Universidade Federal do Rio Grande do Norte
Centro de Tecnologia
Programa de Pós-Graduação em Engenharia Elétrica e de Computação

Industry 4.0-Compliant Artificial Intelligence-Based Power Transformer Fault Classification Method During Data Missing Conditions

Ingrid Thaís Azevêdo Dantas

Advisor: Prof. Dr. Flavio Bezerra Costa

Co-Advisor: Prof. Dr. Rodrigo Prado de Medeiros

Master's Dissertation presented to the Graduate Program in Electrical and Computer Engineering at UFRN (concentration area: Automation and Systems) as part of the requirements for obtaining the title of Master of Science.

PPgEEC order number: M686
Natal, RN, March 2025

Universidade Federal do Rio Grande do Norte - UFRN
Sistema de Bibliotecas - SISBI
Catalogação de Publicação na Fonte. UFRN - Biblioteca Central Zila Mamede

Dantas, Ingrid Thaís Azevêdo.
Industry 4.0-Compliant Artificial Intelligence-based power transformer fault classification method during data missing conditions / Ingrid Thaís Azevêdo Dantas. - 2025.
73 f.: il.

Dissertation (master's) - Federal University of Rio Grande do Norte, Center of Technology, Graduate Program in Electrical and Computer Engineering, Natal, 2025.
Orientação: Prof. Dr. Flavio Bezerra Costa.
Coorientação: Prof. Dr. Rodrigo Prado de Medeiros.

1. Power transformers - Dissertation. 2. Fault classification - Dissertation. 3. Wavelet transform - Dissertation. 4. Missing data - Dissertation. 5. Machine learning - Dissertation. 6. Industry 4.0 - Dissertation. I. Costa, Flavio Bezerra. II. Medeiros, Rodrigo Prado de. III. Título.

RN/UF/BCZM CDU 621.3

Elaborado por Jackeline dos Santos Pinheiro da Silva Maia Cavalcanti - CRB-15/317

To my family, to my parents and grandparents, for being part of everything I am, and for reminding me every day that I can explore the world, but I know exactly where I belong.

Acknowledgment

Reaching this point has been an intense journey, filled with challenges, doubts, discoveries, and, above all, learning. No achievement is ever truly solitary, and this dissertation only became possible thanks to the support and presence of many dear people by my side.

I thank God for granting me health, courage, and resilience to keep moving forward, even in the most difficult moments.

To my advisor, Prof. Dr. Flavio Bezerra Costa, and my co-advisor, Prof. Dr. Rodrigo Prado de Medeiros, my heartfelt thanks for your patience, guidance, and careful attention throughout this journey. Thank you for believing in my potential, even when I had doubts myself.

To the professors of the graduate program in Electrical and Computer Engineering at UFRN, I'm deeply grateful for all the knowledge shared and the ongoing support throughout the course. I extend a special thanks to the members of the examination committee for their valuable feedback, which helped improve this work.

To my family, my safe harbor. Thank you for always being there for me, even in silence, with love and unconditional support. Everything I am owes so much to you.

To the friends who walked with me during this path, even from afar: thank you for the encouraging messages and for reminding me, in moments of exhaustion, why it was worth continuing.

Finally, to everyone who, in one way or another, was part of this journey, my deepest and most sincere gratitude.

Resumo

Transformadores de potência são elementos fundamentais em sistemas elétricos, responsáveis pela transferência eficiente de energia entre diferentes níveis de tensão. Apesar de sua robustez, esses equipamentos estão sujeitos a falhas ao longo do tempo, como faltas elétricas internas, que comprometem não apenas o próprio transformador, mas também a estabilidade de todo o sistema interligado.Diante desse cenário, torna-se cada vez mais relevante o uso de soluções inteligentes que viabilizem o monitoramento contínuo e a classificação precisa das faltas, contribuindo para diagnósticos mais ágeis, para assim reduzir o tempo de indisponibilidade dos equipamentos. Este trabalho propõe uma abordagem inovadora para a classificação de faltas internas em transformadores de potência, com foco especial naquelas que ocorrem nas buchas. A proposta baseia-se no desenvolvimento de um método que alia técnicas matemáticas avançadas a modelos de aprendizado de máquina supervisionado. Um dos diferenciais da abordagem está na sua capacidade de operar mesmo com dados incompletos, uma condição frequentemente observada em ambientes industriais reais, em que falhas de comunicação, defeitos em sensores ou ruídos nos sinais, por exemplo, podem afetar a integridade medições e consequentemente, dos diagnósticos. A metodologia combina a Transformada Wavelet Estacionária com Bordas em Tempo Real (RT-BSWT) com modelos de aprendizado de máquina. Após a aplicação dessa transformada, são extraídas as energias dos coeficientes que formam a base para a etapa seguinte do método utilizando aprendizado de máquina. Três algoritmos supervisionados foram utilizados na implementação do sistema: Árvore de Decisão, Floresta Aleatória e Regressão Logística. A escolha desses modelos se deu com base em sua boa performance em tarefas de classificação e por sua interpretabilidade. O sistema foi avaliado a partir de uma base de dados simulada, composta por diferentes tipos de faltas internas na bucha do transformador, incluindo faltas monofásicas, bifásicas, trifásicas e fase-terra, aplicadas tanto no lado primário quanto no secundário. Os testes também consideraram variações de ângulo e resistência de falta, condições de carga, presença de ruído e, principalmente, a perda de uma das correntes, com o objetivo de simular condições operacionais realistas. Os resultados demonstraram que a abordagem proposta é altamente eficaz, alcançando taxas de acurácia superiores a 95% em cenários ideais. Entre os algoritmos testados, a Floresta Aleatória apresentou o melhor desempenho. A eficácia do método também foi validada por meio de uma comparação direta com a abordagem baseada apenas em processamento por wavelets, sem uso de aprendizado de máquina. Embora o método tradicional tenha apresentado desempenho razoável em cenários ideais (acurácia de até 85%), sua performance foi severamente afetada por perda de dados e presença de ruído, chegando a 36,9% de acurácia em condições adversas. Além disso, ele foi limitado na identificação de certos tipos de falta, como bifásica com terra do lado secundário ou trifásica na falta de um dado. Por outro lado, o método com aprendizado de máquina demonstrou maior precisão e adaptabilidade, classificando corretamente todos os 20 tipos de faltas avaliados, mesmo sob perdas de sinal e com diferentes níveis de ruído. Além da contribuição técnica, este trabalho está alinhado aos conceitos da Indústria 4.0, ao integrar técnicas de análise inteligente, capacidade de resposta em tempo real e robustez frente à perda de dados. Isso amplia consideravelmente o potencial de aplicação da metodologia em ambientes industriais modernos. O sistema desenvolvido pode ser incorporado a arquiteturas existentes de proteção e monitoramento, oferecendo suporte direto à tomada de decisão operacional. Em síntese, este trabalho oferece uma contribuição significativa para a área de diagnóstico e classificação de faltas internas em transformadores de potência. A capacidade de operar sob condições adversas e de manter alta acurácia mesmo com dados incompletos torna o método promissor tanto para aplicações acadêmicas quanto industriais. Espera-se que esta proposta sirva de base para o desenvolvimento de sistemas de proteção mais inteligentes, contribuindo com a segurança e eficiência no setor elétrico.

Palavras-chave: Transformadores de potência, Classificação de Faltas, Perda de Dados, Aprendizado de Máquina, Wavelets, Indústria 4.0.

Abstract

Power transformers are fundamental components in electrical systems, responsible for the efficient transfer of energy between different voltage levels. Despite their robustness, these devices are subject to failures over time, such as internal electrical faults that can compromise not only the transformer itself but also the stability of the entire interconnected system. In this context, the use of intelligent solutions that enable continuous monitoring and accurate fault classification becomes increasingly relevant, contributing to faster diagnostics and, thus, reducing equipment downtime. This work proposes an innovative approach for classifying internal faults in power transformers, with a particular focus on those occurring in the bushings. The proposed method is based on the development of a classification system that combines advanced mathematical techniques with supervised machine learning models. A key feature of the approach is its ability to operate effectively even with incomplete data, a common condition in real industrial environments where communication failures, sensor defects, or signal noise can compromise measurement integrity and, consequently, the diagnostic process. The methodology integrates the Real-Time Bounded Stationary Wavelet Transform (RT-BSWT) with machine learning models. After applying this transform to the current signals, energy values are extracted from the coefficients and used as the input features for the classification stage. Three supervised algorithms were implemented: Decision Tree, Random Forest, and Logistic Regression. These models were selected for their strong performance in classification tasks and for their interpretability. The system was evaluated using a simulated dataset containing different types of internal faults in transformer bushings, including single-phase, two-phase, three-phase, and phase-to-ground faults, applied to both the primary and secondary sides. The tests also considered variations in fault resistance, fault angle, load conditions, noise, and, most importantly, the absence of one phase current, aiming to simulate realistic operational scenarios. The results demonstrated that the proposed approach is highly effective, achieving accuracy rates above 95% under ideal conditions. Among the tested models, the Random Forest algorithm performed the best. The effectiveness of the method was also validated through a direct comparison with a traditional wavelet-only approach, without machine learning. While the conventional method showed reasonable performance in ideal conditions, with up to 85% accuracy, its effectiveness was severely affected by data loss and noise, dropping to 36.9% accuracy in adverse scenarios. It also struggled to identify specific fault types, such as secondary-side two-phase-to-ground faults or three-phase faults under missing data. In contrast, the machine learning-based method showed higher accuracy and adaptability, correctly classifying all 20 fault types evaluated, even under signal loss and different noise levels. Beyond its technical contribution, this work aligns with the principles of Industry 4.0 by integrating intelligent data analysis, real-time responsiveness, and robustness against data loss. These characteristics significantly expand the potential for applying the proposed methodology in modern industrial environments. The system can be integrated into existing protection and monitoring architectures, providing direct support for operational decision-making. In summary, this work presents a significant contribution to the field of diagnosis and classification of internal faults in power transformers. Its ability to perform under adverse conditions and maintain high accuracy even with incomplete data makes it a promising solution for both academic research and practical applications in the electric power sector. This approach may serve as a foundation for the development of smarter protection systems, contributing to increased safety and operational efficiency.

Keywords: Power Transformers, Fault classification, Wavelet Transform, Missing Data, Machine Learning, Industry 4.0.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Relevance of the Topic

Power transformers play a vital role in power systems as they enable the integration among the generation, transmission, distribution, and consumption sectors. They are essential for adjusting voltage levels, minimizing losses, and ensuring the delivery of electricity at the appropriate voltage for the efficient operation of the electrical grid (Sudha et al. 2022).

Similarly to other power system components, power transformers are exposed to the adverse effects of fault-induced electromagnetic transients. In the event of a fault, for example, power transformers may require repair or replacement, which can cause major financial losses as well as electric power supply outages to consumers due to their temporary unavailability (Abbasi 2022). Therefore, it is of utmost importance the rapid fault elimination in power transformers (Shaban et al. 2019).

According to Ahmadi et al. (2021), internal faults in power transformers, such as turn-to-turn, phase-to-phase, and phase-to-ground short circuits, are among the main causes of transformer failure. These faults are typically related to insulation degradation and can remain undetected if their intensity is low, potentially evolving into more severe conditions when protection systems fail to operate in time.

Ideally, the power transformer protection should operate with high sensitivity and speed for internal fault detection, ensuring no operation is necessary for external faults and energization maneuvers. The most commonly used protection for transformers with power ratings from 10 MVA is the differential protection, with percentage differential relays being the most common for this purpose (Medeiros, Costa, Silva, Muro, Júnior & Popov 2021).

In addition to transformer protection, classifying faults is also an essential task since it provides valuable information about which phases are involved in the fault, which ensures better monitoring of the transformer's health over time. Identifying the type of internal fault in power transformers is as important as detecting the fault itself because quickly identifying the type of fault can reduce the maintenance period (Ananwattanaporn et al. 2017). When the specific type of internal fault is determined, repairs can be carried out more promptly, minimizing the transformer downtime and helping to prevent additional damage.

With the advent of Industry 4.0, the use of real-time communication between the most varied devices has become increasingly established in the industrial sector. Devices for collecting current signals are generally installed at the terminals of power transformers in industrial plants. These devices are known as current transformers and have the basic function of transforming the high currents of the power system into lower values that can be read by sensors and digital relays. However, the process of collecting and obtaining a dataset of electrical currents for diagnosing and monitoring transformers in modern industrial power plants, for example, is not always a trivial task. In practice, monitoring and data acquisition systems are vulnerable to electromagnetic and environmental interference, which can result in abnormal readings or even data loss, thus impacting the correct operation of the equipment (Cai et al. 2024). Ensuring the industrial plant keeps its operation even in the event of a lack of data is crucial to maintaining operational continuity, avoiding interruptions, and preventing potential financial losses. Recent studies have investigated methods to mitigate the effects of data scarcity (Dzaferagic et al. 2021, Li et al. 2022, Cai et al. 2024). In the particular case of the transformer, ensuring the robustness of your monitoring and fault classification system even when not all phase currents are available is essential to prevent operational disruptions, enhance system reliability, and ensure continuous power delivery.

Several techniques based on digital signal processing, artificial intelligence, and fuzzy logic have recently been developed for fault detection and classification (Barbosa et al. 2011, Shah & Bhalja 2013, Bhatnagar & Yadav 2020, Costa 2014). Among of them, the wavelet transform has assumed a prominent role as it allows the proper analysis of non-stationary signals, such as fault currents (Costa 2014). For fault classification purposes, the stationary wavelet transform (SWT) is the most used version and has usually been used as a preprocessing routine for methods that employ logic fuzzy or artificial neural networks (Youssef 2004, Vanamadevi & Santhi 2013, Lala et al. 2017, Ananwattanaporn et al. 2017, Qiu 2022, Venkatasami & Latha 2016, Rajaraman et al. 2016)

Although the SWT has a good performance in detecting faults in power systems, it is quite affected by noise, and it is inaccurate in detecting overdamped fault-induced transients, such as in high-impedance faults, being so far unsuitable for practical applications (Costa 2014). However, the real-time boundary stationary wavelet transform (RT-BSWT), proposed in (Costa 2014), considers the border effects of a one-cycle sliding window in computing the wavelet signals. Consequently, it is sensitive to detecting fault-induced transients and overcurrents, even in noisy signals. Therefore, the RT-BSWT has been successfully used for fault detection (Costa 2014) and power system protection purposes (Costa et al. 2017, Leal et al. 2019, Medeiros, Costa, Silva, Muro, Júnior & Popov 2021). In protection area, for example, differential protection schemes based on high-frequency content of the currents using the RT-BSWT tool have demonstrated excellent results in power transformers (Medeiros & Costa 2018, Medeiros et al. 2016) and in transmission lines (Prado et al. 2024). Although RT-BSWT has been applied in some studies for fault classification, there is limited research in this area, and many approaches still rely on pre-established thresholds, which can restrict their adaptability. Artificial intelligence (AI) emerges as a promising alternative to overcome this limitation by eliminating the need for fixed threshold definitions. However, AI models require a specific and well-designed training process with representative and labeled datasets to achieve high accuracy and generalization capability. AI, through techniques such as machine learning, can identify complex patterns and classify faults accurately. Therefore, the combination of advanced wavelet transform and artificial intelligence has the potential for fast and accurate fault classification in power transformers even when the faults occur under data-missing conditions, which has not been yet proposed in the literature for the authors' best knowledge.

1.2 Motivation

Power transformers are essential components of the power transmission system. If a transformer failure occurs, this can lead to major interruptions in the power delivery. In addition, it is crucial to minimize power transformer failures due to their frequent maintenance requirements. However, many transformers are not properly maintained for various reasons, such as being located on the outskirts of big urban centers or in remote areas. By installing a wireless monitoring system, it is possible to monitor any transformer (Vinothkumar et al. 2021) effectively.

The significant rise in connectivity of physical objects to the Internet, known as the Internet of Things (IoT), is materializing at an unprecedented pace. In this context, sensors and actuators embedded in physical devices are interconnected through the network, enabling these objects to perform tasks intelligently and communicate with each other to visualize, capture information, process data, and carry out specific actions(Al-Fuqaha et al. 2015).

The robustness of fault monitoring and diagnosis systems is crucial to ensure success in several aspects. Sensors, which are as prone to failure as other components, can fail and present data loss, which may affect the operation of fault monitoring and diagnosis systems. Autonomous systems can face missing data for various reasons, including unexpected sensor failures. Conventional methods and model-based reasoning can reach incorrect conclusions if they infer from obsolete data instead of recognizing that the data is unavailable (Scarl 1989).

In this context, the development of robust artificial intelligence-based monitoring and fault diagnosis systems becomes fundamental. With the rapid advancement of the Internet of Things (IoT), the use of sensor networks and real-time data analysis demands robust systems. Implementing systems capable of fault classification, even in scenarios with missing or incomplete data, provides greater security in transformer diagnostics, reducing repair time and consequently minimizing the impact of interruptions in power supply. Such robustness is especially important in remote areas, where limited maintenance access due to logistical constraints makes continuous monitoring more challenging.

There is a growing demand for innovative methodologies to enhance operational reliability in electrical power systems. This underscores the importance of research and development for improving transformer monitoring technologies using artificial intelligence.

1.3 Objectives

- Design a method based on wavelet processing to classify internal faults in power transformers, involving phase-to-ground and phase-to-phase faults.
- Design a method that combines wavelet processing with different machine learning techniques for the classification of internal faults.
- Implement and evaluate different machine learning techniques, such as Decision Tree, Random Forest, and Logistic Regression, with the goal of identifying the most effective technique for fault classification.
- Improve fault detection accuracy by integrating wavelet processing with machine learning, surpassing conventional methods.
- Conduct a diverse set of fault simulations in power transformer windings, covering phase-to-ground and phase-to-phase faults, using detailed models in ATP software to generate a realistic and representative dataset for training the algorithms.
- Validate the robustness of the proposed algorithm in two distinct scenarios: 1) with all current data available, and 2) under missing data conditions, ensuring its reliability for real-world applications.

1.4 Contributions

The main contribution of this work is the development of a real-time classification method for phase-to-ground and phase-to-phase faults in power transformers, even in the presence of missing data. This contribution has the potential to significantly improve monitoring and fault diagnosis in power transformers, aligning with the principles of Industry 4.0 for modern industrial power systems.

Regarding the works developed to support this research, so far, the paper entitled ”Industry 4.0-Compliant Wavelet-Based Power Transformer Fault Classification Method During Data Missing Conditions” was published at the 2024 IEEE Power & Energy Society General Meeting (PESGM), in the best papers section. The authors of this paper are I. T. A. Dantas, R. P. Medeiros, F. B. Costa, F. J. P. Sousa, M. Wagner, M. Klar, and J. C. Aurich.

Additionally, a subsequent article, titled ”Power Transformer Fault Classification Using Industry 4.0-Compliant Machine Learning Under Data Missing Conditions”, has been submitted to IEEE Transactions on Power Delivery, further expanding the research by incorporating advanced machine learning techniques to enhance the robustness of fault classification in scenarios with missing data. The authors of this article are I. T. A. Dantas, R. P. Medeiros, F. B. Costa, O. V. Santana Jr., F. J. P. Sousa, M. Wagner, M. Klar, and J. C. Aurich.

1.5 Work Structure

This dissertation is organized into seven chapters:

- Chapter 1 - Introduction: Provides an overview of fault classification in power transformers and its relevance in the context of transformer protection.
- Chapter 2 - State of the Art: A comprehensive review of the literature on internal fault classification in transformers and data loss issues, highlighting existing contributions and potential research gaps.
- Chapter 3 - Mathematical Tools and Computational Implementation: Discussion on the Stationary Wavelet Transform (SWT) and machine learning algorithms used in the study, including their theoretical foundations and computational implementation.
- Chapter 4 - The Proposed Wavelet Processing-Based Method: Description of the wavelet processing approach for internal fault classification, including feature extraction criteria and the methodology applied for fault classification.
- Chapter 5 - The Proposed Machine Learning-Based Method: Introduction to the machine learning-based method for internal fault classification in transformers, detailing the techniques used and the models applied.
- Chapter 6 - Results: Comparative analysis of the performance of the proposed methods, presenting performance metrics and discussions on the effectiveness of the developed approaches.
- Chapter 7 - Conclusions and Future Work: Summary of the main contributions of the research and perspectives for future work aimed at improving internal fault classification in power transformers.

Chapter 2

State-of-the-Art

According to Abbasi (2022), numerous fault diagnostic techniques have been proposed in the literature for various components of transformers. These methods can be classified as knowledge-based, data-driven, or value-based. Data-driven methods model processes using the relationship between fault classes and data patterns. They reduce dimensionality based on multivariate statistics, converting high-dimensional data into specific lower dimensions. Transformers primarily utilize three groups of data-driven methods: quantitative methods of artificial intelligence, statistical analysis, and signal processing. Signal processing extracts fault characteristics in the frequency domain (such as frequency response scanning and low-voltage impulse tests) and in the time domain (such as wavelet transform and Fourier analysis).

Over the years, several methods have been developed to detect faults in transformers. Traditional methods include Dissolved Gas Analysis (DGA), which detects faults by examining dissolved gases in insulating oil; Frequency Response Analysis (FRA), which identifies mechanical and electrical changes in windings and core through responses at different frequencies and Computational Intelligence (CI) methods for fault classification. This chapter reviews the methods developed for fault classification in transformers.

2.1 Methods for Fault Classification

A frequency response analysis (FRA) to detect deformations in the windings and core of power transformers is performed in Shaban et al. (2019). The authors simulated different types of faults in a transformer modeled in MATLAB and divided the frequency band into four regions. They observed that the severity and type of fault affect statistical indicators such as mean, standard deviation, skewness, and kurtosis, which help identify the faults. The results showed that faults like turn-to-turn short circuits and axial displacement cause significant changes in all the parameters analyzed. It was concluded that, although FRA is a promising technique for monitoring transformers, its interpretation is still challenging. The reliance on visual analysis and expert opinion highlights the need for further research to standardize the process and define additional characteristics to improve interpretation accuracy, especially for minor faults.

The research by Venkatasami & Latha (2016) proposes a method of Extreme Learning Machine (ELM), a recent second-generation neural network algorithm, to classify faults in power transformers with high accuracy using the enthalpy of dissolved gas in transformer oil (dissolved gas analysis, DGA) as the input feature, with simulations conducted in MATLAB software. The ELM method is tested with two databases: DB1 and DB2. DB1 contains 151 records from IEC TC10, including faults such as partial discharge, low and high energy discharges, thermal faults, and normal conditions. DB2 contains 219 records from utilities in India, covering faults such as partial discharge, overheating, arcing, and electrical and thermal faults. The test efficiency for DB1 was 84.4% with gas concentration and 93.3% with enthalpy, while for DB2 it was 97.5% with gas concentration and 100% with enthalpy.

Rajaraman et al. (2016) present an algorithm for fault classification in three-phase transmission lines using wavelet multiresolution analysis (MRA). The researchers used MATLAB Simulink to simulate a 300 km transmission line, where different types of faults (L-G, L-L, L-L-G, and L-L-L), fault distances, and fault inception angles were analyzed. The study compared the effectiveness of 16 different wavelets in fault classification, which were based on the absolute differences and sums of the phase wavelet coefficients at two points along the line, near buses 1 and 2. If the sum of the coefficients of all phases is zero, there is no ground involvement in the fault (L-L or L-L-L). Otherwise, there is ground involvement (L-G or L-L-G). Specific faults are identified by the magnitude of the absolute differences. The db(4) wavelet filter was the most suitable for application in real-time digital relays, enabling quick and accurate detection of anomalies.

Faiz et al. (2016) propose a method for detecting, locating, and estimating the severity of interturn faults (ITFs) in power transformers. ITFs occur when a partial short circuit develops between turns of the same winding, causing electromagnetic asymmetries and potentially leading to more severe transformer damage. The study introduces an index based on negative sequence current, which is used to identify ITFs even under adverse conditions, such as unbalanced loads, asymmetric supply voltages, and external short circuits.

The proposed approach offers several advantages over traditional detection methods, such as DGA and differential relay protection, which may struggle to detect low-severity ITFs. Additionally, by leveraging negative sequence currents, this method enables online fault detection without requiring transformer shutdown. However, despite these advantages, the method has a significant limitation: it requires measurements of all three phase currents to compute the negative sequence current. If data from one of the phases is unavailable—whether due to sensor failure or communication issues—the calculation becomes infeasible, compromising fault detection.

In the research conducted by Qiu (2022) a method was proposed to identify faults in power transmission lines. The fuzzy logic was used in combination with wavelet transform to process the input data. Simulation tests showed that the method can distinguish among single-phase-to-ground faults, phase-to-phase faults, phase-to-phase-to-ground faults, and three-phase faults with specific fuzzy rules. Regarding accuracy, the proposed method presented greater performance when compared to techniques such as Support Vector Machines (SVM), Artificial Neural Networks (ANN), and Genetic Algorithm + Graph Neural Network (GA-GNN), the proposed method stands out for its ease of implementation and lower dependency on complex adjustments.

A method combining both discrete wavelet transform and fuzzy logic was presented in Ananwattanaporn et al. (2017) for internal fault classification in power transformers. The proposed algorithm used the wavelet coefficients of the differential currents in all phases during the first one-quarter of a cycle after the fault onset to determine the fault type. The method's effectiveness was assessed in a 50 MVA and 115/23 kV power transformer, and a global success rate of only 80% was achieved in the correct fault classification, which suggests this algorithm requires further improvement.

The study by Bera et al. (2018) investigates the identification of internal faults in Indirect Symmetrical Phase Shift Transformers (ISPSTs) using machine learning. To achieve this, an ISPST was modeled in the PSCAD/EMTDC software, where 40 different types of internal faults were simulated. These faults were classified by combining four factors: the fault location (exciting transformer or series transformer), the fault side (primary or secondary), the phases involved (A, B, or C individually or combined), and the type of short circuit (phase-to-ground, phase-to-phase, phase-phase-ground, or three-phase fault), totaling 10 types of short circuits for each of the four possible fault locations (EP, ES, SP, SS). In total, considering variations in tap position, direct and reverse phase shifts, load, and the percentage of the winding affected, 38,400 fault cases were generated, with 960 samples per fault type. Features from the phase signals were extracted in the time, frequency, and information theory domains and, after selecting the most relevant ones, were used to train five machine learning algorithms. The best performances were obtained by ensemble learning classifiers, with Extremely Random Trees (ERT) achieving 98.76% accuracy and Random Forest (RFC) reaching 97.54%, outperforming Multilayer Perceptron (96.13%), Logistic Regression (93.54%), and Support Vector Machines (92.60%). The study highlights the potential of machine learning in improving fault detection in ISPSTs but also emphasizes challenges such as the dependence on data quality and the lack of an investigation into the robustness of the extracted features in the presence of noise.

Patil et al. (2024) present a machine learning-based approach to predicting faults in power transformers using DGA. The study utilizes a dataset of 197 DGA samples from the IEEE database and tests different machine learning algorithms, including K-Nearest Neighbors (KNN), Decision Tree, Support Vector Machine (SVM), Random Forest, and Naive Bayes. The analyzed faults are classified into six types: partial discharge (PD), spark discharge (D1), arc discharge (D2), low-temperature thermal overheating (T1), medium-temperature thermal overheating (T2), and high-temperature thermal overheating (T3). Among the tested models, Naive Bayes achieved the best performance, with an accuracy of 69.23%, followed by Random Forest and KNN, both at 64.10%.

The results indicate that machine learning can be a great alternative for improving transformer fault diagnosis, overcoming the limitations of traditional DGA interpretation methods. However, the accuracy of the models is still moderate, suggesting the need for more robust datasets to enhance prediction reliability. Additionally, the dependency on data quality and the occurrence of misclassifications indicate that these models require further refinements before they can be widely adopted in the industry.

2.2 Challenges in Using Incomplete Data for Fault Analysis in Transformers

Ensuring the availability of all data is crucial for accurate fault classification in power transformers. This chapter explores the challenges of data scarcity and examines approaches to mitigate the effects of incomplete data, enabling reliable fault classification analysis in transformers.

The authors in Scarl (1989) investigated how automated diagnostic systems handle sensor failures, comparing traditional methods with an approach known as model-based reasoning. They found that conventional methods struggle to accurately identify sensor failures and manage situations where sensor data is missing. In contrast, model-based reasoning utilizes detailed system representations to test hypotheses about failures, effectively adapting to the absence of sensor data.

Monitoring systems often experience data loss due to sensor failures or environmental interferences. The work of Cai et al. (2024) addresses the challenge of filling missing data in power transformer monitoring systems under DC bias conditions, caused by stray currents from subway systems. The RCBiLSTM model is proposed, which combines Residual Networks (ResNet), Convolutional Neural Networks (CNN), and Bidirectional Long Short-Term Memory (BiLSTM) to enhance data filling accuracy. Experimental validation was conducted using data from a 220 kV transformer. The results demonstrated that the method outperformed other evaluated models in metrics such as Mean Squared Error (MSE), Mean Absolute Error (MAE), and R-squared (R2), especially in scenarios with high rates of missing data (10-40%).

Although the RCBiLSTM method demonstrates improvements in the accuracy of time series data filling in transformers, its viability in real operational environments may be compromised due to prolonged processing time. This limitation hinders the adoption of the model in transformer monitoring systems that require quick responses for real-time decision-making.

The work by Dzaferagic et al. (2021) addresses fault detection and classification in Industrial Internet of Things (IIoT) systems in the presence of missing sensor data, utilizing Generative Adversarial Networks (GANs) for data imputation. Based on the Tennessee Eastman Process dataset, the proposed methodology demonstrates that GANs can effectively generate accurate estimates for missing data, thereby enhancing the efficiency of fault detection and classification modules. The developed system integrates an autoencoder for anomaly detection and a Deep Neural Network

(DNN) for fault classification, resulting in a robust approach that mitigates the impact of persistent sensor failures.

The results indicate that this approach holds promise for maintaining the reliability of industrial monitoring systems even in scenarios with incomplete data. However, the research also acknowledges significant challenges: the imputation time increases as the number of observations and the amount of missing data grow. Furthermore, real-time application of such methods faces difficulties as the imputation process currently operates offline and requires enhancements for effective real-time implementation.

The study by Li et al. (2022) proposed improvements in dissolved gas analysis (DGA) for diagnosing faults in power transformers, addressing the common challenge of imbalanced sample datasets by generating synthetic data similar to real data to augment minority faults. They developed a generative adversarial network (GAN) model consisting of a generator and discriminator. Three fault diagnosis models - BPNN, SVM, and DBN - were established to assess diagnostic accuracy before and after data augmentation. The results showed an average increase of 30%, 25%, and 50% in recognition accuracy of minority samples for BPNN, SVM, and DBN models, respectively, after applying the GAN model. Compared to the SMOTE method, GAN demonstrated a 43.6% superior improvement in recognizing imbalanced DGA samples. The study highlights a drawback concerning majority samples: although GAN balanced the less frequent fault samples effectively, there were observations of a slight decrease in accuracy for majority samples.

2.3 Summary

This chapter provides a review of fault classification techniques in power transformers. Section 2.1 explores data-driven approaches such as DGA, FRA, and Computational Intelligence methods, which employ artificial intelligence and signal processing to detect and classify faults. Section 2.2 addresses the issue of data scarcity in fault analysis, discussing methodologies such as data imputation techniques using neural networks and GANs to enable classification even with missing data. In Tables 2.1 and 2.2, respectively, summarize Section 2.1 and Section 2.2 covered in this chapter.

Table 2.1: Summary of Literature Review of Methods for Fault Classification

Reference	**Methods**	**Element**
(Shaban et al. 2019)	FRA	Power Transformer
(Venkatasami & Latha 2016)	DGA + ELM	Power Transformer
(Rajaraman et al. 2016)	MRA	LT
(Qiu 2022)	Fuzzy	LT
(Ananwattanaporn et al. 2017)	MRA + Fuzzy	Power Transformer
(Bera et al. 2018)	Machine Learning (ERT, RFC, SVM, MLP, LR)	ISPST
(Patil et al. 2024)	DGA + Machine Learning (KNN, DT, SVM, RF, NB)	Power Transformer

Abbreviations Key: FRA - Frequency Response Analysis; DGA - Dissolved Gas Analysis; ELM - Extreme Learning Machine; MRA - Wavelet Multiresolution Analysis; LT - Transmission Line; Fuzzy - Fuzzy Logic; ERT - Extremely Random Trees; RFC - Random Forest Classifier; SVM - Support Vector Machine; MLP - Multilayer Perceptron; LR - Logistic Regression; KNN - K-Nearest Neighbors; DT - Decision Tree; NB - Naive Bayes.

Table 2.2: Summary of Literature Review of Challenges in Using Incomplete Data for Fault Analysis in Transformers

Reference	**Methods**	**Element**
(Scarl 1989)	Model-based Reasoning	Sensor Data Failure
(Cai et al. 2024)	RCBiLSTM	Power Transformer Data Imputation
(Dzaferagic et al. 2021)	GANs + Autoencoder + DNN	Fault Detection in IIoT Systems
(Li et al. 2022)	GANs	Imbalanced DGA Dataset Augmentation

Abbreviations Key: RCBiLSTM - Residual Convolutional BiLSTM; GANs - Generative Adversarial Networks; IIoT - Industrial Internet of Things; DGA - Dissolved Gas Analysis; DNN - Deep Neural Network.

Chapter 3

Mathematical Tools

This chapter presents the rationale behind the mathematical tools used in the development of this work: the wavelet transform, which constitutes the preprocessing step of the current signals to serve as inputs for the proposed fault classifier; and the machine learning algorithms, such as random forest, decision tree, and logistic regression, which perform the event classification.

3.1 The Wavelet Transform

The Wavelet Transform traces its origins to the Haar transform, proposed in 1910, but its mathematical formulation was only consolidated in the 1980s with significant contributions from Daubechies (1992) and Mallat (1989), the latter being responsible for the development of multiresolution analysis. The discrete wavelet transform (DWT) became widely used for signal decomposition into scaling and wavelet coefficients, enabling detailed analysis at different time-frequency resolutions. To enhance this technique, the stationary wavelet transform (SWT) was developed, providing a more stable signal representation and improved performance in transient detection.

Feature extraction from signals often requires advanced methods to facilitate data interpretation. In this context, machine learning has been widely applied to data classification, enabling the automatic detection of patterns from pre-processed datasets. Various algorithms can be used for this purpose, ranging from linear models, such as logistic regression, to more sophisticated techniques, such as decision trees and neural networks. To implement these approaches, libraries like Scikit-Learn provide a comprehensive set of tools for model training, validation, and optimization, along with performance evaluation metrics.

This chapter presents the mathematical and computational tools employed in this study. First, the fundamentals of the SWT and its variation, the real time boundary stationary wavelet transform (RT-BSWT), are discussed. Next, the fundamental concepts of machine learning tools used in this research, such as random forest, decision tree, and logistic regression, are discussed.

3.2 The Stationary Wavelet Transform (SWT)

The DWT and its variation, the SWT, are techniques used to decompose signals in the time-frequency domain. Both methods divide a discrete signal into scaling and wavelet coefficients, using specific filters: a low-pass filter (h_φ), which retains low-frequency components, and a high-pass filter (h_ψ), which preserves high-frequency details. While DWT performs decomposition with subsampling, SWT avoids the effect of subsampling by two in the signal samples (Costa 2014, Costa & Driesen 2012).

Figure 3.2 illustrates the SWT decomposition process of an arbitrary signal x into scaling and wavelet coefficients in TWDR. In this process, the (h_φ) filter extracts the scaling coefficients $s_1, s_2, ...$, which represent the low-frequency components of the signal. In parallel, the (h_ψ) filter generates the wavelet coefficients $w_1, w_2, ...$, responsible for capturing high-frequency variations. The decomposition occurs across multiple levels, and at each stage, the resulting scaling coefficients are used as input for a new filter bank, allowing for a more detailed signal analysis.

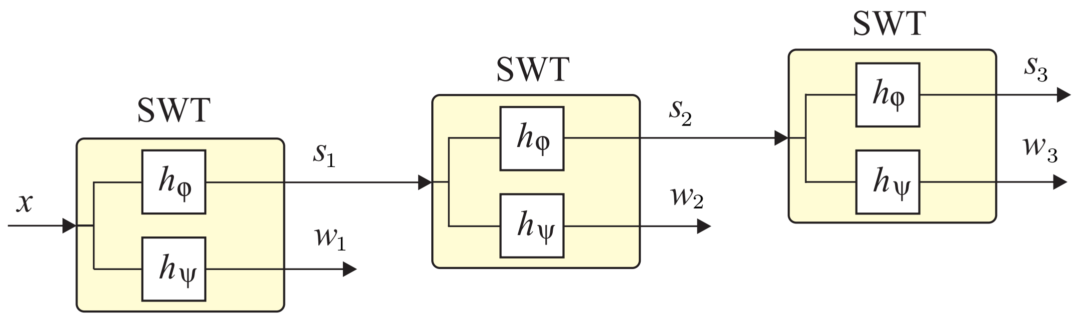

Figure 3.1: Block diagram depicting the first three decomposition levels of the SWT(Medeiros 2017).

The scaling coefficients s_j and the wavelet coefficients w_j in the SWT method, at a given resolution level j, are obtained through the convolution of the coefficients s_{j-1} from the previous level $j-1$ with the filters h_φ and h_ψ, respectively. This process is described by the following equations:

$$s_j(k) = \frac{1}{\sqrt{2}} \sum_{n=-\infty}^{\infty} h_\varphi(n-k) s_{j-1}(n), \tag{3.1}$$

$$w_j(k) = \frac{1}{\sqrt{2}} \sum_{n=-\infty}^{\infty} h_\psi(n-k) s_{j-1}(n), \tag{3.2}$$

in which s_0 corresponds to the original signal, $j \geq 1$, and the filters h_φ and h_ψ act as low-pass and high-pass filters, respectively.

The calculation of the scaling coefficients s_1 and wavelet coefficients w_1 in SWT, at the first decomposition level and for a series of k_t samples, can be represented in matrix form using the SWT pyramidal algorithm, as described by (Percival & Walden 2000):

$$s_1 = \mathbf{H}_\varphi x, \quad w_1 = \mathbf{H}_\psi x.$$

The matrices $\mathbf{H}_\varphi$ and $\mathbf{H}_\psi$ are square, with an order equal to k_t, and are constructed from circular shifts of the filter coefficients h_φ and h_ψ, respectively.

At the first resolution level of SWT, using the db(4) basis, the first three wavelet and scaling coefficients are influenced by the initial and final samples of the signal. Since these coefficients are computed using sliding windows, the signal's edges have fewer available samples for filter application, leading to the so-called border effect. This phenomenon can cause distortions, as the first computed values inside the window tend to have different amplitudes from the others. At the first decomposition level, this effect primarily affects the first $L - 1$ coefficients.

When analyzing signals using sliding windows, it is observed that the *wavelet* coefficients computed within these windows do not always exactly correspond to the coefficients of the sliding window itself. This discrepancy arises due to border effects, which can introduce inconsistencies in the calculations and impact the accuracy of the analysis.

For instance, when wavelet coefficients are computed using the pyramidal SWT algorithm in an offline manner, the last $\Delta k - L + 1$ coefficients of the sliding window match those obtained from the pyramidal algorithm. However, the first $L - 1$ coefficients differ, leading to inconsistencies between the sliding window of the signal and the wavelet analysis window (Costa 2014).

To address this issue and ensure proper alignment between the sliding wavelet window and the signal window, (Costa 2014) proposed including the first $L - 1$ border coefficients in the energy calculation, replacing the initial coefficients of the wavelet sliding window. This approach corrects for border effects and improves the accuracy of the computed coefficients.

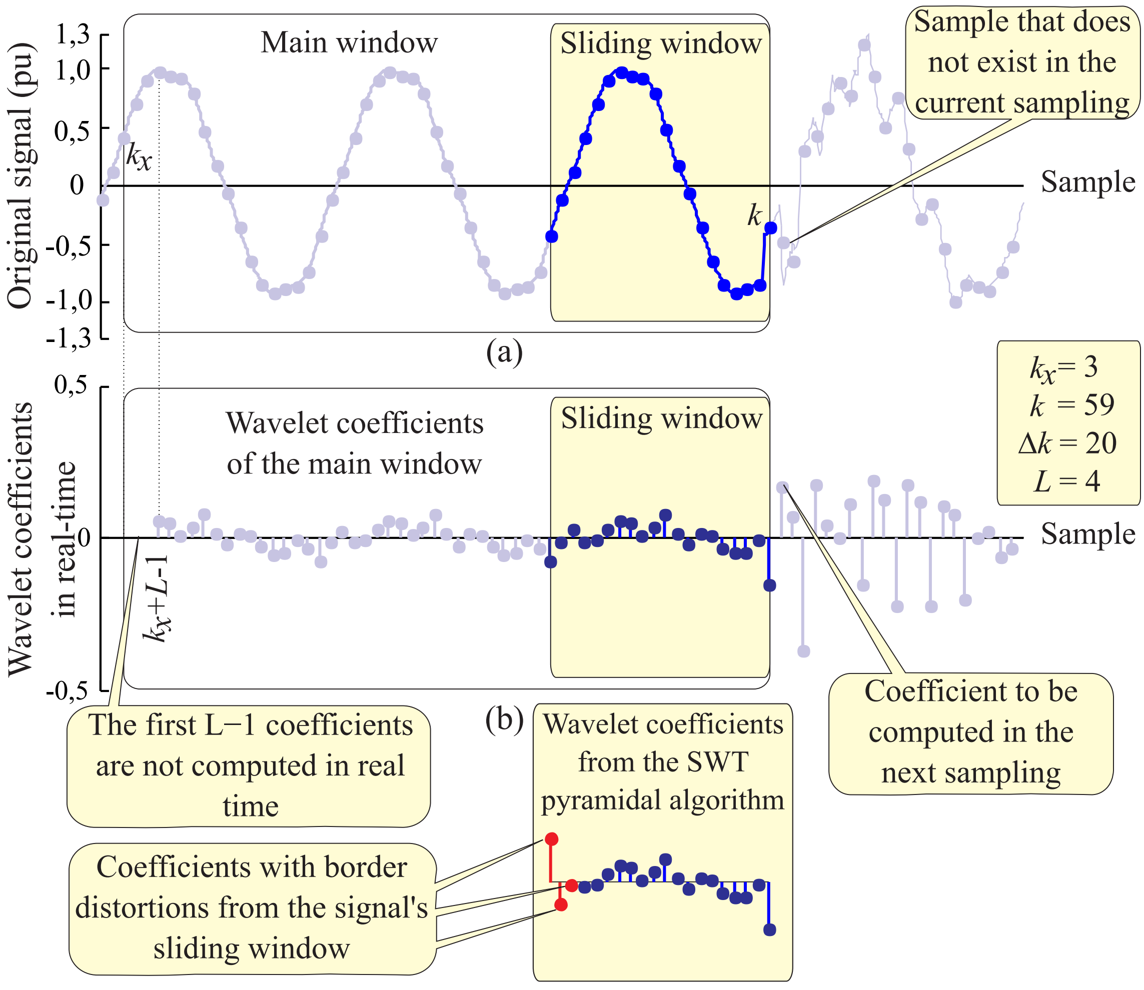

Figure 3.2: Calculation of wavelet coefficients in real-time: (a) Original signal; (b) Wavelet coefficients of the SWT pyramidal algorithm; (c) Wavelet coefficients of the SWT pyramidal algorithm related to the sliding window of the signal(Costa 2014).

In real-time applications, SWT scaling and wavelet coefficients are computed while accounting for border effects. Thus, for each sampling point k, the first and last $L-1$ samples of the sliding window are utilized. These coefficients are obtained through the inner between the samples of the signal and the scaling and wavelet filters, as described by (Costa 2014):

$$s(l,k) = \frac{1}{\sqrt{2}} \sum_{n=0}^{L-1} h_{\varphi}(n) i(k - L + n + 1 + l), \tag{3.3}$$

$$w(l,k) = \frac{1}{\sqrt{2}} \sum_{n=0}^{L-1} h_{\psi}(n) i(k - L + n + 1 + l), \tag{3.4}$$

where $k \geq \Delta k - 1$ represents the current sampling point, $0 \leq l < L$ denotes the index of the border coefficient, L is the filter length, and $\Delta k \geq L$ determines the sliding window size. Additionally, the term $i(k+m) = i(k-\Delta k+m)$, with $m \in N^*$, accounts for signal periodicity over Δk samples.

Although the sliding window spans Δk samples and shifts one sample at a time, the coefficients are computed only for the L samples with indices less than or equal to k. This ensures the algorithm's suitability for real-time applications.

In the SWT framework, the signal i is decomposed into L wavelet coefficients per sampling point. The coefficient $w(0, k)$, often written as $w(k)$, corresponds to the conventional *wavelet* coefficient in the real-time version of the pyramidal SWT algorithm (Mallat 1999). In contrast, coefficients $w(l, k)$, where $l \neq 0$, are referred to as border coefficients, as they are influenced by distortions caused by the boundaries of the sliding window.

Thus, the proposed method not only extracts conventional SWT wavelet coefficients but also introduces $L - 1$ additional coefficients that account for border effects (Costa 2014). This modification enhances signal analysis accuracy, making the results more reliable for real-time applications.

3.3 Energy of Wavelet Coefficients in SWT

According to Parseval's theorem (Percival & Walden 2000), the spectral energy ε of the current i, within a sliding window of size Δk, is decomposed into scale and wavelet coefficients. At the first decomposition level, the energy of the wavelet coefficients is given by (Medeiros et al. 2014):

$$\dot{\varepsilon}(k) = \sum_{n=k-\Delta k+1}^{k} w^2(n), \tag{3.5}$$

For $k \geq \Delta k + L - 2$, this energy considers all wavelet coefficients within the sliding window but does not account for the $L-1$ wavelet coefficients at the borders. Thus, the energy preservation condition is not satisfied, as:

$$\varepsilon \neq \dot{\varepsilon} + \ddot{\varepsilon}.$$

To ensure energy preservation, the energy of the wavelet coefficients (ε^w) in a sliding window is defined according to (Costa & Driesen 2012):

$$\varepsilon^w(k) = \varepsilon^{wa}(k) + \varepsilon^{wb}(k), \tag{3.6}$$

For $k > \Delta k - 1$, the energy ε^{wa} corresponds to the $L - 1$ wavelet coefficients at the borders of the sliding window and is defined as:

$$\varepsilon^{wa}(k) = \sum_{l=1}^{L-1} w^2(l, k), \tag{3.7}$$

whereas, the energy ε^{wb} is computed using the others wavelet coefficients of the sliding window, named as conventional wavelet coefficients, as follows:

$$\varepsilon^{wb}(k) = \sum_{n=k-\Delta k+L}^{k} w^2(n). \tag{3.8}$$

3.4 Machine Learning Concepts

This section presents the main concepts of machine learning, focusing on three widely used algorithms for classification: Logistic Regression (LR), Decision Tree (DT), and Random Forest (RF).

With technological advancements and the increasing availability of data, machine learning has become an alternative to traditional programming methods. Instead of following fixed rules defined by a programmer, models adjust their parameters automatically based on examples, allowing for greater flexibility in problem-solving (Mitchell 1999).

Machine learning can be divided into three main categories (Russell & Norvig 2016):

Supervised learning: The model is trained using labeled data, where each input is associated with a known output. This enables the algorithm to learn patterns and make predictions on new data. It is widely used in classification and regression tasks.

Unsupervised learning: The model does not receive labels and must identify patterns and groupings within the data on its own. This approach is commonly used in customer segmentation and anomaly detection.

Reinforcement learning: The algorithm learns through trial and error, interacting with an environment and receiving rewards or penalties based on its decisions. It is applied in fields such as robot control, gaming, and process optimization.

In the context of this study, the objective is to classify samples based on a previously prepared dataset. To achieve this, the supervised learning approach will be employed, where each instance is associated with a class, allowing the model to learn patterns and make predictions based on these examples.

3.4.1 Logistic Regression

Logistic regression is a statistical model used to analyze dichotomous variables, meaning those that can take only two values. Cox (1970) and Walker Duncan (1967) proposed this model based on the idea that the logarithm of the odds ratio between two categories has a linear relationship with the explanatory variables.

In this context, the dependent variable takes binary values to indicate the presence or absence of a characteristic. Logistic regression, which is part of the Generalized Linear Models (McCullagh Nelder, 1989), is described by the following equation:

$$\ln\left(\frac{\pi(x)}{1-\pi(x)}\right) = \beta x \tag{3.9}$$

In which $\pi(x)$ represents the probability of the event occurring, and βx corresponds to a linear combination of the explanatory variables weighted by their coefficients. To ensure that the model produces values between 0 and 1, the sigmoid function is applied:

$$\pi(x) = \frac{\exp(\beta' x)}{1+\exp(\beta' x)} \tag{3.10}$$

Logistic regression estimates the probability of an event occurring, that is:

$$\pi(x) = P(Y = 1 \mid x) \tag{3.11}$$

This means that, given a set of explanatory variables x, the model provides the probability that the dependent variable Y takes the value 1. This probability can be rewritten in the form of the sigmoid function:

$$\pi(x) = \frac{\exp(\beta' x)}{1+\exp(\beta' x)} \tag{3.12}$$

This equation indicates that as the value of the linear combination $\beta' x$ increases, the probability $\pi(x)$ approaches 1; if it decreases, the probability approaches 0.

The model can be more generally expressed as:

$$Y = \pi(x) + \varepsilon \tag{3.13}$$

The error ε in the model is random and depends on the predicted probability $\pi(x)$. Since Y can only take values 0 or 1, the error also has two possible values: when $Y = 1$, the error is given by $\varepsilon = 1 - \pi(x)$ with probability $\pi(x)$, and when $Y = 0$, it is $\varepsilon = -\pi(x)$ with probability $1 - \pi(x)$.

Thus, ε has a mean of 0 and variance $\pi(x)(1 - \pi(x))$, and since Y follows a Bernoulli distribution, the probability of success is given by $\pi(x)$. The model coefficients β are estimated using the Maximum Likelihood method, which adjusts the parameters to maximize the probability of the observed data.

To classify a new observation x_c, a decision threshold is typically set at 0.5. If $\pi(x_c) \geq 0.5$ the observation is classified as 1, indicating the occurrence of the event; otherwise, it is classified as 0, indicating the absence of the event. This threshold can be adjusted depending on the application needs. (Hosmer et al. 1989)

3.4.2 Decision Trees

Decision trees (DT) are methods used for classification and regression, organizing decisions in a hierarchical structure. They start at the root node, where data is analyzed, pass through internal nodes, which apply tests on attributes, and end at leaf nodes, where the final classification is determined. (Breiman 2001)

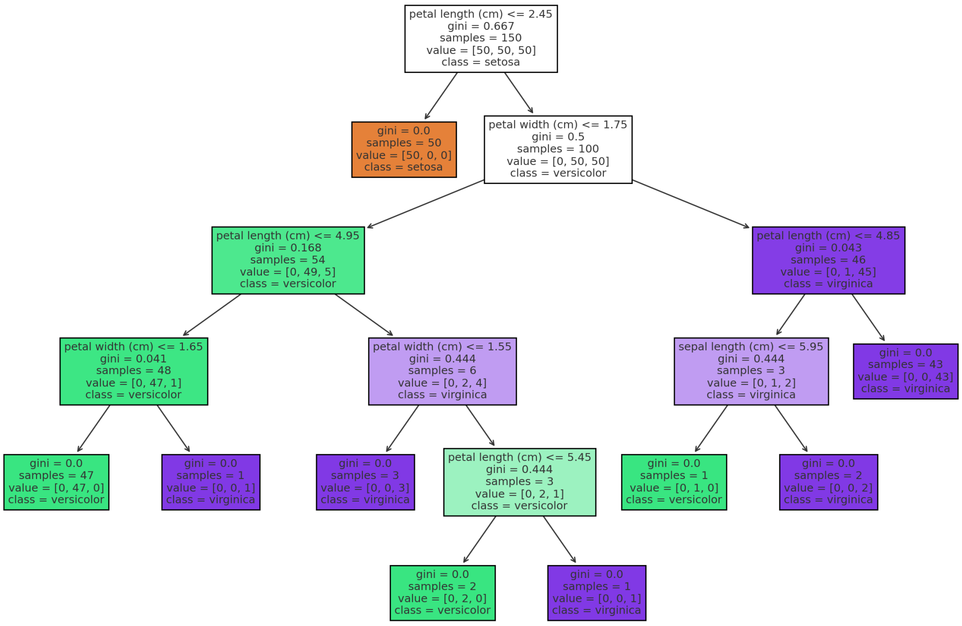

Figure 3.3: Decision tree of the Iris dataset (Scikit-Learn 2025).

The tree is built by recursively splitting the data based on an impurity metric, such as entropy or the Gini index, until the subsets become homogeneous. This process generates if-then rules, making the model interpretable and enabling direct decision-making. Due to their efficiency and clarity, decision trees are widely applied in machine learning. (Quinlan 1986)

The decision tree shown in the Figure 3.3 was generated from an official example in Scikit-Learn, available in the library's documentation. The example uses the Iris dataset, which contains information on three species of flowers (Iris setosa, Iris versicolor, and Iris virginica), classified based on four features: petal length, petal width, sepal length, and sepal width. The decision tree model learns patterns from this data and makes predictions based on hierarchical rules.

The tree starts with a decision based on petal length. If it is less than or equal to 2.45 cm, the flower is classified as Iris setosa. If it is greater, further splits occur, analyzing other features until the correct species is determined. Each node represents a question about the flower's attributes, and the final nodes indicate the classification.

3.4.3 Random Forest

Random Forests are supervised learning algorithms that combine multiple decision trees to improve accuracy and prevent overfitting. Overfitting occurs when a single tree learns specific patterns from the training data, reducing its ability to generalize. To mitigate this, Random Forest constructs multiple trees using randomly sampled subsets of the data and selects a random subset of features for each split. This algorithm belongs to the ensemble methods, which combine several simple models to achieve a stronger one. Each tree makes independent decisions, and the final prediction is determined by majority vote in classification tasks or by averaging individual predictions in regression, making the model more accurate and stable (Breiman 2001).

Each tree in the Random Forest is trained using a bootstrap sampling process, where a subset of the data is randomly selected with replacement. Some samples are left out of training and used to estimate the model's error without needing a separate validation set—these are called Out-of-Bag (OOB) samples. Additionally, Random Forest evaluates the importance of each variable by testing how accuracy changes when the values of a variable are randomly shuffled. If accuracy drops significantly, the variable is considered highly influential. The model's performance depends on the diversity among trees and the individual strength of each tree: if they are too similar, the error increases; if they are well-trained and diverse, the model becomes more accurate and reliable.

Figure 3.4 illustrates this concept with a Random Forest trained on the Iris dataset, one of the most widely used in machine learning. This dataset contains information about three species of flowers (setosa, versicolor, and virginica) and their four main characteristics: sepal length, sepal width, petal length, and petal width. Each of the five trees represented in the figure was trained on slightly different subsets of the data and randomly selected attributes for splitting nodes, ensuring varied decision-making and reducing the risk of overfitting.

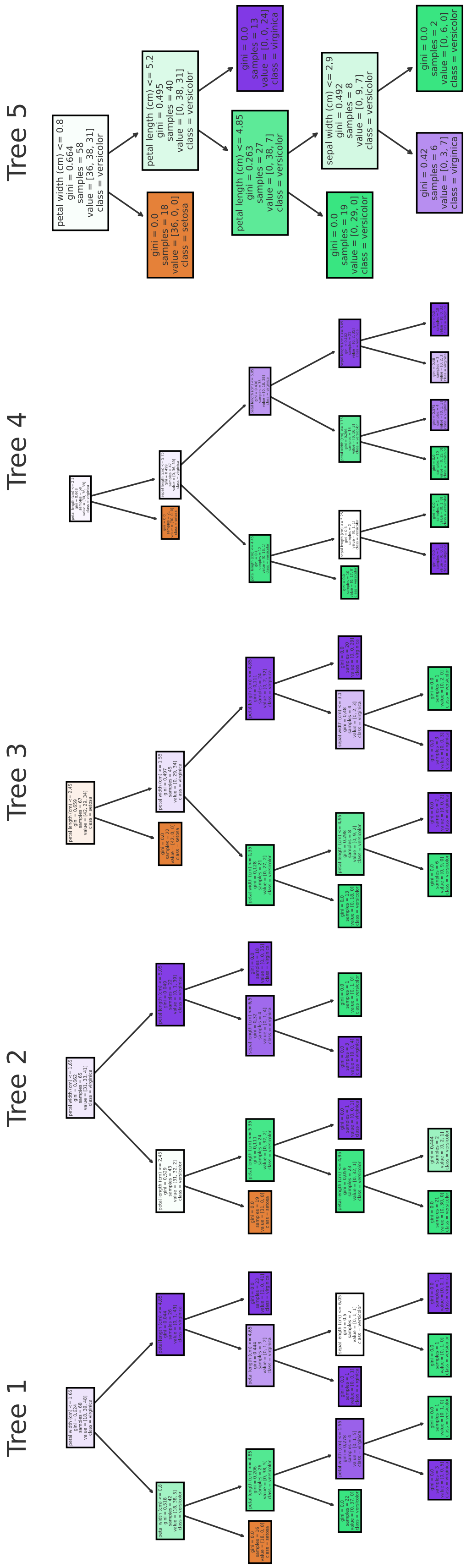

Figure 3.4: Random Forest applied to the Iris dataset.

3.5 Summary

This chapter presented the main mathematical tools used in this study, with an emphasis on the Wavelet Transform and machine learning algorithms applied to event classification.

Initially, the fundamentals of the Wavelet Transform were discussed, focusing on the SWT. The chapter covered the decomposition of the signal into scaling and wavelet coefficients. The border effects and strategies for their mitigation were also analyzed. Furthermore, the calculation of wavelet coefficient energy was explored.

Next, the machine learning algorithms employed for classification were introduced, specifically logistic regression, decision trees, and random forests.

Chapter 4

The Proposed Wavelet Processing-Based Method

In recent years, wavelet-based processing methods have emerged as powerful signal analysis and processing tools in various applications. In the special case of faults in power transformers, these transient phenomena usually present high-frequency and short-duration components, whose analysis can be properly performed through the wavelet transform in several resolution levels. In the literature, the wavelet transform gained notoriety from the contributions presented by Daubechies (1992), who conceptualized the discrete wavelet transform (DWT), and Mallat (1989), who proposed the concept of multiresolution analysis. Currently, a variant of the DWT, the SWT (stationary wavelet transform), has been widely used since it presents a better performance in real-time detection of transients in power systems than the previous version (Costa 2014).

This chapter presents a detailed approach to the proposed wavelet processing-based method, utilizing the Real-Time Boundary Stationary Wavelet Transform (RT-BSWT) to compute the energies of the scaling and wavelet coefficients of the currents measured by the current transformers installed on both sides of the power transformer. This technique enables the classification of internal faults in the primary and secondary windings of a power transformer using fixed rules.

4.1 The Proposed Real-Time Power Transformer Fault Classification Method

Fig. 4.1 depicts the flowchart of the proposed power transformer fault classification method, which is executed at each sampling time (T_s=$1/f_s$), where f_s corresponds to the sampling frequency. The proposed fault classifier was designed for currents sampled at f_s=15.36 kHz, a typical sampling frequency used in industrial fault recorders (Medeiros, Costa, Silva, Muro, Júnior & Popov 2021).

The wavelet filter chosen is the Daubechies with four coefficients (db(4)) due to its high efficiency in capturing significant signal features present in the faulted currents in this sampling frequency (Costa 2014), where the digital acquisition of three-

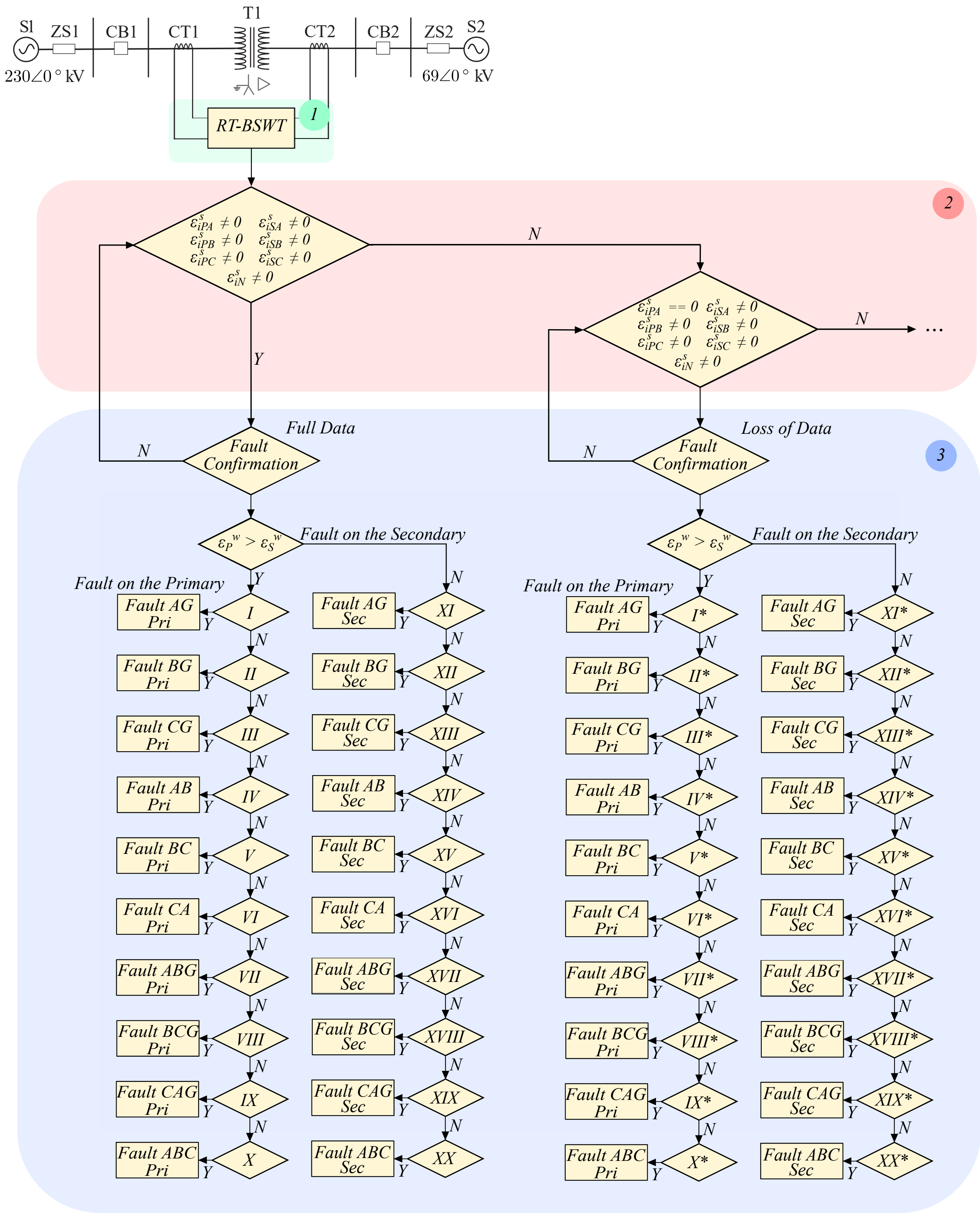

Figure 4.1: Flowchart of the proposed real-time power transformer fault classification method.

phase and neutral currents from current transformers (CTs) is performed through anti-aliasing filters and AD converters to get the following time-discrete currents: i_P={i_{PA}, i_{PB}, i_{PC}}, i_N, and i_S={i_{SA}, i_{SB}, i_{SC}}. The sub-indices P, S, and N refer to the primary transformer winding, secondary transformer winding, and the neutral point, respectively, whereas A, B, and C refer to phases A, B, and C, respectively. Details of each sub-routine presented in the flowchart are addressed in the remainder of this section.

4.1.1 Wavelet Coefficients Computation

The proposed method applies the real-time boundary stationary wavelet transform (RT-BSWT) in the currents measured by CTs installed in the primary and secondary side windings of the power transformer. The wavelet coefficients of the RT-BSWT are given by (Costa 2014):

$$w_i(l,k) = \frac{1}{\sqrt{2}} \sum_{n=0}^{L-1} h_\psi(n) \mathring{i}(k - L + n + 1 + l), \tag{4.1}$$

where $k \geq \Delta k - 1$ is the current time index associated with the current sample time k/f_s; $0 \leq l < L$ is the border index; h_φ and h_ψ are low-pass scaling and high-pass wavelet filters, respectively; L is the filters length; $\Delta k \geq L$ is the sliding window length; $\mathring{i}(k+m) = i(k - \Delta k + m)$ with $m \in N^*$, which is a periodized current in Δk samples; $w_i = \{w_{i_{PA}}, w_{i_{PB}}, w_{i_{PC}}, w_{i_N}, w_{i_{SA}}, w_{i_{SB}}, w_{i_{SC}}\}$ are related to $i = \{i_{PA}$, i_{PB}, i_{PC}, i_N, i_{SA}, i_{SB}, and $i_{SC}\}$.

4.1.2 Wavelet Coefficients Energies Computation (Block 1)

Based on Medeiros, Costa, Silva, Muro, Júnior & Popov (2021), the proposed method uses the RT-BSWT to compute the scaling and wavelet coefficient energies of the CT currents as follows:

$$\mathcal{E}_i^s(k) = \sum_{l=1}^{L-1} [s_i(l,k)]^2 + \sum_{n=k-\Delta k+L}^{k} [s_i(0,n)]^2, \tag{4.2}$$

$$\mathcal{E}_i^w(k) = \sum_{l=1}^{L-1} [w_i(l,k)]^2 + \sum_{n=k-\Delta k+L}^{k} [w_i(0,n)]^2, \tag{4.3}$$

where $\mathcal{E}_i^s = \{\mathcal{E}_{i_{PA}}^s, \mathcal{E}_{i_{PB}}^s, \mathcal{E}_{i_{PC}}^s, \mathcal{E}_{i_N}^s, \mathcal{E}_{i_{SA}}^s, \mathcal{E}_{i_{SB}}^s, \mathcal{E}_{i_{SC}}^s,\}$ and $\mathcal{E}_i^w = \{\mathcal{E}_{i_{PA}}^w, \mathcal{E}_{i_{PB}}^w, \mathcal{E}_{i_{PC}}^w$, $\mathcal{E}_{i_N}^w, \mathcal{E}_{i_{SA}}^w, \mathcal{E}_{i_{SB}}^w, \mathcal{E}_{i_{SC}}^w,\}$ are, respectively, the scaling coefficient energies and wavelet coefficient energies of all CT currents.

4.1.3 The Data Missing Condition Identification (Block 2)

According to Medeiros, Costa, Silva, Muro, Júnior & Popov (2021), the level of the scaling coefficient energies of all primary-side and secondary-side currents feeding the data acquisition system can be used to monitor the transformer operation during steady-state, i.e., before any fault event, as depicted in Fig. 4.1. For instance, when all phase currents (all data) are present, $\mathcal{E}^s_{i_P} > 0$ and $\mathcal{E}^s_{i_S} > 0$. Conversely, when there is a loss of any data due to a communication failure in the data acquisition system, either $\mathcal{E}^s_{i_P} = 0$ or $\mathcal{E}^s_{i_S} = 0$. For instance, in the flowchart depicted in Fig. 4.1, the lack of data was considered by admitting $i_{PA} = 0$, resulting in $\mathcal{E}^s_{i_{PA}} = 0$.

4.1.4 The Fault Classification (Block 3)

The proposed fault classification method is enabled when:

$$\mathcal{E}^w_i(k) > \epsilon, \tag{4.4}$$

in which $\epsilon = \{\epsilon_1 \text{ and } \varepsilon_2,\}$ are thresholds related to the faulted and healthy transformer windings, respectively, being statistically defined as follows:

$$\epsilon = \frac{N}{k_2 - k_1 + 1} \sum_{n=k_1}^{k_2} \mathcal{E}^w_i(n), \tag{4.5}$$

where $[k_1/f_s \; k_2/f_s]$ is a previous steady-state time range; $N = 50,000$ and $N = 30$ are empirical constants used to define ϵ_1 and ϵ_2, respectively.

When a fault is confirmed, the classifier initially identifies the transformer winding in which the fault occurred. A primary-side fault is identified when the sum of the wavelet coefficient energies of all primary-side currents is greater than the sum of the wavelet coefficient energies of all secondary-side currents. Otherwise, it is identified as a secondary-side fault, as depicted in Fig. 4.1.

Table 4.1 presents the classification based on the energy thresholds of the wavelet coefficients ($\mathcal{E}^w_i$) for different types of faults in a transformer. Faults can occur on both the primary and secondary sides of the transformer, identified by the energies of the wavelet coefficients of the currents on the primary side and the energies of the wavelet coefficients of the currents on the secondary side. The colored cells in the table indicate whether the energy of the wavelet coefficient exceeds the specified threshold (ϵ_1 or ϵ_2), which are the fault and no-fault thresholds, respectively.

Table 4.1: Threshold sorting

	$\mathcal{E}^w_{iPA}$	$\mathcal{E}^w_{iPB}$	$\mathcal{E}^w_{iPC}$	$\mathcal{E}^w_{iN}$	$\mathcal{E}^w_{iSA}$	$\mathcal{E}^w_{iSB}$	$\mathcal{E}^w_{iSC}$
AG_P (I)	ϵ_1			ϵ_1	ϵ_2	ϵ_2	
BG_P (II)		ϵ_1		ϵ_1		ϵ_2	ϵ_2
CG_P (III)			ϵ_1	ϵ_1	ϵ_2		ϵ_2
AB_P (IV)	ϵ_1	ϵ_1			ϵ_2	ϵ_2	ϵ_2
BC_P (V)		ϵ_1	ϵ_1		ϵ_2	ϵ_2	ϵ_2
CA_P (VI)	ϵ_1		ϵ_1		ϵ_2	ϵ_2	ϵ_2
ABG_P (VII)	ϵ_1	ϵ_1		ϵ_1	ϵ_2	ϵ_2	ϵ_2
BCG_P (VIII)		ϵ_1	ϵ_1	ϵ_1	ϵ_2	ϵ_2	ϵ_2
CAG_P (IX)	ϵ_1		ϵ_1	ϵ_1	ϵ_2	ϵ_2	ϵ_2
ABC_P (X)	ϵ_1	ϵ_1	ϵ_1		ϵ_2	ϵ_2	ϵ_2
AG_S (XI)	ϵ_2		ϵ_2		ϵ_1		
BG_S (XII)	ϵ_2	ϵ_2				ϵ_1	
CG_S (XIII)		ϵ_2	ϵ_2				ϵ_1
AB_S (XIV)	ϵ_2	ϵ_2	ϵ_2		ϵ_1	ϵ_1	
BC_S (XV)	ϵ_2	ϵ_2	ϵ_2			ϵ_1	ϵ_1
CA_S (XVI)	ϵ_2	ϵ_2	ϵ_2		ϵ_1		ϵ_1
ABG_S (XVII)	ϵ_2	ϵ_2	ϵ_2		ϵ_1	ϵ_1	
BCG_S (XVIII)	ϵ_2	ϵ_2	ϵ_2			ϵ_1	ϵ_1
CAG_S (XIX)	ϵ_2	ϵ_2	ϵ_2		ϵ_1		ϵ_1
ABC_S (XX)	ϵ_2	ϵ_2	ϵ_2		ϵ_1	ϵ_1	ϵ_1

$\mathcal{E}^w_i > \epsilon_1$, $\mathcal{E}^w_i > \epsilon_2$

The columns of the table represent the energies of the wavelet coefficients of the currents in phases A, B, C, and the neutral on the primary side of the transformer ($\mathcal{E}^w_{iPA}$, $\mathcal{E}^w_{iPB}$, $\mathcal{E}^w_{iPC}$, and $\mathcal{E}^w_{iN}$) and the energies of the wavelet coefficients of the currents in phases A, B, and C on the secondary side of the transformer ($\mathcal{E}^w_{iSA}$, $\mathcal{E}^w_{iSB}$ and $\mathcal{E}^w_{iSC}$).

The rows of the table correspond to different types of faults that can occur in the transformer, including single-phase faults (AG, BG, CG), two-phase faults (AB, BC, CA), two-phase faults with ground (ABG, BCG, CAG), and three-phase faults (ABC). Each type of fault is indicated with an index (I, II, III, etc.) and specified whether it occurs on the primary side (P) or the secondary side (S) of the transformer.

For example, for an AG fault on the primary side of the transformer (AG_P), the table indicates that the energies of the wavelet coefficients of the currents in phase A ($\mathcal{E}^w_{iPA}$) and the neutral ($\mathcal{E}^w_{iN}$) on the primary side exceed the threshold ϵ_1, while on the secondary side, the energies of the currents in phases A ($\mathcal{E}^w_{iSA}$) and B ($\mathcal{E}^w_{iSB}$) exceed the threshold ϵ_2. The colored cells help to quickly visualize this information: red cells indicate that the energy exceeds ϵ_1, while light blue cells indicate that the energy exceeds ϵ_2.

The fault threshold ϵ_1 is applied on the side where the fault occurs, while the threshold ϵ_2 is used on the side where the fault does not occur. If there is a fault on the secondary side of the transformer, the threshold ϵ_1 will be applied on that side, while the threshold ϵ_2 will be used on the primary side. This ensures accurate detection and classification of faults on both sides of the transformer.

Figures 4.2, 4.3, 4.4, 4.5, 4.6, 4.7, 4.8, and 4.9, illustrate, respectively, the primary-side and secondary-side currents as well as their respective wavelet energies for: a phase-to-ground fault, a phase-to-phase fault, a phase-to-phase-to-ground fault, and a three-phase fault, all of them occurring on both sides of the transformer. For the sake of simplification, only currents and wavelet energies of the AG, AB, ABG, and ABC faults on both sides of the transformer were depicted, since the other faults involving B and C phases present similar patterns as shown in Table 4.1. The selected waveforms are presented in the following order: currents on the transformer primary-side (i_{PA}, i_{PB}, and i_{PC}); wavelet coefficient energies of the transformer primary-side currents ($\mathcal{E}^w_{i_{PA}}$, $\mathcal{E}^w_{i_{PB}}$, $\mathcal{E}^w_{i_{PC}}$, and $\mathcal{E}^w_{i_N}$); currents on the transformer secondary-side (i_{SA}, i_{SB}, and i_{SC}); and wavelet coefficient energies of the transformer secondary-side currents ($\mathcal{E}^w_{i_{SA}}$, $\mathcal{E}^w_{i_{SB}}$, and $\mathcal{E}^w_{i_{SC}}$).

According to Figure 4.2, for example, when an AG fault occurs on the transformer's primary side, the wavelet coefficient energies of just phase A and neutral currents are expected to be sensitized. Due to the transformer magnetic coupling, the wavelet coefficient energies of secondary-side phases A and B currents are also expected to be sensitized, such as depicted in Figure 4.2(d). Thereby, it is expected the classifier enables the output "I" for an AG fault on the primary side of the transformer by the following conditions (Table 4.1): $\mathcal{E}^w_{i_{PA}} > \epsilon_1$, $\mathcal{E}^w_{i_{PB}} < \epsilon_1$, $\mathcal{E}^w_{i_{PC}} < \epsilon_1$, $\mathcal{E}^w_{i_N} > \epsilon_1$, $\mathcal{E}^w_{i_{SA}} > \epsilon_2$, $\mathcal{E}^w_{i_{SB}} > \epsilon_2$, and $\mathcal{E}^w_{i_{SC}} < \epsilon_2$. The same analysis procedure is performed to classify all the another faults in accordance with Table 4.1.

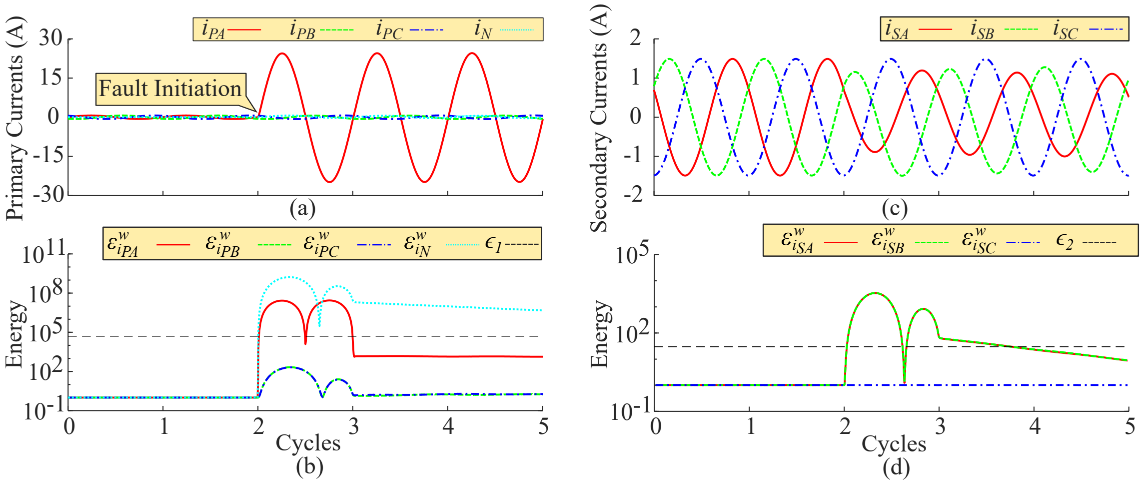

Figure 4.2: Single phase-to-ground fault in the primary side (AG_P fault): (a) Primary-side currents; (b) Energy of the primary-side currents; (c) Secondary-side currents; (d) Energy of the secondary-side currents.

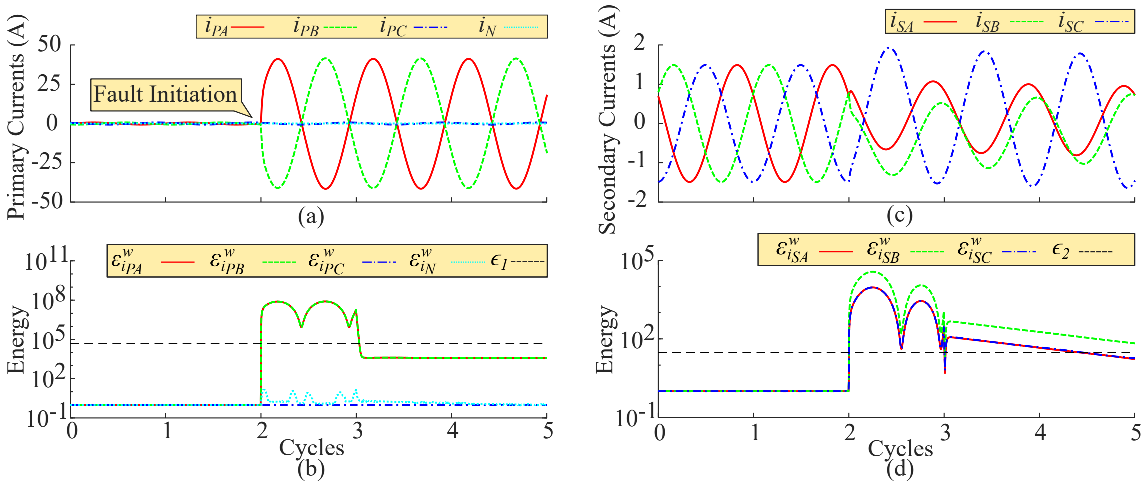

Figure 4.3: Phase-to-phase fault in the primary side (AB_P fault): (a) Primary-side currents; (b) Energy of the primary-side currents; (c) Secondary-side currents; (d) Energy of the secondary-side currents.

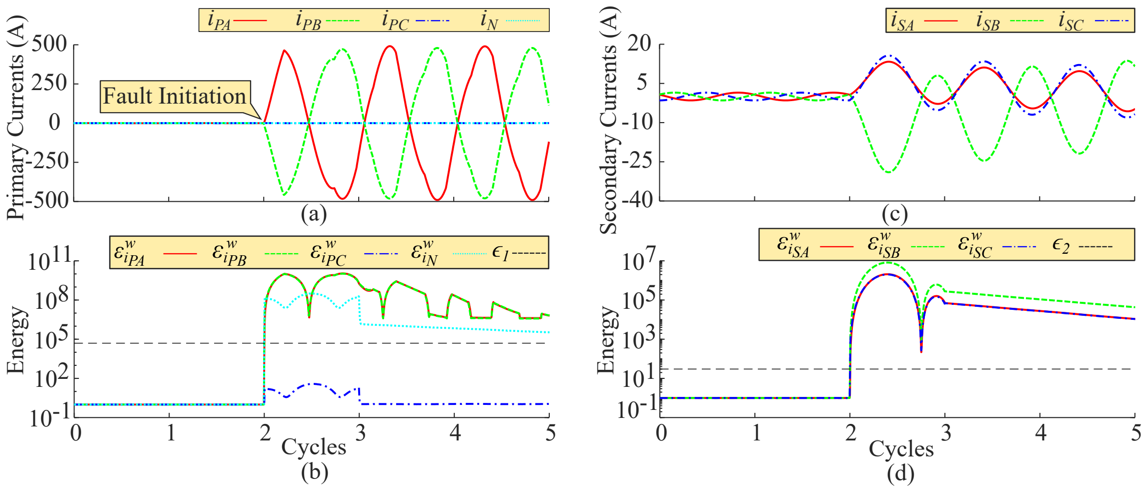

Figure 4.4: Phase-to-phase-to-ground fault in the primary side (ABG_P fault): (a) Primary-side currents; (b) Energy of the primary-side currents; (c) Secondary-side currents; (d) Energy of the secondary-side currents.

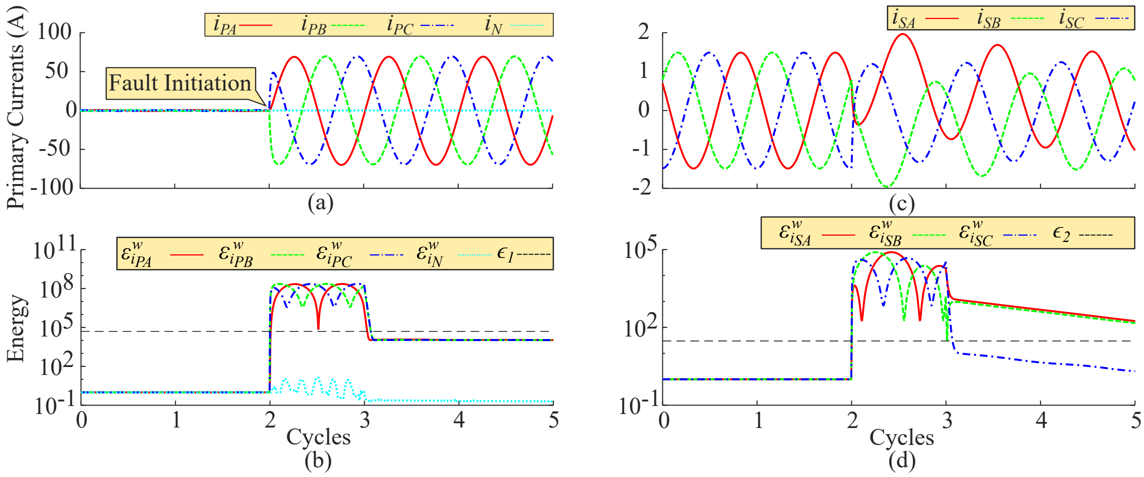

Figure 4.5: Three-phase fault in the primary side (ABC_P fault): (a) Primary-side currents; (b) Energy of the primary-side currents; (c) Secondary-side currents; (d) Energy of the secondary-side currents.

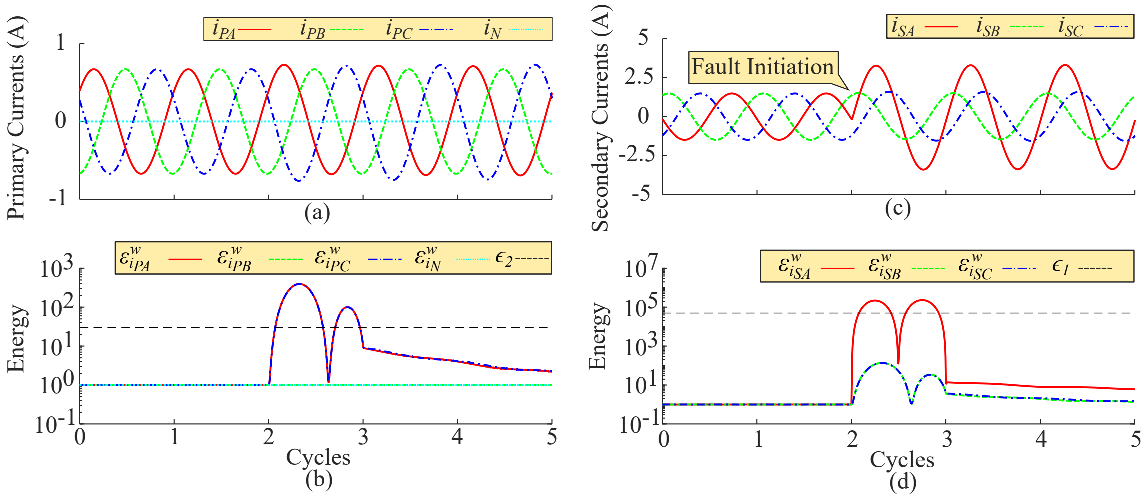

Figure 4.6: Single phase-to-ground fault in the secondary side (AG_S fault): (a) Primary-side currents; (b) Energy of the primary-side currents; (c) Secondary-side currents; (d) Energy of the secondary-side currents.

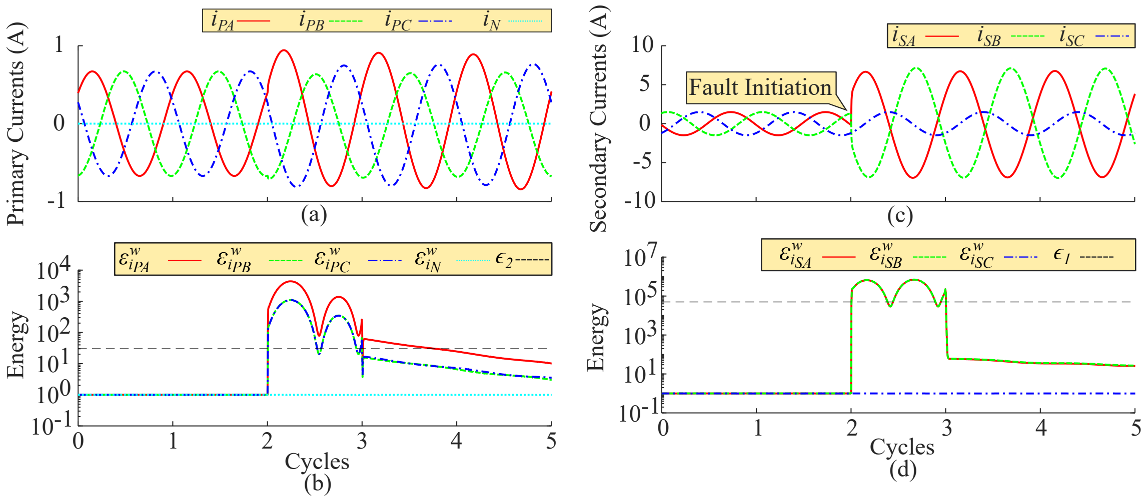

Figure 4.7: Phase-to-phase fault in the secondary side (AB_S fault): (a) Primary-side currents; (b) Energy of the primary-side currents; (c) Secondary-side currents; (d) Energy of the secondary-side currents.

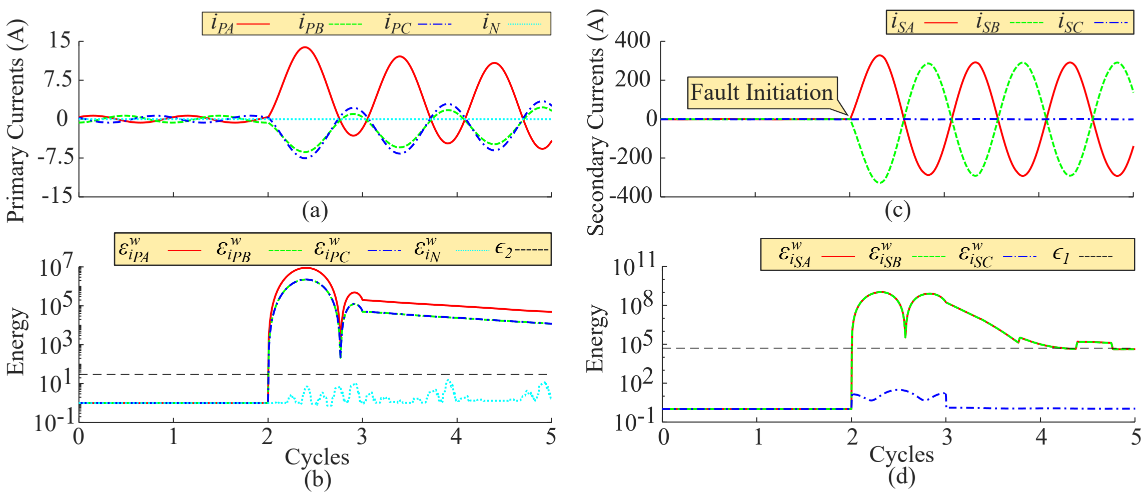

Figure 4.8: Phase-to-phase-to-ground fault in the secondary side (ABG_S fault): (a) Primary-side currents; (b) Energy of the primary-side currents; (c) Secondary-side currents; (d) Energy of the secondary-side currents.

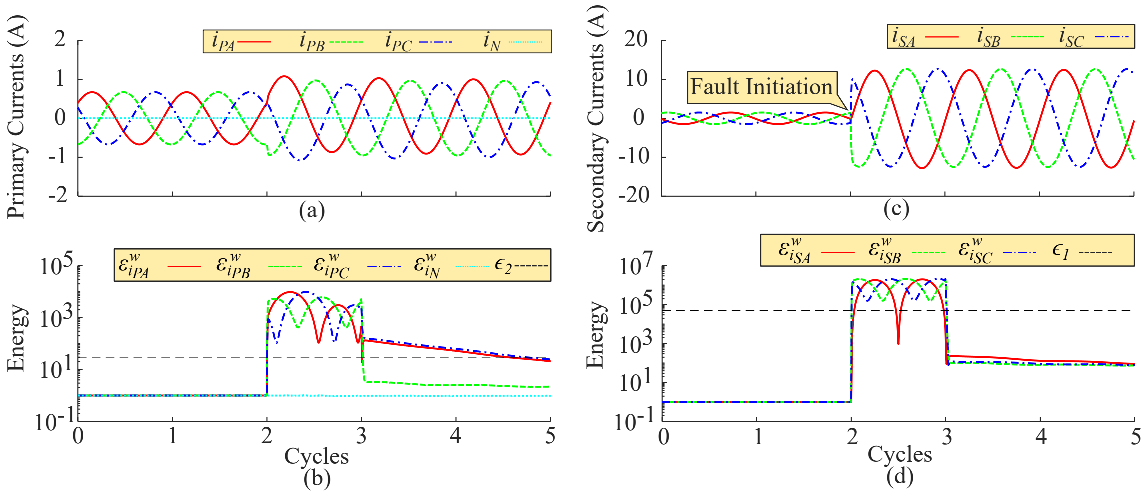

Figure 4.9: Three-phase fault in the secondary side (ABC_S fault): (a) Primary-side currents; (b) Energy of the primary-side currents; (c) Secondary-side currents; (d) Energy of the secondary-side currents.

4.2 Summary

The proposed method employs the RT-BSWT to compute the energy of the scaling and wavelet coefficients of the transformer currents, enabling the classification of different types of faults in the primary and secondary windings. Fault classification is based on empirically determined energy thresholds, where each fault is uniquely characterized by the values of the energies of the wavelet coefficients of the currents that exceed or do not exceed these thresholds on both sides of the transformer. Additionally, the system is capable of detecting data absence and accurately distinguishing most internal faults, even in the presence of missing data, ensuring a reliable fault diagnosis in power transformers.

Chapter 5

The Proposed Machine Learning-Based Method

Machine learning is a key tool for pattern recognition and data-driven decision-making across various fields. Unlike traditional approaches, these models learn from examples without explicitly programmed rules. Among the most used techniques, supervised learning relies on labeled datasets to train models that map inputs to expected outputs.

This chapter presents the machine learning-based method, using features extracted from signal processing. The processing is performed using RT-BSWT, as described in Chapter 4, and the features generated from the wavelet coefficient energies are used as inputs for classification.

5.1 Implementation of the Models

The implementation of the models in this study was carried out using computational tools that facilitated their development and evaluation. This section presents the main libraries used and their contributions to this process.

The preprocessing stage was conducted in MATLAB, where essential attributes for modeling were extracted and processed. The environment enabled the application of signal analysis and numerical processing techniques, ensuring that the data was properly prepared before model implementation. These input attributes were then used in Google Colab, where the machine learning algorithms were trained and evaluated.

5.1.1 Google Colab

The experiments were conducted on Google Colab, a platform based on Jupyter notebooks that allows Python code execution in the cloud. Support for GPUs and TPUs accelerates model training, while the availability of pre-installed libraries eliminates the need for local configuration, making development more efficient.

5.1.2 Libraries Used

For model implementation, the Scikit-Learn library was used, providing algorithms for classification, regression, and clustering, as well as tools for data preprocessing, hyperparameter optimization, and model evaluation. Additionally, NumPy and Pandas were employed for data managment, and Matplotlib and Seaborn for result visualization, among others. In this study, three machine learning models were tested: logistic regression (LR), decision tree (DT), and random forest (RF), allowing for performance comparison across different approaches.

5.2 Signal Processing

The signal processing used in this machine learning-based method is the same as in Chapter 4, based on RT-BSWT, but without considering the predefined classification rules. The CT currents are decomposed, and the wavelet coefficients are calculated according to Equation 4.1. The energy of these coefficients is then obtained using Equation 4.3, which sums the squared values of the coefficients along of an one-cycle sliding window.

Fig. 5.1 illustrates this process in a flowchart. Initially, the wavelet coefficients of the currents are calculated. Then, the energy of these coefficients is extracted and used as input for the machine learning models.

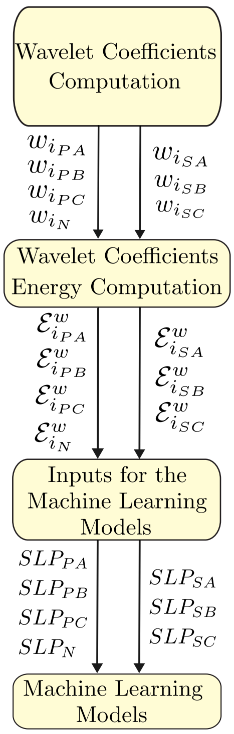

Figure 5.1: Signal processing flowchart based on RT-BSWT for the machine learning-based method.

5.3 Inputs for the Machine Learning Models

In this work, the wavelet coefficient energies of the CT currents are inputs for the machine learning models. For this purpose, two points were chosen for the wavelet coefficient energies of each current: the first one representing the pre-fault period and collected before the first faulted sample, and the second one representing the post-fault period and collected, in turn, in one-quarter of a cycle after the first faulted sample. For each selected point, the two coordinates are: the x-axis coordinate, regarding the collected sample index, and the y-axis coordinate, regarding the value of the wavelet coefficient energy of the current taken in this sample. From these two points, it is possible to compute the angular coefficient or slope (SLP) as follows:

$$SLP = \frac{\mathcal{E}i^w(kPF) - \mathcal{E}i^w(kSS)}{k_{PF} - k_{SS}}, \tag{5.1}$$

in which k_{PF} and k_{SS} correspond, respectively, to the samplings in the post-fault and steady-state periods and $SLP = \{SLP_{PA}, SLP_{PB}, SLP_{PC}, SLP_N, SLP_{SA}, SLP_{SB}, SLP_{SC}\}$ are the slopes for each one of the seven wavelet coefficient energy signals, which are inputs for the machine learning models.

Fig. 5.2 depicts some waveforms for an AG fault occurring in the transformer primary-side, at the order: the transformer primary-side currents (i_{PA}, i_{PB}, i_{PC}, and i_N); the wavelet coefficient energies of the primary-side currents ($\mathcal{E}^w_{i_{PA}}$, $\mathcal{E}^w_{i_{PB}}$, $\mathcal{E}^w_{i_{PC}}$, and $\mathcal{E}^w_{i_N}$); the transformer secondary-side currents (i_{SA}, i_{SB}, and i_{SC}); the wavelet coefficient energies of the secondary-side currents ($\mathcal{E}^w_{i_{SA}}$, $\mathcal{E}^w_{i_{SB}}$, and $\mathcal{E}^w_{i_{SC}}$). After the fault inception time, a hard increase in both phase A and neutral wavelet coefficient energies is expected, providing very high values for SLP_{PA} and SLP_N. In contrast, the energies of phases B and C are less sensitive, which is reflected in the low values for SLP_{PB} and SLP_{PC}, respectively [Fig. 5.2(b)]. Due to the magnetic coupling of the transformer, the phases A and B energies of the secondary side currents are also hard sensitized (SLP_{SA} and SLP_{SB} with high values), such as shown in Fig. 5.2(d), which is a challenge for a fault classification method because the current at phase B was affected in a fault that affected only phase A.

5.4 Summary

This chapter proposed the machine learning-based method, using features extracted from the primary and secondary side currents of the transformer. Signal processing was performed using RT-BSWT, as described in Chapter 4, but without considering predefined classification rules.

The method implementation was carried out in two stages. Signal preprocessing and feature extraction were performed in MATLAB, and the extracted data was transferred to the Google Colab environment for computational modeling and evaluation. During this stage, the Scikit-Learn library was used for data manipulation, while NumPy, Pandas, Matplotlib, and Seaborn assisted in data management and result visualization.

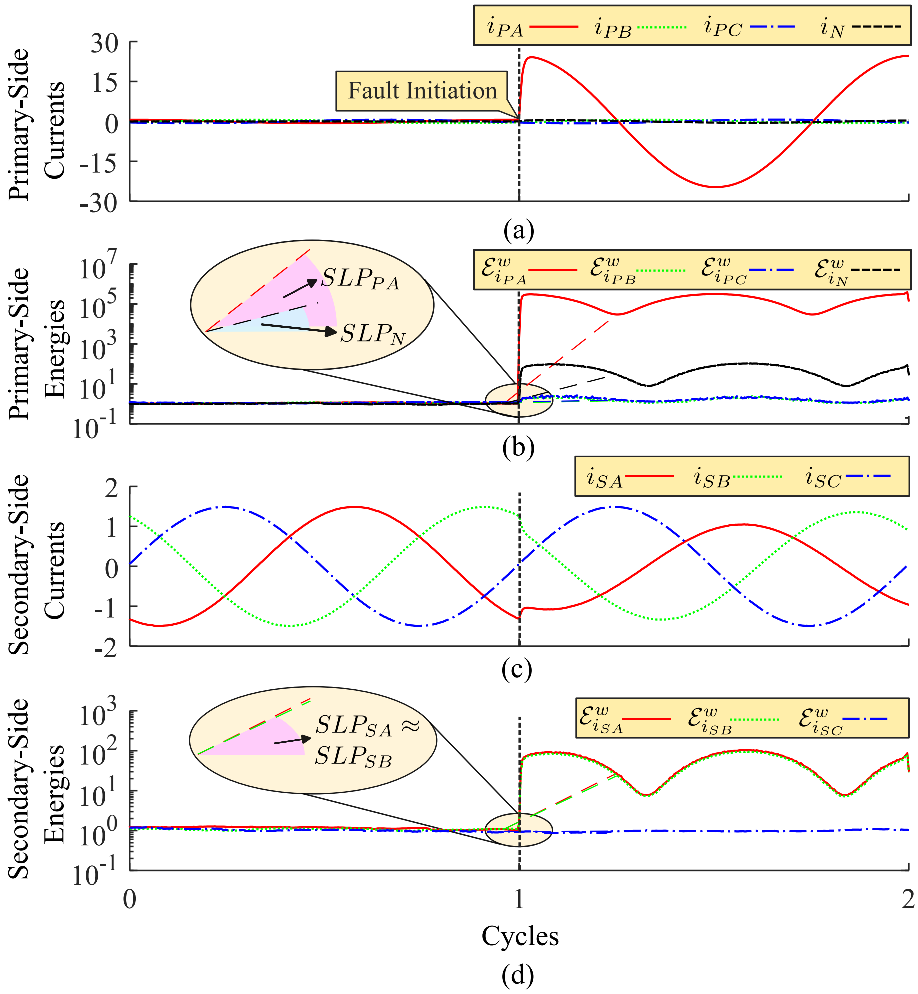

Figure 5.2: AG fault on the primary side of the transformer.: (a) Prymary-side CT currents; (b) Prymary-side energies ; (c) Secondary-side CT currents; (d) Secondary-side energies.

Chapter 6

Results

This chapter presents the analysis of the proposed methods for the classification of internal faults in transformers. Initially, the parameters of the electrical system and the transformer used in the modeling are described. Then, the performance evaluation of the wavelet-based and machine learning-based approaches is carried out. The analysis considers an extensive database of simulated faults, covering different operating conditions and scenarios with missing data.

No comparisons were made with existing methods, as no approaches were identified in the literature that consider internal fault classification in transformers in the presence of missing data.

6.1 Power System Description

The power system was modeled using the Alternative Transients Program (ATP). Two Thévenin equivalents, represented by sources S1 and S2 and their respective impedances, were employed. These equivalents are connected to the primary and secondary windings of the power transformer T1 through circuit breakers CB1 and CB2, as well as current transformers TC1 and TC2. The values of the resistances and reactances of the Thévenin equivalents are presented in Table 6.1.

The power transformer T1, described in Fig. 6.1, has a nominal power of 100 MVA and a transformation ratio of $V_H : V_X = 230 : 69$ kV. The high-voltage windings (V_H) are connected in a grounded wye, while the low-voltage windings (V_X) are connected in a delta configuration. The impedances of the primary and secondary windings of the transformer are, respectively, $Z_H = 2.04 + j12.54$ e $Z_X = 1.44 + j38.04$.

To model the power transformer nonlinearity, the Hevia Hysteresis routine from ATP was used. Regarding current transformers, their saturation curves were modeled by the Saturation routine. Table 6.2 presents the description of the current-flux (i, ϕ) pairs for the power transformer and for the current transformers.

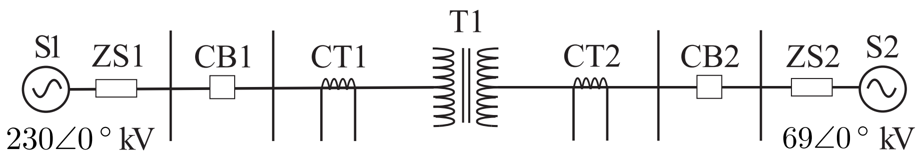

Figure 6.1: The Power Transformer

Table 6.1: Thevenin impedances of the transmission system (Medeiros 2017)

Fontes	$R_0(\Omega)$	$X_0(\Omega)$	$R_1(\Omega)$	$X_1(\Omega)$
S1	16.07	25.04	12.05	18.78
S2	5.52	8.61	4.02	6.26

Table 6.2: Nonlinear characteristic of magnetizing branches of power transformer and current transformers (Medeiros 2017)

T1		CT1 (800-5 A)		CT2 (1200-5 A)	
i (A)	φ (Wb)	i (A)	φ (Wb)	i (A)	φ (Wb)
0.14	498.14	0.052	0.112	0.054	0.338
0.48	523.04	0.075	0.225	0.132	1.606
1.21	547.95	0.135	0.450	0.175	1.876
2.54	572.86	0.165	1.125	0.189	2.251
6.45	579.08	0.301	1.501	0.341	2.626
8.95	585.31	0.555	1.688	0.561	2.926
15.59	591.54	0.687	1.876	0.976	3.001
35.46	603.99	44.856	2.251	9.440	3.477

6.2 Database

An extensive database was analyzed to evaluate the performance of the proposed method, which covered ten different types of short circuits (faults) on both the primary and secondary sides of the transformer. For each type of failure, different fault inception angles $\theta = \{0^o, 30^o, 60^o, ..., 180^o\}$ and different values for the fault resistance $r = \{10, 20, 30, ..., 100\}$ Ω were considered, resulting in a total of 70 scenarios for each type of fault. The database comprised 1,400 cases, providing a comprehensive and detailed analysis of the method's performance in various situations. The proposed fault classification method was designed to operate with all currents available and also in challenging cases where data in one of the currents is missing. Two scenarios were considered: 1) all currents available, and 2) a challenging situation where phase-A data in the primary was missing.

Figure 6.2 illustrates a scenario in which the current i_{PA} on the primary side of the transformer is absent during an AG fault on the same side. The image shows the

CTs installed on both sides of the transformer, measuring their respective currents, but the current i_{PA} is not recorded in the fault classification system.

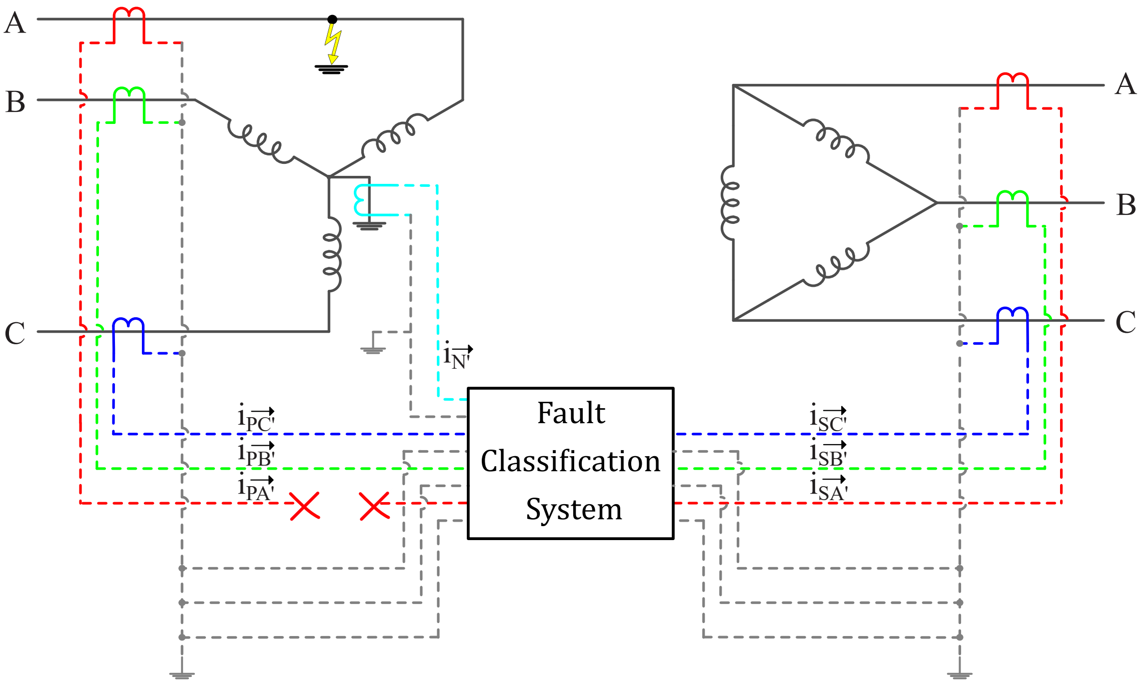

Figure 6.2: AG fault on the transformer's primary side, missing i_{PA} current.

6.3 Training and Testing Databases

To apply machine learning algorithms, the dataset was slightly expanded, and the modifications made will be detailed throughout the text. The ten types of faults analyzed are the same as those previously mentioned (AG, BG, CG, AB, BC, AC, ABG, BCG, ACG, and ABC), occurring in both the primary and secondary windings of the transformer, totaling 20 fault patterns. Unlike the proposed threshold-based wavelet processing method, presented in Chapter 4, which cannot distinguish between phase-to-phase and phase-to-phase-to-ground faults on the secondary side, the proposed machine learning-based method allows the classification of all 20 fault types. This is because machine learning algorithms analyze more complex patterns in the data, whereas the traditional wavelet method relies exclusively on predefined thresholds for classification.

Faults were simulated within the transformer's differential protection zone, i.e., between the CTs and the windings. As in the wavelet-based processing method, different fault inception angles $\theta = 0^o$, 30^o, 60^o, 90^o, 120^o, 150^o, and 180^o, as well as fault resistance values $r_f = 10$, 20, 30, 40, 50, 60, 70, 80, 90, and 100 Ω, were considered. The main difference is that, in this case, the analysis included transformer operation under both light and heavy load conditions, increasing the number of scenarios per fault type. As a result, 140 scenarios were generated for each fault type, leading to a dataset of 2,800 fault cases.

Differential protection algorithms for power transformers are well known for their ability to detect internal faults from external faults and transformer energization efficiently (Medeiros, Costa, Silva, Popov, Chavez Muro & Lima Junior 2021). Therefore, in a practical application, the proposed internal fault classification method would be triggered only when an internal fault is detected by the differential protection mechanism. As a consequence, the analysis of external faults and transformer energization events is beyond the scope of this work and was not considered.

The proposed power transformer fault classification method was implemented using random forest, decision tree, and logistic regression. These models were designed to operate even in challenging scenarios, such as critical faults combined with situations where data from one of the current signals is unavailable. Such data loss could occur due to physical interruptions in the connection between the CT secondary and the relay, or as a result of communication failures in transmitting current signals to the proposed algorithm. By addressing these potential challenges, the proposed method aims to provide reliable fault classification under real-world conditions. Moreover, in industrial applications, it facilitates the implementation of Industry 4.0 technologies, such as the Internet of Things and digital twins, among others, which rely on a continuous data flow.

For each of the 2,800 fault simulations, a data file was generated containing the following main columns: SLP_{PA}, SLP_{PB}, SLP_{PC}, SLP_N, SLP_{SA}, SLP_{SB}, and SLP_{SC}, which are inputs for the machine learning models as defined in Section III.D. The simulations were conducted using $f_s = 15360$ Hz, resulting in a time step (T_s) of about 65 μs. Each data file recorded only one sample for each current corresponding to 65 μs of fault duration, stored in the rows of the file. From the original dataset containing 2,800 faults, 2,240 faults (80%) and 560 faults (20%) were reserved for training and testing datasets, respectively. At this stage, the original datasets contained no instances of missing data, focusing exclusively on challenging fault scenarios.

For the training dataset, an approach was applied to include scenarios involving missing data from one of the current signals. Specifically, the column corresponding to the slopes calculated for one particular current was systematically removed, resulting in a new dataset with missing data for that current. This process was repeated for each of the seven currents, yielding seven additional datasets, each with missing data from a specific current. These modified datasets were then combined with the original training dataset with no missing data, expanding the training data to simulate real-world scenarios with data loss. Consequently, the training dataset increased eightfold, from the original 2,240 cases to $8 \times 2,240 = 17,920$ cases. The rows of this expanded training dataset were randomly mixed to ensure diversity and minimize any potential order bias during model training.

The 20% (560 cases) of the dataset reserved for testing, which did not contain missing data, was modified to include scenarios with missing current data. For instance, one column corresponding to the slopes associated with a particular current signal was removed to simulate missing data for that current. This process was repeated for all seven currents, generating seven additional datasets with missing data for each specific current. These modified datasets were then combined with the original testing dataset, resulting in an expanded testing dataset of $8 \times 560 = 3,920$ cases.

6.3.1 Training Performance

The proposed fault classification method was implemented using three machine learning models: RF, DT, and LR.

For training and validation, the extended training dataset consisting of 17,920 cases was used. This dataset encompasses scenarios where all data are available and situations with missing data, and a signal-to-noise ratio (SNR) of 55 dB was included. This approach allows the model to understand and handle the possibility of missing information, enabling it to better adjust to such cases. Thus, it becomes possible to develop a method that is more adaptable to real-world applications, where the model needs to classify even in the absence of certain input data.

The RF, DT, and LR models were trained using the hyperparameters defined in the scikit-learn library (Raschka et al. 2022).Tables 6.3, 6.4, and 6.5, present the hyperparameters used for each model.

Table 6.3: Default hyperparameters for Random Forest (RF)

Hyperparameter	**Default Value**
bootstrap	True
criterion	gini
max_depth	None
max_features	sqrt
n_estimators	100
min_samples_split	2
min_samples_leaf	1
random_state	20

Table 6.4: Default hyperparameters for Decision Tree (DT)

Hyperparameter	**Default Value**
criterion	gini
max_depth	None
splitter	best
min_samples_split	2
min_samples_leaf	1
random_state	20

Table 6.5: Default hyperparameters for Logistic Regression (LR)

Hyperparameter	Default Value
C	1.0
penalty	l2
solver	lbfgs
max_iter	100
random_state	None

Table 6.6 presents a performance comparison of the used machine learning algorithms (RF, DT, and LR) with respect to the following performance metrics: accuracy, precision, recall, and F1-score. Accuracy reflects the proportion of correct predictions, while precision measures the reliability of positive predictions. Recall indicates the model's ability to identify all positive cases, and the F1-score balances precision and recall.

Classification Metrics (SNR 55 dB, Missing Data)				
Algorithm	**Accuracy**	**Precision**	**Recall**	**F1-score**
Random Forest	0.953	0.95	0.95	0.95
Decision Tree	0.929	0.93	0.93	0.93
Logistic Regression	0.745	0.77	0.75	0.74

Table 6.6: Comparison of Classification Metrics for Training and Validation Data.

According to Table 6.6, the random forest achieved the best results, outperforming the other algorithms across all analyzed metrics. These findings highlight the robustness of the random forest, confirming its suitability for transformer fault classification under various conditions and missing data.

Fig. 6.3 presents the learning curves for the random forest, decision tree, and logistic regression models, enabling an analysis of each model's performance as a function of the training dataset size. The blue full curve represents the model's average accuracy on the training dataset, while the red dotted curve shows the average accuracy obtained through 5-fold validation. The shaded areas around each curve correspond to the standard deviation intervals, indicating the variability of results at each point.

The training curve reflects how well the model learns patterns in the training data. In contrast, the validation curve evaluates the model's performance on unseen data, providing a measure of its generalization ability. If the two curves are close together and have high values, this indicates that the model is well-fitted and generalizes well to new data, such as in Fig. 6.3a). On the other hand, a significant gap between the curves (with the validation curve being lower) suggests a model with overfitting, meaning it learns the training data well but struggles to generalize. Conversely, when both curves have low values, the model may be underfitted, indicating that it has not effectively captured the patterns in the data.

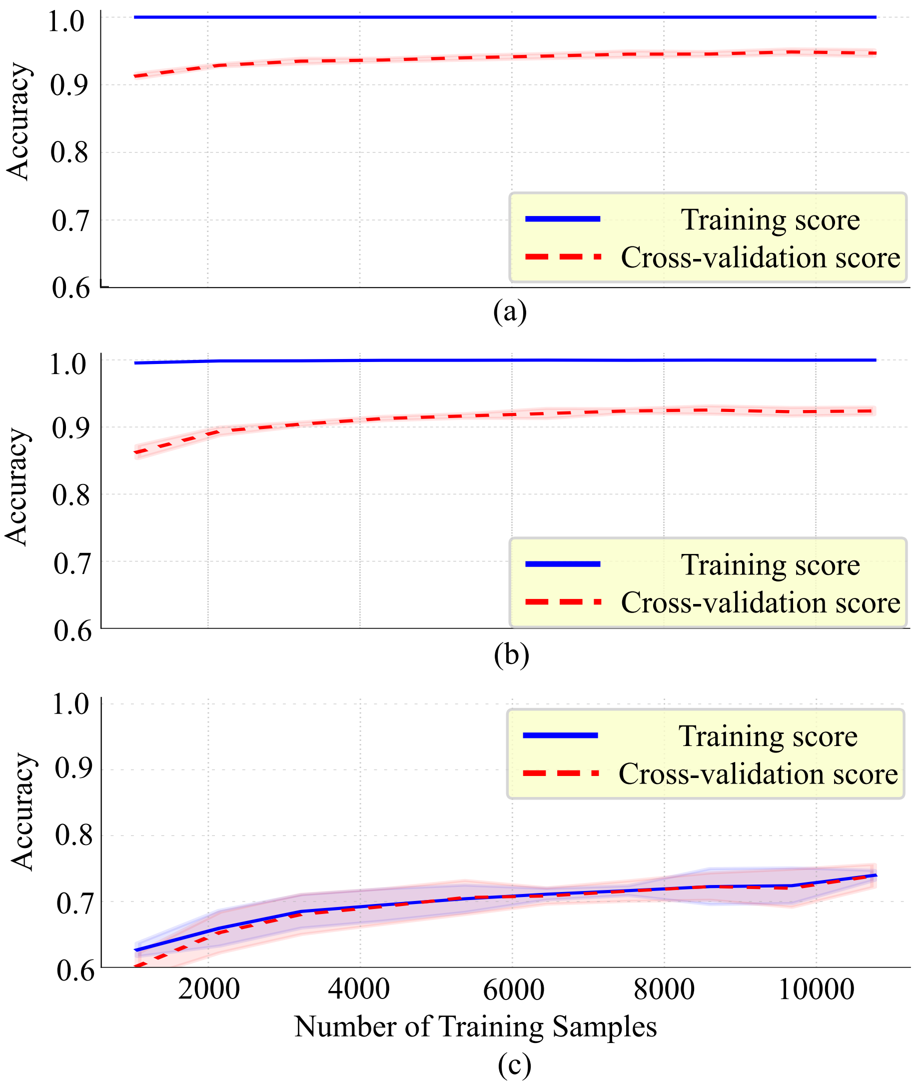

Figure 6.3: Learning curve for the proposed machine-learning-based models: (a) random forest; (b) decision tree; (c) logistic regression.

6.4 Performance Assessment of the Proposed Wavelet-Based Fault Classification Method

6.4.1 Overall Performance with All Data Available

Fig. 6.4 presents the confusion matrix generated from the 1,400 cases, considering full data availability. The wavelet-based method achieved an accuracy of 85%, demonstrating good performance in classifying most faults.

The main errors occurred on the secondary side of the transformer, where phase-to-phase-to-ground faults were classified as phase-to-phase faults. This pattern of misclassification highlights a limitation of the method in distinguishing these conditions, leading to some incorrect classifications.

Table 6.7 provides the numbering corresponding to each fault type, assisting in the interpretation of the confusion matrix (Figs. 6.4 and 6.5).

Number	**Fault**
0	AG - primary
1	BG - primary
2	CG - primary
3	AB - primary
4	BC - primary
5	CA - primary
6	ABG - primary
7	BCG - primary
8	CAG - primary
9	ABC - primary
10	AG - secondary
11	BG - secondary
12	CG - secondary
13	AB - secondary
14	BC - secondary
15	CA - secondary
16	ABG - secondary
17	BCG - secondary
18	CAG - secondary
19	ABC - secondary

Table 6.7: Labeling or codification of the fault types for method performance assessment.

6.4.2 Overall Performance With Data Missing

Fig. 6.5 presents the confusion matrix generated from the 1,400 cases considering the absence of data related to the phase-A current. Even with the lack of these data, the classifier demonstrated solid results, achieving an accuracy rate of 80%, showcasing its robustness even in challenging scenarios with missing data. However, in addition to the errors associated with phase-to-phase-to-ground faults on the secondary side, which were classified as their respective phase-to-phase faults, the proposed method exhibited misclassification in only one additional fault type.

With the absence of the phase-A current, the three-phase ABC fault on the primary side was incorrectly classified. When $i_{PA} = 0$, this fault was identified as a two-phase BC fault on the primary side. The absence of the I_a current on the primary side caused the classification (X) to be interpreted as (V) (Table 4.1), since the condition $\mathcal{E}^w_{i_{PA}} > \epsilon_1$ was not satisfied.

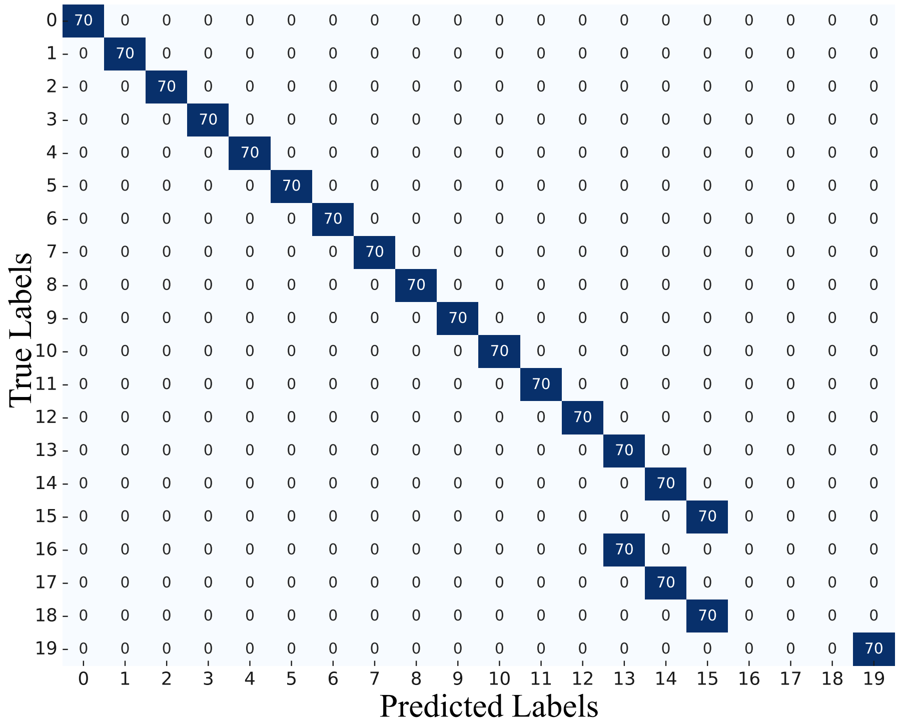

Figure 6.4: Confusion matrix with the performance assessment of the wavelet-based fault classification method: full data available.

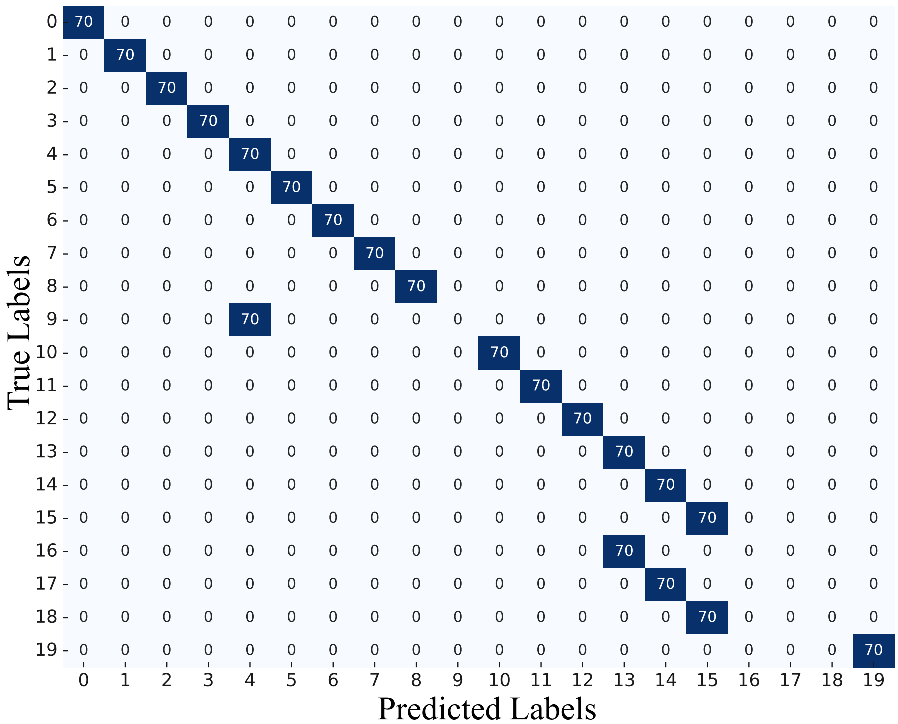

Figure 6.5: Confusion matrix with the performance assessment of the wavelet-based fault classification method: data missing.

6.5 Performance Assessment of the Proposed Machine Learning-Based Fault Classification Method

Table 6.8 summarizes the performance metrics for the three machine learning-based models used in this paper, evaluated using the extended testing dataset. The evaluation considered four distinct scenarios:

- Scenario 1: full data available with an SNR of 55 dB;
- Scenario 2: data missing considering an SNR of 55 dB;
- Scenario 3: full data available with an SNR of 40 dB;
- Scenario 4: data missing with SNR of 40 dB.

Therefore, in addition to addressing various challenging fault scenarios and missing data conditions, the assessment also accounted for the effects of noise, which is a critical factor in practical applications which require electrical quantities measurement and processing.

Classification Metrics (scenario 1)				
Algorithm	**Accuracy**	**Precision**	**Recall**	**F1-score**
Random Forest	0.998	1.0	1.0	1.0
Decision Tree	0.986	0.99	0.99	0.99
Logistic Regression	0.828	0.83	0.83	0.82
Classification Metrics (scenario 2)				
Algorithm	**Accuracy**	**Precision**	**Recall**	**F1-score**
Random Forest	0.929	0.93	0.93	0.93
Decision Tree	0.906	0.91	0.91	0.91
Logistic Regression	0.711	0.73	0.71	0.70
Classification Metrics (scenario 3)				
Random Forest	1.0	1.0	1.0	1.0
Decision Tree	0.975	0.98	0.98	0.97
Logistic Regression	0.827	0.83	0.82	0.82
Classification Metrics (scenario 4)				
Random Forest	0.938	0.94	0.94	0.94
Decision Tree	0.903	0.90	0.90	0.90
Logistic Regression	0.714	0.74	0.72	0.71

Table 6.8: Comparison of classification metrics for the testing step by considering scenarios 1, 2, 3, and 4.

According to Table 6.8, the random forest achieved the best performance regarding the assessed metrics, both for the scenario with all available data and for the scenario with incomplete data. Figs. 6.6, 6.7, 6.8, and 6.9 depict, respectively, the performance and the efficiency of the random-forest algorithm through confusion matrices for the scenarios 1, 2, 3, and 4.

In the testing scenarios with complete datasets (scenarios 1 and 3), the random forest algorithm achieved outstanding accuracy rates of 99.8% and 100% for SNRs of 55 dB and 40 dB, respectively. The confusion matrices in Figs. 6.6 and 6.8 depict these performances. In scenario 1 (Fig. 6.6), only one misclassification occurred, where a CG fault on the secondary side was incorrectly identified as a CG fault on the primary side. Meanwhile, in scenario 3 (Fig. 6.8), no misclassifications were observed, with all fault types correctly identified, as evidenced by the perfect alignment of rows and columns in the confusion matrix.

In data missing scenarios (scenarios 2 and 4), the random forest algorithm showed a slight decrease in performance across all evaluated metrics compared to scenarios with full data availability. For SNRs of 55 dB and 40 dB, the algorithm achieved accuracy rates of 92.9% and 93.8%, respectively. These results are remarkable given the complexity of the problem, which includes 20 distinct fault types and the challenge of missing data. Furthermore, the minimal impact of reduced SNR from 55 dB to 40 dB further emphasizes the robustness of the algorithm against noise. This robustness is attributed to the preprocessing step, which computes the boundary wavelet coefficient energies of the faulted current signals as model inputs. As addressed in (Costa 2014, Medeiros et al. 2016), these energies are minimally affected by noise, enabling reliable fault classification even under challenging conditions. Table 6.7 presents the labels linked to each fault type for further machine learning-based fault classification method performance assessment in a confusion matrix (Figs. 6.6, 6.7, 6.8 and 6.9).

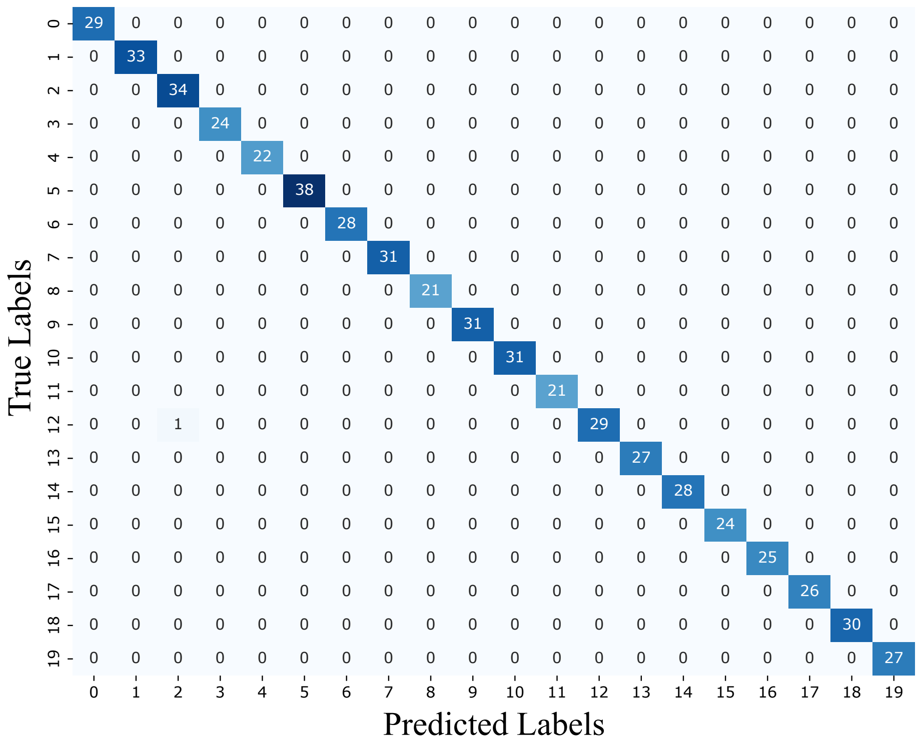

Figure 6.6: Confusion matrix for the scenario 1: full data available with SNR of 55 dB.

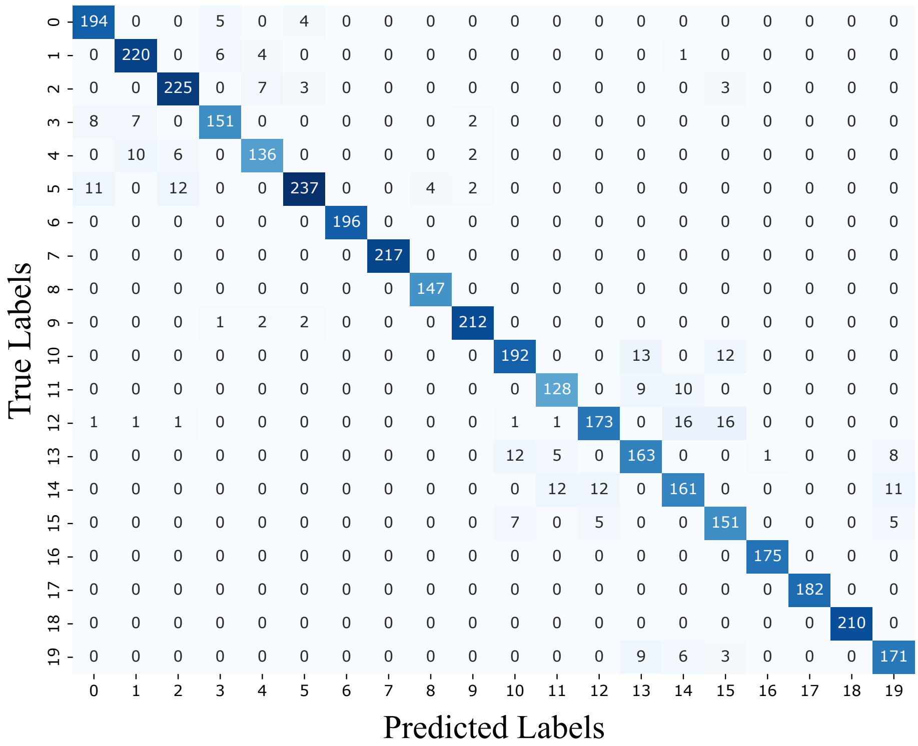

Figure 6.7: Confusion matrix for scenario 2: data missing for a single feature with SNR of 55 dB.

6.6 Performance Comparison Between The Wavelet-Based and the Machine Learning-Based Fault Classification Methods

The proposed wavelet-based and machine learning-based algorithms are compared to each other regarding two conditions: data availability and SNR, as summarized in Table 6.9. Using the testing dataset with all data available and 20 fault types, the RF-based method achieved a success rate of 100% across SNRs of 55 dB and 40 dB. In contrast, the wavelet-based method achieved an accuracy of 85% for an SNR of 55 dB, which dropped significantly to 42.5% at 40 dB.

The limitations of the wavelet-based fault classification method stem from its inability to distinguish between line-to-line and double line-to-ground faults on the transformer secondary side, as double line-to-ground faults were classified as their respective line-to-line faults. This limitation significantly affected its performance, whereas the machine learning-based method demonstrated a greater ability to distinguish between these types of faults, even under reduced SNR conditions.

When evaluated on the testing dataset with missing data and an SNR of 55 dB, the wavelet-based method achieved an accuracy rate of 80%, while the Random Forest algorithm maintained a robust performance, reaching an accuracy of 92.9%. However, when the SNR was reduced from 55 dB to 40 dB, the performance of the

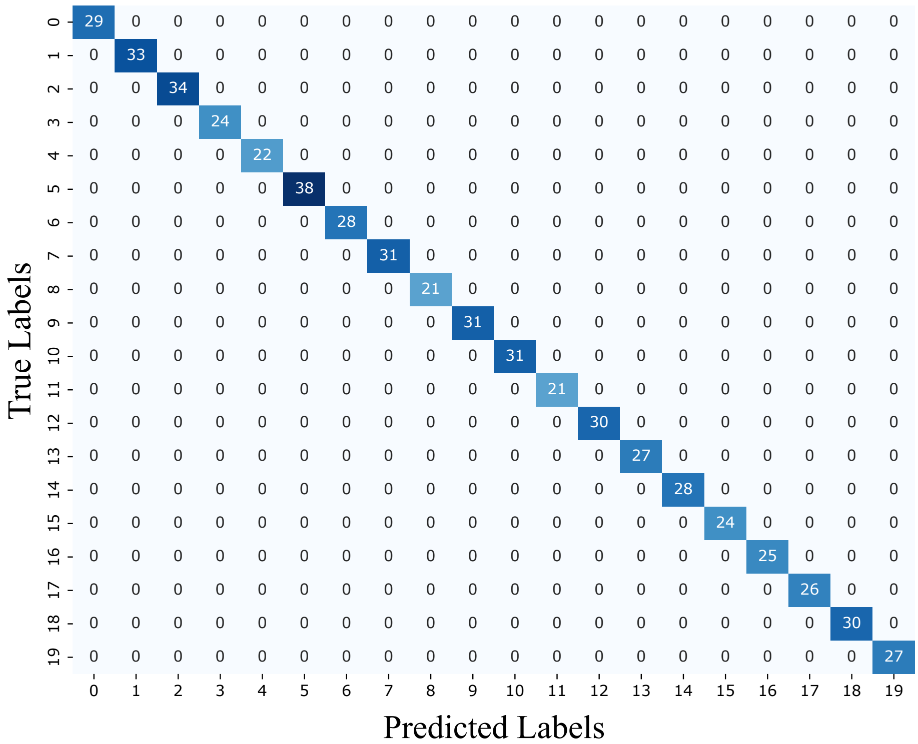

Figure 6.8: Confusion matrix for scenario 3: full data available with SNR of 40 dB.

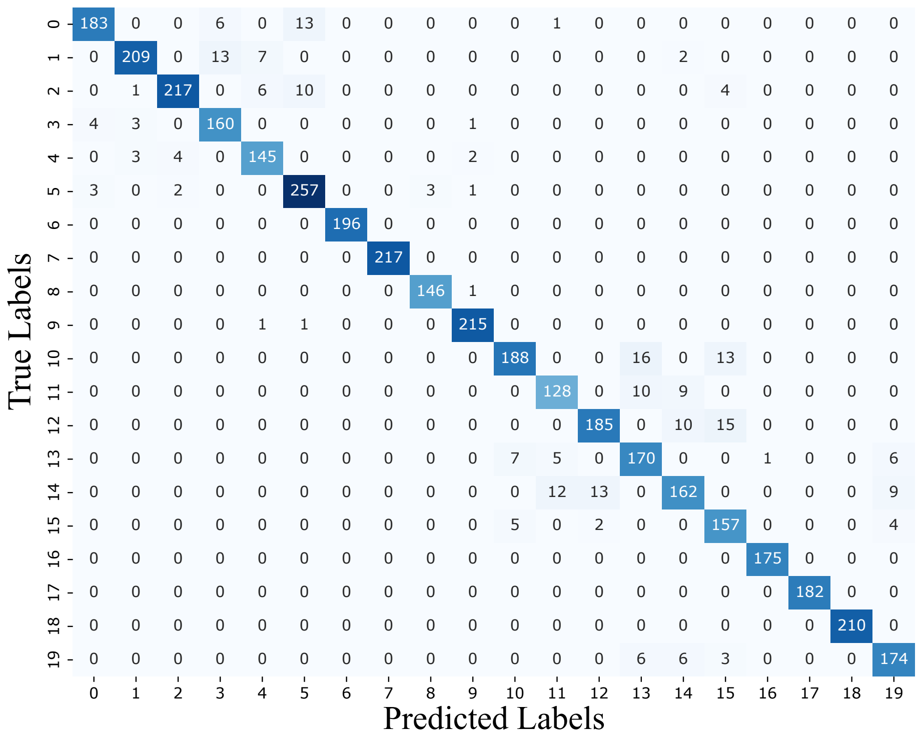

Figure 6.9: Confusion matrix for scenario 4: data missing for a single feature with SNR of 40 dB.

wavelet-based method dropped from 80% to 36.9%. In contrast, the Random Forest algorithm demonstrated resilience, achieving an accuracy of 93.8%, highlighting its robustness and effectiveness in scenarios involving missing data and significant noise.

The proposed wavelet-based method's strong dependence on threshold parameterization explains its low performance. In contrast, the proposed Random Forest-based algorithm leverages a machine learning framework combined with advanced wavelet transform preprocessing, eliminating reliance on thresholds or manual parameterization. This independence makes the algorithm more robust in scenarios with high noise levels or missing data, ensuring greater reliability in real-world applications. Furthermore, the proposed machine learning-based method accurately classifies all 20 fault types in a transformer, while the wavelet-based method is limited to recognizing only 17 fault types. This additional capability reinforces the versatility and effectiveness of the proposed machine learning-based algorithm in fault classification.

Table 6.9: Performance comparison of machine learning-based method and wavelet processing-based method under different conditions

Method	**Data Availability**	**SNR (dB)**	**Accuracy (%)**
Machine Learning-Based Method	Full Data	55	100
Wavelet Processing-Based Method	Full Data	55	85
Machine Learning-Based Method	Full Data	40	100
Wavelet Processing-Based Method	Full Data	40	42.5
Machine Learning-Based Method	Missing Data	55	93.8
Wavelet Processing-Based Method	Missing Data	55	80
Machine Learning-Based Method	Missing Data	40	92.9
Wavelet Processing-Based Method	Missing Data	40	36.9

6.7 Summary

This chapter presents the evaluation of two fault classification methods: one based on wavelet processing and the other on machine learning, both analyzed under various fault conditions.

The wavelet-based method achieved an accuracy rate of 85% when all data were available and 80% when the phase A current on the primary side was missing, both considering an SNR of 55 dB. However, it faced difficulties in distinguishing certain fault types due to its reliance on thresholds, failing to differentiate between phase-to-phase and double line-to-ground faults on the secondary side. These limitations directly impacted its performance, and when evaluated under an SNR of 40 dB, its accuracy dropped to 42.5% with all data available and to 36.9% when a data point was missing.

The machine learning-based method, specifically using the Random Forest algorithm, consistently outperformed the wavelet-based approach, maintaining an accuracy of 100% for both 55 dB and 40 dB SNR when all data were available. Additionally, it demonstrated robustness even in the absence of a data point, achieving accuracy rates of 93.8% for an SNR of 55 dB and 92.9% for an SNR of 40 dB. Unlike the wavelet-based method, which was limited to identifying only 17 fault types, the machine learning model correctly classified all 20 fault types.

Chapter 7

Conclusions and Next Steps

7.1 Conclusions

This work proposes two fault classification methods for power transformers: one using wavelet processing and the other based on machine learning. Both were tested under different and challenging fault conditions, evaluating their ability to correctly identify 20 types of internal faults.

The wavelet-based processing method achieved an accuracy of 85% when all data were available and 80% when data loss occurred in the phase A current on the primary side, considering an SNR of 55 dB. However, it faced challenges in classifying certain faults, particularly in differentiating between line-to-line and double line-to-ground faults on the secondary side, which were incorrectly classified as their respective line-to-line faults. Additionally, it struggled with the correct identification of three-phase faults when a data point was missing. These limitations affected its performance, which further deteriorated with the reduction of the SNR to 40 dB, resulting in accuracy rates of 42.5% with all data available and 36.9% when data was missing.

The proposed machine learning-based method, using the Random Forest algorithm, maintained an accuracy of 100% when all data were available and demonstrated robustness even in the presence of missing data and noise. In scenarios where one data point was missing, its accuracy was 92.9% with an SNR of 55 dB and 93.8% with an SNR of 40 dB, highlighting its adaptability to different conditions. Moreover, unlike the wavelet-based method, which was limited to identifying only 17 fault types, its reliance on predefined thresholds limits its adaptability to more complex scenarios. The machine learning-based model, on the other hand, correctly classified all 20 faults, demonstrating greater accuracy and versatility in internal transformer fault classification.

7.2 Next Steps

The next steps of this work involve the enhancement and expansion of the proposed methods, making them more robust and adaptable to different conditions. One key aspect to be explored is the application of the models to different types of transformers and operational scenarios, allowing for a more comprehensive evaluation of their effectiveness.

Additionally, the impact of removing multiple data points on classification accuracy will be investigated. This will help determine the extent to which the model can maintain acceptable performance under significant data loss and how many missing inputs it can tolerate before its accuracy is compromised.

To address these limitations, data imputation techniques will be explored, evaluating different approaches for reconstructing missing information and their impact on classification accuracy. Furthermore, other machine learning models, including more advanced alternatives, will be tested to assess their potential to improve system robustness and adaptability to various challenging operating conditions.

Finally, an essential step will be applying the method in a real or real-time simulated environment. This will allow for an assessment of its practical feasibility and potential challenges in implementing it within a transformer protection system. These advancements will contribute to the development of more reliable transformer protection strategies, aligned with the demands of Industry 4.0.

Bibliography

Abbasi, Ali Reza (2022), 'Fault detection and diagnosis in power transformers: a comprehensive review and classification of publications and methods', *Electric Power Systems Research* **209**, 107990.

Ahmadi, Hossein, Behrooz Vahidi & Amin Foroughi Nematollahi (2021), 'A simple method to detect internal and external short-circuit faults, classify and locate different internal faults in transformers', *Electrical engineering* **103**, 825–836.

Al-Fuqaha, Ala, Mohsen Guizani, Mehdi Mohammadi, Mohammed Aledhari & Moussa Ayyash (2015), 'Internet of things: A survey on enabling technologies, protocols, and applications', *IEEE communications surveys & tutorials* **17**(4), 2347–2376.

Ananwattanaporn, Santipont, Monthon Leelajindakrairerk, Chaiyan Jettanasen, Chaichan Pothisarn, Atthapol Ngaopitakkul & Dimas Anton Asfani (2017), Internal fault classification algorithm in power transformer based on discrete wavelet transform and fuzzy logic, *em* '2017 6th IIAI International Congress on Advanced Applied Informatics (IIAI-AAI)', IEEE, pp. 886–891.

Barbosa, D., U.C. Netto, D.V. Coury & M. Oleskovicz (2011), 'Power transformer differential protection based on clarke's transform and fuzzy systems', *IEEE Trans. Power Del.* **26**(2), 1212–1220.

Bera, Pallav Kumar, Rajesh Kumar & Can Isik (2018), Identification of internal faults in indirect symmetrical phase shift transformers using ensemble learning, *em* '2018 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)', IEEE, pp. 1–6.

Bhatnagar, Maanvi & Anamika Yadav (2020), Fault detection and classification in transmission line using fuzzy inference system, *em* '2020 5th IEEE International Conference on Recent Advances and Innovations in Engineering (ICRAIE)', pp. 1–6.

Breiman, Leo (2001), 'Random forests', *Machine learning* **45**, 5–32.

Cai, Zhichao, Zhixi Tang, Bichuan Xu & Tangbing Li (2024), 'Research on the filling of missing monitoring data under dc bias condition of power transformer', *Electric Power Systems Research* **231**, 110343.

Costa, F. B. (2014), 'Fault-induced transient detection based on real-time analysis of the wavelet coefficient energy', *IEEE Trans. Power Del.* **29**(1), 140–153.

Costa, F. B., A. Monti & S. C. Paiva (2017), 'Overcurrent protection in distribution systems with distributed generation based on the real-time boundary wavelet transform', *IEEE Trans. Power Del.* **32**(1), 462–473.

Costa, Flavio B. (2014), 'Boundary wavelet coefficients for real-time detection of transients induced by faults and power-quality disturbances', *IEEE Transactions on Power Delivery* **29**(6), 2674–2687.

Costa, Flavio B & Johan Driesen (2012), 'Assessment of voltage sag indices based on scaling and wavelet coefficient energy analysis', *IEEE Transactions on Power Delivery* **28**(1), 336–346.

Daubechies, I.. (1992), *Ten Lectures on Wavelets.*
URL: *http://books.google.com.br/books?id=Nxnh48rS9jQCprintsec=frontcoverdq=ten+lectur*
BRsa=Xei=EgZUU7aaAofLsATEv4GQDAredir$_e$*sc = yv = onepageqf = false*

Dzaferagic, Merim, Nicola Marchetti & Irene Macaluso (2021), 'Fault detection and classification in industrial iot in case of missing sensor data', *IEEE Internet of Things Journal* **9**(11), 8892–8900.

Faiz, Jawad, J Gharaeei & S Lotfifard (2016), Detection, location, and estimation of severity of interturn faults in power transformers, *em* '2016 10th International Conference on Compatibility, Power Electronics and Power Engineering (CPE-POWERENG)', IEEE, pp. 39–44.

Hosmer, David W, Borko Jovanovic & Stanley Lemeshow (1989), 'Best subsets logistic regression', *Biometrics* pp. 1265–1270.

Lala, Himadri, Subrata Karmakar & Sanjib Ganguly (2017), Fault diagnosis in distribution power systems using stationary wavelet transform and artificial neural network, *em* '2017 7th International Conference on Power Systems (ICPS)', IEEE, pp. 121–126.

Leal, Mônica Maria, Flavio Bezerra Costa & João Tiago Loureiro Sousa Campos (2019), 'Improved traditional directional protection by using the stationary wavelet transform', *International Journal of Electrical Power & Energy Systems* **105**, 59–69.

Li, Yuan, Yaoyu Xu, Xinghui Li, Rui Li, Jinshan Lin & Guanjun Zhang (2022), 'Addressing imbalance of sample datasets in dissolved gas analysis by data augmentation: Generative adversarial networks', *IET Generation, Transmission & Distribution* **16**(22), 4494–4504.

Mallat, S. (1999), *A Wavelet Tour of Signal Processing*, Wavelet Tour of Signal Processing, Elsevier Science.

Mallat, S. G. (1989), 'A theory for multiresolution signal decomposition: the wavelet representation', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **11**(7), 674–693.

Medeiros, R. P. & F. B. Costa (2018), 'A wavelet-based transformer differential protection with differential current transformer saturation and cross-country fault detection', *IEEE Trans. Power Del.* **33**(2), 789–799.

Medeiros, R. P., F. B. Costa & J. F. Fernandes (2014), 'Differential protection of power transformers using the wavelet transform', *IEEE PES Gen. Meeting* .

Medeiros, R. P., F. B. Costa & K. M. Silva (2016), 'Power transformer differential protection using the boundary discrete wavelet transform', *IEEE Trans. Power Del.* **31**(5), 2083–2095.

Medeiros, Rodrigo Prado de (2017), 'Proteção diferencial de transformadores de potência utilizando a transformada wavelet com efeitos de borda'.

Medeiros, Rodrigo Prado, Flavio Bezerra Costa, Kleber Melo Silva, Jose de Jesus Chavez Muro, Jose Raimundo Lima Júnior & Marjan Popov (2021), 'A clarke-wavelet-based time-domain power transformer differential protection', *IEEE transactions on power delivery* **37**(1), 317–328.

Medeiros, Rodrigo Prado, Flavio Bezerra Costa, Kleber Melo Silva, Marjan Popov, Jose de Jesus Chavez Muro & Jose Raimundo Lima Junior (2021), 'A clarke-wavelet-based time-domain power transformer differential protection', *IEEE Transactions on Power Delivery* pp. 1–1.

Mitchell, Tom M (1999), 'Machine learning and data mining', *Communications of the ACM* **42**(11), 30–36.

Patil, Gayatri S, Uma S Patil & Priyanka P Shinde (2024), Enhancing power transformer reliability through machine learning-based fault prediction using dissolved gas analysis, *em* '2024 Third International Conference on Power, Control and Computing Technologies (ICPC2T)', IEEE, pp. 72–76.

Percival, D. B. & A. T. Walden (2000), *Wavelet Methods for Time Series Analysis*, Cambridge series on statistical and probabilistic mathematics, Cambridge University Press.

Prado, Igor F, Flavio B Costa, Kleber M Silva, Rodrigo P Medeiros & Bruce A Mork (2024), 'High-frequency-based transmission line percentage differential protection with traveling wave alignment', *IEEE Access* .

Qiu, Guangping (2022), 'Application of wavelet packet and fuzzy algorithm in power system short circuit fault classification', *Mathematical Problems in Engineering* **2022**(1), 2882456.

Quinlan, J. Ross (1986), 'Induction of decision trees', *Machine learning* **1**, 81–106.

Rajaraman, P, NA Sundaravaradan, Rounak Meyur, M Jaya Bharatha Reddy & DK Mohanta (2016), 'Fault classification in transmission lines using wavelet multiresolution analysis', *IEEE Potentials* **35**(1), 38–44.

Raschka, Sebastian, Yuxi Hayden Liu & Vahid Mirjalili (2022), *Machine Learning with PyTorch and Scikit-Learn: Develop machine learning and deep learning models with Python*, Packt Publishing Ltd.

Russell, Stuart J & Peter Norvig (2016), *Artificial intelligence: a modern approach*, pearson.

Scarl, Ethan A (1989), 'Diagnostic tolerance for missing sensor data', *Telematics and Informatics* **6**(3-4), 251–258.

Scikit-Learn (2025), 'Plot the decision surface of decision trees trained on the iris dataset'. Accessed: 2025-03-01.
URL: *https://scikit-learn.org/stable/auto$_e$xamples/tree/plot$_i$ris$_d$tc.html*

Shaban, Ahmed E, Khalid H Ibrahim & Tamer M Barakat (2019), Modeling, simulation and classification of power transformer faults based on fra test, *em* '2019 21st International Middle East Power Systems Conference (MEPCON)', IEEE, pp. 908–913.

Shah, A. M. & B. R. Bhalja (2013), 'Discrimination between internal faults and other disturbances in transformer using the support vector machine-based protection scheme', *IEEE Trans. Power Del.* **28**(3), 1508–1515.

Sudha, B, LS Praveen & Anusha Vadde (2022), 'Classification of faults in distribution transformer using machine learning', *Materials Today: Proceedings* **58**, 616–622.

Vanamadevi, N & S Santhi (2013), Impulse fault detection and classification in power transformers with wavelet and fuzzy based technique, *em* 'Recent Advancements in System Modelling Applications: Proceedings of National Systems Conference 2012', Springer, pp. 261–273.

Venkatasami, Athikkan & Pitchai Latha (2016), 'Application of extreme learning machine in fault classification of power transformer', *Circuits and Systems* **7**(10), 2837–2845.

Vinothkumar, S, K Saravanakumar, M Sudarsan et al. (2021), Online transformer monitoring system, *em* 'Recent Trends in Intensive Computing', IOS Press, pp. 824–829.

Youssef, O.A.S. (2004), 'Combined fuzzy-logic wavelet-based fault classification technique for power system relaying', *IEEE Transactions on Power Delivery* **19**(2), 582–589.

Appendix A

The Random Forest Model Implementation

```
from google.colab import drive
import pandas as pd
import joblib
from sklearn.model_selection import train_test_split, learning_curve
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, classification_report,
    confusion_matrix
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns

drive.mount('/content/drive')

df = pd.read_csv('/content/drive/MyDrive/Documentos/tmp/
    planilhas_combinadas_55db_80.csv')
df['idF'] = df['b5']*32 + df['b4']*16 + df['b3']*8 + df['b2']*4 + df['b1']*2 + df
    ['b0']

X = df[['Ew_iaH', 'Ew_ibH', 'Ew_icH', 'Ew_inH', 'Ew_iaX', 'Ew_ibX','Ew_icX']].
    values
y = df['idF'].values

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.25,
    random_state=20)

rf = RandomForestClassifier(random_state=42)
rf.fit(X_train, y_train)
y_pred = rf.predict(X_test)
print("Accuracy:", accuracy_score(y_test, y_pred))
print(classification_report(y_test, y_pred))

importances = rf.feature_importances_
indices = np.argsort(importances)[::-1][:7]
selected_features = np.array(['Ew_iaH', 'Ew_ibH', 'Ew_icH', 'Ew_inH', 'Ew_iaX', '
    Ew_ibX', 'Ew_icX'])[indices]

plt.figure(figsize=(10, 6))
```

```
plt.bar(selected_features, importances[indices], color='lightblue')
plt.xlabel("Features")
plt.ylabel("Importance")
plt.title("Feature Importances")
plt.show()

cm = confusion_matrix(y_test, y_pred)
plt.figure(figsize=(10, 8))
sns.heatmap(cm, annot=True, fmt="d", cmap="Blues", xticklabels=range(len(cm)), yticklabels=range(len(cm)), cbar=False)
plt.xlabel('Predicted Class')
plt.ylabel('True Class')
plt.title('Confusion Matrix')
plt.savefig('/content/CM5501.pdf')
plt.show()

train_sizes, train_scores, test_scores = learning_curve(rf, X_train, y_train, train_sizes=np.linspace(0.1, 1.0, 10), cv=5, scoring='accuracy', n_jobs=-1, shuffle=True, random_state=42)
train_mean, train_std = train_scores.mean(axis=1), train_scores.std(axis=1)
test_mean, test_std = test_scores.mean(axis=1), test_scores.std(axis=1)

plt.figure(figsize=(8, 6))
plt.plot(train_sizes, train_mean, label="Training score", color="b")
plt.plot(train_sizes, test_mean, label="Cross-validation score", color="r")
plt.fill_between(train_sizes, train_mean - train_std, train_mean + train_std, alpha=0.1, color="b")
plt.fill_between(train_sizes, test_mean - test_std, test_mean + test_std, alpha=0.1, color="r")
plt.ylim(0.6, 1.01)
plt.title("Learning Curve - Random Forest")
plt.xlabel("Number of Training Samples")
plt.ylabel("Accuracy")
plt.legend(loc="best")
plt.grid(True)
plt.savefig('learning_curve_random_forest.svg', format='svg')
plt.show()

joblib.dump(rf, '/content/drive/MyDrive/Documentos/random_forest.pkl')

datasets = [
    ('/content/drive/MyDrive/Documentos/tmp/base_nova_55db_20.csv', '/content/CM5502.pdf'),
    ('/content/drive/MyDrive/Documentos/tmp/planilhas_combinadas_55db_20_semdados.csv', '/content/CM5503.pdf'),
    ('/content/drive/MyDrive/Documentos/tmp/base_nova_40db_20.csv', '/content/CM4002.pdf'),
    ('/content/drive/MyDrive/Documentos/tmp/planilhas_combinadas_40db_20_semdados.csv', '/content/CM4003.pdf')
]

rf = joblib.load('/content/drive/MyDrive/Documentos/random_forest.pkl')
```

```
for file_path, save_path in datasets:
    df_test = pd.read_csv(file_path)
    df_test['idF'] = df_test['b5']*32 + df_test['b4']*16 + df_test['b3']*8 + df_test['b2']*4 + df_test['b1']*2 + df_test['b0']
    X_test, y_test = df_test[['Ew_iaH', 'Ew_ibH', 'Ew_icH', 'Ew_inH', 'Ew_iaX', 'Ew_ibX', 'Ew_icX']].values, df_test['idF'].values
    y_pred = rf.predict(X_test)
    print(f"Accuracy for {file_path}:", accuracy_score(y_test, y_pred))
    print(classification_report(y_test, y_pred))
    cm = confusion_matrix(y_test, y_pred)
    plt.figure(figsize=(10, 8))
    sns.heatmap(cm, annot=True, fmt="d", cmap="Blues", xticklabels=range(len(cm)), yticklabels=range(len(cm)), cbar=False)
    plt.xlabel('Predicted Class')
    plt.ylabel('True Class')
    plt.title(f'Confusion Matrix - {file_path}')
    plt.savefig(save_path)
    plt.show()

print(rf.get_params())
```