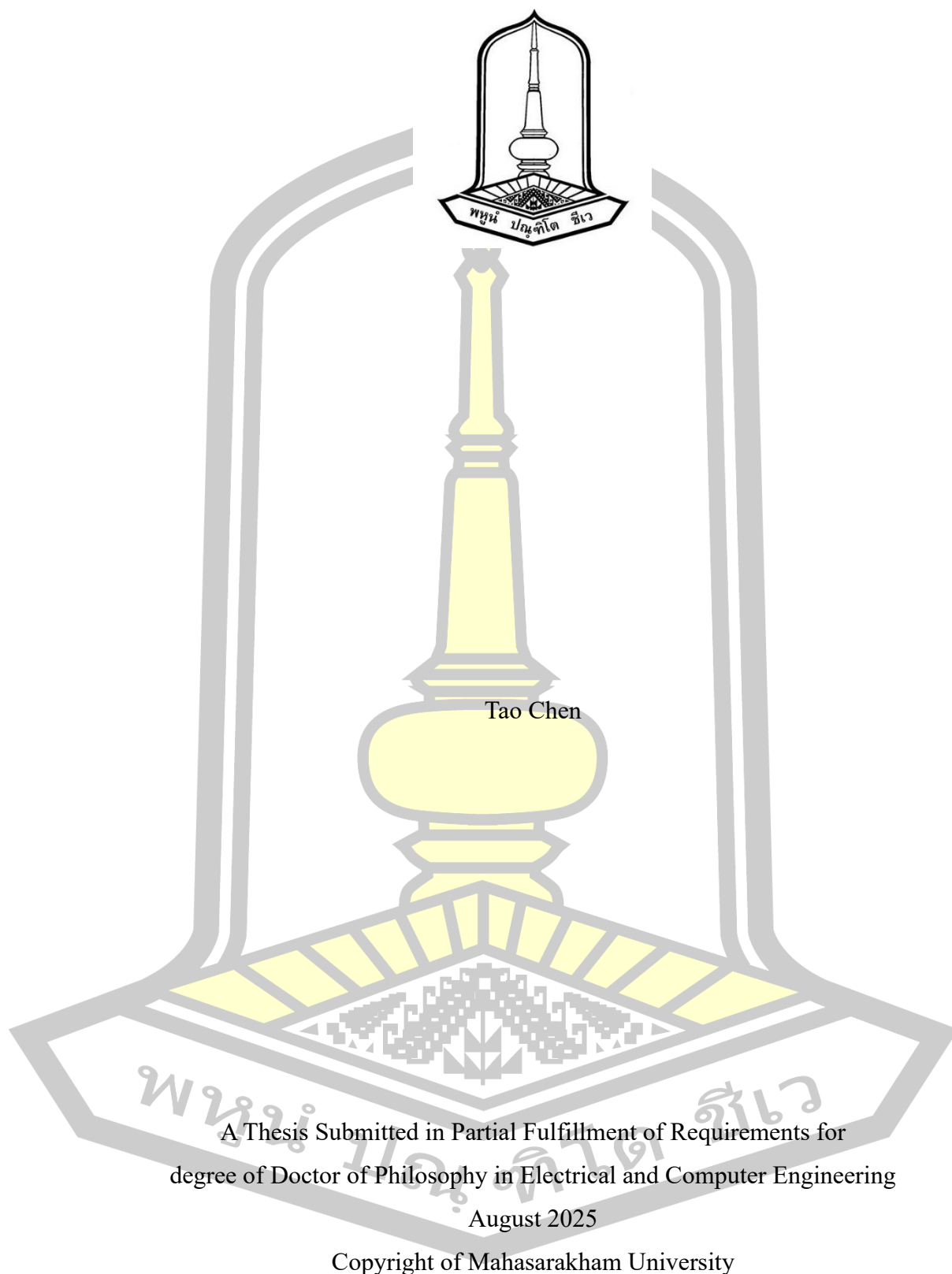


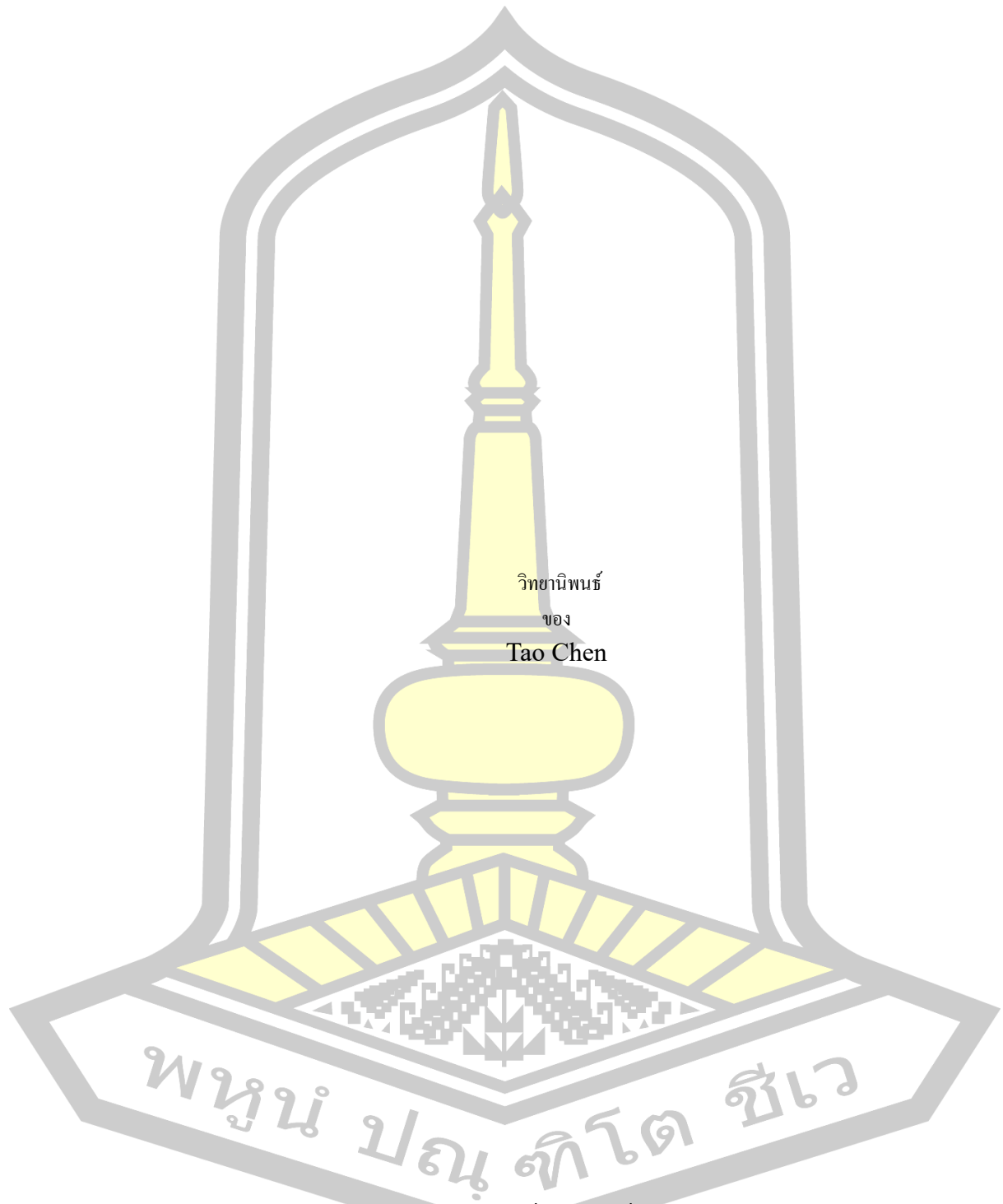


Tao Chen

A Thesis Submitted in Partial Fulfillment of Requirements for
degree of Doctor of Philosophy in Electrical and Computer Engineering
August 2025

Copyright of Mahasarakham University

Wind Turbine Blade Failure Diagnosis Based on SCADA Data-Driven Approach



วิทยานิพนธ์

ของ

Tao Chen

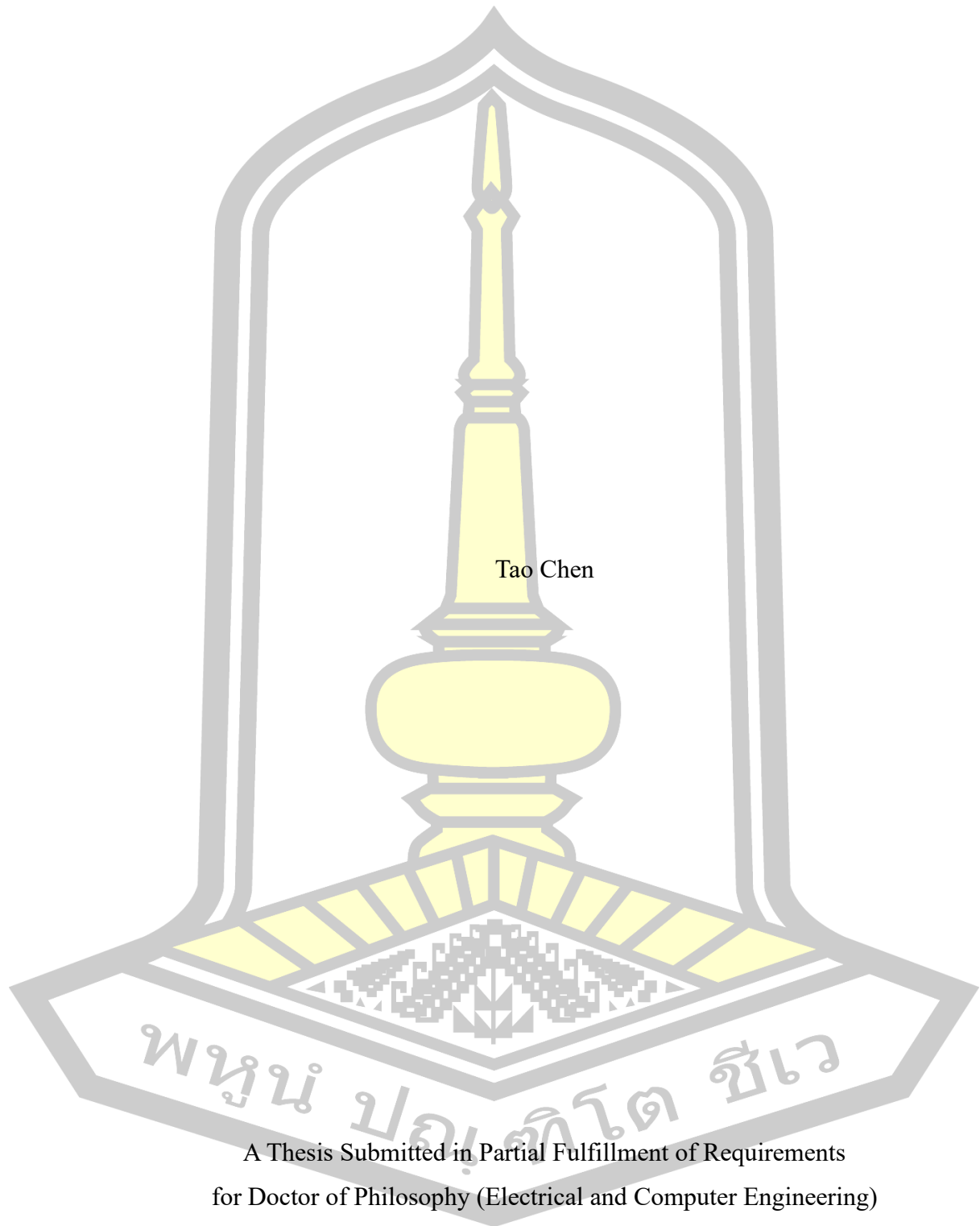
เสนอต่อมหาวิทยาลัยมหาสารคาม เพื่อเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

ปริญญาปรัชญาดุษฎีบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

สิงหาคม 2568

ลิขสิทธิ์เป็นของมหาวิทยาลัยมหาสารคาม

Wind Turbine Blade Failure Diagnosis Based on SCADA Data-Driven Approach



Tao Chen

A Thesis Submitted in Partial Fulfillment of Requirements
for Doctor of Philosophy (Electrical and Computer Engineering)

August 2025

Copyright of Mahasarakham University



The examining committee has unanimously approved this Thesis, submitted by Ms. Tao Chen , as a partial fulfillment of the requirements for the Doctor of Philosophy Electrical and Computer Engineering at Mahasarakham University

Examining Committee

Chairman

(Asst. Prof. Adirek Jantakun , Ph.D.)

Advisor

(Assoc. Prof. Chonlatee Photong ,
Ph.D.)

Committee

(Prof. Worawat Sa-Ngiamvibool ,
Ph.D.)

Committee

(Assoc. Prof. Supannika Wattana ,
Ph.D.)

Committee

(Assoc. Prof. Niwat
Angkawisittpan , Ph.D.)

Mahasarakham University has granted approval to accept this Thesis as a partial fulfillment of the requirements for the Doctor of Philosophy Electrical and Computer Engineering

(Assoc. Prof. Juckamas Laohavanich ,
Ph.D.)

Dean of The Faculty of Engineering

(Asst. Prof. Pondej Chaowarat , Ph.D.)
Dean of Graduate School

TITLE	Wind Turbine Blade Failure Diagnosis Based on SCADA Data-Driven Approach		
AUTHOR	Tao Chen		
ADVISORS	Associate Professor Chonlatee Photong , Ph.D.		
DEGREE	Doctor of Philosophy	MAJOR	Electrical and Computer Engineering
UNIVERSITY	Maharakham University	YEAR	2025

ABSTRACT

This research introduces an innovative data-driven approach for diagnosing wind turbine blade faults, specifically icing, using Supervisory Control and Data Acquisition (SCADA) system data. Wind turbine blades are particularly vulnerable to icing during cold weather conditions, which can severely impact their efficiency, performance, and overall reliability. Early detection of icing faults is crucial to avoid unscheduled downtimes, costly repairs, and extended periods of reduced energy production.

In this study, we propose a deep learning-based fault detection system designed to accurately identify and predict icing conditions on wind turbine blades. The system integrates Convolutional Neural Networks (CNN) and Bidirectional Gated Recurrent Units (BiGRU) to analyze large-scale SCADA data. The CNN is employed to extract spatial features, while the BiGRU captures temporal dependencies in the dataset, making the system robust against varying operational conditions over time.

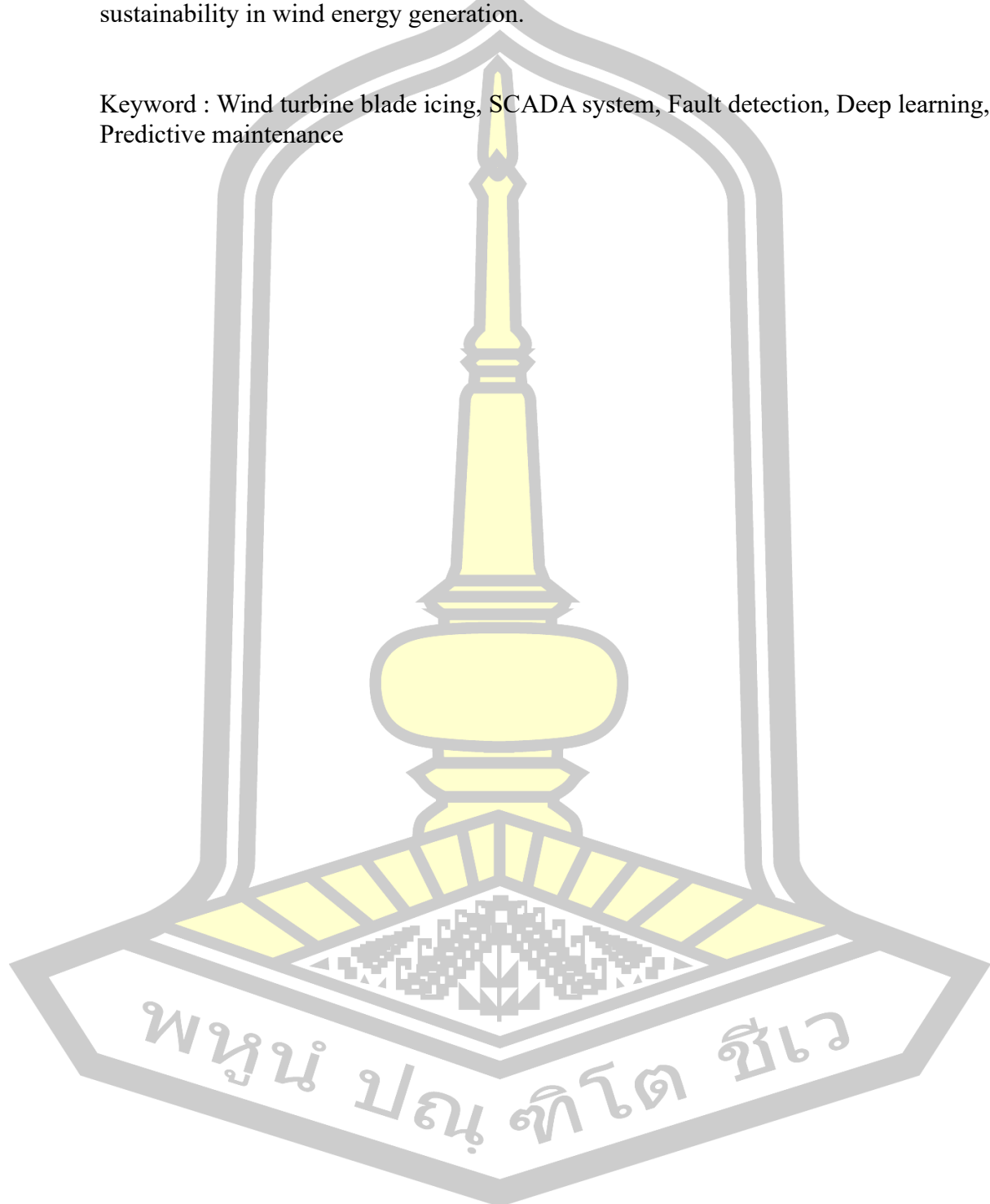
The methodology includes preprocessing the SCADA data to extract relevant features, followed by training and testing the deep learning models to classify and predict blade icing conditions. We compared the performance of this data-driven approach with traditional machine learning techniques such as support vector machines and decision trees, demonstrating significantly improved accuracy, robustness, and scalability in real-world applications.

The proposed system offers real-time monitoring capabilities and the ability to issue early warnings, allowing operators to take proactive measures to prevent severe blade damage. Furthermore, the integration of SCADA data ensures that the system can be seamlessly implemented in existing wind farms, eliminating the need for additional hardware. The results indicate that this approach leads to significant reductions in maintenance costs and enhances the overall operational efficiency of wind turbines.

This research contributes to the field of predictive maintenance in renewable energy systems by offering a practical and scalable solution for intelligent fault

diagnosis. It represents an important step toward the future of autonomous wind turbine monitoring systems, combining the power of deep learning and SCADA data to optimize wind energy production, improve turbine reliability, and ensure long-term sustainability in wind energy generation.

Keyword : Wind turbine blade icing, SCADA system, Fault detection, Deep learning, Predictive maintenance



ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my supervisor, Professor Chonlatee Photong, for his continuous guidance, encouragement, and invaluable support throughout my doctoral research. His expertise, dedication, and insightful advice have played a pivotal role in shaping this work.

I also wish to thank my dissertation committee members, Professor Worawat Sa-Ngiamvibool, Associate Professor Supannika Wattana, and Associate Professor Niwat Angkawisittpan, for their constructive feedback and helpful suggestions during the various stages of my research. Their perspectives have significantly enriched the quality of this work.

Special thanks to the engineering team at the wind turbine manufacturers and the SCADA data providers for sharing critical operational data that made this research possible. The collaboration with industry partners has been instrumental in the practical application of this research.

Finally, I would like to acknowledge Maharakham University for providing the resources and academic environment that facilitated the completion of this research. The financial support from [funding agency, if applicable] is also gratefully acknowledged, as it enabled me to undertake this important project.

Tao Chen

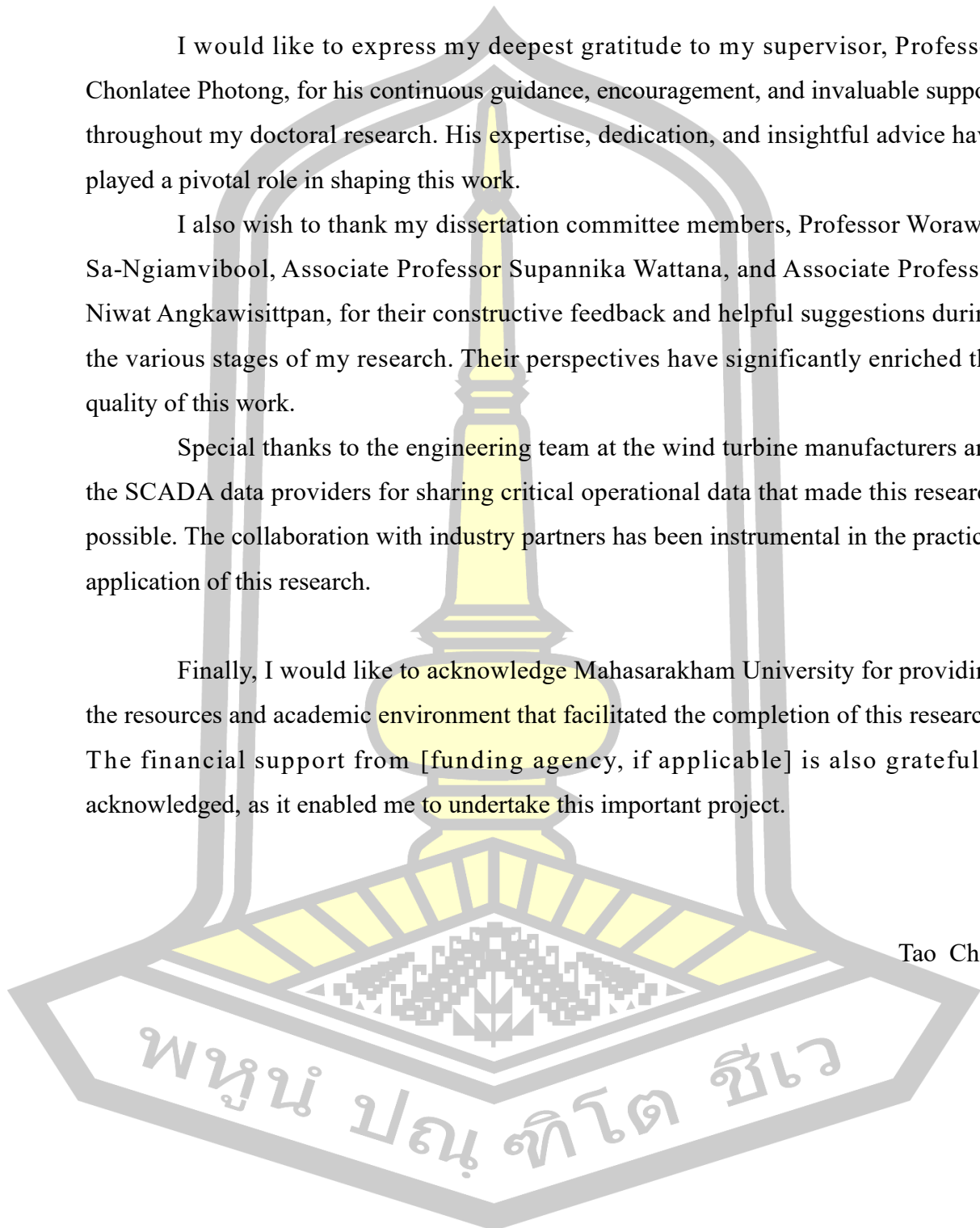
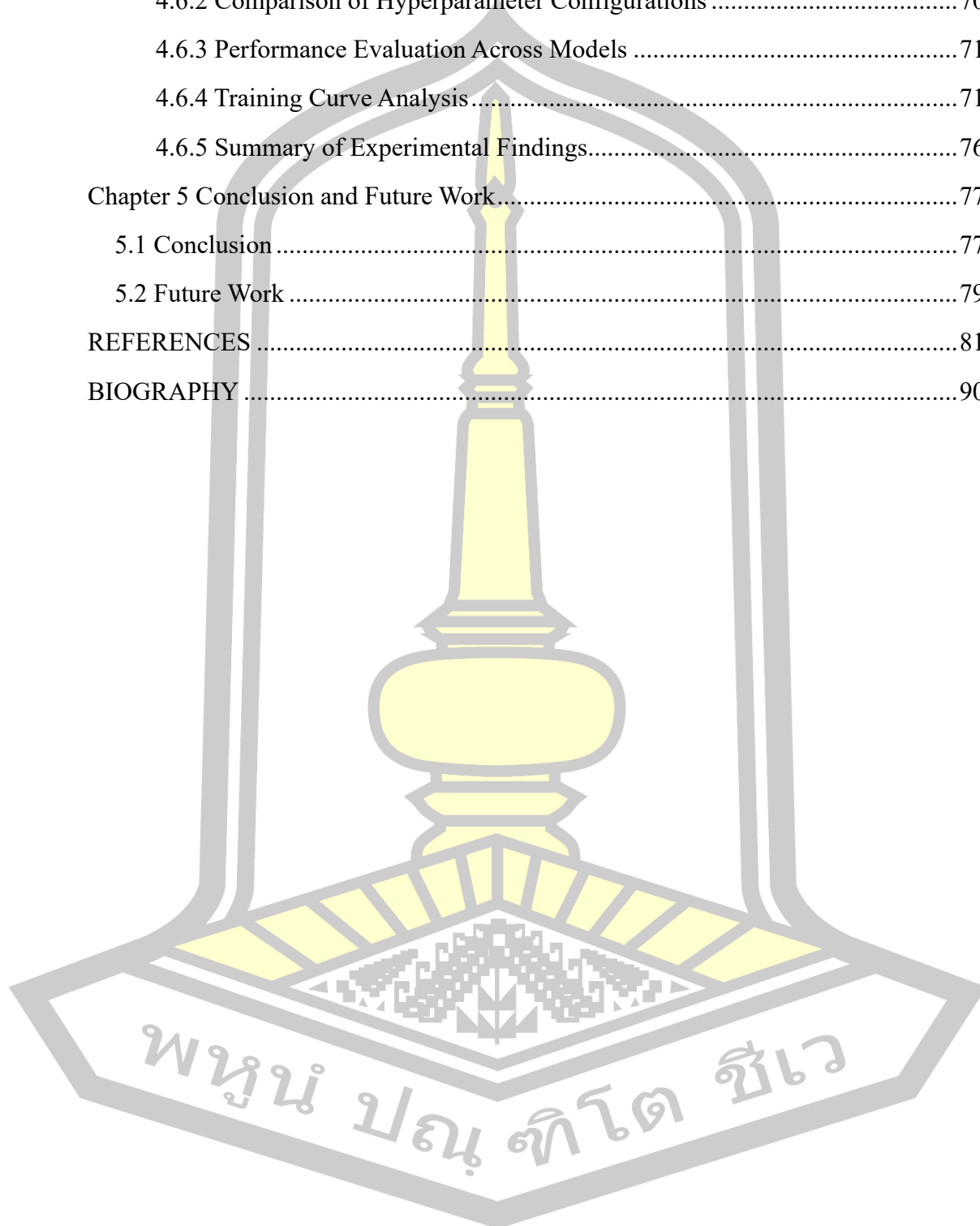


TABLE OF CONTENTS

	Page
ABSTRACT.....	D
ACKNOWLEDGEMENTS.....	F
TABLE OF CONTENTS.....	G
LIST OF TABLES	J
LIST OF FIGURES	K
Chapter 1 Research Background.....	1
1.1 Introduction.....	1
1.2 Research Objectives.....	3
1.3 Research Scope	5
1.4 Benefits of the research.....	6
1.5 Research Plan.....	9
Chapter 2 Literature Reviews	12
2.1 Overview of Wind Turbine	12
2.1.1 The classification of wind turbines.....	12
2.1.2 Structure of the Wind Turbine	14
2.1.3 Main Fault Analysis of Wind Turbine	15
2.2 Research and analysis of wind turbine blades	16
2.2.1 Structure of wind turbine blades	16
2.2.2 Main Faults and Diagnostic Methods of Wind Turbine Blades	17
2.2.3 Factors Contributing to Wind Turbine Blade Icing	19
2.3 Overview of the SCADA System	23
2.3.1 Description of Internal Data in SCADA Systems	24
2.4 Intelligent Algorithms for Wind Turbine Icing Prediction	26
2.4.1 Machine Learning algorithm	27
2.4.2 Machine Learning Approaches for Wind Turbine Icing Prediction	27

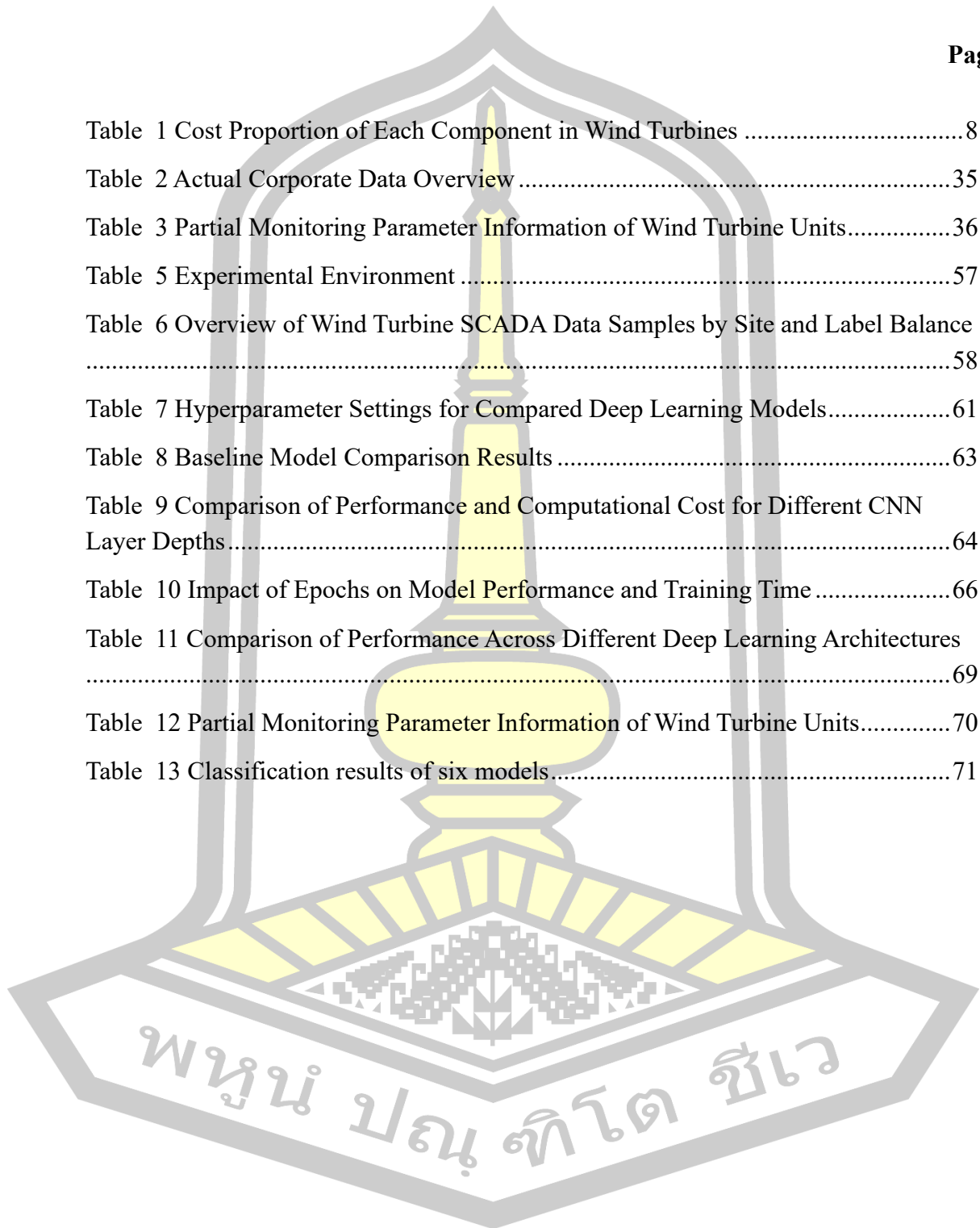
2.4.3 Deep learning algorithm.....	30
2.4.4 Deep Learning in Wind Turbine Blade Icing Prediction.....	31
2.5 Summary.....	33
Chapter 3 Research Methodology.....	34
3.1 Data Preprocessing	34
3.2 Feature Engineering.....	37
3.2.1 Random forest classification theory	38
3.2.2 CNN-Based Power Feature Enhancement	39
3.3 Deep Learning Model Architecture	43
3.3.1 LSTM and BiLSTM Network Architecture Models	44
3.3.2 GRU and BiGRU Network Architecture Models.....	47
3.3.3 Joint CNN-BiGRU mechanism.....	49
3.3.4 Optimization model and loss function selection	50
3.3.5 Temporal Attention Mechanism	51
3.3.6 CNN-BiGRU-TPA model	53
3.4 Summary.....	54
Chapter 4 Experimental Setup and Model Evaluation Framework.....	56
4.1 Experimental Configuration and Data Preparation.....	56
4.1.1 Definition of Evaluation Metrics.....	56
4.1.2 Experimental Environment.....	57
4.1.3 Dataset Splitting Strategy.....	57
4.1.4 Hyperparameter Selection and Optimization Strategy.....	59
4.2 Feature Correlation Analysis and Redundant Feature Removal.....	60
4.3 Baseline Comparison.....	62
4.4 Sensitivity experiments.....	64
4.4.1 Impact of CNN Depth on Model Performance	64
4.4.2 Effect of training rounds on model performance	66
4.5 Deep Learning Architecture Comparison	67
4.6 Experiments with TPA Attention Mechanism.....	69

4.6.1 Hyperparameter Optimization Process.....	69
4.6.2 Comparison of Hyperparameter Configurations	70
4.6.3 Performance Evaluation Across Models	71
4.6.4 Training Curve Analysis.....	71
4.6.5 Summary of Experimental Findings.....	76
Chapter 5 Conclusion and Future Work.....	77
5.1 Conclusion.....	77
5.2 Future Work	79
REFERENCES	81
BIOGRAPHY	90



LIST OF TABLES

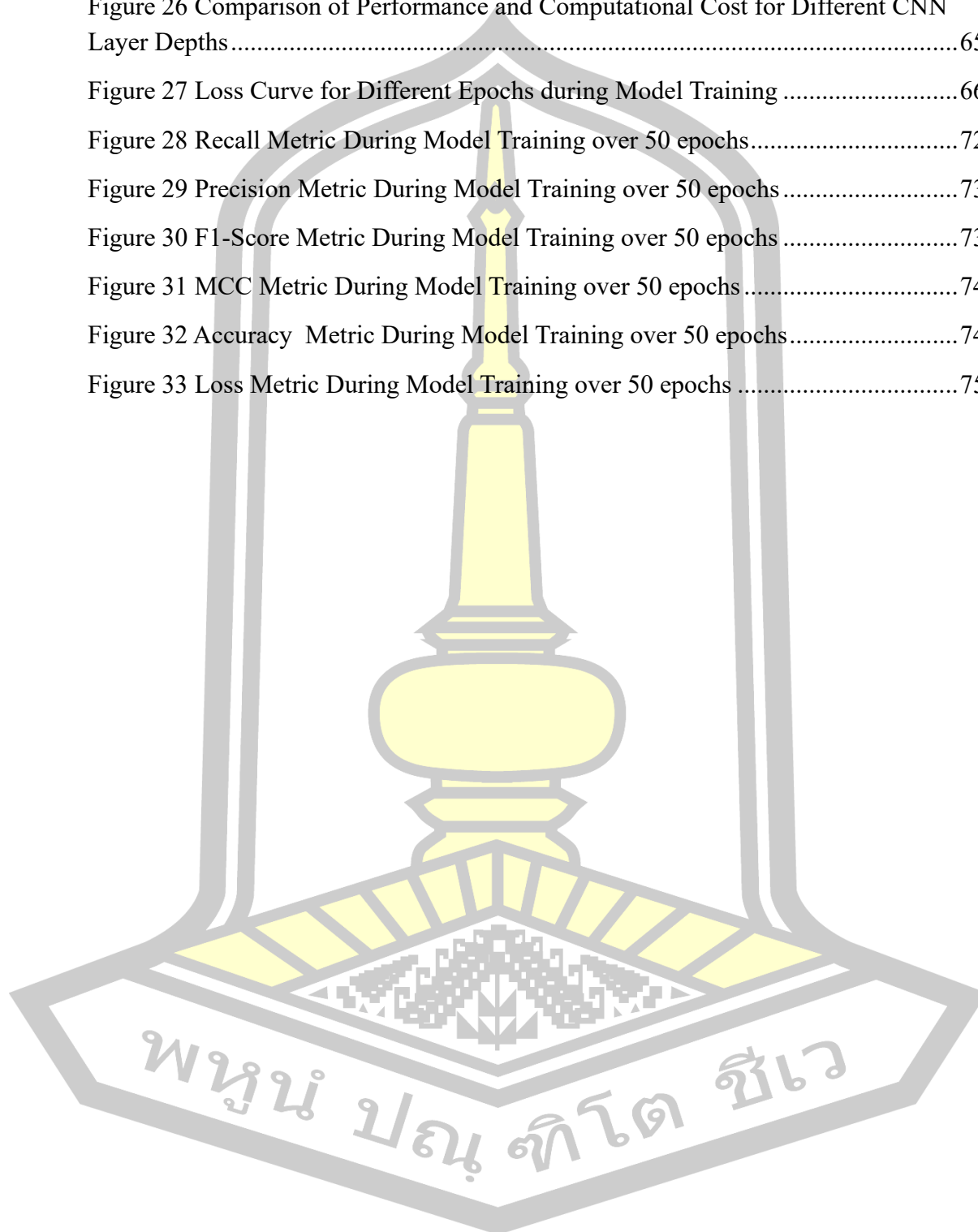
	Page
Table 1 Cost Proportion of Each Component in Wind Turbines	8
Table 2 Actual Corporate Data Overview	35
Table 3 Partial Monitoring Parameter Information of Wind Turbine Units.....	36
Table 5 Experimental Environment	57
Table 6 Overview of Wind Turbine SCADA Data Samples by Site and Label Balance	58
Table 7 Hyperparameter Settings for Compared Deep Learning Models.....	61
Table 8 Baseline Model Comparison Results	63
Table 9 Comparison of Performance and Computational Cost for Different CNN Layer Depths.....	64
Table 10 Impact of Epochs on Model Performance and Training Time	66
Table 11 Comparison of Performance Across Different Deep Learning Architectures	69
Table 12 Partial Monitoring Parameter Information of Wind Turbine Units.....	70
Table 13 Classification results of six models.....	71



LIST OF FIGURES

	Page
Figure 1 Wind power density potential of the global [1]	1
Figure 2 Statistics of Wind Turbine Installed Capacity in China from 2017 to 2022[4]	2
Figure 3 Thesis Framework	11
Figure 4 Wind turbine axis classifications:(a)Vertical-axis structure and (b) Horizontal-axis structure[10]	12
Figure 5 Structure frame of DFIG with gearbox wind turbine [11].....	13
Figure 6 Structure frame of direct drive PMSG wind turbine [11].....	13
Figure 7 Structure frame of multibrid wind turbine [11]	14
Figure 8 Structure and Components of a Wind Turbine[12].....	15
Figure 9 Wind turbine components and associated faults [13]	16
Figure 10 Percentage failures of wind turbine components.....	16
Figure 11 Internal Structure and Material Distribution Diagram of Wind Turbine Blade [15].....	17
Figure 12 Types of Wind Turbine Blade Faults [16]	18
Figure 13 Display interface for SCADA system [35].....	24
Figure 14 Blade Ice Detection Flowchart	38
Figure 15 Random Forest Classification Algorithm[74]	39
Figure 16 Convolutional Neural Network Algorithm[75]	39
Figure 17 Framework of LSTM [76]	44
Figure 18 Framework of BiLSTM [77]	46
Figure 19 Framework of GRU [39]	47
Figure 20 Framework of BiGRU	48
Figure 21 CNN-BiGRU Algorithm Architecture [80]	50
Figure 22 Framework of TPA	52
Figure 23 Framework of CNN-BIGRU-TPA.....	54
Figure 24 Workflow of the Proposed Blade Icing Detection Framework	59

Figure 25 SCADA Feature Correlation Heatmap	62
Figure 26 Comparison of Performance and Computational Cost for Different CNN Layer Depths	65
Figure 27 Loss Curve for Different Epochs during Model Training	66
Figure 28 Recall Metric During Model Training over 50 epochs.....	72
Figure 29 Precision Metric During Model Training over 50 epochs	73
Figure 30 F1-Score Metric During Model Training over 50 epochs	73
Figure 31 MCC Metric During Model Training over 50 epochs.....	74
Figure 32 Accuracy Metric During Model Training over 50 epochs.....	74
Figure 33 Loss Metric During Model Training over 50 epochs	75



Chapter 1 Research Background

1.1 Introduction

In 2023, wind power further solidified its position in the global energy landscape, emerging as the second-largest renewable energy source after solar photovoltaics (see in the Figure 1). Its key advantages—including zero emissions, sustainability, and declining operational costs—have driven widespread adoption. Technological advancements have enhanced the efficiency of wind power generation across diverse terrains and climates. As a result, many countries have implemented proactive policies to accelerate wind power deployment. Notably, wind-rich regions offer high energy density potential, reinforcing wind power's growing role in the global transition to clean energy.

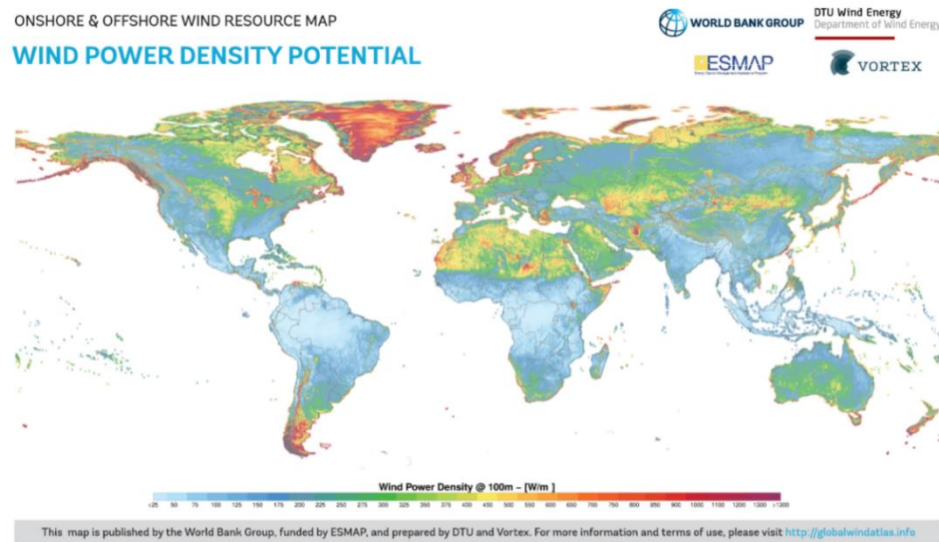


Figure 1 Wind power density potential of the global [1]

In order to realize the national goal of 2030 carbon peak 2060 carbon neutral, the development of wind power industry is in full swing, and the safety of wind turbines has become more and more important. With the growing global demand for energy and the aggravation of environmental pollution, more and more people have begun to pay attention to the issue of energy security and actively promote the development of renewable and clean energy. In particular, our government proposed on September 22, 2020, the goal of carbon emissions to peak in 2030 and achieve carbon neutrality in 2060, and re-emphasized on March 15, 2021, to do a good job in

carbon peaking and carbon neutrality, and the advancement of the "3060" plan[2] signals the rapid development of renewable energy.

Wind power, as one of the main force of renewable energy generation, has great potential for development with the continuous maturation of technology. of which China's new installed capacity of wind power accounted for about 48.5%. Affected by the epidemic and other factors, although the new wind power installed capacity in 2022 year-on-year decline, but is still showing an upward trend (Figure 2). However, with the increasing scale of wind power installed capacity, the operation and maintenance of wind turbines has become more and more burdensome, and the cost of troubleshooting has increased. Coupled with the limitation of wind energy resources, wind farms are usually constructed in high altitude areas such as plateaus, mountains and other high-altitude areas as well as the sea, the operating environment is complex and harsh, resulting in the operation and maintenance work is difficult to carry out[3].

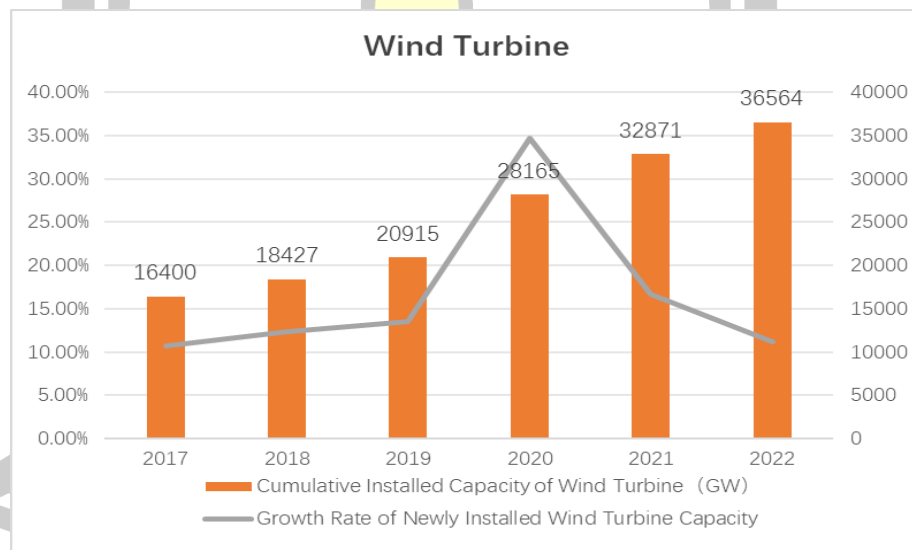


Figure 2 Statistics of Wind Turbine Installed Capacity in China from 2017 to 2022[4]

As one of the key components in the operation of wind turbines, the failure of wind turbine blades causes long downtime, which is often accompanied by high O&M costs and loss of power generation. Therefore, effective condition monitoring and fault diagnosis of the blades during the operation of WTGs, and early detection and elimination of potential safety hazards are of great significance and application

value to improve the reliability of WTGs, enhance the utilization rate of WTGs, and reduce the cost of O&M and downtime loss of power generation [5].

In recent years, with the development of artificial intelligence and the advancement of industrial digital transformation, blade fault diagnosis technology has also been developed accordingly, with a variety of signal processing methods emerging in an endless stream, and data-driven methods based on the wind power industry are also widely used in fault diagnosis tasks[6]. With the continuous advancement of algorithms, high fault diagnosis accuracy can be obtained in simulation experiments under ideal conditions, but more attention should be paid to the complex problems existing in practical engineering, for example, the difficulty of obtaining sufficient labeled data, the noise interference brought by the complex and changeable operating environment of wind turbines, and the difficulty of meeting the independent and homogeneous distribution assumptions of the training set and the test set in the application[7]. The solution of these practical problems is of great practical significance to promote the application of wind turbine blade fault diagnosis system.

1.2 Research Objectives

The overarching goal of this research is to develop a practical and robust fault diagnosis framework for wind turbine blades, with a particular focus on icing-related faults, by leveraging deep learning techniques in combination with SCADA system data. This goal addresses the growing industrial demand for intelligent, efficient, and real-time monitoring solutions that ensure operational safety, reduce maintenance costs, and extend the service life of wind turbines, especially under complex and harsh environmental conditions.

To achieve this goal, the study defines the following five key objectives:

1. To explore and design deep learning-based models for wind turbine blade fault diagnosis.

The research first analyzes the limitations of traditional fault detection methods, including high cost, limited sensitivity to complex failure patterns (such as blade icing), and reliance on manual inspection. To address these issues, statistical techniques such as frequency analysis and association rule mining are employed to identify critical operational patterns. These patterns guide the development of a hybrid

model architecture that integrates both spatial and temporal feature learning to enhance the model's ability to handle complex, nonlinear signals from SCADA data.

2.To implement comprehensive data preprocessing and feature engineering techniques tailored for SCADA datasets.

SCADA data typically exhibits high dimensionality, noise, and non-stationary characteristics. Therefore, this study establishes a rigorous preprocessing pipeline, including missing value imputation, normalization, time-series segmentation, and data balancing to handle class imbalance between icing and normal conditions. Feature selection is performed using Random Forest to isolate key influencing variables. Subsequently, CNN-based feature extraction layers are applied to enhance spatial characteristics prior to temporal modeling.

3.To construct and optimize a deep learning architecture that integrates CNN, BiGRU, and attention mechanisms.

A core contribution of this research is the development of a hybrid CNN-BiGRU model enhanced with attention modules to better capture spatiotemporal dependencies within multivariate SCADA time-series data. The model architecture is optimized through hyperparameter tuning, including network depth, dropout rate, and learning rate scheduling (e.g., cosine annealing). CrossEntropyLoss is adopted to manage multi-class classification and avoid overfitting.

4.To evaluate and benchmark the proposed model against traditional and state-of-the-art diagnostic methods.

The performance of the CNN-BiGRU-attention model is assessed using real-world SCADA data collected from wind turbines located in Hunan Province, China. Comparative experiments are conducted against classical machine learning algorithms (e.g., Random Forest) and other deep learning models (e.g., CNN-only, LSTM, BiLSTM). Evaluation metrics include accuracy, precision, recall, F1-score, and ROC-AUC to comprehensively validate the effectiveness and generalization ability of the proposed method.

5.To develop a practical fault diagnosis system suitable for real-time deployment in wind farm SCADA platforms.

The final objective is to bridge theoretical development with practical application. The proposed model is designed to be lightweight and compatible with

real-time deployment in existing SCADA systems, enabling early detection of blade faults such as icing. This supports timely maintenance or de-icing interventions, thereby minimizing downtime, reducing power losses, and improving turbine operational safety. The findings are also expected to contribute to the broader field of intelligent condition monitoring in renewable energy systems and provide a technical foundation for future smart-grid-integrated fault management platforms.

In summary, this research systematically integrates advanced SCADA data analysis, deep learning model innovation, and engineering deployment feasibility, thereby providing a comprehensive solution for wind turbine blade fault diagnosis in both academic and industrial contexts.

1.3 Research Scope

In response to the increasing complexity of wind turbine blade fault diagnosis—especially under harsh environmental conditions such as icing—this study systematically investigates a novel approach grounded in data-driven techniques. Unlike traditional monitoring systems that rely heavily on physical inspections and manual analysis, the proposed method integrates SCADA system data with advanced deep learning models to enable real-time, intelligent fault detection[8]. The research is structured around three core objectives, each addressing a critical technical challenge:

To address the limitations of traditional condition monitoring methods for detecting wind turbine blade faults—such as challenging installation, high operational cost, and insufficient stability—this study proposes a condition monitoring approach based on Supervisory Control and Data Acquisition (SCADA) systems. By leveraging built-in turbine sensors, real-time operational data from various components is collected without the need for additional external hardware. This ensures that the data used during model training and testing accurately reflects real-world turbine conditions, thereby enhancing the reliability of blade icing fault diagnosis.

To overcome the inefficiencies in processing large-scale and complex raw data encountered in traditional fault diagnosis approaches—particularly those relying on manual feature extraction and signal-based machine learning methods—this research introduces a deep learning-based intelligent diagnostic framework. Initially, the raw SCADA data undergoes comprehensive preprocessing, including data augmentation,

segmentation, normalization, and standardization. These steps are critical to address data imbalance and enhance the extraction of informative features. Subsequently, a Random Forest algorithm is employed for dimensionality reduction and feature selection, which transforms the data into multiple sets of multivariate time series fragments. These sequences are then input into a CNN-BiGRU deep learning model, where spatial and temporal features are learned. Multiple rounds of experiments are conducted to fine-tune model parameters and identify optimal configurations that maximize blade fault detection accuracy while preserving the integrity of SCADA-derived information.

This research further explores optimization strategies for enhancing the deep learning architecture. Specifically, the convolutional neural network (CNN) component is improved by selecting an appropriate number of layers and filter sizes to effectively extract spatial features from the input data. In parallel, the recurrent component—Gated Recurrent Units (GRU)—is optimized to capture temporal dependencies, and its performance is enhanced using a bidirectional structure (BiGRU), allowing the model to learn both forward and backward time-series correlations. By combining CNN and BiGRU into an end-to-end deep learning architecture, the proposed model significantly improves the accuracy of blade fault detection. This optimized fusion architecture not only elevates model robustness but also provides a practical and reliable tool for real-time wind turbine maintenance decision-making.

Collectively, these objectives form the methodological foundation of this research. In the following chapters, the study elaborates on data acquisition, preprocessing techniques, model architecture design, and performance evaluation strategies, all aimed at realizing a reliable and efficient fault diagnosis framework for wind turbine blades. Special emphasis is placed on the integration of spatial and temporal feature learning, as well as the use of real-world SCADA data to validate the practical applicability of the proposed approach.

1.4 Benefits of the research

With the continuous advancement of wind power generation technology and the steady expansion of global installed capacity, the long-term operation of wind

turbines is increasingly approaching their designed service limits. As a result, reducing fault incidence and improving power generation efficiency have become pivotal to ensuring the safe, stable, and cost-effective operation of modern wind farms. Most wind turbines are installed in regions with abundant wind resources—such as high-altitude mountainous areas and offshore environments—where they are subjected to extreme environmental conditions, including persistently low temperatures, high humidity, and frequent wind turbulence. These harsh conditions accelerate the wear and degradation of components, thereby increasing the risk of critical system failures.

Although considerable research efforts have been devoted to components such as gearboxes, generators, and bearings, relatively limited attention has been given to wind turbine blades—the primary element responsible for wind energy capture and mechanical-to-electrical conversion. Blades are not only positioned externally, making them more susceptible to environmental damage, but they also represent a substantial portion of the turbine's total cost[9]. As illustrated in Table 1, blade-related costs account for approximately 23.3% of the overall turbine cost—the highest among all major components—followed by the tower (18.9%) and gearbox (16.2%). This highlights the strategic importance of blade integrity in the context of both operational performance and economic sustainability[10].

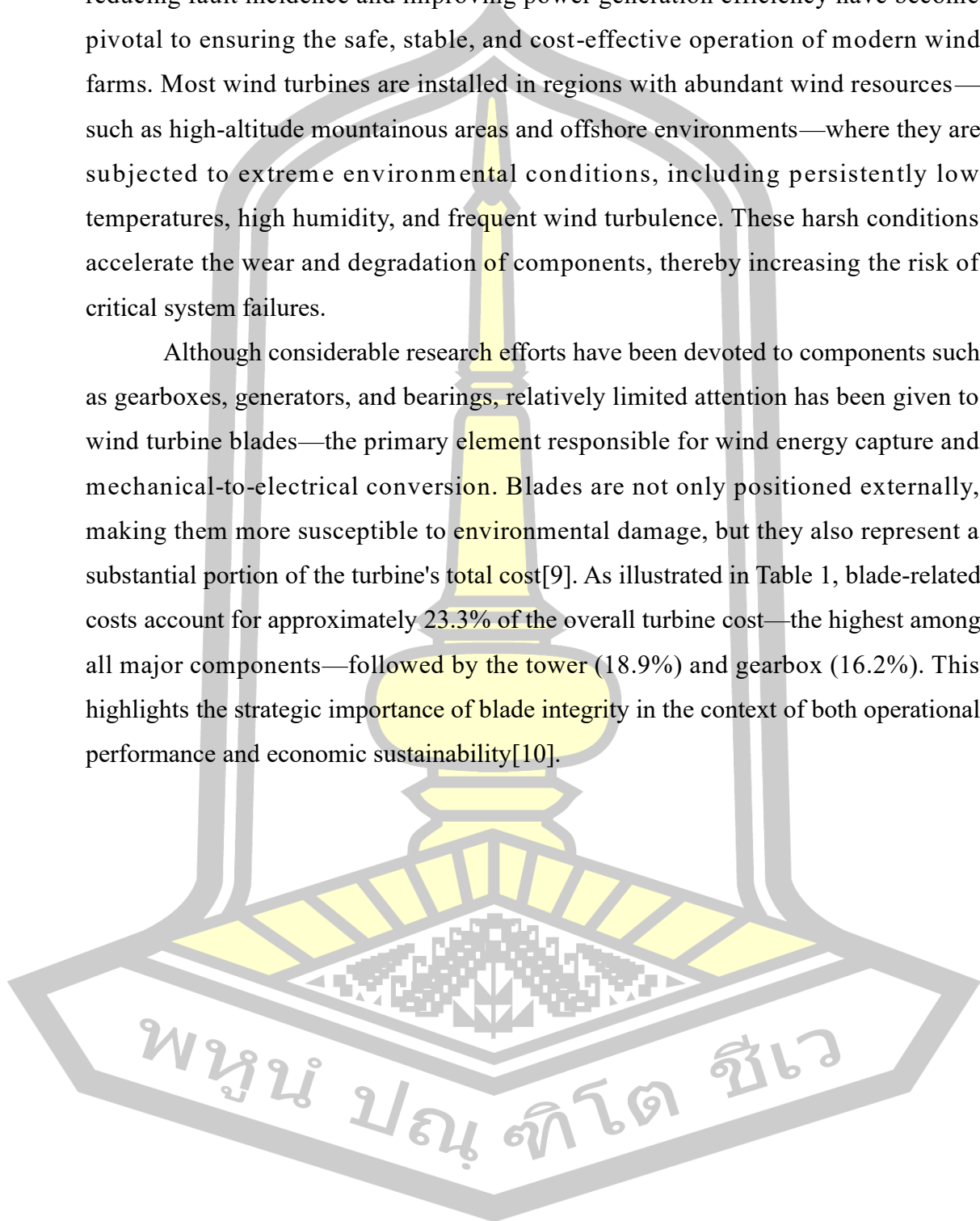


Table 1 Cost Proportion of Each Component in Wind Turbines

Component Name	Cost Proportion (%)	Component Name	Cost Proportion (%)	Component Name	Cost Proportion (%)
Blade	23.3	Transformer	3.59	Nacelle Cover	1.9
Tower	18.9	Generator	3.4	Heat Exchange System	1.3
Gearbox	16.2	Hub	3.0	Bearing	1.22
Converter	7.3	Frame	2.8	Bolt	1.04
Controller	5.0	Main Shaft	2.2	Cable	0.96
Pitch System	3.9	Yaw	2.0	Braking System	0.6

Given the high cost and vulnerability of turbine blades, failure to monitor their operational condition in real-time can result in undetected micro-defects escalating into severe structural failures. Among all fault types, blade icing has emerged as one of the most frequent and critical challenges—particularly in cold-climate regions. Ice accumulation modifies the blade's aerodynamic structure by increasing the trailing-edge thickness, thereby reducing the lift-to-drag ratio and substantially lowering power output. Furthermore, the added mass and aerodynamic asymmetry lead to increased mechanical loads, excessive nacelle vibration, and accelerated fatigue of drivetrain components. In extreme cases, prolonged icing may lead to catastrophic blade fracture or large-scale ice shedding, posing serious threats to personnel, infrastructure, and overall turbine safety.

In light of these risks, the timely and accurate diagnosis of blade icing faults is vital for ensuring uninterrupted turbine operation and extending component lifespan. However, traditional condition monitoring methods—such as manual inspection, vibration analysis, and acoustic emission techniques—suffer from notable limitations, including high cost, low responsiveness, poor scalability, and dependence on external

sensors. These shortcomings hinder their practical effectiveness, particularly in large-scale wind farm applications.

To address this gap, this research proposes a novel data-driven intelligent fault diagnosis framework that integrates SCADA system data with advanced deep learning models. By leveraging the built-in sensor infrastructure of SCADA systems, combined with the automated feature extraction and temporal modeling capabilities of Convolutional Neural Networks (CNN) and Bidirectional Gated Recurrent Units (BiGRU), the proposed method enables early detection of icing-related anomalies without the need for additional instrumentation. Furthermore, the deep learning model is designed for seamless integration into turbine control systems, enabling intelligent linkage with de-icing mechanisms and supporting proactive maintenance strategies.

This work contributes to the field in the following key aspects:

It offers a cost-effective, real-time, and scalable solution for wind turbine blade condition monitoring based on operational data.

It improves fault diagnosis accuracy and generalization ability through an end-to-end deep learning architecture optimized for SCADA time-series data.

It supports the intelligent digital transformation of wind turbine operation and maintenance (O&M) systems.

It establishes a technical foundation for the integration of predictive analytics and autonomous control in broader renewable energy applications.

Ultimately, this research aligns with global efforts toward carbon neutrality and sustainable energy development. By enhancing the reliability, safety, and efficiency of wind energy systems, the proposed methodology not only addresses a critical engineering challenge but also provides transferable insights for intelligent fault detection across other components and renewable energy platforms.

1.5 Research Plan

To tackle the pressing issue of wind turbine blade icing faults, this dissertation proposes a SCADA data-driven fault diagnosis approach grounded in deep learning techniques. The research is systematically organized into five chapters (see in the Figure 3), each focusing on a distinct yet interconnected phase of the study, ensuring a

comprehensive and coherent exploration from problem identification to model implementation and validation.

Chapter 1 introduces the background and motivation of the research, emphasizing the operational challenges posed by blade icing in cold climates and the urgent need for efficient and intelligent diagnostic methods. It critically examines the limitations of traditional monitoring systems and highlights the advantages of leveraging SCADA data combined with deep learning. The chapter also clarifies the research objectives, significance, and methodological framework, laying the foundation for the rest of the thesis.

Chapter 2 provides an in-depth review of existing literature related to wind turbine condition monitoring and fault diagnosis. It discusses the structural characteristics and common failure modes of wind turbine components, as well as the development and application of SCADA-based monitoring systems. The chapter further analyzes classical diagnostic techniques and recent advances in machine learning and deep learning, identifying the gaps in current research—particularly the lack of effective and timely diagnostic models for blade icing based on operational data.

Chapter 3 focuses on the operational characteristics of wind turbines and the formal definition of the problem. Real-world SCADA datasets are analyzed to identify key performance indicators such as power output, wind speed, rotor speed, and blade pitch angle. Through statistical methods and domain expertise, the chapter defines the multi-class classification problem of blade icing severity. It also outlines data preprocessing procedures, including time-series segmentation, outlier elimination, normalization, and feature selection, thereby ensuring that the data is well-prepared for model training.

Chapter 4 presents the core methodological contribution of this dissertation: the construction and evaluation of a hybrid CNN-BiGRU deep learning model for blade fault diagnosis. The model is designed to extract both spatial and temporal features from SCADA time-series data, enabling more accurate identification of blade icing events. This chapter details the model architecture, training strategy, hyperparameter tuning, and dropout optimization. Furthermore, extensive experiments are conducted to compare the proposed method with traditional algorithms such as

SVM, Random Forest, and LSTM, using multiple evaluation metrics and real-case validation to verify its effectiveness and generalization ability.

Chapter 5 concludes the dissertation by summarizing the key findings and contributions of the research. It confirms the feasibility and robustness of the proposed deep learning-based diagnostic framework and discusses its practical implications for intelligent wind turbine maintenance. The chapter also reflects on current limitations and proposes directions for future work, such as incorporating additional sensing modalities, enabling real-time model deployment, and expanding the methodology to other critical wind turbine components. This comprehensive research plan provides a solid technical foundation for advancing intelligent fault diagnosis in wind energy systems.

This five-chapter research framework enables a structured, data-centric, and practical solution to the pressing challenge of blade icing fault diagnosis in wind turbines.

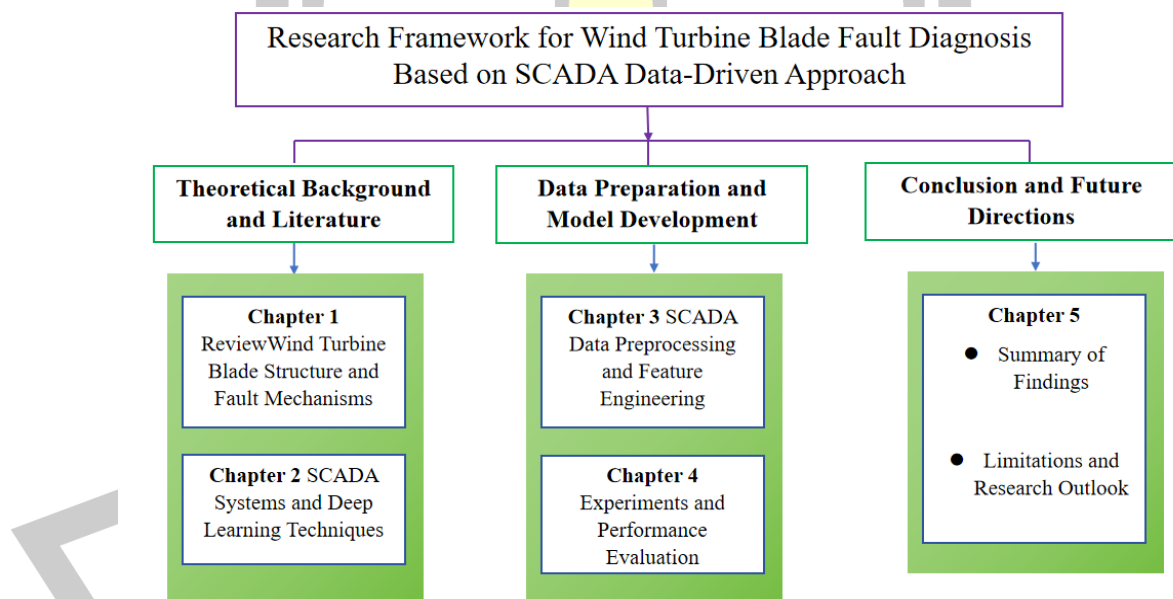


Figure 3 Thesis Framework

Chapter 2 Literature Reviews

2.1 Overview of Wind Turbine

Wind turbine is a complex and typical electromechanical liquid coupling system, through the rotation of the wind energy acting on the blades into mechanical energy, the use of generators will be converted into electrical energy, through the relevant control equipment will be the output of the unit power fed into the transmission network, and delivered to the end user.

2.1.1 The classification of wind turbines

Wind turbines can be classified according to different criteria, with the most common being based on the axis and generator type:

1.Axis Classification

Wind turbines have been built horizontally or vertically. As it can be seen in Figure 4(a), in vertical wind turbines, the blades are installed vertically [10]. One of the advantages of this kind of wind turbines is that the conversion systems and the gearboxes are located on the ground, while, the disadvantage of this type is that the maintenance is somehow complicated as it normally requires rotor removal. Besides, the efficiency of converting wind energy to electrical one is not high in compared to horizontal turbines. For these reasons, nowadays, the modern wind turbines usually have been built horizontally, as shown in Figure 4(b).

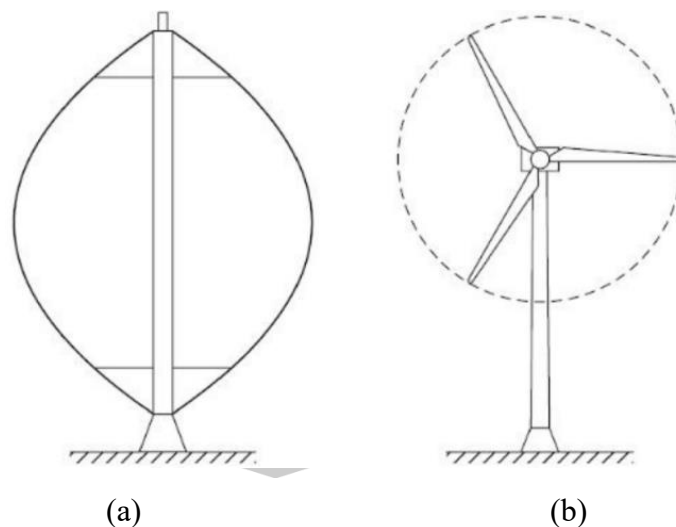


Figure 4 Wind turbine axis classifications:(a)Vertical-axis structure and (b) Horizontal-axis structure[10]

2. Generator Type Classification

In the Chinese market, variable-pitch wind turbines are mainly available in three configurations[11]:

Gearbox-equipped Doubly-Fed Induction Generator (DFIG)

This configuration uses a gearbox to increase the rotation speed of the wind rotor to the speed required by the generator. The DFIG can operate effectively under variable speeds, making it suitable for a wide range of wind speeds.

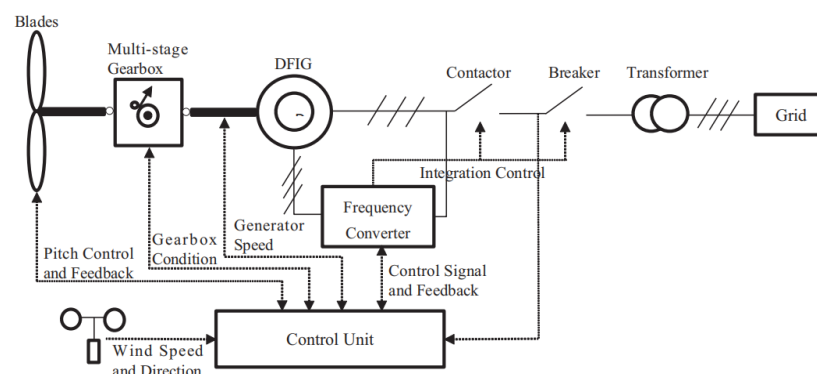


Figure 5 Structure frame of DFIG with gearbox wind turbine [11]

i. Direct-Drive Permanent Magnet Synchronous Generator (PMSG)

This setup does not include a gearbox, with the generator directly connected to the wind rotor, thereby reducing mechanical losses and maintenance requirements. The permanent magnet synchronous generator is favored for its high efficiency and reliability.

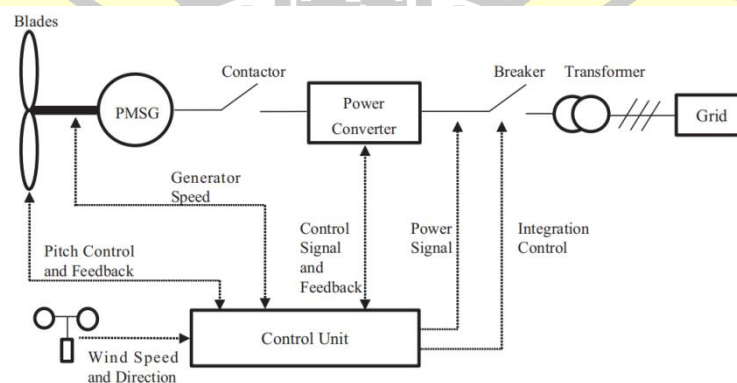


Figure 6 Structure frame of direct drive PMSG wind turbine [11]

ii. Single-stage/Twin-stage Gearbox-equipped Permanent Magnet Synchronous Generator (PMSG)

This configuration combines the high efficiency of the permanent magnet synchronous generator with the speed matching capabilities of a gearbox. Depending on the specific design, either a single-stage or a twin-stage gearbox can be used to accommodate different wind speeds and load requirements.

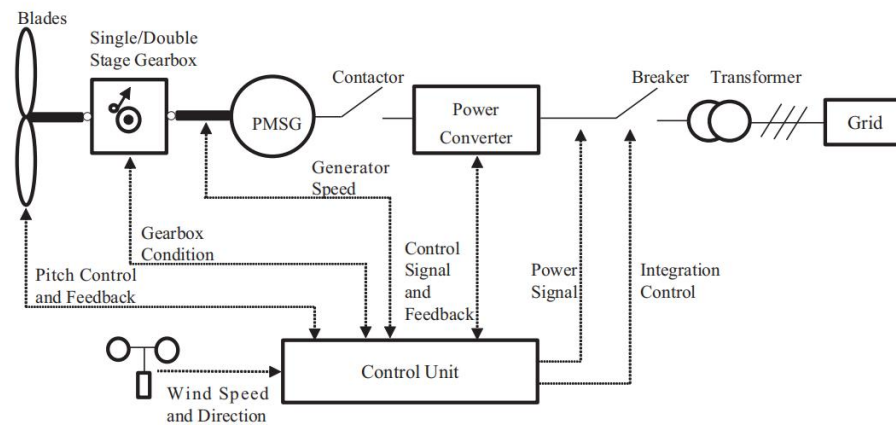


Figure 7 Structure frame of multibrid wind turbine [11]

Each configuration has its characteristics and applicable scenarios. The gearbox-equipped doubly-fed induction generator (DFIG) is widely used for its flexible control strategies and cost-effectiveness.

The main research object unit of this paper is the 2MW variable speed doubly-fed wind turbine developed by HARBIN ELECTRIC WIND POWER, model XE96D-2000, which is a H-bladed, upwind, blade pitch-variable, active yaw, wind turbine, with an impeller diameter of 96 meters, a rated capacity of 2000kW, and a design service life of 20 years. The machine utilizes a doubly-fed induction generator(DFIG), which allows the wind turbine to operate over a relatively wide range of wind turbine rotor speed variations to obtain more electrical energy.

2.1.2 Structure of the Wind Turbine

Wind turbines are sophisticated aerodynamic machines with complex control systems, consisting of various components, subcomponents, and parts such as foundation connection to the electric grid, tower access ladder, wind orientation control (yaw control), nacelle, generator, anemometer, electric or mechanical brake, gearbox, rotor blade, blade pitch control, rotor hub etc. Introduction to the main

subsystems and components of a gearbox-equipped Doubly-Fed Induction Generator wind turbine system are as follows:

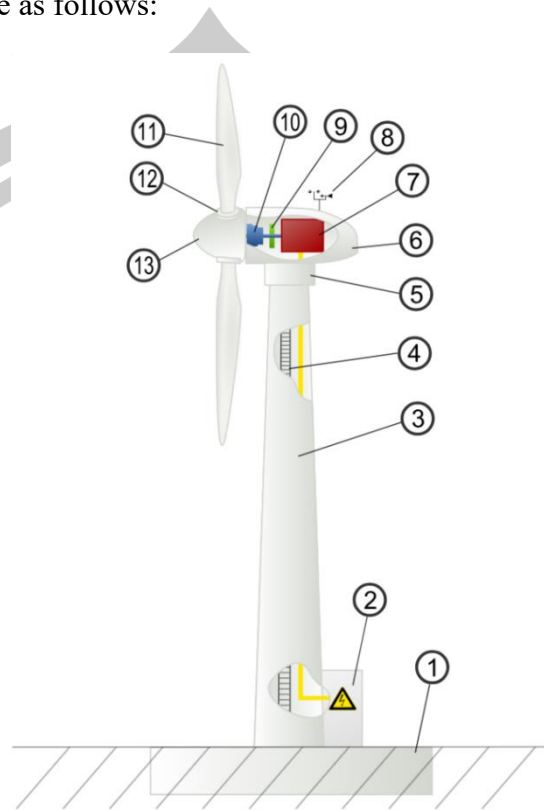


Figure 8 Structure and Components of a Wind Turbine[12]

①Foundation ②Connection to the electric grid ③Tower ④Access ladder ⑤Wind orientation control (yaw control) ⑥Nacelle ⑦Generator ⑧Anemometer ⑨Electric or mechanical brake ⑩Gearbox ⑪Rotor blade ⑫Blade pitch control ⑬Rotor hub

2.1.3 Main Fault Analysis of Wind Turbine

The stable operation of a wind turbine system relies on the components being in a healthy state and working closely with each other. However, because wind turbines often operate in more complex environments such as the sea, mountains, prairies and the Gobi, they are subjected to extreme weather conditions such as acid rain, blizzards and hailstorms, storms and sandstorms. As a result, these components have a greater or lesser probability of abnormalities or even failures during operation.

Depending upon external environment, and turbine condition itself, failure can occur anywhere in the wind turbine (Figure 9). However, the frequency and extent to which the particular failure affects the system performance varies. Gearbox failures

are less frequent but are most costly to repair and replace, whereas, electrical system of the wind turbine often fails, but can be easily repaired. A description of failure associated with turbine components are shown in Figure 10.

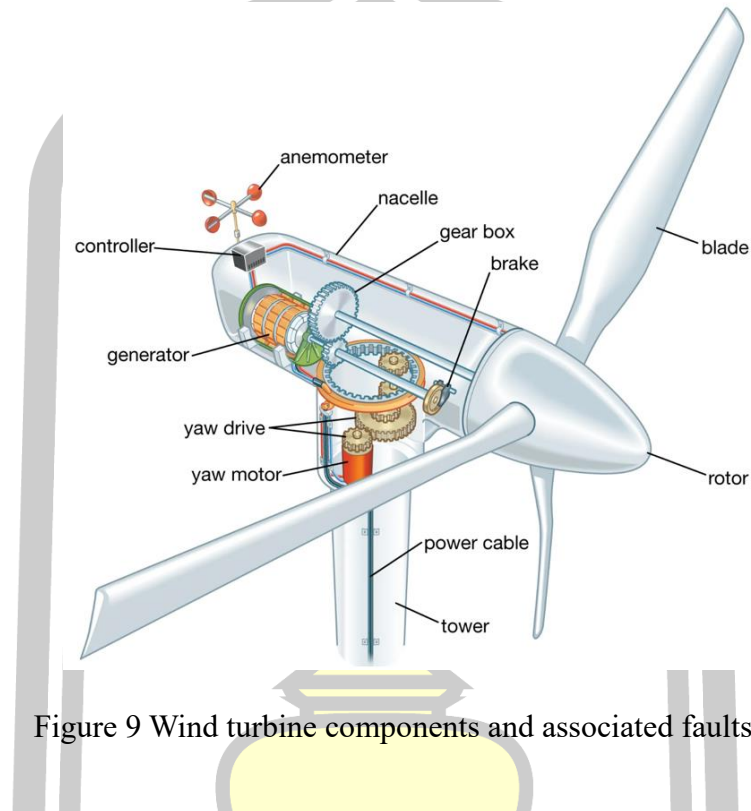


Figure 9 Wind turbine components and associated faults [13]

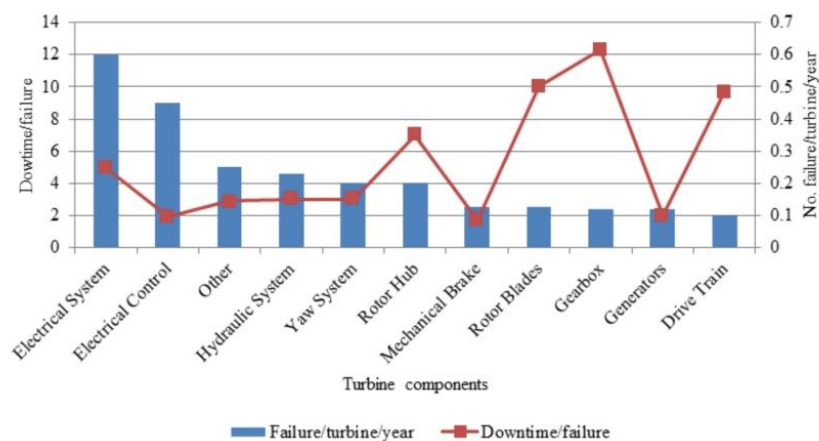


Figure 10 Percentage failures of wind turbine components

2.2.1 Structure of wind turbine blades

Figure 11 illustrates the internal structure of wind turbine blade, primarily consisting of the skin and the main spar. The primary elements of the main spar

include the spar cap and the shear web. The spar cap, primarily made of unidirectional glass fiber, serves as the first main component, while the shear web, constructed from bi-axial glass fiber, acts as the second. The shear web is specifically designed to handle shear-induced loads, thereby enhancing the blade's stability[15].

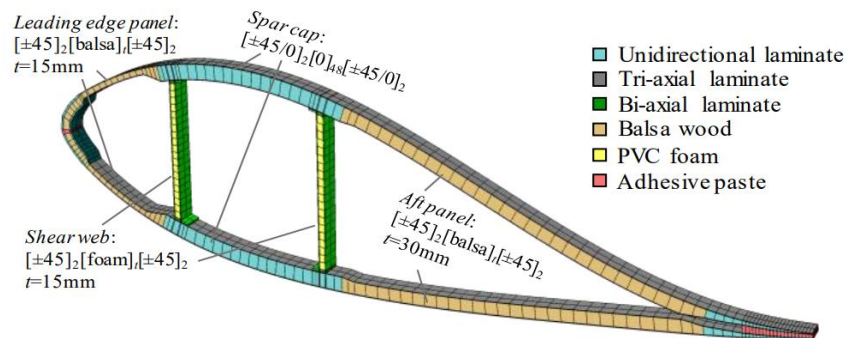


Figure 11 Internal Structure and Material Distribution Diagram of Wind Turbine Blade [15]

2.2.2 Main Faults and Diagnostic Methods of Wind Turbine Blades

Wind turbine blades are operated in harsh areas for a long time, which is prone to various surface defects and threatens the healthy operation of wind turbines. Since there are many factors that affect the health status of the blade, various types of failures occur in the blade. (as shown in the Figure 12).

i. Blade Cracking

Blades can develop cracks due to shifts in the center of gravity, thermal expansion and contraction, or impacts from external objects. Initially minor cracks, if left unchecked, can worsen under continual stress, leading to further crack expansion. In severe cases, this can result in blade breakage, posing significant risks.

ii. Localized Deterioration

Typically, the areas most susceptible to environmental wear and tear include the leading edge, the windward side, and the trailing edge of the blade. These regions are prone to surface cracking, abrasion, and fractures due to their direct exposure to environmental elements.



(a) Blade Cracking



(b) Blade Breakage



(c) Electrochemical Corrosion



(d) Blade Icing

Figure 12 Types of Wind Turbine Blade Faults [16]

iii. Electrochemical Corrosion

Galvanic corrosion of blades refers to the process of deterioration caused by the electrochemical reaction that occurs between the material and the environment of a wind turbine blade, usually made of metal or composite materials. This corrosion can occur when blades are exposed to moisture, oxygen, salts, or other corrosive substances, especially in harsh environments such as coastal areas or industrial areas. The electrochemical process results in a loss of material on the blade's surface, weakens its structure, and can lead to blade failure over time. This process is very important for wind turbines as it affects the efficiency, safety and longevity of the generator, and therefore requires regular maintenance and monitoring to prevent serious damage.

iv. Blade Icing

Many wind farms are located in regions of high altitude and latitude where low temperatures prevail throughout the year, leading to frequent blade icing. This icing process occurs in three phases: initial icing, noticeable icing, and severe icing. Thus, identifying and addressing icing at its initial stage through timely fault detection and implementing appropriate de-icing methods are crucial. This can prevent further ice accumulation and ensure the continuous operation of wind turbines. This document further explores and analyzes the phenomenon of blade icing.

2.2.3 Factors Contributing to Wind Turbine Blade Icing

Wind turbine blade icing refers to the phenomenon where moisture accumulates and freezes on the surface of turbine blades under conditions of low temperature and high humidity. Icing has negative effects on the operation and performance of wind turbines, including reduced aerodynamic efficiency of the blades, increased load and vibration, decreased power generation, and higher maintenance and operational costs.

The ice formed on turbine blades can vary in size, shape, and characteristics depending on several factors such as climate conditions, wind speed, and humidity. These factors are interrelated and interact with one another, making the causes of blade icing complex and diverse.

The possible factors contributing to blade icing include:

- i. Temperature. It is a key factor influencing the heat exchange characteristics of water droplets. The lower the temperature, the greater the tendency for droplets to freeze. At sufficiently low temperatures, droplets may freeze directly on the blade surface, forming rime ice or glaze ice.
- ii. Wind Speed. It affects the convective heat transfer rate on the blade surface. The higher the wind speed, the more ice accumulates on the blades. High wind speeds increase the impact velocity of water droplets and the rate of convective heat transfer, promoting the formation of thicker ice layers on the blade surface.

iii. Liquid Water Content. The higher the liquid water content in the atmosphere, the thicker the ice on the blades. When the air contains a large amount of liquid water, droplets gather on the blade surface and freeze, forming a thick ice layer.

iv. Droplet Diameter. The influence of droplet diameter on blade icing is similar to that of liquid water content. Larger droplets result in thicker ice layers. Large water droplets significantly affect the adhesion and freezing process on the blade surface.

v. Airfoil Shape. The airfoil of the turbine blade determines its aerodynamic characteristics. Different airfoil shapes influence how ice forms and the type of ice that accumulates. Poorly designed airfoils may be more prone to icing.

vi. Surface Roughness. The roughness of the blade surface affects both aerodynamic properties and heat transfer. A rougher surface increases the likelihood of ice adhering to the blade.

The factors influencing blade icing are not limited to the six listed above. Due to the complexity of the causes, traditional methods rely on a comprehensive analysis that takes multiple factors into account, often combining them with expert knowledge to detect icing. However, building such mechanism-based models is challenging and often lacks generalization ability. Data-driven approaches can help alleviate these limitations.

2. Diagnostic Methods of Wind Turbine Blades

Addressing the issue of icing on wind turbine blades promptly can significantly reduce further fault development, thus extending the lifespan of the blades, enhancing energy production efficiency, and decreasing maintenance expenses. Currently, various technical approaches are employed to monitor icing conditions on wind turbine blades, including acoustic emission monitoring, vibration analysis, drone surveillance, non-destructive evaluation, and SCADA system data analysis.

i. Acoustic Emission Monitoring Technology

Acoustic emission monitoring is particularly effective for early-stage or potential faults due to its high signal-to-noise ratio, making it ideal for environments with significant background noise. For instance, research documented in Reference[17]

enhances the detection of crack propagation in wind turbine blades by optimizing wavelet basis function bandwidth parameters through minimum Shannon entropy. This process calculates the wavelet rescaled spectrum, boosting the time-frequency resolution of acoustic emission signals related to cracks, thus enabling precise localization and focused monitoring of these defects. Moreover, as detailed in Reference [18], applying wavelet analysis to decompose the acoustic emission signals of wind turbine blades helps in extracting vital features such as the maximum spectral coefficient layer and reconstructed signal spectra, aiding in the accurate identification of surface damages on the blades.

ii. Vibration Monitoring Technology

Vibration monitoring stands out as the foremost method for diagnosing faults in engineering equipment and is prevalently utilized in commercial wind turbine monitoring systems. This method employs vibration sensors to translate the anomalies in wind turbine blades into discernible vibration signals. Commonly used sensors include accelerometers, velocity, and displacement sensors, with accelerometers being the most favored due to their broad frequency range of 1 to 30,000 Hz, making them particularly effective for wind turbine fault diagnosis [19].

Reference [20] introduces an innovative approach where a piezoelectric thin-film sensor is positioned between the hammer head and body. This setup, upon impacting the wind turbine blade, transforms mechanical vibrations into electrical signals, thereby enabling accurate assessment of damage depth by comparing the generated voltage against known standards.

iii. Unmanned Aerial Vehicle (UAV) Monitoring Technology

In recent times, the industrial sector has witnessed rapid advancements in computer vision algorithms, particularly in detecting defects on wind turbine blades. These methods, leveraging image capture, have garnered extensive interest globally. Employing an optimized Canny edge detection method [21] enhances the precision in identifying surface cracks. Further refinement is achieved by integrating Canny with Sobel operators, improving the identification accuracy.

Furthermore, according to Reference [22], integrating GABOR filter and OSTU threshold segmentation methods enhances the image detection algorithms for blade defects. This hybrid approach facilitates a more detailed examination of defect characteristics, significantly boosting the detection algorithm's accuracy.

iv. Non-destructive Monitoring Technology

Non-destructive monitoring encompasses techniques like ultrasound, infrared thermography, and X-ray for inspecting wind turbine blades. These methods utilize contactless detectors to reflect various signals from the blades, identifying potential defects and assessing damage levels. However, despite their effectiveness, these non-destructive techniques often come with high costs and usually require turbine shutdown during inspection. Typically applied as static tests before turbines leave the factory, the real-time fault detection of operational wind turbine blades remains challenging. Moreover, for most non-destructive techniques, experienced inspectors are essential for accurate defect analysis.

v. SCADA Detection Technology

In recent years, the Supervisory Control and Data Acquisition (SCADA)-based approach has been recognized as a cost-effective method, and has been extensively studied^[27]. For large wind turbines, SCADA systems have become standard and are dedicated to collecting operating and status data to provide rich environmental, electrical, and mechanical information about the operating status of wind turbines. At present, SCADA data has been widely used in the main components of wind turbines, including the condition monitoring of blades[23, 24], sensors[25], generators[26], and gearboxes [27].

Zhao[28] monitored the wind turbine blades by detecting the voltage, using the data from the voltage sensor to find the best detection sensitivity in a variety of ways. However, there are certain defects in the application of this research method, and it is not perfect in online detection. Jiang[29] used the B-spline surface to study the quality inspection process of offshore wind blades, focusing on a protocol for the location of control points, increasing the initial surface fitness and thus reducing the time required for the optimizer to converge to the ideal position. Lee[30] combined random

forests (RFs) with extreme gradient boosting (XGBoost) to establish a data-driven framework for wind turbine fault detection. In its proposed design, RFs are used to rank features by importance. Compared with the current emerging technologies, RFs have certain shortcomings, such as algorithm computing power, computing speed, and processing of mutually exclusive characteristics. Jiang Guoqian et al. [31] from Yanshan University proposed a denoising autocoding network model integrating time series information to model and mine the spatiotemporal correlation of multivariable SCADA data, which reduced the influence of input noise and interference on the model and improved the accuracy of fault detection. A wavelet-based fully convolutional neural network model [32] was proposed to detect blade icing faults, and the SCADA data of wind turbine blade icing was mined through the wavelet multi-scale characteristics combined with the spatial feature extraction characteristics of the convolutional neural network, and the correlation between spatial variables was extracted, and a good diagnostic accuracy was obtained through the actual wind field blade icing data. However, wavelet fully convolutional neural networks do not take into account the inherent but important time dependence of SCADA data. Zhang, et al. [33] studied a condition monitoring method for wind turbines based on data analysis, and proposed a condition monitoring method based on the correlation of SCADA variables, and at the same time, to evaluate the overall operation of the wind turbine under different working conditions. A DRL-based UC model [34] was proposed by formulating the problem as an MDP and training it offline using PPO-D, achieving faster computation and robustness against wind uncertainty. Simulation on 5-unit and IEEE 118-bus systems validated its efficiency and generalization.

2.3 Overview of the SCADA System

The development of SCADA systems, as depicted in Figure 13, has undergone significant evolution since their inception in the 1950s, initially aimed at monitoring and controlling power and water systems. Advancements in computer and network communication technologies have transformed SCADA into comprehensive, multifunctional systems capable of remote monitoring, data acquisition, process control, and fault diagnosis. By the 1990s, the integration of Internet technologies and advancements in information technology enhanced SCADA systems with open

standards and network protocols, boosting their flexibility and compatibility. Entering the 21st century, the emphasis shifted towards real-time data processing, analytics, and decision support, leading to widespread application across various sectors like electric power, water treatment, petrochemicals, and manufacturing.



Figure 13 Display interface for SCADA system [35]

Specifically, for wind farms, SCADA systems are crucial for the efficient operation and management of turbine clusters. They provide functionalities such as data gathering, analysis, storage, and control, along with advanced features for predictive maintenance, such as power and wind speed forecasts, condition assessments, and fault diagnostics. This evolution has significantly improved wind farm operational efficiency.

In China, the progress in wind power SCADA systems began relatively late and is still developing. International manufacturers like Vestas, General Electric, and Siemens have created proprietary systems suited to their models, prompting the rise of third-party systems like those from Garrad Hassan in the UK. Chinese companies, including Goldwind Technology and Sinovel Wind Power, are also making strides in SCADA research and development, indicating a growing domestic market and technological advancement in this area.

2.3.1 Description of Internal Data in SCADA Systems

The wind turbine is a whole integrated system composed of multiple components, and there is interaction and coupling between the various components on the unit at the same time, and when an abnormality occurs in a certain component of the wind turbine, it is very easy to cause the failure of the whole wind turbine system.

SCADA system is a large-scale wind turbine pre-installed data acquisition and monitoring and control system, the system collects the time series data of each sensor pre-installed in the time dimension, the dimension of the data is higher, the sampling interval is generally a few seconds to a few minutes, and the content of the record is the basic situation of the operation of the wind turbine itself and some change information of the surrounding environment. Among them, the information of each part of the generator speed, active power, motor temperature and other parts such as gearbox and fan blades are recorded in the state of the fan itself; The external information is recorded with basic information such as the ambient temperature, wind speed, blades and the overall pressure of the fan.

i. Wind Condition Parameters

This category includes variables such as wind speed and wind direction, which are typically measured using anemometers and wind vanes. These parameters are essential as they represent one of the primary driving factors for wind turbine operations. They have a significant correlation with other operational parameters like power output and pitch angle.

ii. Energy Conversion Related Parameters

These are variables integral to the process of energy conversion in wind turbines. They include active power, generator speed, pitch angle, and pitch speed. The accumulation of ice on the blades has been demonstrated to adversely affect the power performance of the turbines.

iii. Temperature Parameters

This group encompasses temperatures from critical components such as the pitch motor and battery cabinet, as well as the internal cabin temperature and external environmental temperature. Blade icing can disturb the aerodynamic force and mass balance, leading to fluctuations in the temperatures of key components like the pitch motor and battery cabinet.

iv. Vibration Parameters

These include measurements of cabin acceleration in the X and Y directions. It has been documented that ice accumulation on the blades can increase the levels of vibration within the cabin, indicating the onset of icing issues.

The SCADA data contains multi-parameter characteristics related to the icing state of the blades. However, monitoring the operating status of the blades with SCADA data is an extremely challenging task, as SCADA data often does not contain measurements related to the condition of the wind turbine blades, such as blade vibration and mechanical loads. Second, the existing SCADA variables do not directly reflect the health of the wind turbine blades. Moreover, most of the existing methods rely on artificial feature extraction and shallow machine learning methods, and these methods have shallow network structure and limited modeling ability, and cannot achieve satisfactory performance. On the other hand, because wind turbines are driven randomly by the wind and are subject to various disturbances and noises, SCADA data often have non-stationary characteristics. Due to the multivariate, multi-scale, dynamic time-varying and non-stationary characteristics of SCADA data, the existing methods cannot deal with these problems well. In SCADA, most methods only consider the global characteristics of each variable of the sensor, ignoring the local characteristics of the different variables. In view of the strong temporal correlation of SCADA data, machine learning is introduced in order to fully explore the correlation information in the temporal dimension between different data.

2.4 Intelligent Algorithms for Wind Turbine Icing Prediction

Wind turbine blade icing poses significant operational challenges, particularly in cold-climate regions where the accumulation of ice on blades can lead to performance degradation, increased mechanical stress, and even system failure. To address these risks proactively, the development of accurate and efficient prediction models has become a key research area. With the advent of large-scale SCADA (Supervisory Control and Data Acquisition) systems, a wealth of operational data has become available, offering new opportunities for data-driven fault diagnosis. Intelligent algorithms—especially machine learning and deep learning techniques—have demonstrated considerable potential in modeling the complex, nonlinear, and temporal patterns embedded in such data. This section introduces the foundations of

machine learning and reviews the state-of-the-art approaches used for wind turbine icing prediction. It further analyzes how these algorithms are adapted and optimized for time-series data to enhance prediction performance in real-world applications.

2.4.1 Machine Learning algorithm

Machine learning algorithms are computational methods that enable systems to automatically learn patterns and make decisions or predictions based on data, without being explicitly programmed for specific tasks. The concept of machine learning originates from the broader field of artificial intelligence (AI), with its foundations laid in the mid-20th century. Arthur Samuel, an American pioneer in AI, first coined the term "machine learning" in 1959, defining it as the "field of study that gives computers the ability to learn without being explicitly programmed."

The development of machine learning has evolved through several stages: from early rule-based systems to modern data-driven approaches powered by statistical theory and computational advancements. With the exponential growth of data and computing power, various algorithms—such as decision trees, support vector machines, neural networks, and ensemble methods—have been developed to handle tasks in classification, regression, clustering, and anomaly detection. In recent years, deep learning, a subfield of machine learning involving multi-layered neural networks, has significantly advanced the capabilities of algorithmic learning, particularly in image recognition, natural language processing, and time-series prediction.

Today, machine learning algorithms are widely applied across diverse domains such as healthcare, finance, energy, and engineering, driving innovation through data-informed decision-making.

2.4.2 Machine Learning Approaches for Wind Turbine Icing Prediction

With the increasing deployment of wind energy systems in cold climate regions, blade icing has become a critical issue affecting both safety and operational efficiency. Consequently, predictive modeling of wind turbine icing using SCADA (Supervisory Control and Data Acquisition) data has emerged as a prominent research focus. A range of machine learning techniques—both traditional and deep learning-based—have been explored for this purpose. This section provides a critical review of

relevant studies, highlighting model architectures, capabilities, and prediction performance.

Early work, such as that reported in [36], employed a shallow or conventional neural network for icing prediction, yielding an accuracy of approximately 83% on the evaluated dataset. Although shallow networks are computationally efficient, they lack the capacity to effectively model the nonlinear dependencies inherent in SCADA data and fail to capture the temporal dynamics characteristic of time-series inputs.

To address this limitation, Elman Neural Networks (ENN) were introduced in [37, 38], achieving prediction accuracies of 95% and 98%, respectively. ENNs incorporate recurrent connections within the hidden layer, allowing information from previous time steps to influence the current output. This design makes ENNs well-suited for sequence modeling. Training is typically conducted using the backpropagation (BP) algorithm to minimize prediction errors.

Deep Belief Networks (DBNs), probabilistic generative models that are pre-trained in an unsupervised manner and fine-tuned using supervised learning, were investigated in [39]. A reported prediction accuracy of around 94% was achieved in [40]. However, DBNs lack mechanisms for modeling temporal dependencies and often require prolonged training times due to their complex training procedures, especially on large datasets.

In [41], a one-dimensional Convolutional Neural Network (1D-CNN) was applied to icing prediction tasks. Designed for sequential or time-series data, the model achieved a notable prediction accuracy of 97% without requiring explicit feature selection or extraction. Nevertheless, while 1D-CNNs capture local temporal patterns, they do not explicitly model inter-feature relationships.

Traditional machine learning approaches have also been investigated. For instance, [42] proposed a Decision Tree (DT) model coupled with manual feature selection, which resulted in an impressive prediction accuracy of 99%. Similarly, [43]. Although such models are robust and interpretable, they do not consider the sequential nature of SCADA data.

In [44], A global model was formed by aggregating local updates, resulting in an accuracy of approximately 70%. While FL addresses data privacy concerns, it also fails to incorporate temporal structures within time-series data.

In[45], clustering was initially used to divide the data samples into multiple classes to reduce the computational complexity, and then the machine learning approach of Light Gradient Boosting Machine (LightGBM) was applied for refinement of the outcome. In general, the accuracy of combined or ensemble models is relatively high as they take into consideration both the temporal and spatial aspects of data.

Xu et al. [46] introduced a prediction method based on a support vector machine (SVM) optimized via particle swarm optimization (PSO), which improved prediction accuracy through refined data preprocessing and feature engineering.

Meng et al. [47] utilized recursive feature elimination combined with random forest (RFE-RF) and SVM to monitor icing. The integration of model fusion (stacking) allowed for accurate fault diagnosis and robust performance.

Tao et al. [48] incorporated physical icing characteristics into a Stacked-XGBoost model. This hybrid approach considered both short- and long-term icing effects, leading to improved diagnostic precision.

Extreme Gradient Boosting (XGBoost), a highly optimized ensemble learning algorithm, was used in [49, 50]. By iteratively training weak learners with a regularization-enhanced objective function, XGBoost achieved a high accuracy of 99% on limited datasets. However, like other tree-based methods, it lacks temporal modeling capability.

To model the complex interactions among SCADA features as well as temporal dependencies, a Graph Temporal Attention Network (GTAN) was proposed in [51]. This approach constructs a graph where nodes represent SCADA features and edges encode their relationships. A Graph Convolutional Network (GCN) was employed to evolve feature representations, while temporal dependencies were captured through an attention mechanism. An average prediction accuracy of 75% was reported across three datasets, reflecting a 5% improvement over baseline models such as CNN and LSTM.

Classical machine-learning (ML) algorithms remain attractive for their computational efficiency and transparent decision rules [52, 53]. XGBoost combined with exponentially weighted moving averages (EWMA) provides early-warning capability by tracking deviations from learned normality [54]. Semi-supervised tri-

training with XGBoost alleviates label scarcity and class imbalance via pseudo-labelling and Focal Loss [55], and RFECV-guided TSVM pipelines for small-sample scenarios [56] further illustrate the breadth of ML strategies available.

More recently, hybrid and ensemble models have gained attention. These architectures typically integrate one subnetwork designed to capture temporal dynamics (e.g., RNNs, GRUs) with another network that focuses on learning inter-feature relationships (e.g., CNNs, GATs). Various integration strategies—such as serial, parallel, or attention-based fusion—have been explored to simultaneously leverage spatial and temporal information in SCADA time-series data.

2.4.3 Deep learning algorithm

Deep learning algorithms are a subset of machine learning methods based on artificial neural networks with multiple layers (hence "deep"). These algorithms are designed to automatically extract hierarchical features from data, enabling the modeling of complex, non-linear relationships. Deep learning is inspired by the structure and function of the human brain, particularly the interconnection of neurons, and it aims to mimic human-like learning in tasks such as perception, pattern recognition, and decision-making.

The conceptual foundation of deep learning traces back to the 1940s and 1950s with the development of the first artificial neuron model—the McCulloch-Pitts neuron—and the perceptron. However, due to limitations in computational resources and algorithmic efficiency, early neural networks were restricted in capability. It wasn't until the 2000s, with the advent of powerful GPUs, the availability of large-scale datasets, and the development of more advanced training techniques (such as backpropagation and dropout), that deep learning began to flourish.

Notable deep learning algorithms include:

Convolutional Neural Networks (CNNs): Primarily used for image and spatial data processing.

Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM): Designed for sequential and time-series data.

Gated Recurrent Units (GRUs): A simplified alternative to LSTM with similar performance.

Autoencoders: Used for unsupervised learning, data compression, and feature learning.

Generative Adversarial Networks (GANs): A framework for generating new data samples similar to a given dataset.

Transformers: State-of-the-art architecture for natural language processing and sequential modeling tasks.

Deep learning has demonstrated remarkable success across various domains such as computer vision, speech recognition, medical diagnostics, and renewable energy forecasting. Its ability to model high-dimensional data and adapt to diverse application scenarios makes it a cornerstone technology in modern artificial intelligence research.

2.4.4 Deep Learning in Wind Turbine Blade Icing Prediction

Owing to its robust capability to model complex and nonlinear systems, deep learning has been increasingly applied in the prediction and diagnosis of wind turbine blade icing. Recent studies highlight a clear trend towards integrating multiple models to improve both the accuracy and robustness of icing forecasts.

Several significant contributions in this area include:

Kreutz et al.[36] proposed a convolutional neural network (CNN) model with dual-input channels and one-dimensional convolution filters. This approach effectively identified blade icing events using historical SCADA data and meteorological forecasts.

Yang et al. [57] combined SCADA data with random forest and k-nearest neighbors (KNN) algorithms to enhance feature selection. The resulting hybrid model significantly outperformed traditional backpropagation (BP) neural networks in icing diagnosis.

Several recent studies have advanced the integration of deep learning models for icing prediction:

Dazhong et al. [58] proposed a deep fully connected neural network, which achieved superior prediction accuracy when compared with conventional KNN, SVM, and unoptimized BP neural networks.

Xiong et al. [59] developed a short-term icing prediction model that combines Bidirectional LSTM (Bi-LSTM) and SVM, with PCA-based dimensionality reduction and multi-group data training, achieving high predictive accuracy.

Tao et al. [60] employed a hybrid CNN-Attention-GRU model with a focal loss function, which significantly improved both the accuracy and the F1-score of icing prediction.

In addition, several advanced architectures have further enhanced the handling of imbalanced and high-dimensional SCADA data:

Spatiotemporal attention mechanisms [61] and the GTAN model [51] have been utilized to extract and weight temporal and spatial features, enhancing the robustness of icing predictions.

Tian et al. [62] introduced a Multi-level Convolutional Recurrent Neural Network (MCRNN) specifically tailored for blade icing detection.

Wang et al. [63] proposed a Wavelet-LSTM model that integrates wavelet-based multi-scale feature extraction with temporal modeling, effectively capturing both local and global time-series patterns to enhance fault detection performance.

In [64], a LSTM network was utilized to predict future feature values followed by a Support Vector Machine (SVM) to determine the occurrence of icing based on the predicted values. In the studies reported in [65], a CNN and a Gated Recurrent Unit (GRU) network were stacked together. In [66], an attention mechanism was added to this combined network. In the latest paper in [45], Temporal Convolutional Network (TCN) was utilized to capture temporal patterns and dependencies within SCADA data by using one-dimensional convolutional layers. TCN employs dilated convolutions to increase the receptive field without increasing the number of parameters, thus capturing long-term dependencies more effectively. A prediction accuracy of 80% was obtained for the dataset examined.

These advancements reflect the growing importance of deep learning in the field of wind turbine blade icing prediction, offering more adaptive, reliable, and accurate solutions in real-world operational environments.

Deep-learning (DL) methods now dominate the state of the art because they can uncover nonlinear relations hidden in high-dimensional SCADA streams, reducing reliance on manual features and auxiliary sensors [67] [68]

[69]. Representative examples include a rotor-icing framework that marries dimensionality-reduced sensor data with power-curve embeddings to achieve 96.4 % classification accuracy while lowering power-loss estimation error by 18.7 % [70] ; an unsupervised β -VGATAE model that attains 90.9–93.4 % accuracy by combining a β -VAE with graph-attention networks [71] ; a time-attention CNN that surpasses existing methods on SCADA data [72] ; and a ReliefF-enhanced 1D-CNN-SBiGRU architecture whose weighted accuracy exceeds traditional SVM, CNN, and BiLSTM baselines by 14.44–43.08 % [73] . Nevertheless, DL models still grapple with class imbalance, limited generalization, and real-time constraints, motivating work on interpretability, data augmentation, and lightweight inference.

2.5 Summary

The internal structure of motor blades and the types of common defects provide the basis for understanding blade failures. Next, a detailed comparison of the current leading blade condition monitoring technologies, including acoustic emission monitoring, vibration analysis, drone inspection, non-destructive testing, and SCADA-based methods, was made, and the advantages and disadvantages of each in terms of data processing complexity and installation and maintenance costs were evaluated. The chapter further elaborates on the method of extracting fault features, and examines the principles and benefits of using various data processing techniques such as time-frequency analysis. Then, the SCADA detection technology for blade fault detection was analyzed in depth, using artificial neural networks, machine learning, and deep learning, and comparing their capabilities in processing large and complex blade operation data. Finally, deep learning was selected as the preferred technology, which laid a solid theoretical foundation for the construction of blade fault diagnosis model and data preprocessing. This series of discussions not only deepened the understanding of wind turbine blade fault diagnosis, but also provided strong technical support for the implementation of wind turbine fault detection and prevention strategies.

Chapter 3 Research Methodology

As described in Chapter 2, the SCADA data contains multi-parameter characteristics related to the icing state of the blades. However, monitoring the operating status of the blades with SCADA data is an extremely challenging task, as SCADA data often does not contain measurements related to the condition of the wind turbine blades, such as blade vibration and mechanical loads. Second, the existing SCADA variables do not directly reflect the health of the wind turbine blades. Moreover, most of the existing methods rely on artificial feature extraction and shallow machine learning methods, and these methods have shallow network structure and limited modeling ability, and cannot achieve satisfactory performance. On the other hand, because wind turbines are driven randomly by the wind and are subject to various disturbances and noises, SCADA data often have non-stationary characteristics. Due to the multivariate, multi-scale, dynamic time-varying and non-stationary characteristics of SCADA data, the existing methods cannot deal with these problems well. In SCADA, most methods only consider the global characteristics of each variable of the sensor, ignoring the local characteristics of the different variables. In view of the strong temporal correlation of SCADA data, deep learning is introduced in order to fully explore the correlation information in the temporal dimension between different data.

3.1 Data Preprocessing

This study employs SCADA data provided by a wind power enterprise in Hunan Province, encompassing the operational behavior of ten wind turbines over the course of 2020 to 2021 across various seasonal conditions. These data, recorded at a frequency of 1 Hz via a real-time monitoring platform (Figure 13), form the foundational input for model training. The acquisition configuration of each turbine is detailed in Table 2.

Table 2 Actual Corporate Data Overview

Location A (Collection Frequency:1s)		
Turbine ID	Data Samples	Collection Time
wind turbine 28	44639	2021.8.7-2021.9.6
wind turbine 29	44639	2021.8.7-2021.9.6
wind turbine 39(1)	44373	2021.8.1-2021.8.31
wind turbine 39(2)	44639	2021.12.15-2022.1.14
wind turbine 45	44373	2021.8.1-2021.8.31
Location B (Collection Frequency:1s)		
Turbine ID	Data Samples	Collection Time
wind turbine 4	42822	2021.3.6-2021.4.8
wind turbine 10	8201	2021.1.21-2021.1.27
wind turbine 14	44581	2021.10.14-2021.11.13
wind turbine 22	42674	2021.5.13-2021.6.12
wind turbine 28	44639	2021.8.7-2021.9.6
wind turbine 29	44639	2021.8.7-2021.9.6

Given the high dimensionality of raw SCADA data—containing hundreds of variables—most features were irrelevant to blade icing. Through consultations with wind turbine manufacturers and domain experts, 26 highly relevant continuous variables were identified. These features were categorized into four main groups: wind field indicators, yaw system dynamics, blade pitch control parameters, and environmental conditions (Table 3).

To ensure high-quality inputs for model learning, a multi-stage preprocessing pipeline was implemented. Short-term missing values were imputed using linear interpolation, while long-duration sensor failures led to data segment exclusion. Standardization and normalization procedures were then applied to the selected features to eliminate scale disparities caused by heterogeneous measurement units (e.g., wind speed in m/s, power in kW, temperature in °C). This step was crucial for accelerating convergence and avoiding feature dominance during model training.

Table 3 Partial Monitoring Parameter Information of Wind Turbine Units

Serial Number	Variable Name	Serial Number	Variable Name
1	Wind_speed	14	Pitch1_moto_tmp
2	Generator_speed	15	Pitch2_moto_tmp
3	Power(kw)	16	Pitch3_moto_tmp
4	Wind_direction(°)	17	Acc_xx
5	Wind_direction_mean	18	Acc_yy
6	yaw_position	19	environment_tmp
7	yaw_speed	20	int_tmp
8	Pitch1_speed	21	Pitch1_ng5_tmp
9	Pitch2_speed	22	Pitch2_ng5_tmp
10	Pitch3_speed	23	Pitch3_ng5_tmp
11	Pitch1_angle	24	Pitch1_ng5_DC
12	Pitch2_angle	25	Pitch2_ng5_DC
13	Pitch3_angle	26	Pitch3_ng5_DC

The data utilized in this study was obtained from the SCADA system; however, it contained anomalies, redundant entries, and missing values. In order to mitigate the impact of these issues on model training and evaluation, the following integrated data preprocessing process was implemented:

(1) Data reading and preliminary screening

Iterate through all files in the data directory and perform preliminary screening based on specific keywords in the file names. Only eligible CSV files are retained for further analysis.

(2) Data Labeling and Missing Value Filling

Experienced wind turbine engineers labeled the raw data as either "normal" or "iced". To ensure data continuity and integrity, missing values were filled in using the fillna() method. This prevents gaps in the time series from affecting subsequent analysis and model training.

(3) Data Standardization and Segmentation

Since SCADA data are time series data collected from sensors at fixed intervals, they need to be standardized and segmented to meet the input requirements of the CNN-BiGRU wind turbine blade icing prediction model:

- Standardization: Min-Max standardization method is used to scale the feature values to the range of $[0, 1]$ to reduce the impact of unit and magnitude differences between different features on the model. The standardization formula is as follows Eq. (3.1).

$$X' = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \quad (3.1)$$

Where X' represents the pre-processed feature data, X refers to the original data, and $X_{\max}$ and $X_{\min}$ indicate the maximum and minimum feature values, respectively.

- Segmentation: Segment the normalized time series data into fixed length segments (500 data points per segment) to fit the input format of the model.

(4) Icing data processing

The same preprocessing steps, including segmentation and normalization, were applied to the icing data. In addition, fault data from other data sources are extracted, preprocessed using a consistent methodology, and integrated into the overall dataset.

Through the preprocessing steps described above, the SCADA data is converted into a structured, clean, and standardized format suitable for model training and prediction, thereby improving the reliability and accuracy of the wind turbine blade icing prediction system.

3.2 Feature Engineering

Following data preprocessing and segmentation, effective feature engineering is critical to ensure high-performance classification of blade icing states. In order to eliminate the influence of redundant parameters and highlight the most informative variables, this section introduces a feature enhancement strategy that integrates Random Forest-based feature selection with CNN-based power characteristic enhancement. The complete blade icing detection workflow is depicted in Figure 14.

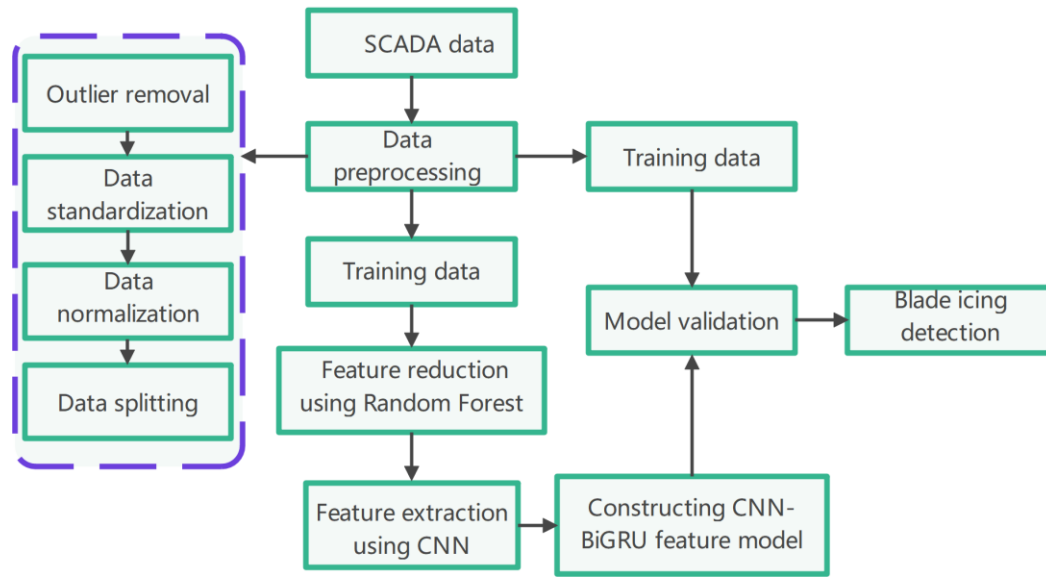


Figure 14 Blade Ice Detection Flowchart

3.2.1 Random forest classification theory

In machine learning, Random Forest algorithms are frequently utilized for classifying and extracting features from large datasets, offering robust data handling capabilities. Concurrently, due to their significant data fitting capacity, Artificial Neural Networks are extensively applied in fault detection research.

Random Forest Classification (RFC) involves an ensemble of multiple decision tree models, denoted as [74], with the parameter set $\{\Theta_k\}$ consisting of independently and identically distributed random vectors. The fundamental concept behind Random Forest classification includes: (1) Utilizing bootstrap sampling to extract N samples from the initial training set, ensuring these samples have the same size as the original dataset. (2) Each sample is linked to a unique decision tree model, resulting in N distinct classification outcomes. (3) The final classification is determined by conducting a majority vote across the N outcomes.

To address the multidimensional features present in wind turbine blade icing recognition and to enhance the identification model's efficiency, this section employs the Random Forest approach to simplify the multidimensional features found in SCADA data. The flow of this classification method is depicted in Figure 15.

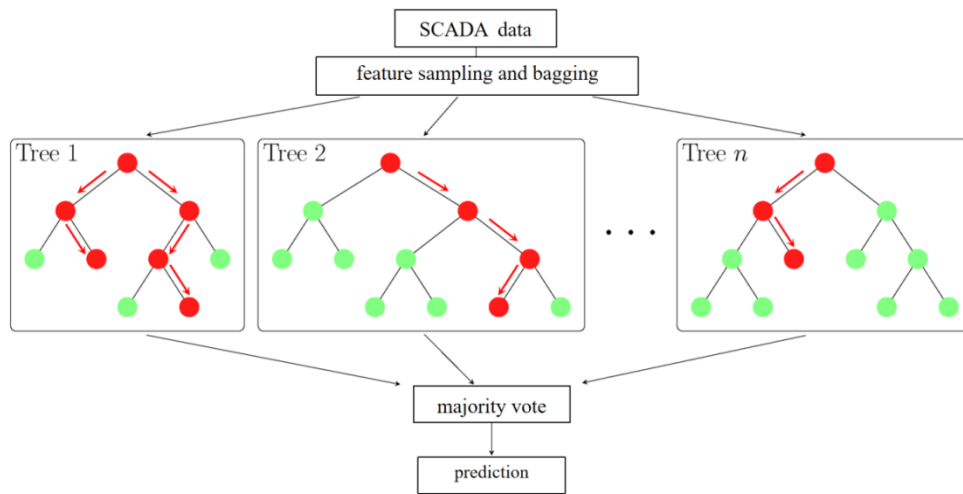


Figure 15 Random Forest Classification Algorithm[74]

3.2.2 CNN-Based Power Feature Enhancement

Convolutional Neural Networks (CNN) [75] is a neural network with a feedforward structure, which is characterized by increasing the number of convolutional neural network layers and using convolution operations to extract the global features of the data, as shown in Figure 16. In the convolutional neural network, the feature extraction of the input signal is completed by the filtering level, and the classification of the learned feature parameters is completed by the classification level, and the two layers of parameters can be subdivided into four parts: convolutional layer, pooling layer, activation layer and fully connected layer. The first three component filter levels and the last one are classification levels.

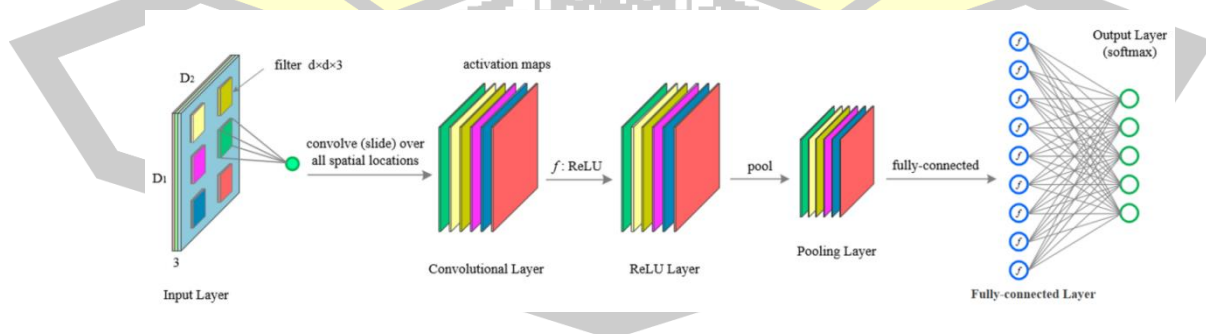


Figure 16 Convolutional Neural Network Algorithm[75]

1.Convolutional layer

Convolutional layer operations generally use convolutional kernels, an input image, which is divided into small pieces, and each small piece after the division is convoluted separately, and the corresponding features will be after the operation. This operation reduces the number of parameters in the entire network, prevents the occurrence of overfitting, and reduces the memory occupied by the entire network, resulting in a smaller amount of computation. Generally in practice, because the convolution operation needs to flip the convolution kernel over during backpropagation, the correlation operation is generally used to replace the convolution operation, which is as shown in the formula (3.2):

$$y^{l(i,j)}=K_i^l * X^{l(j')} = \sum_{j'=0}^{W-1} K_i^{l(j')} X^{l(j+j')} \quad (3.2)$$

where $K_i^{l(j')}$ is the j' weight of the i th convolutional kernel of l -layer; $X^{l(j+j')}$ Represents the j -th convoluted local region of the i th convolution kernel of l -layer; W represents The width of the convolution kernel.

2.Activation Layer:

After completing the convolution operation, a Logits value is generated, which requires a nonlinear transformation; this is where activation functions are used. The commonly utilized activation functions include the following types:

i. Sigmoid functions:

The output of the Sigmoid function has a certain range, but because the Sigmoid function has soft saturation and offset, which will lead to the problem of vanishing gradient, the Sigmoid function is generally no longer used as the activation function (see in the Eq. 3.5) of the model.

$$f(x)=\frac{1}{1+e^{-x}} \quad (3.3)$$

ii. Tanh functions:

$$f(x)=\frac{1-e^{-2x}}{1+e^{-2x}} \quad (3.4)$$

The Tanh function converges faster than the sigmoid function, and because its mean value is 0, which compensates for the offset of the sigmoid, the performance is

better than that of the sigmoid, but as the number of Tanh function layers increases, the gradient disappears, so it is rarely used in neural networks today.

iii. ReLU functions:

$$f(x) = \begin{cases} 0, & x \leq 0 \\ x, & x > 0 \end{cases} \text{ or } f(x) = \max(0, x) \quad (3.5)$$

In the Eq. (3.5), when the value $X \leq 0$ is entered, this neuron is inhibited; When $X > 0$, let the neurons remain linearly activated. In convolutional neural networks, the ReLU function is a common activation function, and its advantage is that its gradient does not decay, which can reduce the occurrence of gradient vanishing.

The advantages of using the ReLU activation function are expressed in detail as follows: first, compared with the Sigmoid function or the Tanh function, when the input value of the ReLU activation function deviates from 0 far, the neurons will not fall into a saturated state, and even have a gradient disappearing during backpropagation; Second, it can further accelerate the speed of network convergence. Third, there is a significant sparsity in neuronal output, so the generalization ability of the model is significantly enhanced.

iv. Shifted ReLU functions:

$$f(x) = \begin{cases} 1, & x \leq 0 \\ x, & x > 0 \end{cases} \text{ or } f(x) = \max(-1, x) \quad (3.6)$$

In the Eq. (3.6), Shifted ReLU is a variation of the ReLU function, similar to Leaky ReLU, Random ReLU, Parametric ReLU, etc. Although it has some advantages that ReLU does not have, such as the output may be negative, it can converge quickly, etc., but as the number of training iterations increases, Shifted ReLU will fall into the hard saturation zone.

Based on the comparison of the above activation functions, this paper finally chooses ReLU as the activation function of the proposed network.

3. Pooling layer

The pooled layer, also known as the downsampling layer, is generally located behind the convolutional layer. Its role is to help optimize the network and reduce the number of parameters in the network. In general, the most commonly used methods are mean pooling and maximum pooling. The mathematical descriptions are shown in (3.7) and (3.8).

i. Maximum pooling

$$p^{l(i,t)} = \max_{(j-1)w+1 \leq t \leq jw} \{\alpha^{l(i,t)}\} \quad (3.7)$$

ii. Average pooling

$$p^{l(i,t)} = \frac{1}{w} \sum_{(j-1)w+1}^{jw} \alpha^{l(i,t)} \quad (3.8)$$

In the formula, $p^{l(i,t)}$ represents the feature value output after pooling of the neuron in the i -th feature map of the l -th layer; $\alpha^{l(i,t)}$ represents the activation value output of the t -th neuron in the i -th feature map of the l -th layer; w is the width of the pooling region.

The pooled sliding window is similar to the convolution kernel in the convolution operation, and the maximum pooling operation with relatively good effect is used in the model in this paper.

4. Fully connected layer

In convolutional neural networks, the fully connected layer occupies the least number of layers and requires the largest amount of computation. The goal is to collect all the features from the previous feature map for classification. The formula for a fully connected layer in a convolutional network is shown in the following equation (3.9):

$$z^{l+1(j)} = \sum_{i=1}^n W_{ij}^l \alpha^{l(i)} + b_j^i \quad (3.9)$$

In the formula, W_{ij}^l represents the weight between the i -th neuron in layer l and the j -th neuron in layer $l+1$ denotes the logit value of the j -th output neuron in layer $l+1$; and b_j^i is the bias term added to the j -th neuron in layer $l+1$, contributed collectively by all neurons in layer l .

When the $l+1$ layer is a hidden layer, the activation function (3.10) is ReLU:

$$\alpha^{l+1(i)} = \max\{0, z^{l+1(i)}\} \quad (3.10)$$

When the $l+1$ layer is the output layer, the activation function (3.11) is Softmax:

$$q^j = \text{softmax}(z^{l+1(j)}) = \frac{e^{z^{l+1(j)}}}{\sum_k e^{z^{l+1(k)}}} \quad (3.11)$$

The CNN-based fault diagnosis method has the following characteristics:

- 1) Because of the structure of the shared convolutional kernel, it is better at processing high-dimensional data.
- 2) Obtain appropriate weights through training, and do not need complex manual selection of features.

For the construction of the CNN-based fault diagnosis model, each basic convolutional block consists of a convolutional layer, a batch normalization layer, and a ReLU activation function. Initially, random weights are assigned to the network parameters. The preprocessed time-series fragments $\{S_j\}, j=1, 2, \dots, N$, obtained through scale transformation and time slicing, are then fed into the CNN. The network is built by stacking three convolutional blocks, with the number of filters in each block set to 128, 256, and 128, respectively. After the convolution operations, max pooling is applied, followed by a fully connected layer. This process enables the extraction of high-level spatial features from the SCADA sequence data, forming a representative feature map for fault diagnosis.

3.3 Deep Learning Model Architecture

A Recurrent Neural Network (RNN) is a model designed to process time series data by retaining previous information through the interconnected structure of nodes at each layer. This stored information is then used to influence the output of subsequent layers, enabling the model to capture temporal dependencies in the data. However, RNNs face the problem of gradient vanishing during the backpropagation process. To address this issue, two extended models were proposed: Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU).

3.3.1 LSTM and BiLSTM Network Architecture Models

(1) LSTM Network Architecture Models

In the structure of an LSTM, as illustrated in the figure 17, each LSTM cell contains three gate mechanisms: the forget gate (f_t), the input gate (i_t), and the output gate (o_t) [76]. The subscript t denotes the current time step, while $t-1$ refers to the state information passed from the previous LSTM cell. The symbol σ represents the sigmoid activation function, and $\tanh$ denotes the hyperbolic tangent activation function. The operator $+$ stands for element-wise addition, and $\otimes$ indicates element-wise multiplication.

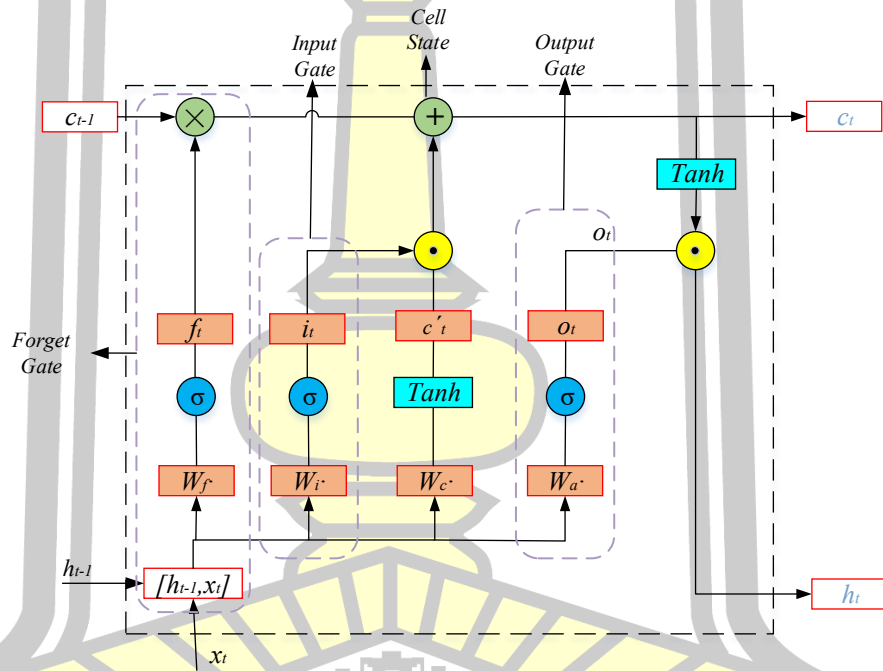


Figure 17 Framework of LSTM [76]

Within an LSTM, the forget gate first determines which information should be retained and which should be discarded. This decision is based on the previous hidden state (h_{t-1}) and the current input (x_t), and is computed through a sigmoid activation function that outputs values in the range $[0, 1]$. This output is then element-wise multiplied with the previous cell state (C_{t-1}) to determine the proportion of information to retain—where 0 means "completely discard" and 1 means "fully retain." The specific computation is given by the following equation (3.12):

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \quad (3.12)$$

In the above equation, W_f denotes the weight matrix of the forget gate, and $[h_{t-1}, x_t]$ represents the concatenation of the previous hidden state and the current input into a single vector. b_f is the bias term for the forget gate, and σ denotes the sigmoid activation function.

Subsequently, the input gate is responsible for determining which new information should be added to the cell state.

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \quad (3.13)$$

In the above equations (3.13), W_i is the weight matrix of the input gate, and b_i is the corresponding bias term. Next, in the next equations (3.14), we calculate the candidate cell state at time step t , which is derived based on the previous hidden state and the current input:

$$\tilde{c}_t = \tanh(W_c[h_{t-1}, x_t] + b_c) \quad (3.14)$$

In the next equations (3.15), The cell state C_t at the current time step is computed by performing an element-wise multiplication of the previous cell state C_{t-1} with the forget gate output, followed by an addition of the element-wise product of the input gate and the candidate cell state.

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \quad (3.15)$$

In this way, the LSTM integrates the short-term memory $\tilde{c}_t$ with the long-term memory C_{t-1} to form the updated cell state C_t . Under the regulation of the forget gate, the network is capable of retaining information from the distant past, while the input gate prevents irrelevant or unimportant information from being incorporated into the memory at the current time step.

Next, in the equations (3.16) the output gate is introduced, which governs the influence of the long-term memory on the current output.

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \quad (3.16)$$

In the equations (3.17), the final output of the LSTM is determined jointly by the output gate and the current cell state.

$$h_t = o_t \odot \tanh(c_t) \quad (3.17)$$

Similar to other neural networks, LSTM updates its network parameters through backpropagation. However, since it processes sequential time-series data, the training relies on a time-based variant of backpropagation, known as Backpropagation Through Time (BPTT).

(2) BiLSTM Network Architecture Models

The BiLSTM network is formed by combining the forward LSTM structure and the backward LSTM structure. This special architecture allows the feature data obtained by the hidden layer to simultaneously have information from both the past and the future. Compared to LSTM, BiLSTM can capture bidirectional dependencies in the input data, effectively improving the feature representation ability of the prediction model for the input data. The network structure is shown in Figure 18.

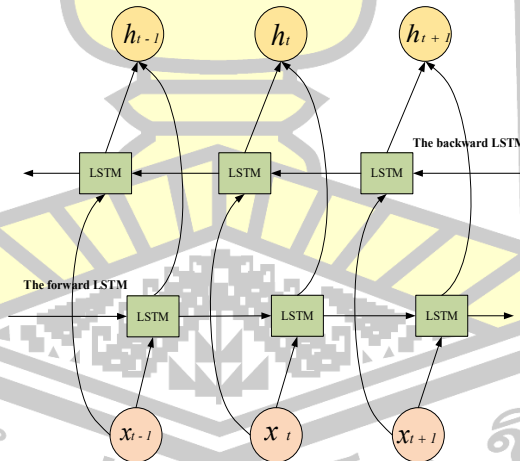


Figure 18 Framework of BiLSTM [77]

As shown in Figure 18, the forward LSTM structure in the BiLSTM network follows a similar computation process to that of a single LSTM. The forward hidden state and the backward hidden state are combined to obtain the hidden state of the BiLSTM network. The calculation formula is as following equations (3.18)-(3.20):

$$\vec{h}_t = LSTM(h_{t-1}, x_t) \quad (3.18)$$

$$\overleftarrow{h}_t = LSTM(h_{t+1}, x_t) \quad (3.19)$$

$$h_t = \alpha \vec{h}_t + \beta \overleftarrow{h}_t \quad (3.20)$$

Where x_t , $\vec{h}_t$ and $\overleftarrow{h}_t$ represent the input data, the output of the forward LSTM hidden layer, and the output of the backward LSTM hidden layer at time t , respectively; α and β are constant coefficients, representing the weights of $\vec{h}_t$ and $\overleftarrow{h}_t$, respectively.

3.3.2 GRU and BiGRU Network Architecture Models

(1) GRU Network Architecture Models

GRU, a simplified version of LSTM, features a more efficient structure. As depicted in Figure 19, GRU[39] is composed of three parts: the reset gate (r_t), the update gate (z_t), and the candidate layer (c_t). Unlike LSTM, GRU eliminates the need for an additional memory cell and uses the update gate (z_t) to regulate which information is retained or discarded, thus reducing computational overhead. The GRU's operations can be computed as following (3.21)-(3.23) equations:

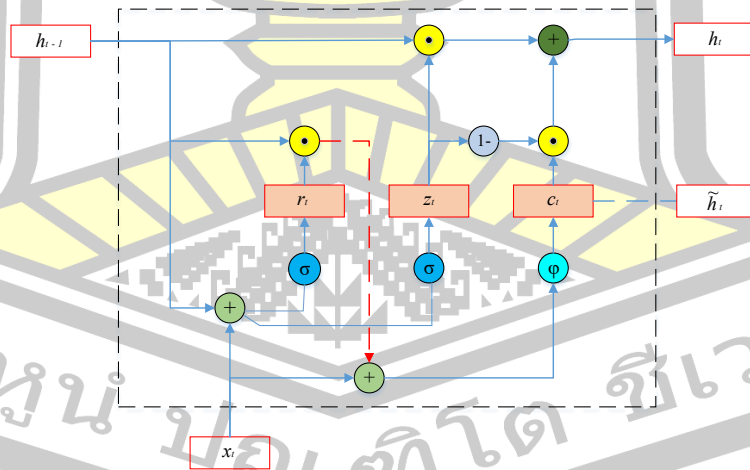


Figure 19 Framework of GRU [39]

$$z_t = \sigma(V_{xz}x_t + U_{hz}h_{t-1} + b_z) \quad (3.21)$$

$$r_t = \sigma(V_{xr}x_t + U_{hr}h_{t-1} + b_r) \quad (3.22)$$

$$c_t = \phi(V_{xc}c_{xt} + U_{hc}(r_t \odot h_{t-1}) + b_c) \quad (3.23)$$

Where V_{xz} , V_{xr} , V_{xc} denote the weights that link the input layer to the GRU units. U_{hz} , U_{hr} , U_{hc} represent the weights associated with the self-connections between the current time step t and the previous time step $t-1$. b_z , b_r denote the biases for the two gates in the GRU unit, while b_c represents the bias for the candidate layer.

(2) BiGRU Network Architecture Models

While GRU is effective in modeling time-series data, it is restricted to learning information solely in the forward direction. In contrast, the Bidirectional GRU (BiGRU) improves upon this by incorporating two layers—one that processes information in the forward direction and another in the backward direction. As illustrated in Figure 3, the forward and backward layers are connected to the output layer, allowing the model to capture both past and future context, making it more effective in processing sequential data with dependencies across different time steps.

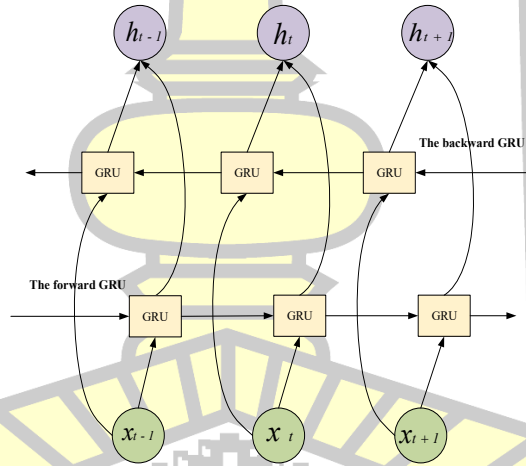


Figure 20 Framework of BiGRU

The forward layer ($\vec{h}_t$) conducts training on the time-series data in the forward direction, while the backward layer ($\overleftarrow{h}_t$) trains the time-series data in the reverse direction. This bidirectional approach allows the model to utilize information from both directions, enhancing its understanding of the sequence. The forward and backward training processes can be formulated as following (3.24)-(3.26) equations: :

$$\vec{h}_t = f(\vec{W} \cdot x_t + \vec{U} \cdot \vec{h}_{t-1} + \vec{b}) \quad (3.24)$$

$$\tilde{h}_t = f(\vec{W} \cdot x_t + \vec{U} \cdot \tilde{h}_{t-1} + \vec{b}) \quad (3.25)$$

$$h_t = \vec{h}_t \oplus \tilde{h}_t \quad (3.26)$$

The weights connecting the input layer to the forward and backward layers are denoted by $\vec{W}$ and $\overleftarrow{W}$, respectively. The self-connection weights for these layers are represented by $\vec{U}$ and $\overleftarrow{U}$. The biases for the forward and backward layers are $\vec{b}$ and $\overleftarrow{b}$, respectively. Additionally, $\oplus$ represents the concatenation of the outputs from both layers.

3.3.3 Joint CNN-BiGRU mechanism

This paper presents a wind turbine icing prediction model that integrates CNN with Bi-directional GRU to enhance predictive performance. CNN is used to extract spatial features of wind turbine SCADA data, while BiGRU processes time series data through its bi-directional structure to capture forward and backward contextual information for better understanding of the temporal properties of the data. CNN firstly performs the input data Convolutional operation is performed to extract the low-level features and reduce the dimensionality through pooling layer to retain the important spatial information. Then, BiGRU partially extracts more comprehensive features from the past and future information of the time series to improve the model's ability to capture temporal dependencies.

In the equations (3.27), the architecture of the model (e.g., Figure 21) consists of five main components: an input layer, a convolutional layer, a BiGRU layer, an output layer, and a final predicted output. The data is first passed through a CNN layer for feature extraction, followed by a BiGRU layer to capture the timing information, and finally an output layer to generate the prediction results. The model's output is typically a fully connected layer that transforms extracted features into target predictions.

$$\hat{y} = W_{OUT} \cdot h + b_{out} \quad (3.27)$$

where $\hat{y}$ represents the predicted output, W_{OUT} is the weight matrix of the output layer, h denotes the features extracted from previous layers (CNN and BiGRU), and b_{out} is the bias term of the output layer.

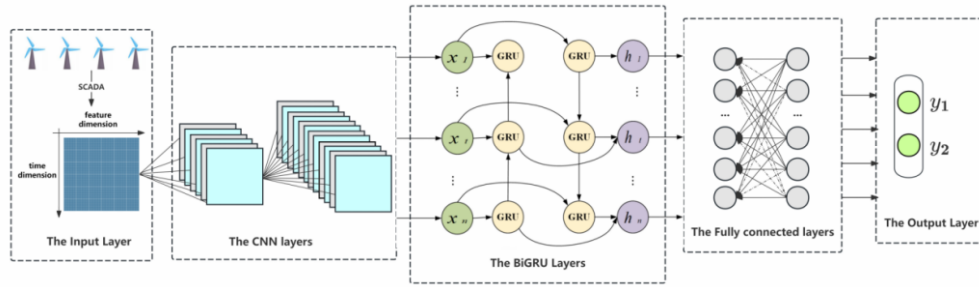


Figure 21 CNN-BiGRU Algorithm Architecture [80]

The joint CNN-BiGRU mechanism demonstrates the flow of data from input to feature extraction to final output, fully utilizing the advantages of convolutional networks in spatial feature extraction and BiGRU in time series modeling.

3.3.4 Optimization model and loss function selection

During debugging, the cosine annealing learning rate scheduler is introduced to dynamically adjust the learning rate, enhancing training efficiency and model performance. Cosine annealing, inspired by Simulated Annealing, gradually decreases the learning rate, helping the model converge more smoothly toward the global optimal solution in the final stages of training[81] .

The cosine annealing learning rate equation (3.28) is given by.

$$\eta_t = \eta_{min} + \frac{1}{2}(\eta_{max} - \eta_{min}) \left(1 + \cos\left(\frac{t}{T_{max}}\pi\right) \right) \quad (3.28)$$

The learning rate at the current training step is denoted as η_t , with η_{max} and η_{min} representing its upper and lower bounds, respectively. Here, t indicates the current training step, while T_{max} defines the annealing period, specifying the total number of steps over which the learning rate is adjusted.

In the icing prediction task, in order to measure the difference between the model's predicted probability distribution and the actual categories, we introduce the CrossEntropyLoss function as the optimization objective function[82] . Since the icing problem can be regarded as a binary categorization problem of icing and non-icing, the properties of the CrossEntropyLoss function are well suited to the needs of such a task. Specifically, the CrossEntropyLoss function (3.29) can accurately calculate the difference between the model's predicted probability distribution and the true category labels to help the model learn the classification boundaries better, especially in the

face of category imbalance, and can effectively avoid the model's prediction biased towards the large category (i.e., the non-icing state).

$$L = -\frac{1}{N} \sum_{i=1}^N \sum_{j=1}^C y_{ij} \log(\hat{y}_{ij}) \quad (3.29)$$

Here, N represents the total number of samples, while C denotes the number of categories within the icing classification. In the wind turbine icing task, this classification is simplified into a binary distinction between icing and non-icing, setting $C=2$. The variable y_{ij} indicates the true label of sample i for category j , and $\hat{y}_{ij}$ represents the model's predicted probability for category j .

Moreover, the cross-entropy loss function helps minimize the misclassification of icing state samples by refining prediction probabilities. In wind turbine blade icing prediction, it enhances the learning of rare icing cases by adjusting the weights of smaller errors, ultimately improving the model's accuracy and robustness in real-world applications. Integrating the cosine annealing learning rate scheduler, the cross-entropy loss function enhances the training process, ensuring efficient learning at each stage and leading to more stable and precise predictions.

3.3.5 Temporal Attention Mechanism

The attention mechanism[40-41] mimics human visual attention by assigning different weights to various features, emphasizing those most relevant to the target task. Typical attention mechanisms focus on extracting important information associated with the current time step. The context vector v_t is derived by weighting the column vectors of the BiGRU hidden state $H = \{h_1, h_2, \dots, h_{t-1}\}$. However, in the case of wind turbine blade icing detection, each time step involves multiple variables that exhibit complex nonlinear relationships. Additionally, each variable exhibits unique characteristics and periodicity, making it challenging to pinpoint a specific time step as the primary focus. To address this, the Temporal Pattern Attention Mechanism(TPA) applies a one-dimensional convolutional neural network (1D-CNN) filter to extract row vector features from the BiGRU hidden state H . This improves the model's capacity to capture the relationships among multiple variables across various time steps. The framework of TPA is shown in Figure 22.

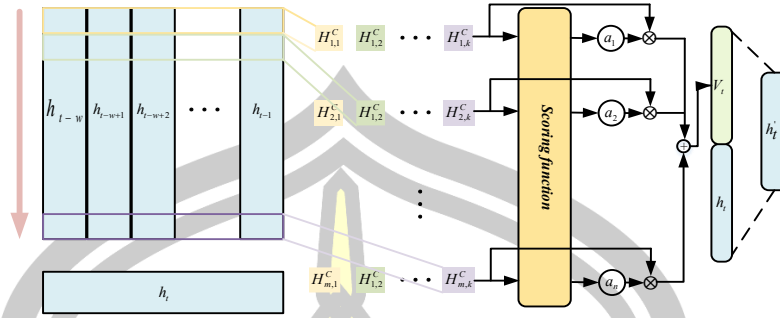


Figure 22 Framework of TPA

For the initial time series H , BiGRU computes a hidden state matrix $H_{i,j}$, and TPA applies k filters to extract features through row-wise convolution operations on each row H_i of H , generating a time pattern matrix $H_{i,j}^C$. Here, in the equation (3.30), $H_{i,j}$ represents the output value corresponding to the i -th row vector and the j -th convolutional kernel, with C representing the convolution process.

$$H_{i,j}^C = \sum_{l=1}^w H_{i,(t-w-l+1)} \times C_{j,T-w+l} \quad (3.30)$$

In the equations (3.31)-(3.32), the predicted nodes are h_t and $H_{i,j}^C$. Each matrix row is processed to calculate a weight, which is then normalized to derive the attention weight a_i , reflecting the influence of each time series on h_t .

$$f(H_i^C, h_t) = (H_i^C)^T W_a h_t \quad (3.31)$$

$$a_i = \text{sigmoid}(f(H_i^C, h_t)) \quad (3.32)$$

In the equation (3.33), the weighted sum of the weights of each time series is used to obtain the weight vector v_t , which takes into account the combined effect of all rows on h_t , i.e., time attention.

$$v_t = \sum_{i=1}^n a_i H_i^C \quad (3.33)$$

For v_t and h_t , the linear mapping and the sum are performed to obtain the time information state h'_t , see in the equation (3.34).

$$h'_t = W_t h_t + W_v v_t \quad (3.34)$$

Where $W_t \setminus W_v$ represents the weight matrix of different variables.

A key advantage of TPA over traditional attention mechanisms is its ability to assign weights not only to time steps, but also to individual characteristic dimensions related to icing of wind turbine blades. Standard attention mechanisms often distribute weights between different time steps, which makes it difficult to pinpoint the specific effect of each variable on leaf icing within a single time step. In contrast, TPA digs deep into the relationship between variables in historical data and current icing conditions, using weighted attention vectors to better capture time series features to generate more accurate predictions. In addition, while RNN models such as BiGRU often encounter problems such as vanishing gradients when dealing with long-term wind turbine SCADA data series, TPA overcomes this challenge by analyzing the correlation between historical icing sequences and future results through its resource allocation system. This strategy emphasizes key contributors, reduces the risk of losing important information, and effectively captures long-term dependencies, making TPA a powerful tool for predicting icing on wind turbine blades, and Table 5 can demonstrate the superiority of its model.

3.3.6 CNN-BiGRU-TPA model

Figure 23 shows the architecture of the CNN-BiGRU-TPA model developed in this study, consisting of five main components: the input layer, CNN layer, BiGRU layer, TPA layer, and output layer. The roles of these layers are as follows:

(1) Input Layer: The model's input consists of historical SCADA icing data and other influencing factors.

(2) CNN Layer: This layer captures the relationships between input factors and current blade icing. The convolutional layers identify feature patterns, while the pooling layers reduce dimensionality by selecting key features. The processed features are then sent to the BiGRU layer via fully connected layers.

(3) BiGRU Layer[42]: The BiGRU layer captures temporal dependencies within the icing sequence, building a dynamic time-series model. It processes bidirectional information and applies dropout regularization to prevent overfitting. The result is a hidden state matrix $H_{i,j} = [h_1, h_2, \dots, h_{t-1}, h_t]$, which is passed to the TPA layer.

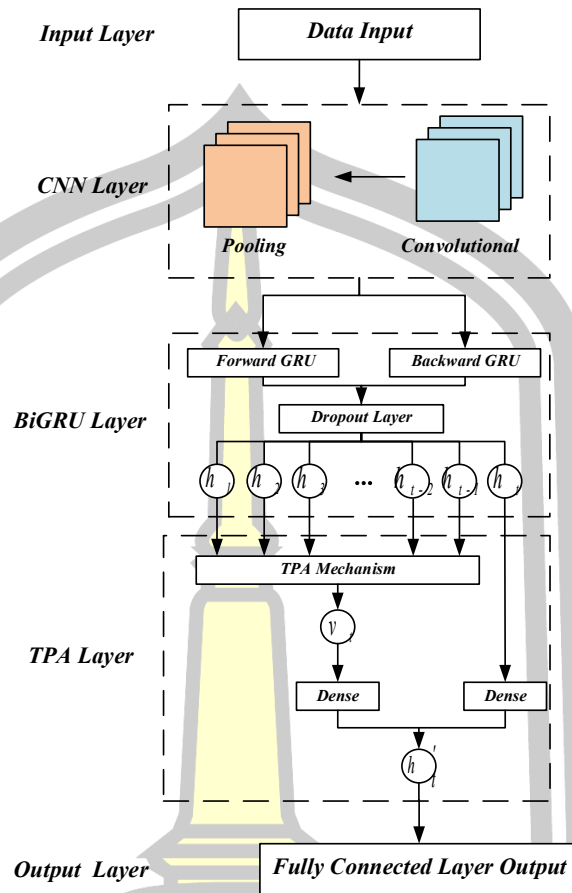


Figure 23 Framework of CNN-BiGRU-TPA

(4) TPA Layer: The Temporal Pattern Attention Mechanism (TPA) assigns adaptive weights to the hidden states from BiGRU, selecting and emphasizing relevant features. The attention mechanism produces a comprehensive temporal feature vector h' by highlighting important features.

(5) Output Layer: The fully connected layer produces the predicted output, indicating whether blade icing will occur in the next time step.

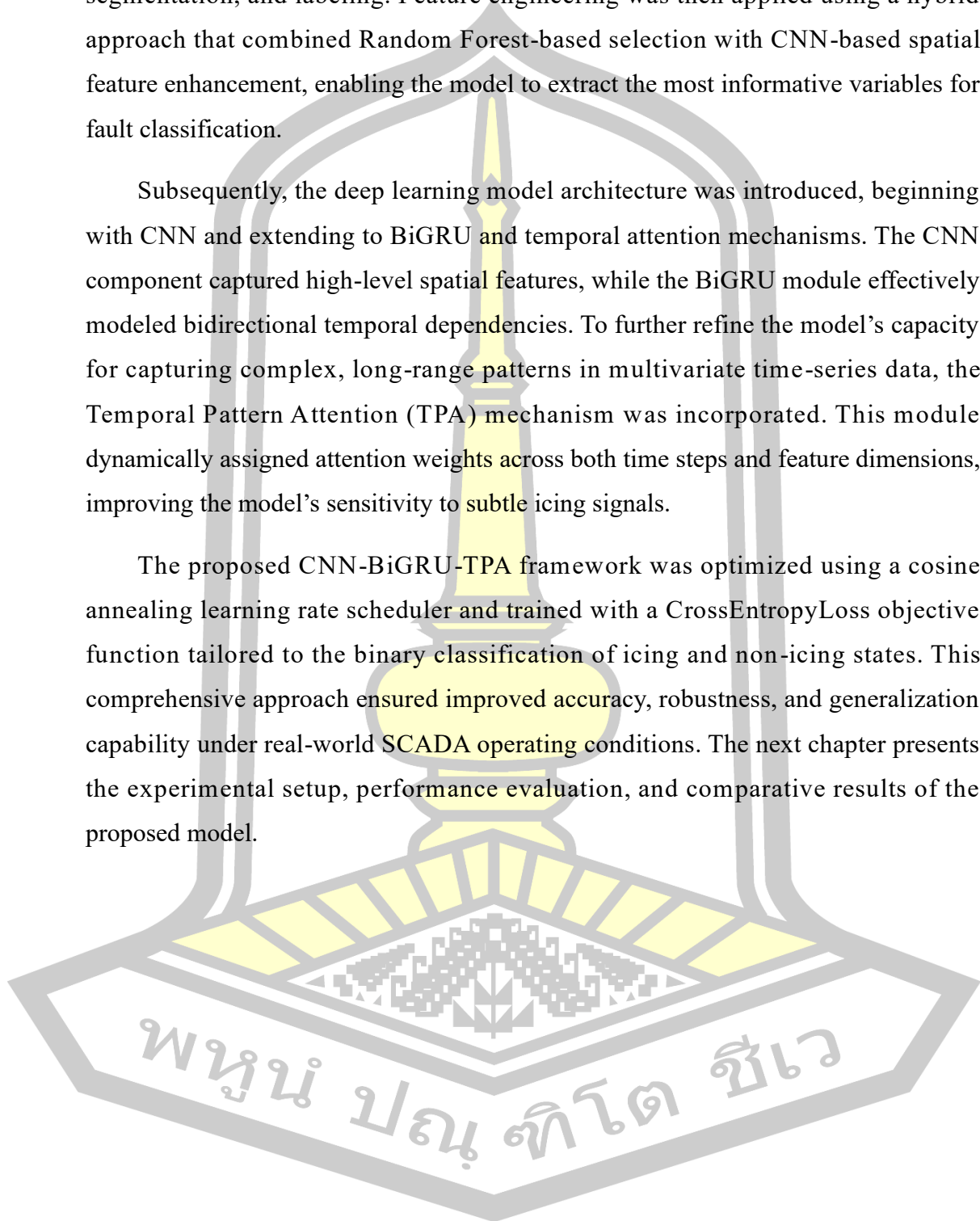
3.4 Summary

This chapter presented the methodological framework for developing a wind turbine blade icing prediction model based on SCADA data and deep learning techniques. First, the challenges of monitoring blade health using SCADA data—such as data noise, missing values, and lack of direct indicators—were addressed through a

robust data preprocessing pipeline that included cleaning, standardization, segmentation, and labeling. Feature engineering was then applied using a hybrid approach that combined Random Forest-based selection with CNN-based spatial feature enhancement, enabling the model to extract the most informative variables for fault classification.

Subsequently, the deep learning model architecture was introduced, beginning with CNN and extending to BiGRU and temporal attention mechanisms. The CNN component captured high-level spatial features, while the BiGRU module effectively modeled bidirectional temporal dependencies. To further refine the model's capacity for capturing complex, long-range patterns in multivariate time-series data, the Temporal Pattern Attention (TPA) mechanism was incorporated. This module dynamically assigned attention weights across both time steps and feature dimensions, improving the model's sensitivity to subtle icing signals.

The proposed CNN-BiGRU-TPA framework was optimized using a cosine annealing learning rate scheduler and trained with a CrossEntropyLoss objective function tailored to the binary classification of icing and non-icing states. This comprehensive approach ensured improved accuracy, robustness, and generalization capability under real-world SCADA operating conditions. The next chapter presents the experimental setup, performance evaluation, and comparative results of the proposed model.



Chapter 4 Experimental Setup and Model Evaluation Framework

4.1 Experimental Configuration and Data Preparation

Building upon the data preprocessing strategies, feature engineering techniques, and model architecture designs introduced in Chapter 3, this chapter details the experimental configuration and evaluation framework used to validate the effectiveness of the proposed wind turbine blade icing prediction models. To ensure fair and rigorous comparisons, a standardized dataset was prepared from real-world SCADA records, and a consistent set of experimental settings was applied across all models. This section outlines the specific evaluation metrics, software and hardware environments, dataset partitioning strategy, and hyperparameter tuning procedures that collectively support the reproducibility and reliability of the experimental results presented in the subsequent sections.

4.1.1 Definition of Evaluation Metrics

This study evaluates the proposed model using imbalanced data, where the number of abnormal (icing) samples is much smaller than the number of normal (non-icing) samples. This imbalance can inflate the precision score, making it unsuitable for accurate model assessment. Given the focus on wind turbine blade icing, with icing and non-icing conditions treated as positive and negative samples, respectively, the paper uses precision, recall, F1-score, and the Matthews correlation coefficient (MCC) as performance metrics. The definitions of these metrics are as following equations (4.1)-(4.5):

$$Precision = \frac{TP}{TP+FP} \quad (4.1)$$

$$Recall = \frac{TP}{TP+FN} \quad (4.2)$$

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (4.3)$$

$$ACC = \frac{TP+TN}{TP+FP+TN+FN} \quad (4.4)$$

$$MCC = \frac{TP \times TN - FP \times FN}{(TP+FP)(TP+FN)(TN+FP)(TN+FN)} \quad (4.5)$$

Where TP, FP, FN, and TN represent true positives, false positives, false negatives, and true negatives, respectively.

4.1.2 Experimental Environment

This experiment was performed in a computing environment with the following Table 5:

Table 4 Experimental Environment

Hardware configuration	Software Configuration
CPU: Intel64 Family 6 Model 186 Stepping 2, GenuineIntel	(1)Operating System: Windows 10 10.0.22631 SP0
(2)Memory: 31.65 GB RAM	(2)Programming language: Python 3.8
(3)Storage: 302.60 GB	(3)Deep learning framework: PyTorch 1.8.1

The experiments were conducted in a local computing environment with detailed hardware and software configurations summarized in Table 4. The system was equipped with an Intel64 Family 6 Model 186 CPU, 31.65 GB of RAM, and 302.60 GB of available storage. The software environment consisted of Windows 10 (version 10.0.22631 SP0) as the operating system, Python 3.8 as the programming language, and PyTorch 1.8.1 as the deep learning framework used for model development and training.

4.1.3 Dataset Splitting Strategy

To facilitate robust and generalizable model training, the curated SCADA dataset was organized with a focus on sample quality, representativeness, and class balance. The data were collected from two geographically distinct wind farm sites, labeled Location A and Location B. Each site contains turbines with varying operational durations and sample sizes, as summarized in Table 5. All data were acquired at a 1-second interval, ensuring consistency in temporal resolution.

Table 5 Overview of Wind Turbine SCADA Data Samples by Site and Label Balance

Location	Total Data Count	Normal Data (Proportion)	Icing data (proportion)	1 : 1 Constructed Sample Set Size
A	222663	204628 (91.9%)	18035 (8.1%)	57254
B	182917	17012 (90.7%)	10,592 (9.3%)	

Given the naturally imbalanced distribution of blade icing events—comprising only approximately 8–9% of the total observations—a balanced sample set was constructed by downsampling the majority class to achieve a 1:1 ratio between icing and non-icing instances. This yielded a refined dataset of 57,254 labeled samples, enhancing model training efficiency and mitigating potential bias toward the dominant class.

Following class balancing, the dataset was partitioned into training (70%), validation (15%), and testing (15%) subsets. This stratification provided sufficient data for both model optimization and generalization assessment, while maintaining temporal integrity to prevent information leakage between subsets.

A comprehensive framework was subsequently implemented to develop the blade icing detection model using SCADA data and machine learning techniques. As depicted in Figure 24, the methodology comprises four sequential stages: data processing, input preparation, model training, and performance evaluation. Initially, SCADA data from ten turbines were aggregated, balanced, and subjected to feature selection using methods such as Random Forest and CNN-based filters. The resulting dataset underwent min-max normalization and was divided into the aforementioned subsets. Over ten models were trained and evaluated through feature correlation analysis and comparative performance testing. The best-performing model was ultimately selected and validated under realistic conditions to assess its accuracy and robustness in detecting blade icing events, contributing to the advancement of intelligent fault diagnosis in wind energy systems.

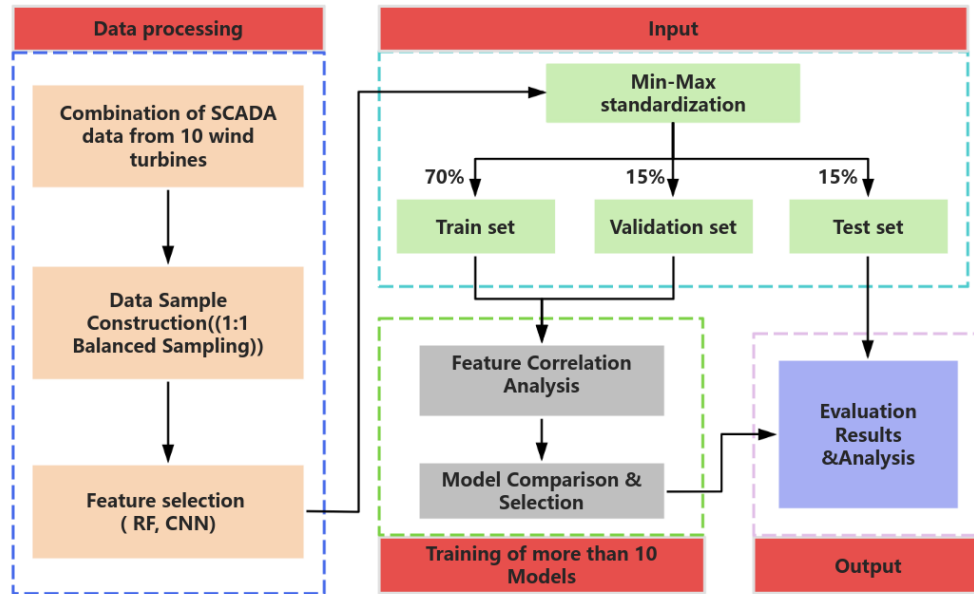


Figure 24 Workflow of the Proposed Blade Icing Detection Framework

4.1.4 Hyperparameter Selection and Optimization Strategy

A well-tuned hyperparameter configuration is essential for enhancing model performance and ensuring training stability. In this study, a systematic hyperparameter optimization approach was adopted to improve the generalization capability of the models on wind turbine SCADA datasets. The key hyperparameters—learning rate, batch size, dropout rate, hidden layer size, filter sizes, and weight decay—were selected and optimized as follows:

(1) Learning Rate and Scheduler

The initial learning rate was set to 0.0003, which was empirically determined through multiple trials as the optimal value for most models. To prevent convergence to local minima and mitigate training oscillations, the Cosine Annealing learning rate scheduler was applied. This dynamic adjustment gradually reduces the learning rate during training, improving convergence stability.

(2) Batch Size

A batch size of 512 was used consistently across all models. This value balances computational efficiency and memory constraints while providing a sufficiently representative gradient estimate in each iteration. The larger batch size

also contributes to more stable updates, particularly for complex models like CNN-BiGRU hybrids.

(3) Dropout Rate

To mitigate overfitting and enhance generalization, the Dropout mechanism was incorporated across all models. Based on comparative experiments:

A dropout rate of 0.1 yielded the best results in CNN and CNN-BiGRU models, effectively balancing information retention and regularization.

For BiGRU and BiLSTM models, a dropout rate of 0.2 provided better robustness against overfitting due to their sequential learning nature.

(4) Hidden Size

The number of hidden units in RNN-based models (LSTM, GRU, BiGRU, BiLSTM) was set to 64 or 128, depending on the model complexity and performance in validation tests. These dimensions provided a good trade-off between representational capacity and computational cost.

(5) Filter Sizes in CNN

For CNN-related models, the convolutional filter sizes were empirically selected from the set $\{5, 7, 9\}$, with 7 chosen as the primary kernel size due to its effectiveness in extracting localized temporal features. Additionally, convolutional layers used 32 or 64 filters to enhance feature representation capabilities.

(6) Weight Decay (L2 Regularization)

A weight decay value of 0.0005 was uniformly applied across all models. This regularization helps prevent overfitting by penalizing large weight magnitudes, thereby encouraging smoother model solutions.

To ensure a fair comparison across models, consistent settings were applied where applicable. The detailed hyperparameter settings are presented in Table 7.

4.2 Feature Correlation Analysis and Redundant Feature Removal

In the wind turbine blade icing prediction task, the SCADA system typically collects hundreds of operational and environmental monitoring variables. To reduce the data dimensionality and minimize the adverse impact of redundant information on model performance, the dataset used in this paper has been pre-selected by domain

experts to include 26 representative key features, covering multiple dimensions such as the turbine's operational status, environmental conditions, and control parameters. However, in practical applications, too many features not only increase the consumption of computational resources but may also introduce noise, reducing the stability and generalization ability of the model, especially in engineering scenarios where real-time performance is required.

Table 6 Hyperparameter Settings for Compared Deep Learning Models

Model	Learning Rate	Weight Decay	Dropout	Hidden Size	Filter Sizes	Batch Size
CNN	0.0003	0.0005	0.1	N/A	[5, 7, 9]	512
LSTM	0.0003	0.0005	0.1	64	N/A	512
GRU	0.0003	0.0005	0.1	64	N/A	512
BiGRU	0.0003	0.0005	0.1	64/128	N/A	512
BiLSTM	0.0001	0.001	0.2	64/128	N/A	512
CNN-BiGRU	0.0003	0.0005	0.1	64/128	[5, 7, 9]	512

Given this, this paper introduces a feature selection step before modeling to improve model training efficiency and prediction performance. First, a visual analysis of the correlation between the 26 key features is performed, and a feature correlation heatmap is generated (as shown in Figure 25) to identify potential linear redundancy relationships between variables. The heatmap uses a color gradient to represent the Pearson correlation coefficients between variables (-1 to 1), where red indicates a strong positive correlation and blue represents a strong negative correlation.

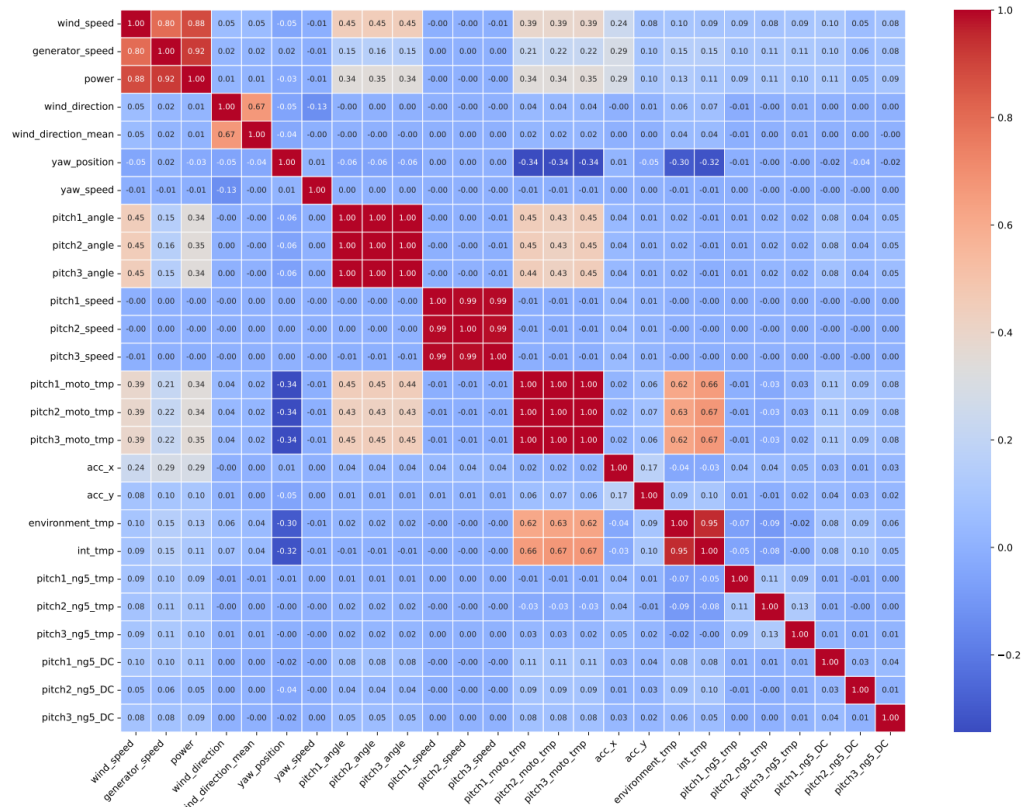


Figure 25 SCADA Feature Correlation Heatmap

As shown in the figure, there is a strong positive correlation between wind speed (wind_speed), generator speed (generator_speed), and power output (power). Additionally, there is significant redundancy between the pitch angle (pitch_angle) and pitch speed (pitch_speed) variables. If these redundant variables are not addressed, they could introduce biases in model learning or computational redundancy. Therefore, by using the heatmap to identify and eliminate highly correlated features, it helps retain core variables with more informative value, thereby improving the model's computational efficiency and generalization ability while maintaining prediction accuracy.

4.3 Baseline Comparison

In order to verify the validity of the proposed model, we selected several common time series modeling methods as baseline models for comparison, including LSTM, BiLSTM, GRU, BiGRU and CNN-BiGRU. The reasons for selecting these baseline models are as follows:

LSTM (Long Short-Term Memory) [83] :LSTM, a well-established Recurrent Neural Network (RNN) variant, is specifically designed to address long-term dependency challenges in sequential data. In this study, a 2-layer LSTM architecture is utilized, with hidden units selected from {8, 16, 32, 64}. The model with the highest performance is ultimately chosen for evaluation.

BiLSTM (Bidirectional LSTM) [84] :BiLSTM extends LSTM by incorporating a bidirectional mechanism, enabling the model to learn both past and future contextual information simultaneously. The same experimental setup as LSTM is applied for comparison.

GRU (Gated Recurrent Unit) and BiGRU (Bidirectional GRU):These models, described in Section 2.2, adopt gating mechanisms to optimize memory usage and improve computational efficiency in sequential learning tasks.

CNN-BiGRU (Convolutional and Bidirectional GRU Combined Model):This model integrates CNN for spatial feature extraction and BiGRU for temporal sequence modeling, enhancing the representation of spatio-temporal dependencies. By effectively capturing complex patterns in large datasets and short time series, CNN-BiGRU demonstrates strong applicability in wind turbine blade icing prediction.

The model training and evaluation were conducted on a Windows 10 platform utilizing the PyTorch 1.8.1 deep learning framework, with Python 3.8 as the programming environment. The corresponding experimental results are presented in Table 8 below.

Table 7 Baseline Model Comparison Results

Model	Accuracy	Precision	Recall	F1-score	MCC
CNN	77.6%	77.6%	77.8%	77.6%	55.4%
LSTM	76.4%	77.7%	75.4%	75.6%	53.1%
BiLSTM	74.1%	74.1%	74.3%	74.1%	48.4%
GRU	69.5%	73.2%	71.0%	69.1%	44.2%
BiGRU	81.6%	81.9%	81.1%	81.3%	63.0%
CNN-BiGRU	85.6%	85.6%	85.9%	85.6%	71.5%

From the results, it can be seen that BiGRU has the best overall performance with an Accuracy of 85.6%, which is better than other time series models. This indicates that BiGRU can capture the time-series features of SCADA data more effectively, which is more advantageous for the task of wind turbine blade icing

prediction. However, the performance of the traditional RNN structure on SCADA data still has some limitations in terms of insufficient feature extraction capability and high model complexity. Therefore, we further explore the performance of different deep learning architectures in feature extraction (see in 4.3).

4.4 Sensitivity experiments

To further assess the influence of various factors on the effectiveness of the wind turbine blade icing prediction model, two sensitivity analyses were performed: examining the effect of a shallow CNN architecture on model performance and evaluating the impact of training iterations on overall accuracy.

4.4.1 Impact of CNN Depth on Model Performance

In order to investigate the performance of CNN networks with different depths in wind turbine blade icing prediction, we designed a CNN model containing 1, 2 and 3 convolutional layers and conducted sensitivity experiments. The experimental results are shown in Table 8, with the increase of the number of CNN layers, there is a significant improvement in the F1-score, but the balance between the performance and the computational cost is of concern.

Table 8 Comparison of Performance and Computational Cost for Different CNN Layer Depths

Number of Layers	Accuracy	F1-score	Number of Parameters	Training Time (seconds/epoch)
CNN_1Layer	94.83%	94.79%	4354	5
CNN_2Layer	96.55%	96.51%	16706	8
CNN_3Layer	95.98%	95.92%	29058	23

(1) 1-layer CNN

This model has the lowest computational cost (training time 5 sec/epoch), but the F1-score is only 94.79%, which indicates that it is deficient in the balance of precision and recall and is prone to misprediction or omission.

(2) 2-layer CNN

This model performs best among all models with an F1-score of 96.51%, while the accuracy is improved to 96.55%. The model strikes an optimal balance between accuracy and recall, and the training time is relatively controllable (8 sec/epoch) for the wind turbine icing prediction task.

(3) 3-layer CNN

After further increasing the number of CNN layers, the F1-score is improved (95.92%), but the increase is limited, while the computational overhead is significantly increased (training time 23 sec/epoch). Although the performance of the model is still high, its advantage over 2-layer CNN is not obvious considering the training cost.

From the visualization results in Figure 26, it can be seen that 2-layer CNN achieves the best performance in terms of accuracy, F1-score, precision and recall, with the F1-score as high as 96.51%. In contrast, the accuracy and recall of 1-layer CNN are low, which is difficult to meet the actual demand, while 3-layer CNN has some improvement, but the computational cost increases more, which is less cost-effective. Therefore, in the wind turbine blade icing prediction task, 2-layer CNN achieves the best balance between performance and computational cost and is a more ideal choice.

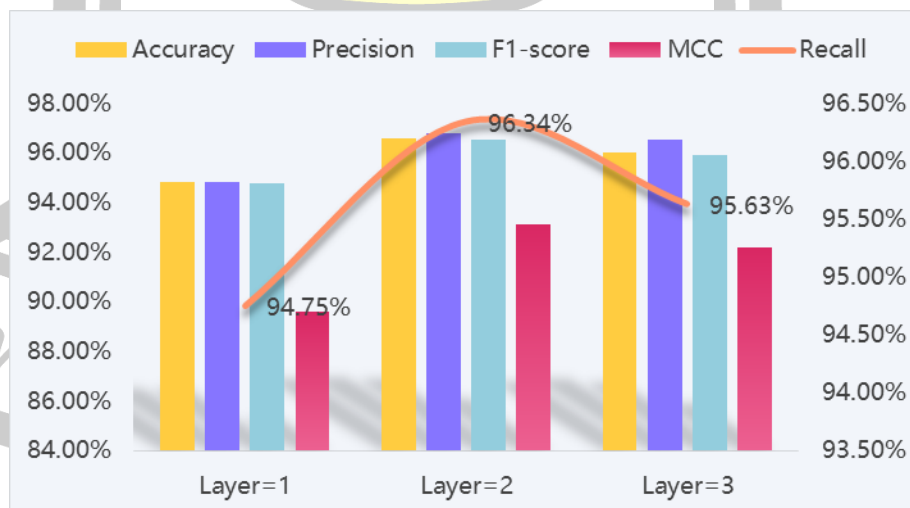


Figure 26 Comparison of Performance and Computational Cost for Different CNN Layer Depths

4.4.2 Effect of training rounds on model performance

To further explore the impact of training iterations (Epochs) on model performance, experiments were conducted with Epoch values of 30, 50, and 80, assessing Accuracy, Precision, Recall, F1-score, and training duration. The corresponding experimental results are presented in the following table.

Table 9 Impact of Epochs on Model Performance and Training Time

Epoch	Accuracy	Precision	Recall	F1-score	TrainingTime (minutes)
30	62.06%	70.99%	64.43%	59.72%	7
50	94.25%	95.19%	93.75%	94.14%	12
80	96.55%	96.79%	96.34%	96.51%	31

In addition, Figure 27 depicts the evolution of the loss function over varying epochs, offering a comprehensive assessment of the model's convergence properties and generalization performance. Experimental results indicate that an increase in training iterations significantly impacts both model accuracy and computational complexity.

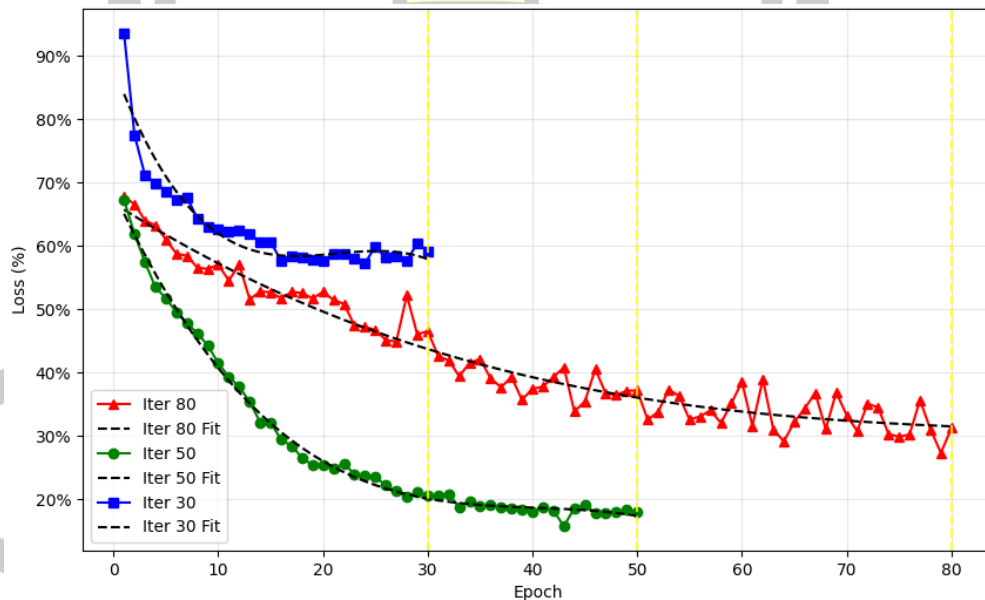


Figure 27 Loss Curve for Different Epochs during Model Training

(1) Epoch=30

The training time is only 7 minutes, but the accuracy (62.06%) and F1-score (59.72%) of the model are low, indicating insufficient training, limited feature extraction capability, and unsatisfactory model performance on the classification task.

(2) Epoch=50

The accuracy (94.25%) and recall (93.75%) of the model are greatly improved, and the F1-score reaches 0.9414, which indicates that the model has basically converged and is able to distinguish the samples efficiently, meanwhile the training time is controlled to be 12 minutes, which is within the permissible range of the computational resources, and it has a high practicability.

(3) Epoch=80

The model performance is further improved (accuracy 96.55%, F1-score 96.51%), but the gain compared to Epoch=50 is limited. In addition, the training time increased significantly to 31 minutes, with a substantial increase in computational cost. From the loss curves, there were more obvious fluctuations during the training process, indicating that the model may have entered the overfitting stage and the generalization ability decreased.

Taking into account the model's performance, computational cost, and generalization capability, Epoch=50 achieves an optimal balance between performance and computational resource consumption. On the one hand, it can effectively reduce the risk of overfitting, while ensuring the stability and reliability of the model in the task of wind turbine blade icing prediction. Therefore, in the subsequent experiments, we choose Epoch=50 as the optimal number of training rounds to ensure the model's generalization ability and practical application value.

4.5 Deep Learning Architecture Comparison

In Section 4.4.1, we examine how the number of CNN layers impacts the effectiveness of wind turbine blade icing prediction. The experimental findings indicate that increasing the CNN layers (to 2 or 3) enhances the model's performance. However, as a feature extraction tool, CNN is limited by the design of network structure, and there is still room for optimization. To further explore better feature

extraction strategies, we introduce multiple deep learning architectures to evaluate the suitability of diverse models for this task.

In this experiment, six distinct architectures were selected to comprehensively explore the temporal information and complex relationships within SCADA data. These architectures encompass a variety of feature extraction methods, including temporal attention, graph neural networks, and residual learning. The specific experimental results are presented in Table 10.

TAPNet (Temporal Attention Network)[84]: This model employs a temporal attention mechanism to enhance focus on critical time points, improving prediction accuracy by assigning adaptive weights to input data from different time steps.

GTAN (Graph Temporal Attention Network)[85]: By integrating Graph Neural Networks (GNN) with Temporal Attention Mechanisms, GTAN effectively captures correlations between turbines, thereby improving the prediction of icing risks in multi-turbine systems.

CBAM (Convolutional Block Attention Module)[86] : CBAM incorporates channel and spatial attention mechanisms into the CNN architecture, enabling the model to prioritize critical features while suppressing irrelevant noise. This design demonstrates exceptional performance in icing state prediction.

ResNet (Residual Network)[87]: ResNet addresses the gradient vanishing problem in deep networks through residual connections, allowing the model to efficiently learn complex features from SCADA data and better model the dynamics of wind turbine blade icing processes.

MLP (Multi-Layer Perceptron)[88] : Although MLP, composed of fully connected layers, exhibits limited capability in time series modeling, it remains effective in capturing underlying data trends and serves as a reliable benchmark model.

CNN_2Layer-BiGRU (Convolutional and Bidirectional GRU Combined Model): This model combines CNN for spatial feature extraction with BiGRU for temporal information processing. The integration enhances the model's ability to represent both spatial and temporal features, making it highly suitable for wind turbine blade icing prediction.

The experimental results discovered indicate that the CNN_2Layer-BiGRU model achieves the best index performance among the four supervised models. The accuracy is 96.55% with recall and precision of above 96%, which is clearly the best chance among others. Hence, it is obvious the approach of joint CNN and BiGRU exhibits remarkable benefits in SCADA data time-series modeling and feature selection, and it also provides a more efficient algorithm for detecting blade icing in wind turbines.

Table 10 Comparison of Performance Across Different Deep Learning Architectures

Model	Accuracy	Precision	Recall	F1-score	MCC
TAPNet	67.82%	69.08%	68.63%	67.75%	69.19%
GTAN	84.48%	84.48%	84.71%	84.46%	37.70%
CBAM	69.64%	66.75%	65.97%	64.75%	32.71%
ResNet	91.38%	91.36%	91.28%	91.31%	82.64%
MLP	85.63%	85.74%	85.96%	85.62%	71.70%
CNN_2Layer-BiGRU	96.55%	96.79%	96.34%	96.51%	93.13%

Moreover, ResNet is one of the best-trained approaches, with an accuracy of 91.38%. This further proves that the deep CNN features extraction is the main strength of this structure. GTAN and TAPNet show a comparative gain in Recall due to the fact that they are tailored for temporal attention, but still fail to reach the performance level of CNN-BiGRU.

4.6 Experiments with TPA Attention Mechanism

To validate the effectiveness of the Temporal Pattern Attention (TPA) mechanism in blade icing classification, this section presents a comprehensive evaluation of six mainstream models, comparing their performance before and after the introduction of TPA. The experiment is structured into several stages, including hyperparameter optimization, model configuration, training behavior analysis, and final performance comparison.

4.6.1 Hyperparameter Optimization Process

To determine optimal model configurations, a controlled variable strategy was applied, where individual hyperparameters were tuned sequentially. Initially, the CNN-BiGRU model was configured with 100 training epochs, a batch size of 100, a

learning rate of 0.001, and 100 hidden units for BiGRU. The performance impact of various CNN kernel sizes and filter numbers was then evaluated. It was observed that smaller kernel sizes (e.g., 3) and fewer filters (e.g., 32) led to reduced model accuracy and precision.

Experiments with increasing CNN layers from one to three revealed that deeper architectures did not necessarily enhance model performance. Consequently, a single-layer CNN was selected for its balance between complexity and effectiveness. Next, the BiGRU component was fine-tuned by adjusting hidden units and layer count. The optimal setting—64 hidden units and one BiGRU layer—was chosen based on minimized prediction error.

Finally, a dropout layer was added to mitigate overfitting, and a dropout rate of 0.2 yielded the best results. These steps ensured that the CNN-BiGRU-TPA model was both efficient and generalizable.

4.6.2 Comparison of Hyperparameter Configurations

The final hyperparameter values for each model are summarized in Table 12. Across all models, the learning rate was tuned within the [0.001, 0.005] range. Cosine Annealing was adopted as the learning rate scheduler. L2 regularization (weight decay between 0.0005 and 0.001) and dropout mechanisms were used to prevent overfitting. Recurrent models (e.g., BiLSTM) employed slightly higher dropout rates (up to 0.3) to improve generalization.

Table 11 Partial Monitoring Parameter Information of Wind Turbine Units

Model	Learning Rate	Weight Decay	Dropout	Hidden Size	Filter Sizes	Batch Size
CNN-TPA	0.0003	0.0005	0.1	N/A	[5, 7, 9]	512
LSTM-TPA	0.0003	0.0005	0.1	64	N/A	512
GRU-TPA	0.0003	0.0005	0.1	64	N/A	512
BiGRU-TPA	0.0003	0.0005	0.1	64/128	N/A	512
BiLSTM-TPA	0.0001	0.001	0.2	64/128	N/A	512
CNN-BiGRU-TPA	0.0003	0.0005	0.1	64/128	[5, 7, 9]	512

All models used a consistent batch size of 512 and cross-entropy as the loss function. CNN-based models utilized multiple kernel sizes (5, 7, 9) to capture multi-scale features. The number of hidden units ranged between 64 and 128 based on complexity and validation performance.

4.6.3 Performance Evaluation Across Models

Table 13 displays the classification results of the six models with the TPA module integrated. As observed, the CNN-BiGRU-TPA model achieved the highest performance, with 88.96% recall, 89.04% precision, 89.08% accuracy, 89.00% F1-score, and 78.00% MCC. This marks a notable improvement from the base CNN-BiGRU model, which had 85.6% accuracy and 71.5% MCC without TPA.

Table 12 Classification results of six models

Models	Recall	Precision	Accuracy	F1-score	MCC
CNN-TPA	79.67%	80.51%	78.74%	78.68%	60.17%
LSTM-TPA	78.07%	79.29%	77.01%	76.90%	57.35%
GRU-TPA	72.10%	72.63%	71.26%	71.20%	44.73%
BiLSTM-TPA	70.44%	72.84%	68.97%	68.46%	43.22%
BiGRU-TPA	74.95%	75.44%	74.14%	74.10%	50.39%
CNN-BiGRU-TPA	88.96%	89.04%	89.08%	89.00%	78.00%

Other models also showed moderate gains after introducing TPA. For example, the GRU model's MCC increased from 44.2% to 44.7%, and its accuracy rose from 69.5% to 71.3%, confirming TPA's general enhancement capability across different architectures.

4.6.4 Training Curve Analysis

Figure 28-33 shows the performance of different models during training across 50 epochs, highlighting key metrics. From the data, we studied the behavior of each model, focusing on the jumping phenomenon, flattening trends, and why some models show less significant loss reduction.

(1) Jumping phenomenon

The curves of the BiGRU-TPA model on multiple indicators, such as Precision (Figure 29) and MCC (Figure 31), showed obvious jumps. This phenomenon might be due to the model's sensitivity to the data, especially in the early training stages, when the model weights were updated drastically, resulting in large fluctuations in the metrics. Since the BiGRU model considered both forward and backward information, if the attention mechanism of the model was unstable, it would cause large changes in the weights of local time steps, leading to fluctuations in prediction accuracy.

Additionally, the jump in the CNN-BiGRU-TPA model was also reflected in the early stage, primarily due to the complexity of the model structure. The model required to balance between extracting local features in the convolutional layer and global temporal features in BiGRU. Early in training, the model might be trying to optimize the weights of these two components, causing the metrics to fluctuate greatly.

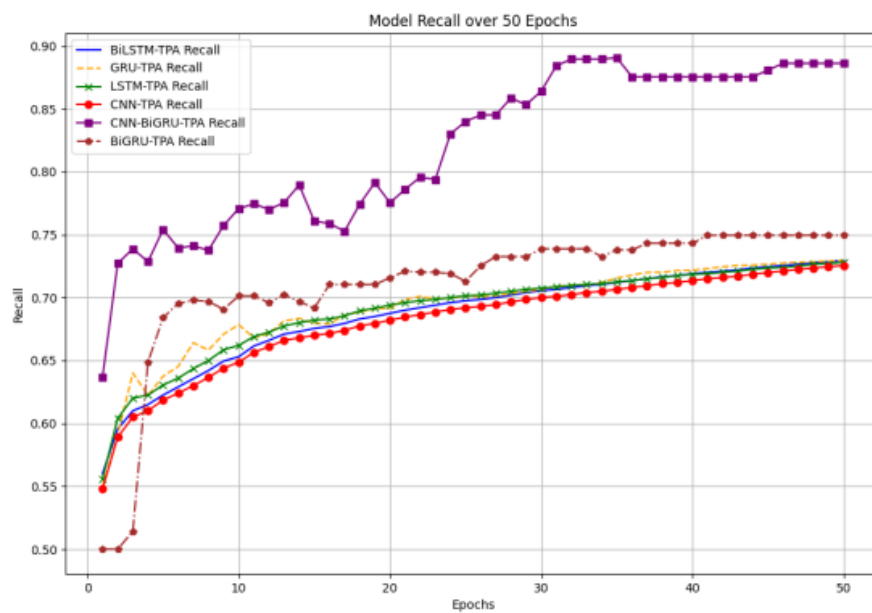
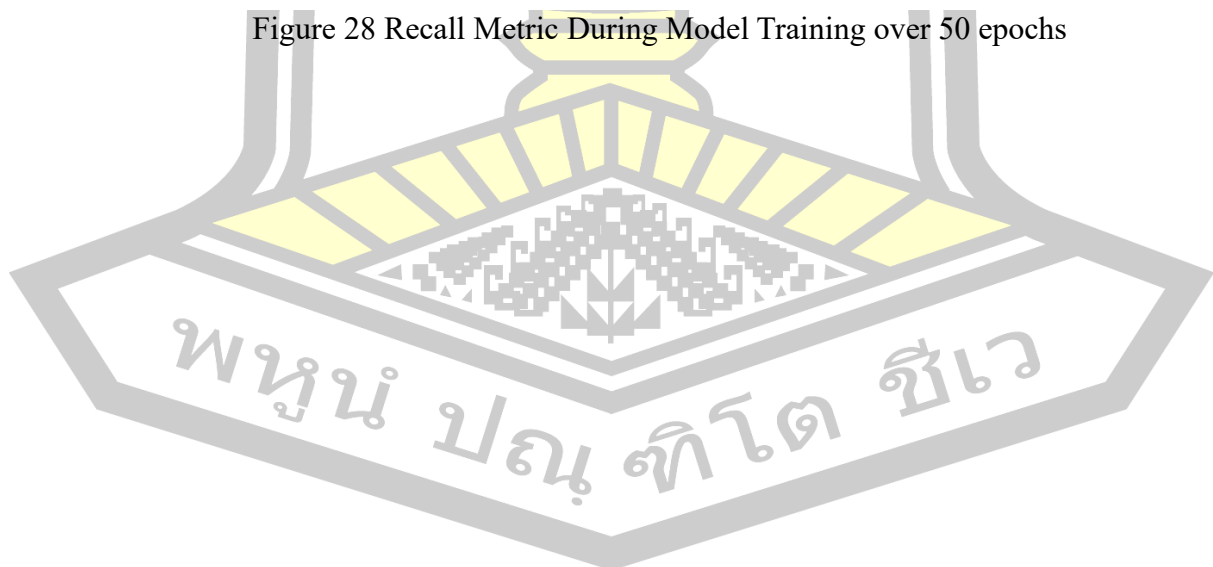


Figure 28 Recall Metric During Model Training over 50 epochs



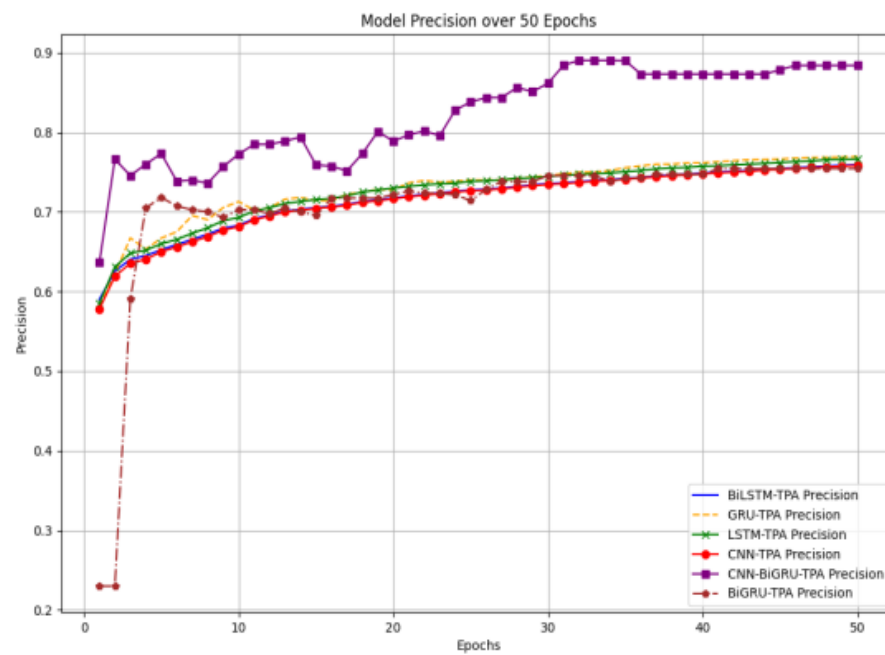


Figure 29 Precision Metric During Model Training over 50 epochs

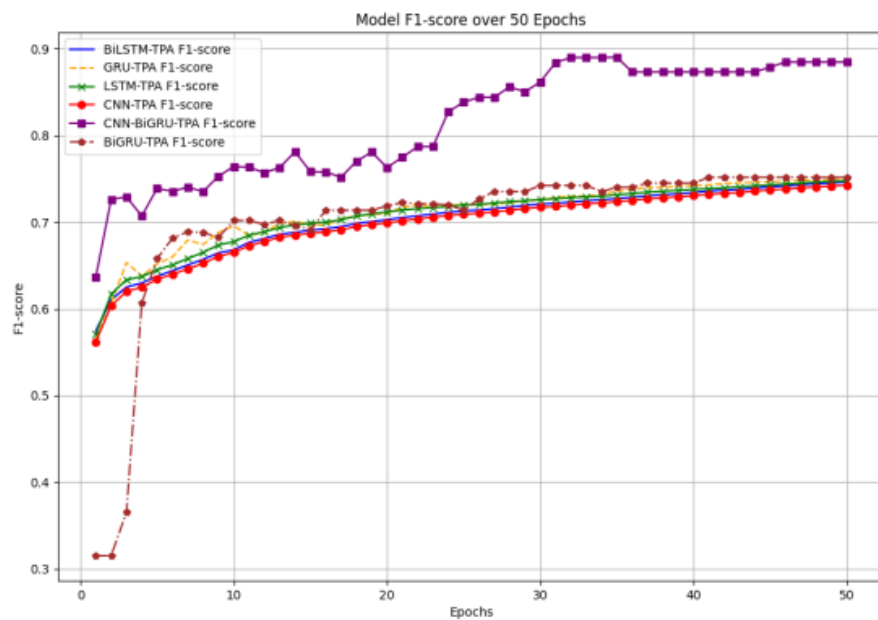


Figure 30 F1-Score Metric During Model Training over 50 epochs

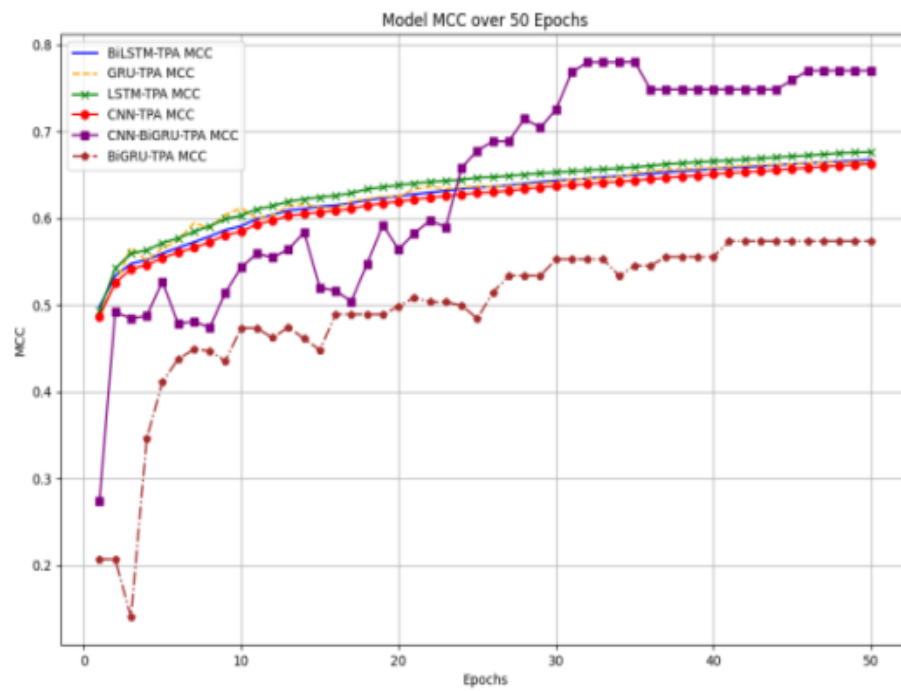


Figure 31 MCC Metric During Model Training over 50 epochs

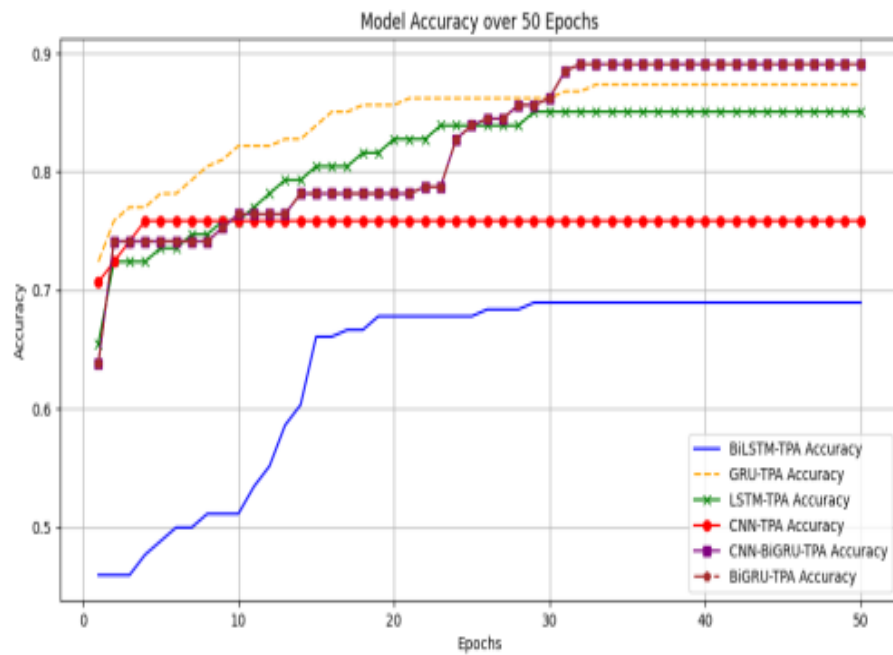


Figure 32 Accuracy Metric During Model Training over 50 epochs

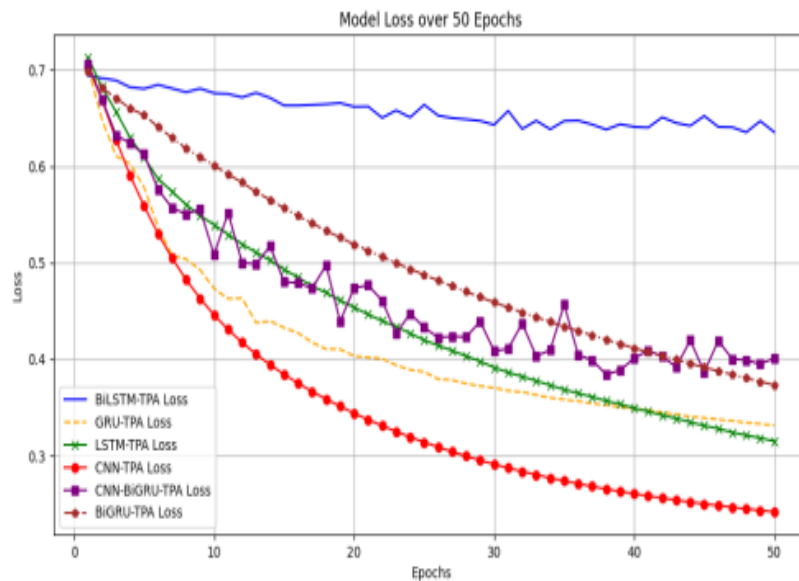


Figure 33 Loss Metric During Model Training over 50 epochs

(2) The flattening phenomenon

The index curves of the LSTM-TPA and GRU-TPA models were relatively stable, as seen in Figure 29 and 30. This might be because these models had simpler structures and focus on one-way temporal features. The optimization direction of these models remained relatively stable in each training cycle, resulting in a slow, steady improvement of the metrics with small fluctuations.

The curve of the CNN-TPA model also showed a relatively flat trend, especially the decrease in the Loss (Figure 33). After about 30 epochs, the loss value decreased by approximately 20%. This indicated that the convolutional network's feature extraction was relatively stable, and the model gradually converged during the training process.

(3) The loss of the CNN-BiGRU-TPA model

Although CNN-BiGRU-TPA excels in terms of Accuracy (Figure 32), Recall (Figure 28), and F1 score (Figure 30), its Loss curve (Figure 33) would not decrease significantly in the later stages of training (between 30 and 50 epochs). Several reasons might account for this:

Complexity of the model: CNN-BiGRU-TPA combined convolutional networks and bidirectional GRUs, along with a time-attention mechanism, resulting in

a large number of parameters during training. While this complex structure could be able to extract richer features, it also caused the model to fall into a local optimum, preventing further significant reductions in the loss value.

The flat and jumping changes in different model curves were closely related to the complexity of the model, optimization difficulty, and the structural characteristics. Although CNN-BiGRU-TPA exhibited superior classification performance, the model's complexity caused the loss value to plateau, reflecting the model's tendency to converge to a local optimum during optimization.

Specifically, the CNN-BiGRU model achieved an accuracy of 85.6%, an F1 score of 85.6%, and an MCC of 71.5% without the TPA module. After incorporating TPA, the CNN-BiGRU-TPA model's accuracy increased to 89.1%, the F1 score reached 89.0%, and the MCC improved to 78.0%, outperforming all other model architectures across the board. This demonstrates that the TPA module has strong adaptability in modeling multi-channel temporal information and can effectively capture dynamic variations in SCADA data that are critical for fault diagnosis.

Furthermore, basic architectures such as GRU and BiLSTM also exhibited noticeable gains with the addition of TPA. For instance, the MCC of the GRU model increased from 44.2% to 44.7%, and the accuracy improved from 69.5% to 71.3%. Although the magnitude of improvement was relatively modest, these results still indicate the general enhancement capability of TPA across different network structures.

4.6.5 Summary of Experimental Findings

In conclusion, the integration of the TPA module significantly improves the performance of various deep learning models in blade icing classification. The CNN-BiGRU-TPA model, in particular, outperforms all others, achieving superior results across all key metrics. TPA enhances the model's sensitivity to temporal patterns and improves consistency in fault classification.

Moreover, the results affirm that TPA is a versatile attention mechanism that can boost performance in both simple and complex network architectures. Its ability to increase recall and MCC highlights its suitability for addressing class imbalance and improving diagnostic reliability in wind turbine monitoring systems.

Chapter 5 Conclusion and Future Work

5.1 Conclusion

This study aimed to design and validate an advanced deep learning model for the fault diagnosis of blade icing in wind turbines using SCADA data. Given the increasing importance of renewable energy in achieving carbon neutrality goals and the growing scale of wind power deployment globally, effective monitoring and timely fault diagnosis of turbine components, especially the blades, have become critical to ensuring operational safety and reducing O&M costs. This research systematically addressed the technical challenges associated with the diagnosis of wind turbine blade icing under practical industrial conditions.

First, a comprehensive review of wind turbine systems, blade structures, fault types, and diagnostic techniques was conducted. The results highlighted the limitations of traditional condition monitoring technologies such as acoustic emission, vibration analysis, and image-based methods, which often involve high costs, installation complexity, or limited real-time applicability. In contrast, SCADA systems offer cost-effective, rich, and continuous data, making them ideal for fault detection through data-driven approaches. However, SCADA data also present challenges such as non-stationarity, high dimensionality, and weak direct indicators of blade health, which limit the effectiveness of shallow machine learning techniques.

To address these challenges, the study proposed a novel CNN-BiGRU-TPA architecture that integrates spatial feature extraction (CNN), temporal sequence modeling (BiGRU), and variable-level time attention weighting (TPA). The research pipeline encompassed SCADA data collection and cleaning, feature selection using Random Forest, spatiotemporal pattern extraction through CNN, temporal sequence modeling via BiGRU, and finally, refinement via TPA to enhance focus on critical temporal patterns.

Experimental results confirmed that the proposed CNN-BiGRU-TPA model outperformed all baseline and existing models. The model achieved an F1-score of 89.0%, accuracy of 89.1%, and MCC of 78.0%, significantly higher than traditional models such as LSTM (F1: 75.6%) and GRU (F1: 69.1%). Notably, when compared with the same CNN-BiGRU architecture without the TPA module, the TPA-enhanced

version showed an increase of 3.5% in accuracy and 6.5% in MCC, demonstrating the importance of variable-level time attention in diagnosing blade icing faults. The inclusion of TPA enabled the model to capture long-range dependencies and assign relevance to features dynamically, thereby improving recall (from 85.9% to 88.9%) and better handling imbalanced data.

The model's robustness was further validated through sensitivity analyses, including tests with different CNN depths and epoch settings. The 2-layer CNN was found to provide the best trade-off between computational cost and predictive performance. Similarly, training with 50 epochs achieved optimal convergence without overfitting, confirming the training stability of the proposed architecture. The model's scalability was also examined by evaluating its performance across different turbine units and seasonal datasets, suggesting strong generalization potential.

Moreover, a comparative analysis of various modern architectures such as TAPNet, GTAN, CBAM, and ResNet further highlighted the superiority of the proposed model. While attention-based models like TAPNet and GTAN performed reasonably well in capturing temporal patterns, their performance lagged behind due to their limited ability in integrating multi-variable SCADA data. ResNet exhibited promising results due to its deep convolutional structure but lacked the temporal modeling strength of BiGRU. Only the proposed CNN-BiGRU-TPA model could effectively integrate spatial, temporal, and variable-level attention features, leading to superior performance.

Through a thorough exploration of SCADA data characteristics, model architecture design, feature engineering, and hyperparameter tuning, this study has successfully demonstrated the feasibility and effectiveness of applying deep learning, particularly attention-enhanced recurrent neural networks, for the blade icing prediction problem. The proposed method not only reduces diagnostic latency but also increases the interpretability of fault features through attention mechanisms.

In practical applications, this model can be deployed within the SCADA systems of operational wind farms, providing real-time predictions and early warnings of icing conditions. By enabling timely de-icing measures, the model has the potential to improve turbine efficiency, reduce unplanned downtime, and prolong component

life. More importantly, this research contributes a valuable methodology to the broader field of data-driven fault diagnosis in renewable energy systems.

5.2 Future Work

While the current research has yielded promising results, several areas offer opportunities for further exploration:

Real-time deployment and edge integration: The current model has been trained and tested in an offline batch-processing environment. Future work will focus on embedding the model into edge devices or SCADA systems for real-time monitoring and diagnosis.

Multimodal data fusion: Incorporating additional data sources such as meteorological data, infrared images, or drone-based visual inspections could enhance the model's robustness and accuracy. A fusion model integrating SCADA and image features via multi-modal deep learning frameworks could be explored.

Transfer learning and generalization: Although the model performed well across different turbines, it is still limited to specific datasets. Transfer learning methods or domain adaptation techniques could be introduced to adapt the model to new turbines or wind farms with minimal retraining.

Explainability and interpretability: Future versions of the model could integrate explainable AI (XAI) techniques to better visualize how specific features contribute to predictions. This would help engineers and operators trust and validate the model's outputs.

Fault type extension: While this study focused on icing faults, the proposed architecture could be extended to other fault types, such as blade cracks, erosion, and pitch system failures. A unified multi-label classification model could enable comprehensive fault diagnosis.

Multi-step prediction and forecasting: Current work focuses on one-step classification (icing or not). Introducing sequence-to-sequence architectures could enable multi-step forecasting of icing trends, allowing predictive maintenance strategies to be implemented in advance.

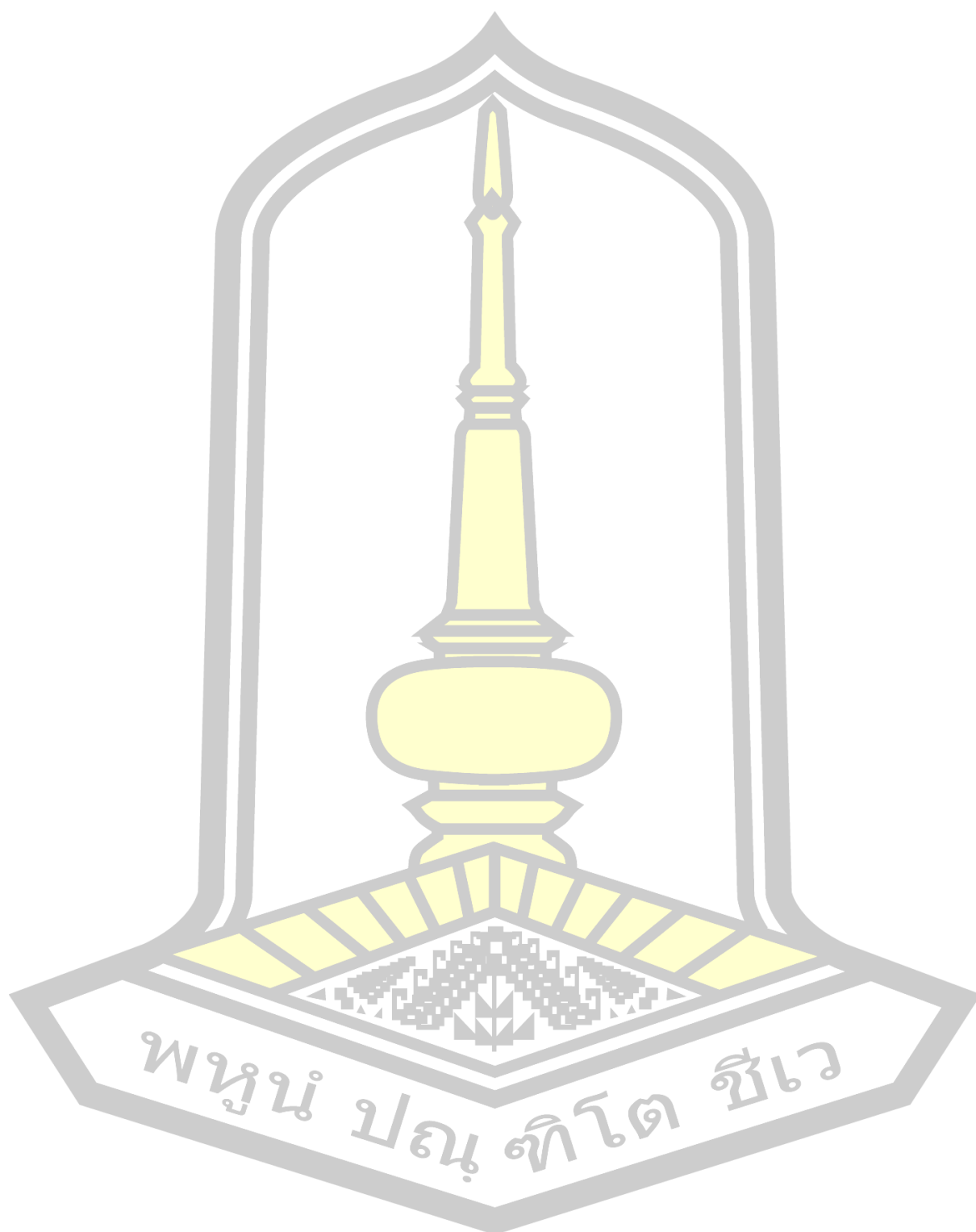
Model compression and lightweight deployment: To facilitate real-time applications in resource-constrained environments, future research could explore

model pruning, quantization, or knowledge distillation to compress the architecture without compromising accuracy.

In summary, the work presented in this thesis not only enhances the reliability of wind power systems but also lays a solid foundation for the continued development of intelligent fault diagnosis frameworks in renewable energy. By integrating deep learning with practical engineering needs, the research bridges the gap between academic innovation and industrial application, offering a viable pathway for intelligent, efficient, and scalable wind turbine monitoring systems in the future.



REFERENCES



1. Ecoclimax. 2018, April 7. Where the wind blows strongest: Exploring global wind power density potential. [Online]. Available: <https://www.ecoclimax.com/2018/04/wind-power-density-potential.html>. (August 14, 2025)
2. State Council of the People's Republic of China. 2021, April 2. [Online]. Available: https://www.gov.cn/xinwen/2021-04/02/content_5597403.html. (August 14, 2025)
3. Zhang, L. (2021). Fault diagnosis of wind turbine blades based on deep learning (Master's thesis, Inner Mongolia University of Science and Technology)
4. Yang, Y. (2020). Research on fault diagnosis system for wind turbine blades based on SCADA data (Master's thesis, North China Electric Power University)
5. Borovykh, A., Oosterlee, C. W., & Bohté, S. M. (2019). Generalization in fully-connected neural networks for time series forecasting. *Journal of Computational Science*, 36, 101020. <https://doi.org/10.1016/j.jocs.2019.101020>
6. Pandya, S. B., Patel, V. P., Patel, R. B., Tanwar, S., & Kumar, N. (2024). Multi-objective snow ablation optimization algorithm: An elementary vision for security-constrained optimal power flow problem incorporating wind energy source with FACTS devices. *International Journal of Computational Intelligence Systems*, 17(1), 33. <https://doi.org/10.1007/s44196-023-00246-w>
7. Nuruzzaman, M., & Rana, S. (2025). IoT-enabled condition monitoring in power distribution systems: A review of SCADA-based automation, real-time data analytics, and cyber-physical security challenges. *Journal of Sustainable Development and Policy*, 1(1), 25–43
8. Guo, Y., Li, X., Wang, Z., Zhang, J., & Chen, Y. (2023). Intelligent manufacturing management system based on data mining in artificial intelligence energy-saving resources. *Soft Computing*, 27(7), 4061–4076. <https://doi.org/10.1007/s00500-023-07853-1>
9. Dao, C., Kazemtabrizi, B., & Crabtree, C. (2019). Wind turbine reliability data review and impacts on levelised cost of energy. *Wind Energy*, 22(12), 1848–1871. <https://doi.org/10.1002/we.2408>
10. Rahimilarki, R., Tahan, M., Eslami, M., & Razavi-Far, R. (2022). Convolutional neural network fault classification based on time-series analysis for

benchmark wind turbine machine. *Renewable Energy*, 185, 916–931.
<https://doi.org/10.1016/j.renene.2021.12.090>

11. Lin, Y., Liu, Y., Sun, X., & Kang, J. (2016). Fault analysis of wind turbines in China. *Renewable and Sustainable Energy Reviews*, 55, 482–490.
<https://doi.org/10.1016/j.rser.2015.10.142>

12. Wikipedia. (n.d.). Components of wind turbines. In Wikipedia. Retrieved August 14, 2025, from https://en.wikipedia.org/wiki/Wind_turbine

13. Encyclopædia Britannica. (n.d.). Renewable energy. In Britannica. Retrieved March 12, 2024, from <https://www.britannica.com/science/renewable-energy>

14. Abouhnik, A., & Albarbar, A. (2012). Wind turbine blades condition assessment based on vibration measurements and the level of an empirically decomposed feature. *Energy Conversion and Management*, 64, 606–613

15. Chen, X., & Eder, M. A. (2020). A critical review of damage and failure of composite wind turbine blade structures. In *IOP Conference Series: Materials Science and Engineering* (Vol. 942, No. 1, 012014). IOP Publishing

16. Baidu Images. n.d. Images of wind turbine blade faults. [Online]. Available: <https://image.baidu.com/search/index?tn=baiduimage&ps=1&ct=201326>

5 9 2 & 1 m = -1 & c l = 2 & n c = 1 & i e = u t f -
 8&lid=af10b1ae003d8d27&dyTabStr=MCwxMiwz

LDEsMiwMyw3LDYsNSw5&word=%E9%A3%8E%E6%9C%BA%E5%8
 F%B6%E7%89%87%E6%96%AD%E8%A3%82. (August 14, 2025)

18. Xu, D., Liu, P., & Chen, Z. (2021). Damage mode identification and singular signal detection of composite wind turbine blade using acoustic emission. *Composite Structures*, 255, 112954

19. Qiao, W., & Lu, D. (2015). A survey on wind turbine condition monitoring and fault diagnosis—Part II: Signals and signal processing methods. *IEEE Transactions on Industrial Electronics*, 62(10), 6546–6557

20. Abdullah, A. (2024). Review of vibration-based damage assessment in wind turbines (Master's thesis, Politecnico di Torino)

21. Shojaei, D., Jafary, P., & Zhang, Z. (2024). Mixed reality-based concrete crack detection and skeleton extraction using deep learning and image processing. *Electronics*, 13(22), 4426
22. Memari, M., et al. (2024). Review on the advancements in wind turbine blade inspection: Integrating drone and deep learning technologies for enhanced defect detection. *IEEE Access*, 12, 33236–33282
23. Pandit, R., & Wang, J. (2024). A comprehensive review on enhancing wind turbine applications with advanced SCADA data analytics and practical insights. *IET Renewable Power Generation*, 18(4), 722–742
24. Castellani, F., et al. (2024). Wind turbine gearbox condition monitoring through the sequential analysis of industrial SCADA and vibration data. *Energy Reports*, 12, 750–761
25. Abbas, M., Chafouk, H., & Ardjoun, S. A. E. M. (2024). Fault diagnosis in wind turbine current sensors: Detecting single and multiple faults with the extended Kalman filter bank approach. *Sensors*, 24(3), 728
26. Ziane, D., et al. (2024). Fuzzy logic-enhanced direct power control for wind turbines with doubly fed induction generators. *Results in Engineering*, 24, 103557
27. Meng, D., et al. (2025). Reliability analysis of wind turbine gearboxes: Past, progress and future prospects. *International Journal of Structural Integrity*, 16(1), 4–38
28. Zhao, S., et al. (2025). Comparison and analysis of major research methods for non-destructive testing of wind turbine blades. *Review of Scientific Instruments*, 96(5)
29. Jiang, T., Lv, P., & Li, D. (2024). A new shape reconstruction method for monitoring the large deformations of offshore wind turbine towers. *Ocean Engineering*, 312, 119253
30. Lee, C. Y., & Maceren, E. D. C. (2024). Wind energy system fault classification and detection using deep convolutional neural network and particle swarm optimization-extreme gradient boosting. *IET Energy Systems Integration*, 6(4), 479–497

31. Jiang, G., et al. (2025). A symmetry-aware pitch feature encoder-decoder network for condition monitoring of pitch bearings in wind turbines. *IEEE Sensors Journal*
32. Wang, L., et al. (2024). A novel approach to wind turbine blade icing detection with limited sensor data via spatiotemporal attention Siamese network. *IEEE Transactions on Industrial Informatics*, 20(6), 8993–9005
33. Zhang, S., Robinson, E., & Basu, M. (2024). Wind turbine condition monitoring based on three fitted performance curves. *Wind Energy*, 27(5), 429–446
34. Xu, G., et al. (2024). Deep reinforcement learning based model-free optimization for unit commitment against wind power uncertainty. *International Journal of Electrical Power & Energy Systems*, 155, 109526
35. HTVCXDNX. n.d. [Online]. Available: <http://www.htvcxdxny.com/public/index.php/index>. (August 14, 2025)
36. Kreutz, M., et al. (2019). Machine learning-based icing prediction on wind turbines. *Procedia CIRP*, 81, 423–428
37. Peng, C., et al. (2019). Icing prediction of fan blade based on a hybrid model. *International Journal of Performability Engineering*, 15(11), 2882
38. Peng, C., et al. (2019). Short-term prediction of generator blade ice fault based on multi-AN. In *2019 Seventh International Conference on Advanced Cloud and Big Data (CBD)*. IEEE
39. Hua, Y., Guo, J., & Zhao, H. (2015). Deep belief networks and deep learning. In *Proceedings of 2015 International Conference on Intelligent Computing and Internet of Things*. IEEE
40. Ma, J., Ma, L., & Tian, X. (2019). Wind turbine blade icing prediction based on deep belief network. In *2019 4th International Conference on Mechanical, Control and Computer Engineering (ICMCCE)*. IEEE
41. Kreutz, M., et al. (2021). Convolutional neural network with dual inputs for time series ice prediction on rotor blades of wind turbines. *Procedia CIRP*, 104, 446–451
42. Zhang, L., et al. (2024). Research on wind turbine icing prediction data processing and accuracy of machine learning algorithm. *Renewable Energy*, 237, 121566

43. Awujoola Olalekan, J., Ogwueleka, F., & Odion, P. (2020). Effective and accurate bootstrap aggregating (Bagging) ensemble algorithm model for prediction and classification of hypothyroid disease. *International Journal of Computer Applications*, 975, 8887
44. Lu, S., et al. (2023). Event-triggered federated learning for fault diagnosis of offshore wind turbines with decentralized data. *IEEE Transactions on Automation Science and Engineering*, 21(2), 1271–1283
45. Li, S., Peng, Y., & Bin, G. (2023). Prediction of wind turbine blades icing based on CJBm with imbalanced data. *IEEE Sensors Journal*, 23(17), 19726–19736
46. Xu, J., Tan, W., & Li, T. (2020). Predicting fan blade icing by using particle swarm optimization and support vector machine algorithm. *Computers & Electrical Engineering*, 87, 106751
47. Meng, H., et al. (2020). Forecast of fan blade icing combining with random forest and SVM. *Electrical Measurement & Instrumentation*, 57, 66–71
48. Tao, T., et al. (2021). Wind turbine blade icing diagnosis using hybrid features and Stacked-XGBoost algorithm. *Renewable Energy*, 180, 1004–1013
49. Wang, L., & Zhao, Y. (2022). Wind turbine blade icing fault prediction based on SCADA data by XGBoost. *Academic Journal of Engineering and Technology Science*, 5(2), 30–35
50. Liu, L., et al. (2021). Data-driven prediction of wind turbine blade icing. In *2021 China Automation Congress (CAC)*. IEEE
51. Ying, L., et al. (2024). Graph temporal attention network for imbalanced wind turbine blade icing prediction. *IEEE Sensors Journal*, 24(6), 9187–9196
52. Huang, C., Sun, X., & Zhang, Y. (2024). Tiny-machine-learning-based supply canal surface condition monitoring. *Sensors*, 24(13), 4124
53. Zhang, Y., Bader, S., & Oelmann, B. (2022). A lightweight convolutional neural network model for concrete damage classification using acoustic emissions. In *2022 IEEE Sensors Applications Symposium (SAS)*. IEEE
54. Guo, P., & Infield, D. (2021). Wind turbine blade icing detection with multi-model collaborative monitoring method. *Renewable Energy*, 179, 1098–1105
55. Man, J., et al. (2023). Semi-supervised blade icing detection method based on tri-XGBoost. *Actuators*. MDPI

56. Bai, X., et al. (2023). Wind turbine blade icing diagnosis using RFECV-TSVM pseudo-sample processing. *Renewable Energy*, 211, 412–419
57. Yang, X., et al. (2020). Diagnosis of blade icing using multiple intelligent algorithms. *Energies*, 13(11), 2975
58. Dazhong, L., Jiarui, L., & Huaying, Z. (2019). Prediction method of fan blade icing based on deep fully connected neural network. *Electric Power Science and Engineering*, 35(4), 39
59. Xiong, C., et al. (2021). Short-term icing status prediction model of wind turbine blades based on Bi-LSTM and SVM models. *Sichuan Electric Power Technology*, 44(3), 88–94
60. Tao, C., et al. (2023). Wind turbine blade icing prediction using focal loss function and CNN-attention-GRU algorithm. *Energies*, 16(15), 5621
61. Jiang, G., et al. (2023). Imbalanced learning for wind turbine blade icing detection via spatio-temporal attention model with a self-adaptive weight loss function. *Expert Systems with Applications*, 229, 120428
62. Tian, W., et al. (2021). A multilevel convolutional recurrent neural network for blade icing detection of wind turbine. *IEEE Sensors Journal*, 21(18), 20311–20323
63. Wang, X., et al. (2022). Detecting wind turbine blade icing with a multiscale long short-term memory network. *Energies*, 15(8), 2864
64. Xiong, C., et al. (2021). Research on icing status forecasting of wind turbine blades based on machine learning. In 2021 6th Asia Conference on Power and Electrical Engineering (ACPEE). IEEE
65. Li, Y., et al. (2022). Prediction of wind turbine blades icing based on feature selection and 1D-CNN-SBiGRU. *Multimedia Tools and Applications*, 81(3), 4365–4385
66. Zhang, Y., et al. (2025). Survey of quantization techniques for on-device vision-based crack detection. *arXiv preprint arXiv:2502.02269*
67. Muthumala, U., et al. (2024). Comparison of tiny machine learning techniques for embedded acoustic emission analysis. In 2024 IEEE 10th World Forum on Internet of Things (WF-IoT). IEEE
68. Zhang, Y., et al. (2025). On-device crack segmentation for edge structural health monitoring. *arXiv preprint arXiv:2505.07915*

69. Ye, F., & Ezzat, A. A. (2024). Icing detection and prediction for wind turbines using multivariate sensor data and machine learning. *Renewable Energy*, 231, 120879
70. Wang, L., et al. (2023). An unsupervised approach to wind turbine blade icing detection based on beta variational graph attention autoencoder. *IEEE Transactions on Instrumentation and Measurement*, 73, 1–12
71. Cheng, X., et al. (2021). Temporal attention convolutional neural network for estimation of icing probability on wind turbine blades. *IEEE Transactions on Industrial Electronics*, 69(6), 6371–6380
72. Tao, C., et al. (2024). Wind turbine blade icing diagnosis using B-SMOTE-Bi-GRU and RFE combined with icing mechanism. *Renewable Energy*, 221, 119741
73. Wei, F., Guo, Z., Han, Q., & Qi, W. (2024). Estimation of wind turbine blade icing volume based on binocular vision. *Applied Sciences*, 15(1), 114
74. Tampinongkol, F. F., Kamila, A. R., Wardhana, A. C., Kusuma, A. W. C., & Revaldo, D. (2024). Implementation of random forest classification and support vector machine algorithms for phishing link detection. *Journal of Informatics Information System Software Engineering and Applications (INISTA)*, 7(1), 127–137
75. Dai, J., Rotea, M., & Kehtarnavaz, N. (2024). An ensemble network for high-accuracy and long-term forecasting of icing on wind turbines. *Sensors*, 24(24), 8167
76. Wang, Q., Si, G., Shan, Z., Fan, D., Li, B., & Wang, X. (2025). Wind turbine blade icing predictive fault warning system based on conditional time series denoising diffusion data restoration model. *IEEE Transactions on Automation Science and Engineering*
77. Wang, L., He, Y., Zhou, Y., Li, L., Wang, J., Zhao, Y., & Du, B. (2024). A novel approach to wind turbine blade icing detection with limited sensor data via spatiotemporal attention Siamese network. *IEEE Transactions on Industrial Informatics*, 20(6), 8993–9005
78. Chen, W., et al. (2022). Wind turbine blade icing detection using a novel bidirectional gated recurrent unit with temporal pattern attention and improved coot optimization algorithm. *Measurement Science and Technology*, 34(1), 014004

79. Guilmeau, T., Chouzenoux, E., & Elvira, V. (2021). Simulated annealing: A review and a new scheme. In 2021 IEEE Statistical Signal Processing Workshop (SSP). IEEE

81. Mao, A., Mohri, M., & Zhong, Y. (2023). Cross-entropy loss functions: Theoretical analysis and applications. In International Conference on Machine Learning. PMLR

82. Sherstinsky, A. (2020). Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. *Physica D: Nonlinear Phenomena*, 404, 132306

83. Abduljabbar, R. L., Dia, H., & Tsai, P.-W. (2021). Unidirectional and bidirectional LSTM models for short-term traffic prediction. *Journal of Advanced Transportation*, 2021(1), 5589075

84. Zhang, X., et al. (2020). Tapnet: Multivariate time series classification with attentional prototypical network. In Proceedings of the AAAI Conference on Artificial Intelligence

85. Ying, L., et al. (2024). Graph temporal attention network for imbalanced wind turbine blade icing prediction. *IEEE Sensors Journal*

86. Lu, Q., Ye, W., & Yin, L. (2023). ResDenIncepNet-CBAM with principal component analysis for wind turbine blade cracking fault prediction with only short time scale SCADA data. *Measurement*, 212, 112696

87. Wang, Z., Yan, W., & Oates, T. (2017). Time series classification from scratch with deep neural networks: A strong baseline. In 2017 International Joint Conference on Neural Networks (IJCNN). IEEE

88. Desai, M., & Shah, M. (2021). An anatomization on breast cancer detection and diagnosis employing multi-layer perceptron neural network (MLP) and convolutional neural network (CNN). *Clinical eHealth*, 4, 1–11

BIOGRAPHY

NAME

DATE OF BIRTH

27 October 1987

PLACE OF BIRTH

ADDRESS

Room 205, Building 3, Forest City, Section 1, Xiangfu
East Road, Yuhua District, Changsha, Hunan
Province, 410007 China.

POSITION

Teacher

PLACE OF WORK

Human Mechanical & Electrical Polytechnic

EDUCATION

2007, Roi-Et Zhiyuan Middle School

2011, Bachelor Degree (B.Eng), Mechanical Engineering,
Hunan Normal University

2014, Master Degree (M.Eng), Mechanical and Electronic
Engineering, Jiangxi University of Science and Technology

2025, Doctor Degree, Electrical and Computer
Engineering, Mahasarakham University

