. 670–682, 2012.

[180] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[181] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE transactions on neural networks*, vol. 5, no. 2, pp. 157–166, 1994.

[182] H. Xu, F. Hu, X. Liang *et al.*, "Attention mechanism multi-size depthwise convolutional long short-term memory neural network for forecasting real-time electricity prices," *IEEE Transactions on Power Systems*, vol. 39, no. 5, pp. 6277–6289, Sep 2024.

[183] H. S. Dhiman, D. Deb, and J. M. Guerrero, "Hybrid machine intelligent SVR variants for wind forecasting and ramp events," *Renewable and Sustainable Energy Reviews*, vol. 108, pp. 369–379, 2019.

[184] M. Yu, D. Niu, T. Gao *et al.*, "A novel framework for ultra-short-term interval wind power prediction based on RF-WOA-VMD and BiGRU optimized by the attention mechanism," *Energy*, vol. 269, p. 126738, 2023.

[185] O. Abedinia, A. Ghasemi-Marzbali, M. Shafiei *et al.*, "Wind power forecasting enhancement utilizing adaptive quantile function and CNN-LSTM: A probabilistic approach," *IEEE Transactions on Industry Applications*, vol. 60, no. 3, pp. 4446–4457, 2024.

[186] Y. Xiao, C. Zou, H. Chi, and R. Fang, "Boosted gru model for short-term forecasting of wind power with feature-weighted principal component analysis," *Energy*, vol. 267, p. 126503, 2023.

[187] J. V. Belle, R. Crevits, D. Caljon *et al.*, "Probabilistic forecasting with modified N-BEATS networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 35, no. 12, pp. 18 872–18 885, 2024.

[188] Ş. T. Özçelik and F. B. Tek, "Integrating the focusing neuron model with N-BEATS and N-HiTS," in *2024 9th International Conference on Computer Science and Engineering (UBMK)*, 2024, pp. 422–425.

[189] H. Wu, J. Xu, J. Wang *et al.*, "Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting," *Advances in neural information processing systems*, vol. 34, pp. 22 419–22 430, 2021.

[190] H. Zhou, S. Zhang, J. Peng *et al.*, "Informer: Beyond efficient transformer for long sequence time-series forecasting," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, no. 12, 2021, pp. 11 106–11 115.

[191] N. Yuan, D. Guan, S. Li *et al.*, "Enhancing neurodegenerative disease diagnosis through confidence-driven dynamic spatio-temporal convolutional network," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 33, pp. 1715–1728, 2025.

[192] A. V. D. Oord, S. Dieleman, H. Zen *et al.*, "Wavenet: A generative model for raw audio," Sep 2016, [Online]. Available: https://arxiv.org/abs/1609.03499.

[193] W. Cao, G. Wang, X. Liang *et al.*, "A STAM-LSTM model for wind power prediction with feature selection," *Energy*, vol. 296, p. 131030, 2024.

[194] S. Afaq and S. Rao, "Significance of epochs on training a neural network," *Int. J. Sci. Technol. Res*, vol. 9, no. 6, pp. 485–488, 2020.

[195] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," Dec 2014, [Online]. Available: https://arxiv.org/abs/1412.6980.

[196] M. E. Torres, M. A. Colominas, G. Schlotthauer, and P. Flandrin, "A complete ensemble empirical mode decomposition with adaptive noise," in *2011 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2011, pp. 4144–4147.

[197] A. Humeau-Heurtier, G. Mahé, and P. Abraham, "Multi-dimensional complete ensemble empirical mode decomposition with adaptive noise applied to laser speckle contrast images," *IEEE transactions on medical imaging*, vol. 34, no. 10, pp. 2103–2117, 2015.

[198] M. Xia, H. Shao, X. Ma, and C. W. de Silva, "A stacked gru-rnn-based approach for predicting renewable energy and electricity load for smart grid operation," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 10, pp. 7050–7059, 2021.

[199] S. Wu, H. Guo, X. Zhang, and F. Wang, "Short-term photovoltaic power prediction based on ceemdan and hybrid neural networks," *IEEE Journal of Photovoltaics*, vol. 14, no. 6, pp. 960–969, 2024.

[200] W. Zhang, Z. Qu, K. Zhang, W. Mao, Y. Ma, and X. Fan, "A combined model based on ceemdan and modified flower pollination algorithm for wind speed forecasting," *Energy conversion and management*, vol. 136, pp. 439–451, 2017.

[201] R. Li and A. Rodríguez, "Neural conformal control for time series forecasting," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, no. 17, 2025, pp. 18 439–18 447.

[202] Southern Regional Power Committee (SRPC), "Commercial operation data of regional generating stations," https://www.srpc.kar.nic.in/html_old/xml-search/commercial.html, 2023, dataset, accessed Oct. 2023.

[203] ENGIE, "Open data renewables – explore," https://web.archive.org/web/20210704061136/https://opendata-renewables.engie.com/explore/index, 2021, dataset retrieved from ENGIE's Open Data Renewables platform.

[204] G. Kougioumtzidis, V. K. Poulkov, P. I. Lazaridis, and Z. D. Zaharis, "Mobile network traffic prediction using temporal fusion transformer," *IEEE Transactions on Artificial Intelligence*, pp. 1–15, 2025.

[205] P. Arora, S. M. J. Jalali, S. Ahmadian, B. K. Panigrahi, P. N. Suganthan, and A. Khosravi, "Probabilistic wind power forecasting using optimized deep auto-regressive recurrent neural networks," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 3, pp. 2814–2825, 2023.

[206] A. Bracale, P. Caramia, P. De Falco, and T. Hong, "Multivariate quantile regression for short-term probabilistic load forecasting," *IEEE Transactions on Power Systems*, vol. 35, no. 1, pp. 628–638, 2020.

[207] W. Zhang, H. Quan, and D. Srinivasan, "An improved quantile regression neural network for probabilistic load forecasting," *IEEE Transactions on Smart Grid*, vol. 10, no. 4, pp. 4425–4434, 2018.

[208] B. Lim, S. Ö. Arık, N. Loeff, and T. Pfister, "Temporal fusion transformers for interpretable multi-horizon time series forecasting," *International journal of forecasting*, vol. 37, no. 4, pp. 1748–1764, 2021.

[209] W. Zhang, H. Zhan, H. Sun, and M. Yang, "Probabilistic load forecasting for integrated energy systems based on quantile regression patch time series transformer," *Energy Reports*, vol. 13, pp. 303–317, 2025.

[210] R. Sivasubramanian, R. CN, S. Kumar *et al.*, "Attention-based spatio-temporal graph neural network for multi-pollutant urban air quality prediction," in *2025 3rd International Conference on Data Science and Network Security (ICDSNS)*. IEEE, 2025, pp. 1–5.

[211] C. Su, Z. Cai, Y. Tian, Z. Chang, Z. Zheng, and Y. Song, "Diffusion models for time series forecasting: A survey," *arXiv preprint arXiv:2507.14507*, 2025.