Department of Electrical and Information Engineering
Electrical and Information Engineering Ph.D. Program
SSD: ING-INF/05 INFORMATION PROCESSING SYSTEMS

Final Dissertation

Trustworthy Machine Learning In Smart Grids

By:
Fatemeh Nazary

Academic Supervisors:
Prof. Carmelo Ardito
Prof. Eugenio Di Sciascio
Company Supervisor:
Eng. Gianluca Sapienza

Coordinator of Ph.D Program:
Prof. Mario Carpentieri

Course n° 35, 01/11/2019 - 31/12/2022

This work has been funded by e-distribuzione S.p.A. company, Italy.

Dedicated to my love (Yashar), my dear **mother**, and my dear brother (Amir).

Acknowledgements

In the beginning, I would like to express my gratitude and appreciation to my supervisor Prof. Carmelo Ardito. I thank him for helping and supporting me whenever I needed it. I thank him for building the trust on me and give me the freedom to do my research.

I would also like to thank my other supervisors Prof. Eugenio Di Sciascio and Eng. Gianluca Sapienza for their support from the university and company sides.

I wish to express my gratitude to Prof. Tommaso Di Noia whom I had his support during my Ph.D. period.

I am particularly grateful to my husband **Yashar**, who supported me and inspired me scientifically and spiritually during this period. I love you Yashar and thank you for everything, especially your love and kindness.

Furthermore, I thank Eng. Luca Delli Carpini, the Engineer in e-distribuzione Smart Grid Lab, explained the overall picture to us in Milan.

Last but not the least, I would like to thank my first and forever friend my mother and my brother who has always been supportive of my family. I would like to thank my grandpa who was my father and supporter throughout my life and left me alone 6 years ago. I would also like to thank my dear mother-in-law for her support; you are always with us.

Abstract

More than a decade after its introduction, the concept of a "smart grid" remains essential to the industry's ongoing digital transformation. A smart grid (SG) is an electricity network that enables the bidirectional flow of electricity and data and can detect, react to, and proactively address changes in demand and a variety of other concerns, all through the use of digital communications technology. Modern SGs designed for the 21st century are required to have *self-healing* capabilities, which are characterized by the capacity to automatically restore and recover the interruption of energy in the grid and to shorten the interruption period for customers, thereby decreasing the likelihood of a more severe disaster, such as one caused by a cascading effect.

A wide range of disciplines, including computer science, electrical engineering, signal processing, statistics, artificial intelligence, and machine learning, have been applied to the study of automatic fault prediction tasks over the past years. This dissertation focuses on the integration of machine learning-based techniques to improve the self-healing capabilities of SGs and examines these ML approaches not only from the standpoint of fault prediction accuracy but also their **trustworthiness**. Among the numerous facets of trust, this study focuses on the *robustness* (against faults and adversarial attacks) and *interpretability* of the proposed fault prediction systems.

This doctoral research project was assigned by the e-distribution Smart

Grid Lab in Milan,[1] where the objective of the project was to "*develop Artificial Intelligence (AI) algorithms with the goal of enabling automatic self-healing characteristics for next-generation smart grids*". Self-healing can be used in a distribution network, e.g., in the smart grid, to detect a fault, localize the fault and diagnose the fault type, to isolate and neutralize it. It should be noted that we followed a Human-Centred Design approach,[2] initially visiting the e-distribution Smart Grid Lab in Milan to interview electrical engineers and study their work, systems, and artifacts. Then, due to the limits imposed by the COVID-19 epidemic, we conducted monthly video sessions with the Lab team in order to discuss the preliminary research findings.

In the context of this doctoral dissertation, methods for predicting faults and identifying them by type and origin have been devised, implemented, and evaluated. In order to extract useful information from the electrical signal and incorporate it into a machine-learning fault prediction system, a number of novel techniques have been proposed in addition to existing ones being improved. These techniques include hand-crafted temporal, frequency, and wavelet features, as well as 2D CNN-based visual spectrogram methods. We also examine the explainability of the various integrated technologies, including the use of *visual explanation*, in order to make the systems more transparent to a wider audience (operators, consumers).

Furthermore, this work enlightens a crucial research area in the security of smart grids, namely what happens to fault prediction methods when they are targeted by malicious actors or *adversarial attacks*. It is demonstrated that state-of-the-art adversarial techniques like FGSM and BIM are capable of learning minor perturbations that can trick the ML models, for example, by misclassifying the fault type or location, hence prolonging or impeding the recovery time of the rescue team.

[1] https://www.e-distribuzione.it/progetti-e-innovazioni/smart-grids.html

[2] ISO 9241-210:2019 - Ergonomics of human-system interaction - Part 210: Human-centred design for interactive systems.

Contents

List of Figures

List of Tables

CHAPTER 1

Introduction

Background. Earlier than a century ago, when energy demands were modest, the concept of our modern electricity was conceived. Traditional electricity grids were unable to adapt to the continuously changing requirements of the 21st century due to their one-way interaction and reliance on technologies from the 1950s. The ability to efficiently integrate and manage diverse energy resources, such as conventional fossil fuel sources, and renewable energy sources such as wind and solar energy, is vital for fulfilling the rising energy demand in the present and the future.

The term **smart electrical grids**, or simply smart grids (SGs), is used to describe the next generation of models for an intelligent electric network that allows the collection of both traditional fossil fuel sources and renewable energy sources such as wind and solar energy. Networks with such characteristics are needed to meet the rising energy demand in the present and future. SGs also take into account the actions of all connected end-users, offering bidirectional interactions between end-users and the grid operator as shown in Figure 1.1, hence enhancing the accessibility of the energy grid. As an illustration, consumers, such as residents and companies, are currently digitally connected to the ICT[1] infrastructures of DSOs[2] via a WAN network and *smart meters*, where the latter facilitates the connection between the DSO-ICT network and the consumer network. The ultimate objective of SGs is to make the electrical grid more efficient, dependable, secure, and environmentally friendly.

One necessity for SGs is the capacity to model *fault* states by their location and types. A fault in electrical grids could be caused by inclement weather, equipment deterioration, aging, or a security attack. A fault in an electrical line can result in a power outage that disrupts business and causes discomfort in homes and neighborhoods. Protection devices and circuit breakers have traditionally been used to monitor faulty lines and locations [49]. However, the power outage investigation report [10] released in 2006 described how undesirable operation of protection relays and circuit breakers might create catastrophic cascading effects and subsequent blackouts. For example, In August 2003, the power cascading failure in the Northeastern United States and Ontario, Canada created a global power outage that lasted four to seven days and left more than 50 million people without electricity, with damages estimated between 4 to 10 billion dollars. These situations call for intelligent, rapid, and precise power fault prediction systems that can analyze the health of the grid and conduct real-time fault analysis.

Approach. Over the past decade, researchers have analyzed autonomous fault detection tasks from a variety of perspectives, combining tools and methods from computer science, electrical engineering, signal processing, statistics, artificial intelligence, and machine learning (ML). Various approaches have been proposed to identify, classify, and localize faults [21, 40, 48, 64, 76, 87], which are described in detail in Chapter 2. This dissertation focuses on the integration of machine learning (ML)-based techniques to improve the self-healing capabilities of SGs and examines these ML approaches not only from the standpoint of fault prediction accuracy but also their **trustworthiness**.

With the increased use of machine learning in daily life, more empha-

[1] Information and Communication Technology

[2] Distribution System Operators (DSO)

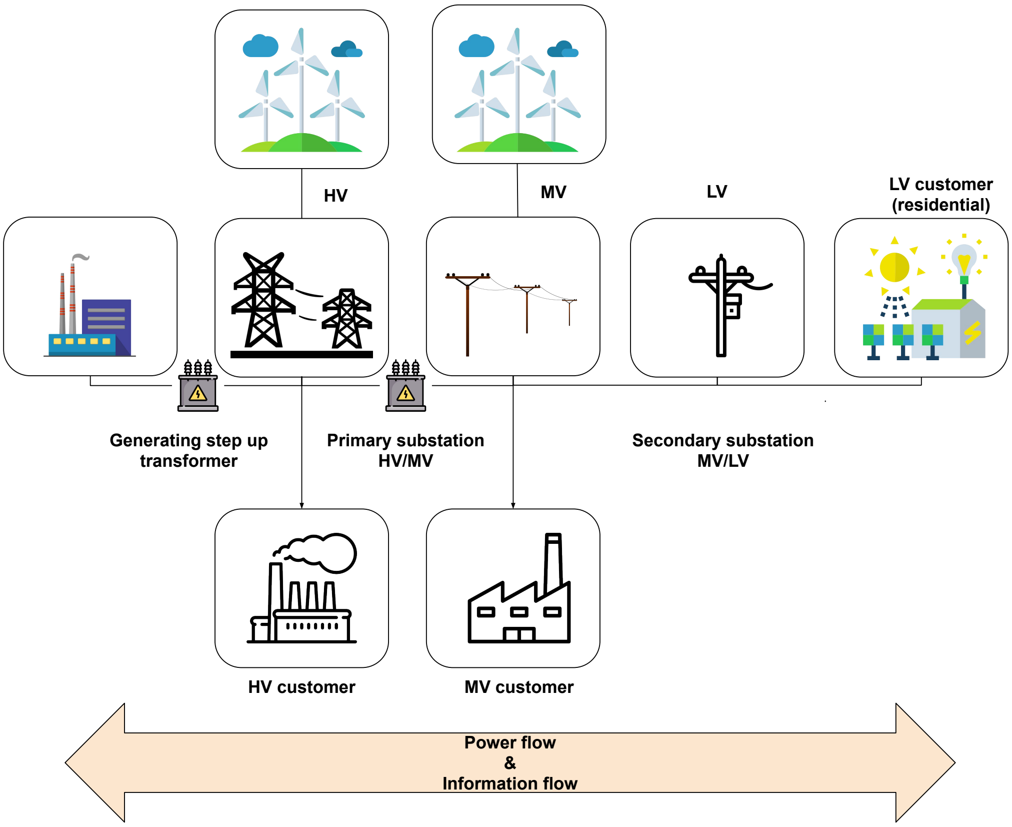

HV
MV
LV
LV customer (residential)
Generating step up transformer
Primary substation HV/MV
Secondary substation MV/LV
HV customer
MV customer
Power flow & Information flow

Figure 1.1: *Smart Grid technology two-way communication*

sis has been placed on its trustworthiness. Although there is no universally agreed notion for responsible machine learning, the major objectives include:

- Generalizability (or basic performance)
- Robustness
- Privacy
- Interpretability
- Fairness

where from the top to the bottom, the definition becomes increasingly obscure; for example, there is no single answer to the question of what fairness is in ML systems, and the response can be viewed from multiple perspectives. We discuss these aspects in more details in the following.

When discussing machine learning, the primary objective is typically to reduce loss across the training set, with the hope of maximizing prediction power on previously unseen data. Consequently, the primary pillar of trust is the *generalizability* of the model, i.e., first and foremost we want to train accurate models, and models that are not accurate are not trustworthy. However, trustworthy ML is not limited to generalizability (or basic performance), and additional aspects begin to emerge when these systems are deployed in high-stake applications involving sensitive data and critical scenarios. *Robustness* refers to a model's ability to withstand noisy and adversarial input, and may often be separated into training-time (also known as poisoning attacks) and test-time data (adversarial attacks). In order to establish trust in these circumstances, we must train models that are resistant to adversarial settings. For what concerns *privacy*, numerous ML models are trained on sensitive and private data. The question is whether or not we can trust models to have access to these sensitive data, and if they are trained on their data, whether or not sensitive data information could be leaked during the training phase.

Concerning accurate and generalizable models, the general trend is for these models to develop in complexity, and the fundamental question is how the model arrived at a particular conclusion. This may be applied to increase the transparency of decision-making processes. This can also be utilized for model troubleshooting, particularly if machine learning techniques are the core component of the system. This capacity to explain the model's decisions is frequently referred to as the model's *interpretability*. While increasing the quality of the ML model on the total dataset, we must also ensure that the model behaves consistently across subgroups, particularly those with sensitive characteristics such as gender and age. As a result, we seek to comprehend the *fairness* of ML models and to develop fair models.

1.1 Research Contributions

As explained in the preceding section, there are many facets of trustworthiness in ML and SGs and it is beyond the scope of a single thesis to cover all aspects. Beyond the generalizability of fault prediction approaches, this study focuses on certain characteristics of trust, such as

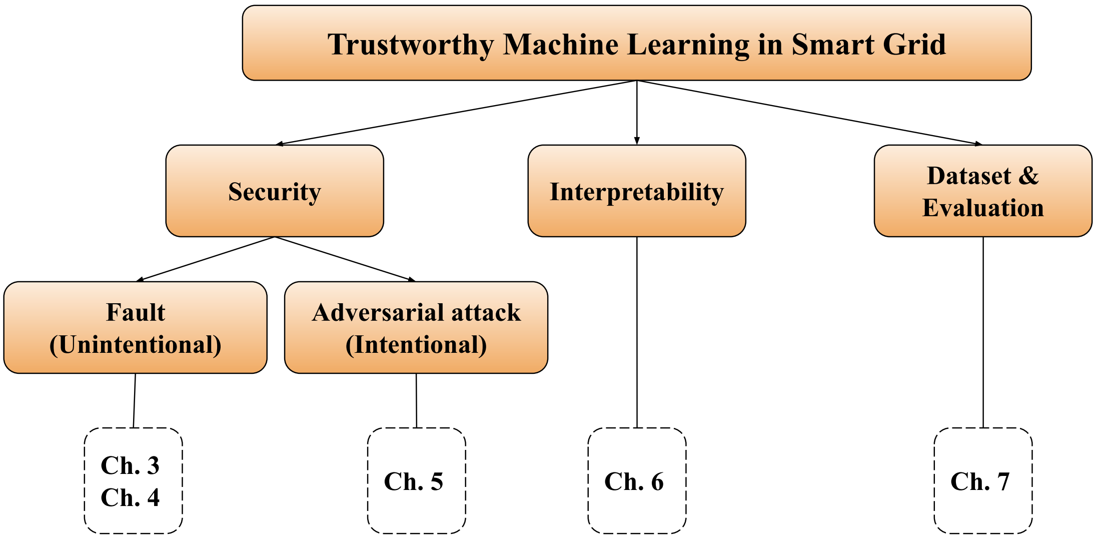


Figure 1.2: *Thesis outline*

robustness against faults (unintentional) and adversarial attacks (intentional) and interpretability, among others.

Figure 1.2 presents a summary of the research contribution in relation to the chapters of this dissertation. Each section discusses one aspect of trustworthy ML in SGs, discussed in more detail below:

1.1.1 Ch. 2: Literature Review

This section covers a comprehensive literature review that studies failure prediction methods in electrical grids from a computational approach and proposes a taxonomy for classifying state-of-the-art literature.

Contribution. Smart grid systems (SGs) are a prominent topic in the literature and contain various subtopics, such as fault management, communication issues, and security. Recently, a number of surveys focusing on diverse viewpoints of SGs have been published. A detailed examination of the algorithms and approaches for the three tasks of fault detection, fault type classification, and fault location prediction for transmission (HV), distribution (MV), and low-voltage (LV) lines has not been presented to our knowledge. We provide a comprehensive analysis and taxonomy of the currently employed strategies for fault prediction jobs in SGs. Our primary focus is on data-driven algorithms. The survey discusses future research directions and SG-related outstanding difficulties highlighted in Chapter 8. The content of the foundations and state-of-the-art described in Chapter 2 has been presented in the article, and the journal *Expert System with Applications* is currently reviewing the complete form of this work.

1.1.2 Ch. 3: Fault Prediction System Using Handcrafted Features

Contribution. In chapter 3 of this dissertation, we focus on three dimensions dependability, serviceability, and accountability, which comprise the security requirements of an SG application. The first two dimensions are covered in Chapter 3 and deal with fault detection and localization, while the final aspect addresses creating the system in a more transparent manner that will be explained in chapter 6. In Ch3 3, we propose a data-driven self-healing system that uses machine learning (ML) approaches to classify fault types and locations automatically. This chapter's research contributions are based on conference papers delivered at the *ITASEC* conference [8].

1.1.3 Ch. 4: Fault Prediction System Using Visual CNN features

Contribution. Due to the vast amounts of data encompassing energy networks, machine-learned (ML) models, especially those based on deep learning, have become more prevalent in the infrastructure of power systems. In order to bridge the gap between previous research and recent breakthroughs in the field of deep learning, in Chapter 4 we propose a spectrogram-convolutional neural network-based representation of the electrical signals, where pre-trained models such as GoogleNet and SqueezeNet are applied. We present simultaneous location and type classification along with an individual prediction of fault type and fault zone, which most previous works focused on. This chapter is extracted from the article published in the *Expert System with Applications* journal [6].

1.1.4 Ch. 5: Adversarial Machine-learned Attack in Smart Grid

Contribution. In Chapter 5 of this dissertation, we examine adversarial attacks against SGs. Adversarial attacks are subtle but non-random perturbations learned by the adversary and inserted into the test data to produce incorrect outputs. We investigate the impact of adversarial attacks on fault prediction systems, focusing on fault type and location classification, by investigating numerous experimental scenarios with varying adversary objectives (e.g., targeted vs. untargeted attacks). This chapter's proposed research contributions are based on conference papers presented at *Adversarial Learning Methods for Machine Learning and Data Mining@KDD'22*, the premier data mining conference.

1.1.5 Ch. 6: Interpretability of Fault Prediction System

Contribution. The exceptional accuracy of machine learning and deep learning methods comes with a cost: their growing complexity and resemblance to black-box models. This dissertation's Chapter 6 focuses on the interpretability of the proposed machine-learned fault prediction methods in SGs. In this chapter, we characterize explanation approaches and discuss them as feature-learned interpretability published in [7], visualizing the impact of pairs of attributes using a decision-tree model [8], and finally, we propose a CNN-based representation of spectrogram-modeled fault data and a visual explanation based on Grad-CAM [6].

The proposed research contributions of this chapter are derived from conference papers presented at the Italian Workshop on Explainable Artificial Intelligence co-located with the 19th International Conference of the Italian Association for Artificial Intelligence, *XAI.it@AIxIA 2020* [7], and the Italian Conference on Cybersecurity, *ITASEC 2021* [8], in addition to Journal article published at the *Expert system with Applications* [6].

1.1.6 Ch. 7: Datasets and Evaluations

Contribution. Despite the vast amount of effort and research published in the field of machine learning for smart grids, reproducibility and lack of available datasets with codes that facilitate the development of ML models and comparison between them pose a significant barrier to the advancement of research in the field.

We describe various kinds of datasets for fault prediction and adversarial attacks thereon and publish the benchmarking code for these scenarios. In particular, we release the following datasets:

- **IEEE13-AdvAttack**: The dataset released here[3] serves two principal purposes: (i) the classifications of faults and the regions where they are most likely to occur, and (ii) the analysis of adversarial machine-learning attacks aimed at fault type and zone classification tasks. This dataset has been released in the resource track of the CIKM'22 conference (main track) [5], which is an annual core-A conference in the field of data mining that attracts top-tier research;

- **Spectrogram-CNN:** The dataset released heree[4] serves two primary purposes: (i) spectrogram-based classifications of faults type and the zones where they are most likely to occur, and (ii) visual interpretation of spectrogram images of voltage signals using Grad-CAM. This dataset has appeared in the journal of *Expert System with Applications*.

1.2 List of Publications

The following articles were published during the course of this research (Corresponding author is Ph.D. Candidate in all accepted papers. Authors are listed in alphabetical order):

Under Review

Ardito, C., Cataldi, A.,Deldjoo, Y., Di Noia, T., Di Sciascio, E., & **Nazary, F.**. (2022). *Fault Prediction in Electrical Smart Grids: State-of-the-art and Future Trends under a Computational Perspective*.

Journal Publications

Ardito, C.,Deldjoo, Y., Di Noia, T., Di Sciascio, E., & **Nazary, F.**. (2022). *Visual inspection of fault type and zone prediction in electrical grids using interpretable spectrogram-based CNN modeling*, Journal of Expert System with application (ESWA), 210, 118368.

[3] `https://bit.ly/3NT5jxG`

[4] `https://github.com/atenanaz/FaultClf_SmartGrids`

Conferences Publications

Ardito, C.,Deldjoo, Y., Di Noia, T., Di Sciascio, E., & **Nazary, F.**. (2022, October). *IEEE13-AdvAttack A Novel Dataset for Benchmarking the Power of Adversarial Attacks against Fault Prediction Systems in Smart Electrical Grid*, In Proceedings of the 31st ACM International Conference on Information & Knowledge Management (CIKM), (pp. 3817-3821).

Ardito, C.,Deldjoo, Y., Di Sciascio, E., **Nazary, F.**, & Sapienza, G. (2021, August). *ISCADA: Towards a Framework for Interpretable Fault Prediction in Smart Electrical Grids*, In IFIP Conference on Human-Computer Interaction (pp. 270-274). Springer, Cham.

Ardito, C.,Deldjoo, Y., Di Sciascio, E., & **Nazary, F.**. (2021). *Revisiting Security Threat on Smart Grids: Accurate and Interpretable Fault Location Prediction and Type Classification*, In The Italian Conference on CyberSecurity (ITASEC) (pp. 523-533).

Ardito, C., Di Sciascio, E., & **Nazary, F.**. (2020). *Improving smart grid self-healing by a graph modeling approach*, 6th Italian Conference on ICT for Smart Cities And Communities (I-CiTies 2020).

Workshops

Ardito, C.,Deldjoo, Y., Di Noia, T., Di Sciascio, E., **Nazary, F.**, & Servedio, G.. (2022, August). *Machine-learned Adversarial Attacks against Fault Prediction Systems in Smart Electrical Grids*, AdvML@KDD'22.

Ardito, C.,Deldjoo, Y., Di Sciascio, E., & **Nazary, F.**. (2020). *Interacting with Features: Visual Inspection of Black-box Fault Type Classification Systems in Electrical Grids*, In XAI.it@AI*IA (pp. 135-141).

Following, we present the background and literature review of smart electrical grid and significant definition about it. Then, moving from Chapter 3 to Chapter 7, we describe in detail the research contributions shown in Figure 1.2. In the end, we review the findings in this dissertation and propose sopen research directions and possible future work fault prediction task in SGs.

CHAPTER 2

AI self-healing methods in Smart Grid: Foundations and State of the Art

Self-healing is one of the primary characteristics of smart electrical grids (SGs). The ability of SGs to automatically restore and recover the interruption of energy in the grid and to decrease the interruption period for customers lessens the likelihood of a more severe disaster, such as one caused by a cascading impact [56]. Self-healing requires both hardware (such as sensors, switches, actuators, and communication networks) and software to be effective (i.e., algorithms capable of providing fault detection and localization). In this chapter, we present the foundation background needed to understand the primary hardware and software components utilized in the literature on fault prediction for power grids.

2.1 Foundations of self-healing features in smart grid

Figure 2.1 depicts the taxonomy for the fault prediction system in SGs for the three tasks of (1) fault detection, (2) fault type classification, and (3) fault location classification.

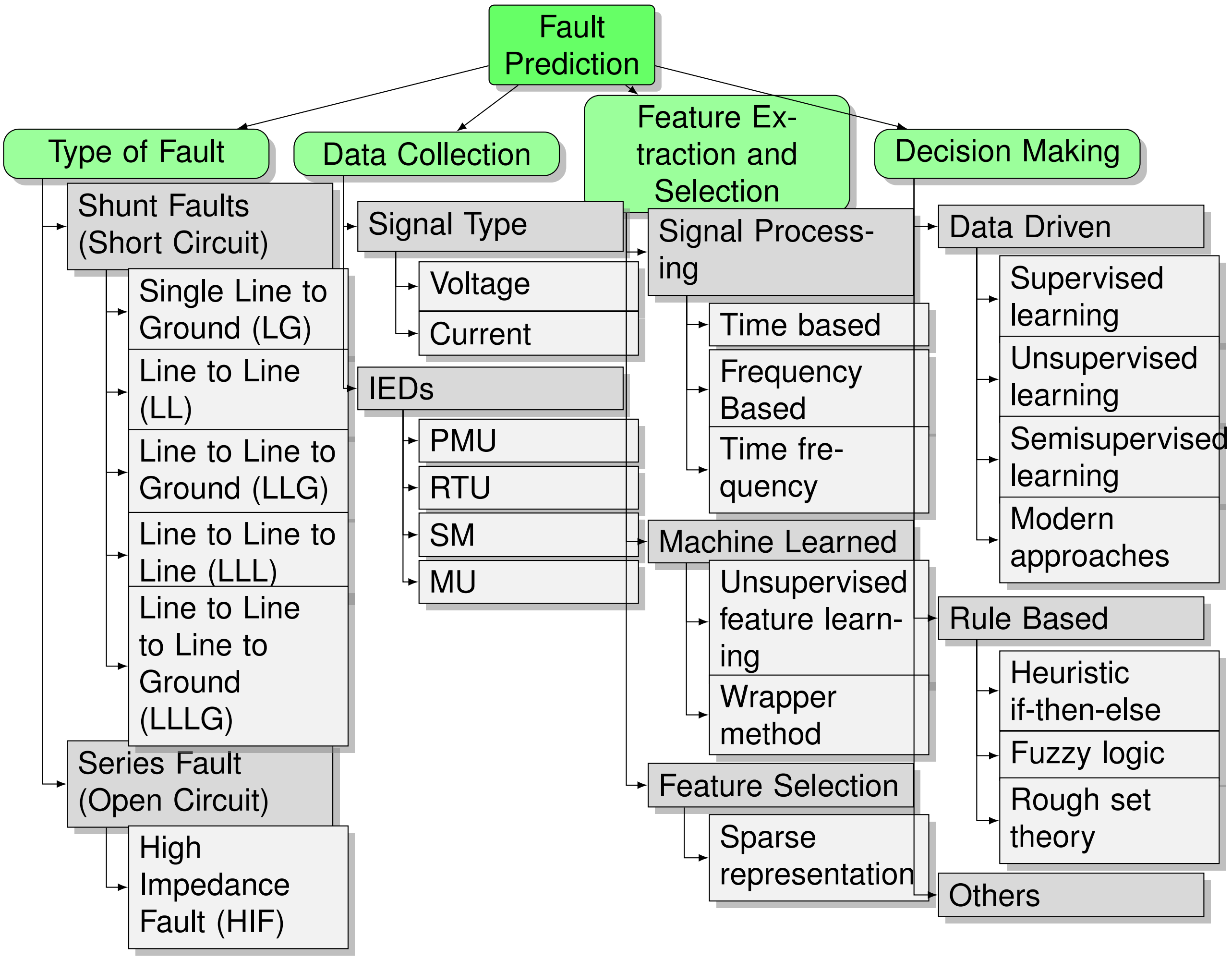


Figure 2.1: *Taxonomy of fault prediction system*

2.1.1 Fault type

There are two types of power grid electrical faults: *open-* and *short-circuit* faults. In addition, they may be either *symmetrical* or *asymmetrical*. In addition, the High Impedance Fault (HIF) [12,20], which does not (necessarily) fall into any of the aforementioned categories, has garnered considerable attention in recent years. This attention is in part owing to the threat posed by HIF faults and the difficulty of identifying them. These types of faults are thoroughly covered in the following:

- *Short-circuit faults.* Short-circuit faults (a.k.a. "shunt faults") are one of the most common types of electrical network failures. A short-circuit fault is an abnormal connection with very low impedance between two locations with different potentials, whether intentional or unintentional. These are the most frequent and dangerous faults that can cause abnormally large currents to flow through the equipment or transmission lines. If short-circuit failures are allowed to continue merely a few times, they can cause substantial equipment damage (such as a fire) due to overheating or arcing problems. Short-circuit faults are caused due to insulation failure between phase conductors or between earth and phase conductors or both. The likelihood of these inaccuracies may be increased by severe weather conditions, such as lightning, intense rain and snow, outdated equipment, and human error [62]. These faults can be classified as symmetric or asymmetric in both transmission and distribution systems, i.e., phase-to-phase (LL), single-phase-to-ground (LG), or two-phase-to-ground (LLG), or symmetric i.e., three-phase-to-ground (LLLG or LLL) faults [1, 75]. Short circuit failures are responsible for the majority of power system malfunctions. The likelihood of an LG fault occurring in the power grid is 85%, whereas the likelihood of a three-phase fault occurring is 2% [62]. To detect and reduce these types of faults, protective devices like fuses, circuit breakers, and protective relays are employed for current and overload protection.

- *Open-circuit faults.* Open-circuit faults, also known as "series faults," can be either symmetrical or asymmetrical when one or more conductors (phases) in the power grid are broken. Joint failures of cables and overhead lines, failure of one or more circuit breaker phases, and melting of a fuse or conductor in one or more phases are the most frequent causes of these faults.

- *Symmetrical and unsymmetrical faults.* In symmetrical faults, all three lines are short-circuited to one another and occasionally to the earth (LLL, LLLG). Due to the fact that all three lines have the same load current magnitude and phase angle, symmetrical faults are also known as balanced faults. They are hazardous since they can produce a large amount of current, but they rarely occur. As a result of these problems, the load current in the three lines is unbalanced.

- *High impedance fault (HIF).* HIFs commonly occur at voltages between 4 kV and 34.5 kV in electric distribution. They occur when a conductor breaks and makes touch with the earth or when a high-impedance object comes into contact with the conductor. The largest problem with HIF is the amount of the fault current, which ranges from 0 to 75 (A) and exhibits flashing and arcing at the place of contact, creating a significant risk of fire or electrical shock [12, 20, 53, 84, 85]. As a result, HIF detection is vital for ensuring safety. However, because the fault current level is often lower than the nominal current, it is difficult for conventional safety systems to detect HIFs (such as over-current and distance relays).

2.1.2 Data collection

Algorithms for fault detection, classification, and localization use some form of data for their task. We can categorize this data into physical data (such as measured voltage and currents), which is connected to the topology of the electrical power grid, environmental data (such as whether), geographical data (such as information on latitude and longitude), and temporal data [22]. Voltage and current are among the most frequent pieces of information that the algorithms can access. In general, data are gathered from intelligent electronic devices and smart sensors (IEDs). These IEDs are integrated into the actual smart power grid or added to the particular grid's topology. These electrical devices/sensors with intelligence include:

- *Smart meter (SM)*. It is a device that captures current, voltage, and electrical energy usage in real-time. Both the electrical consumer and the provider are given access to this information. In the literature, SMs have been employed to extract measurements from voltage and current waveforms, consider for example [12, 41, 63] or to collect outage reports [39].

- *Phasor measurement unit (PMU)*. This device uses GPS as a common time source for synchronization to measure the magnitude and phase angle of current or voltage phasors from the electrical distribution infrastructure system as shown in Figure 2.2. In the literature, they are utilized to collect very accurate time-stamped measurements [20, 29, 33, 46]. [46] propose a PMU placement strategy to improve fault location prediction performance.

- *Frequency Disturbance Recorder (FDR)*. It is a real-time data acquisition tool attached to SGs that is more affordable and simpler to install than PMU. Voltage, phase angle, and frequency are just a few examples of multidimensional data that the FDR can measure in coordinated fashion. As an illustration, in [38], it is used to detect voltage and frequency, which are then transformed into feature vectors and trained using a Hidden Markov Model.

- *Merging Unit (MU)*. With the use of this device, analog signals from traditional current and voltage converters can be transformed into IEC 61850 sampled values. IEC 61850 is an international standard defining communication protocol for intelligent electronic devices at electrical substations [26].

- *Remote Telemetry Unit (RTU)*. It is a remote-installed electronic control system for managing numerous pieces of equipment. Through messages delivered by the master system Supervisory Control and Data Acquisition (SCADA) [67], it controls the connected equipment by transmitting telemetry data from the equipment to SCADA systems and vice versa.

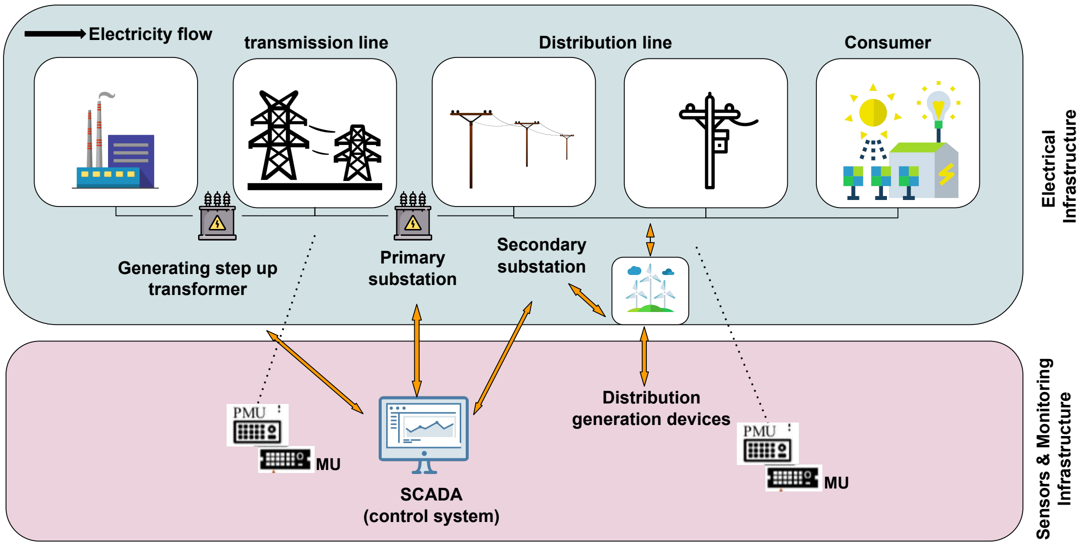


Figure 2.2: *Example of Sensors used to communicate between electrical infrastructure and monitoring infrastructure by collecting data such as PMU and MU*

2.1.3 Feature extraction and selection

Feature extraction aims to produce pertinent and valuable features from the system's raw data acquisition to achieve successful detection. The three subcategories of feature extraction approaches are signal processing, machine learning, and feature selection.

Signal processing methods deal with analyzing, synthesizing, and modifying the signal (voltage or current) to accomplish important tasks such as feature extraction required for the decision-making task. They could be divided into time-based (aka temporal) feature extraction [8, 20], frequency-based to provide uniform spectral resolution [4, 20] and calculating the phase angle of the voltage or current signals [66, 69], and time-frequency approaches, based on discrete-wavelet transform [1, 69, 88], s-transform [73], and Hilbert transform [2].

2.1.4 Decision making

Decision-making refers to the actual fault prediction algorithm, including fault detection (FD), fault type classification (FTC), and fault location prediction (FLD). Finding the distance between a fault and its nearby buses is called fault location detection. Some research projects concentrate on the relatively task of fault zone prediction (FZP), which refers to choosing the location of the fault from a specified list. Data-driven algorithms, rule-based algorithms, or other techniques, such as mathematical circuit modeling, can all be used as fault prediction algorithms as shown in Figur 2.3.

- Data-driven approaches. Data-driven techniques aim to model a system by "training" labeled data or "learning from examples," as the term implies. After a system has been trained, it can be tested with new data to determine how well it functions. Fig. 2 depicts the pipeline of data-driven strategies. The techniques used here can be separated into two categories: (1) traditional techniques like supervised learning and (2) modern techniques like deep neural networks (DNNs). In Section 2.2 we provided more detailed information on the topic.

- Rule-based approaches. Rule-based techniques seek to predict fault by applying a set of instructions and rules in which the knowledge

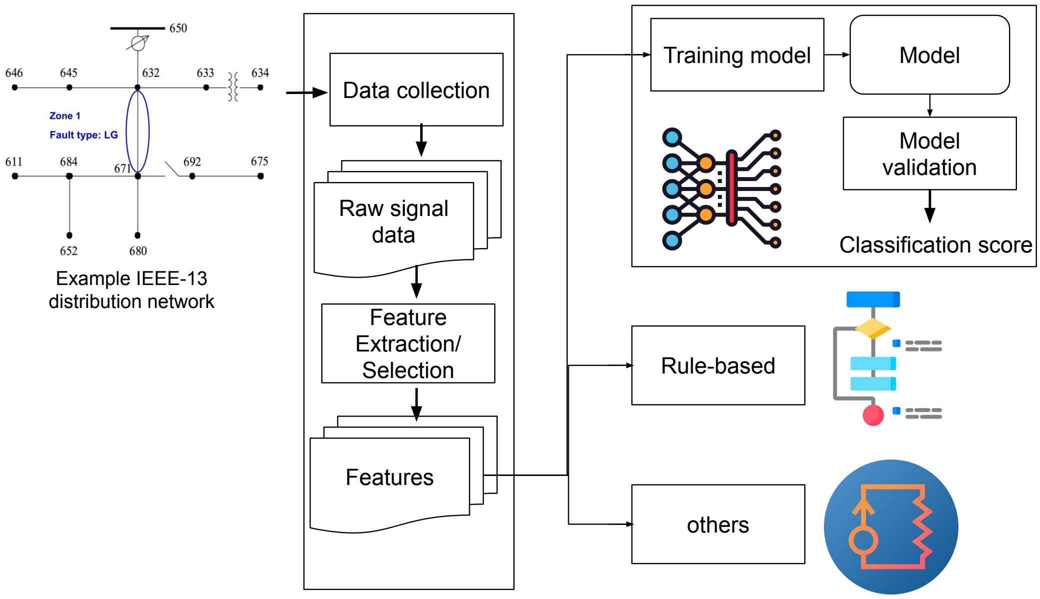


Figure 2.3: *The generic structure of decision-making procedures*

of a human expert has been manually encoded. Heuristic if-then-else, fuzzy logic, and rough set theory are three of these techniques. Section 2.2 has more specific information about these techniques.

- Other approaches. All strategies that come under other methods do not consider data-driven and rule-based decision-making techniques. Here, we focus on electrical circuit modeling, one of the most popular techniques in the literature. This method enables the study of electrical circuit behavior. A mathematical equivalent model of the circuit is built using particular software, and its behavior is then simulated under various circumstances. Circuit modeling, for instance, is used to introduce a fault into the circuit model and test the effectiveness of alternative algorithms for fault detection. For instance, in [32] a model is constructed for investigating faults during power swings. In [84] a simulation of a HIF was performed to evaluate the validity of the proposed detection method (cf. Section 2.2.3).

2.2 State-of-the-Art Approaches

This section widely categorizes decision-making (DMs) for fault prediction methods, as data-driven DMs or non data-driven DMs (e.g., rule-based approaches). Due to focusing on artificial intelligence algorithms, we mainly underline data-driven DM approaches; however, we discuss the second class briefly.

2.2.1 Data-driven decision-making

We categorize data-driven strategies into two groups, as shown in Figure 2.1: (i) classical methods and (ii) modern approaches such as graph-based models, deep learning (DL) models, and various other techniques, as indicated in Table 2.1. The following provides further clarifications.

Classical approaches:

Classical ML algorithms frequently necessitate feature engineering, feature extraction, and processing by ML algorithms. These approaches' key benefit is their simplicity, which often calls for less processing power and computational resources and makes interpretation simpler. In general, we can divide learning into three categories based on the training data $\mathcal{X} = [x_1, x_2, .., x_d, ., x_n]^T$, where n is the number of observations in a T-dimensional space, and $\mathcal{Y}$ represents the labeled observations: supervised learning techniques demand n labeled observations, unsupervised learning techniques require no labeled observations, and semi-supervised learning techniques demand d labeled observations, where $d < n$.

Definition 1 (Supervised Learning - SL)**.** *Given a dataset $\mathcal{D}$ of n pairs $(x, y) \in \mathcal{X} \times \mathcal{Y}$, where x is the input sample, and y is its corresponding class label, the goal is to learn a function $f_\theta : \mathcal{X} \to \mathcal{Y}$ that can predict the class label y for each input sample x, where θ is the model parameter. This leads to solving the following empirical risk minimization (ERM) problem*

$$\min_{\theta} \sum_{(x_i,y_i)\in\mathcal{D}} \ell(f(x_i;\theta), y_i)$$

where $\ell(.)$ is the empirical risk function or loss function. □

Examples of SL algorithms that have been applied in the literature of smart electrical grids for fault prediction we can name of support vector machine (SVM) [47, 64, 82],random forest (RF) [64], logistic regression (LR) [29], decision tree (DT) [1], K-nearest neighbor (KNN) [1], and neural networks [64, 73, 82]. Some research works also adopt mixed neuro-fuzzy inference systems that involve both artificial neural network and fuzzy logic [67, 82] for their addressed task. We will discuss deep neural types in modern approaches.

Definition 2 (Unsupervised Learning - USL)**.** *Given a dataset $\mathcal{D}$ of n input samples $x \in \mathcal{X}$, the goal is to estimate a model that represents the probability distribution $p(x_i \mid \theta)$, where θ is the model parameter. This probability distribution is essential for discovering impressive properties among data.*

An example of USL algorithms that have been used for fault prediction tasks can name intra-class clustering [64], which generates more specific information about fault, such as the most affected areas that require rapid intervention.

Definition 3 (Semisupervised Learning - SSL)**.** *Given a dataset $\mathcal{D}$ of n elements, it can be divided in two parts: $\mathcal{D}_1$ composed by l pairs $(x_l, y) \in \mathcal{X}_l \times \mathcal{Y}$, where x_l is the input sample, and y is its corresponding class label; $\mathcal{D}_2$ is composed by m input samples $x_m \in \mathcal{X}_m$ whose labels are unknown. The goal is to learn a function $f_\theta : \mathcal{X} \to \mathcal{Y}$ that can predict the class label y for each input sample $x \in \mathcal{X}$, with $\mathcal{X} = \mathcal{X}_l \cup \mathcal{X}_m$. In this case the ERM problem is*

$$\min_{\theta} \left\{ \mathbb{E}_{x_m \in D_2} \left[\ell_1 \left(f \left(x_m; \theta \right) \right) \right] + \sum_{(x_l, y) \in D_1} \ell_2 \left(f \left(x_l; \theta \right), y \right) \right\}$$

where $\mathbb{E}$ denotes mathematical expectation, and $\ell_1(.)$. $\ell_1(.)$ and $\ell_2(.)$ are the empirical risk function [61]. □

Different algorithms, including self-training, co-training, generative approaches, Graph-based methods, etc., are available for SSL. Self-training, in which the classifier employs its predictions to train itself, is an illustration of an SSL technique used for fault detection in [20]. Here, the author used SSL to deal with data that resulted from unseen events rather than relying just on a small amount of labeled data for detection. Additionally, there are methods that may be applied in mixed modes, SL, USL, and SSL, like Hidden Markov Model [37, 38].

A multitask logistic low-ranked dirty model (MT-LLRDM) is proposed by Gilanifar et al. in [29], in which the fault classifiers are trained in each location as a separate task. The method utilizes the similarities in the fault data streams among multiple locations across a power distribution network to improve detection performance. The fault types at each location are identified based on fault events obtained from PMUs in various locations. For each classifier task, a logistic regression loss function is ultimately used.

In [64], Raja et al. suggest a method for identifying and classifying nine type of faults. They collect a time series for each station in the grid and represent it as a vector. Then, they perform dynamic anomaly detection: if ‘n’ consecutive outliers are detected in the vector, where ‘n’ is a determined threshold, they assume a fault has occurred. To categorize the fault class, an auto-correlation function creates a small feature representation of the data, which is then passed to a classification technique like SVM, RF, or ANN. They also take into account intra-class analysis by using unsupervised learning to precisely pinpoint the fault.

Transmission line fault location is suggested by Livani et al. in [47]. The authors first utilize an SVM classifier to detect the faulty areas after extracting features from the observed voltages using Discrete Wavelet Transform (DWT). After identifying faulted areas, they use the aerial mode voltage wavelets to calculate the location of the fault.

An approach for HIF detection in distribution lines is proposed by Veerasamy et al. [82]. Discrete Wavelet Transform is used to extract features and train a variety of classifiers, including the Adaptive Neuro-Fuzzy Inference System, Bayesian Neural Network, Fuzzy Inference System, and Support Vector Machine (ANFIS). The study’s findings demonstrate that ANFIS and ANN classifiers perform better than the others.

In [67], Reddy et al. suggest a method for pinpointing transmission line faults. In order to facilitate synchronization, they obtain the fault currents from Remote Telemetry Units (RTUs) using GPS technology. Then they use Discrete Wavelet Transform to extract features and pass them to algorithms such as Adaptive Neuro Fuzzy Inference System (ANFIS) and Artificial Neural Network (ANN) in order to locate faults.

In order to select appropriate wavelet functions and wavelet decomposition levels for precisely classifying faults in transmission lines, Abdelgayed et al. in [1] offer a method based on the Harmony Search Algorithm (HSA). The identified optimal wavelet function is used in DWT to extract features from voltage and current signals. In the end, two machine learning techniques K-Nearest Neighbor (KNN) and Decision Tree (DT), are used for fault classifications.

To detect and classify faults in power grids, Shafiullah et al. in [73] propose a method based on a feed-forward neural network (FFNN). Here, the measured three-phase current signals are processed through s-transform (a generalization of the short-time Fourier transform (STFT)). In contrast to previous processing methods, ST decomposes processed signals into time-frequency components and includes phase information for non-stationary signals. FFNN is then given the extracted features, and it is trained to identify several fault types, including single-line-to-ground (LG), line-to-line-to-ground (LLG), and three-phase-to-ground failures (LLLG).

The purpose of Cui et al. in [20] is to detect and locate high-impedance faults by applying semi-supervised learning (SSL) to consider unlabeled data. They extract a feature pool encompassing time series features, DFT-based features, KF-based features, and other features from several micro-PMUs. Then, in order to choose the ideal feature set and prevent over-fitting, they use a wrapper method. To effectively classify data, the chosen features are passed to an SSL-based detection method. Finally, they propose an approach based on the probability distribution of the fault impedance for locating the fault.

Jiang et al. in [38] collect both frequency and voltage signals by Frequency Disturbance Recorders (FDRs) in order to detect and locate a fault in the transmission line. The Matching Pursuit Decomposition (MPD) technique processes the frequency signal to create a frequency feature vector using a Gaussian dictionary. The Hidden Markov Model (HMM) is then trained to use the frequency features to find and locate systemic flaws. Similarly to this, MPD processes the voltage signal and uses it to identify the fault's location.

To use the resulting spatial-temporal characteristics for fault identification, Jiang et al. train a variety of hidden Markov models (HMMs) in [37]. In the spatial domain, the secondary voltage control (SVC) divides the SG into different zones. A subset of synchrophasor measurement devices is placed within each zone based on an optimal synchrophasor measurement devices selection algorithm (OSMDSA). To properly characterize the signals, an MPD with a Gaussian atom dictionary is utilized in the time domain.

Modern approaches:

We define modern approaches as those that train massive machine-learning models employing vast amounts of data and powerful computing. The prominent examples in this category include deep neural networks (DNNs) [13, 46] and graph-based learning [33]. With great success, these methods have recently gained popularity for visual recognition tasks, and they have now been applied to numerous additional classification problems.

To accurately detect and classify faults, authors in [13] perform classification tasks based on convolutional sparse AutoEncode (CSAE) and softmax. They consider both three-phase voltage and current signals as multi-channel signals and then use Sparse AutoEncoder (SAE) to extract features from that signal automatically.

Locating the faulty lines in power grids is the main goal of the work in [46] in the condition of limited measurement availability. They propose a method based on a four-layer Convolutional Neural Network (CNN) classifier using voltage measurements. Additionally, they recommend a PMU placement strategy based on the CNN classifier's loss function to boost performance.

In [33] a dependency graph approach is proposed for fault detection and localization. They use a decentralized method based on Gaussian Markov Random Fields (GMRFs) in particular to describe the interdependence between many different variables. The proposed approach is useful when measurements coming from different PMUs are incompatible (e.g., not synchronized or at different sampling frequencies).

In [39] a data-driven approach for fault location is proposed. They use the outage reports from smart meters (SMs) to predict the outage region. In order to facilitate decision-making and accurately locate the fault, a model based on Mixed Integer Linear Programming (MILP) is employed once the outage region has been identified.

Authors in [70] defined a problem which is a localized fault as a two-class classification problem by considering temporal, geospatial, physical, and environmental data. To create the training set's partition, K-mean clustering is employed. Then a genetic algorithm is used for classification with an optimized learning rate. Finally, a fuzzy inference system is applied to assess the test set's dependability.

The authors of [42] investigate fault detection in covered conductor overhead lines and take into account the frequency of peaks in the partial discharge (PD) activity signal. As a result, a variety of classifiers, including RF, gradient boosted machine (GBM), naive Bayesian classifier (NB), and SVM are utilized to learn PD-Patterns to interpret a fault or a not faulty situation.

Table 2.1: *Classification of fault-related prediction systems from the perspective of* ***Decision Making***.

Step			Aspect and ApproachAspect and Approach	Highlight and Example WorkHighlight and Example Work
Decision Making	Data Driven	Classical	Supervised Learning:Supervised Learning:	
			• Support Vector Machine• Support Vector Machine	Fault class [64], Fault location [47], HIF detect. [82]Fault class [64], Fault location [47], HIF detect. [82]
			• Neural Network (NN)• Neural Network (NN)	Fault class [64], [45], FFNN [73], BNN [82]Fault class [64], [45], FFNN [73], BNN [82]
				Fault location [67], PD [42]Fault location [67], PD [42]
			• Adaptive Neuro Fuzzy• Adaptive Neuro Fuzzy	Fault location [67], HIF detection [82]Fault location [67], HIF detection [82]
			• Random Forest (RF)• Random Forest (RF)	Fault classifications [64], PD-patterns [42]Fault classifications [64], PD-patterns [42]
			• Decision Tree (DT)• Decision Tree (DT)	Fault classification [1]Fault classification [1]
			• K-Nearest Neighbor (KNN)• K-Nearest Neighbor (KNN)	Fault classification [1], Fault location [70]Fault classification [1], Fault location [70]
			• Logistic Regression (LR)• Logistic Regression (LR)	MT-LLRDM [29]MT-LLRDM [29]
			• Gradient Boosted Machine• Gradient Boosted Machine	PD-patterns [42]PD-patterns [42]
			• Naive Bayesian (NB)• Naive Bayesian (NB)	PD-patterns [42]PD-patterns [42]
			Hidden Markov ModelHidden Markov Model	Fault detection [38], Fault detection/Causal analysis [37]Fault detection [38], Fault detection/Causal analysis [37]
			Unsupervised LearningUnsupervised Learning	Dynamic time warping (DTW) [64]Dynamic time warping (DTW) [64]
			Semisupervised LearningSemisupervised Learning	Data from unseen events [20]Data from unseen events [20]
		Modern	Graph-basedGraph-based	GMRF [33]GMRF [33]
			Deep Neural NetworkDeep Neural Network	CSAE [13], CNN [46]CSAE [13], CNN [46]
			Generative MLGenerative ML	Anomaly detection [55]Anomaly detection [55]
	Rule Based		Heuristic if-then-elseHeuristic if-then-else	EHDI [12], Fault during power swings [32], VMD [85],EHDI [12], Fault during power swings [32], VMD [85], Faults in PV system [14], Inject high-freq. current [63],Faults in PV system [14], Inject high-freq. current [63], Passive overcurrent relay [69], VCCP [84], Hilbert transform [2],Passive overcurrent relay [69], VCCP [84], Hilbert transform [2], Upgraded MU [26]Upgraded MU [26]
			Fuzzy LogicFuzzy Logic	Fuzzy petri net [41], Reliability evaluation [70]Fuzzy petri net [41], Reliability evaluation [70]
			Rough Set TheoryRough Set Theory	DRS [65]DRS [65]
			Greedy AlgorithmGreedy Algorithm	Placement of PMU [46]Placement of PMU [46]
	Others		Sparse Rep.Sparse Rep.	
			• Group SR• Group SR	Automatic feature extraction [75]Automatic feature extraction [75]
			• Compressed Sensing (CS)• Compressed Sensing (CS)	Fault location [49], Bayesian CS [36]Fault location [49], Bayesian CS [36]
			Circuit ModelingCircuit Modeling	HIF [12], HIF [84], Power swing fault [32], Relay [69],HIF [12], HIF [84], Power swing fault [32], Relay [69], Impedance analysis [36], Substitution theory [46], State estimation [51]Impedance analysis [36], Substitution theory [46], State estimation [51]
			Traveling WavesTraveling Waves	Transmission lines [47], Distribution lines [74]Transmission lines [47], Distribution lines [74]
			Optimization ModelOptimization Model	MILP [39]MILP [39]
			PLCPLC	HIF detection/location [53,54]HIF detection/location [53,54]

2.2.2 Rule-based methods

A series of hard-coded instructions, such as if-then-else expressions, serves as the knowledge representation in a rule-based system. This knowledge is based on a curated set of rules that are frequently non-adaptive to the environment or new changes, reflecting the understanding of a human expert in the field. We identified the following rule-based approaches as shown in Table 2.1: *(i)* heuristic if-then-else approaches, *(ii)* fuzzy-logic and rough-set approaches as decision-making methods for fault detection in electrical grids. The articles discussed in this part use various signal representation and feature extraction techniques, but they all use rule-based decision-making as their common denominator.

Heuristic if-then-else approaches: Heuristics are methods for solving problems that might not be optimal but are nonetheless viable. It is employed to find an approximative solution for issues without a precise solution or for which finding one would take a lot of work. If-then-else refers to the notion that the solution is found using a succession of conditional statements.

By computing even harmonics in voltage measurements, Chakraborty et al. [12] suggest a unique application of SMs for identifying HIF in distribution networks. Standard power electronic loads are supposed to produce a significant number of odd harmonic components during steady-state operation, whereas HIF makes both even and odd harmonic components. The even harmonic components present in the voltage waveforms are measured by each SM using the even harmonic distortion index (EHDI). If that index consistently surpasses a threshold over a given period of time, HIF is recognized. To avoid switching or other brief transients being mistaken for HIF, time is set. When HIF is found, SM uses the communication channel to alert the distribution substation.

The aim of Hashemi et al. in [32] is to detect both symmetrical and asymmetrical faults during power swings by considering fundamental frequency phasors of voltage and current. The magnitudes of the voltage and current phasors oscillate with the swing frequency during power swings. In the absence of faults, the oscillatory magnitudes of voltage and current phasors are out of phase. If a fault occurs, instead, they become in phase. That is why they propose two algorithms capable of detecting phase differences. A delta-based algorithm measures delta values by subtracting the present value of phasor magnitude from its corresponding value at one power cycle earlier. Then they compare the current delta value with the voltage value. The admittance, which includes both an oscillating AC and a DC component, is calculated by the admittance-based algorithm. After removing the DC component with a full-cycle Discrete Fourier Transform (DFT), they compare the result with a threshold.

The transient zero sequence currents (TZSCs) are extracted by Weng et al. in [85] using the variational mode decomposition (VMD) approach to provide a series of intrinsic mode functions (IMFs). Then the kurtosis value is calculated for each IMF, and the IMF with the greater kurtosis value is selected. Teager-Kaiser Energy Operators (TKEOs) are calculated for the selected IMF. Subsequently, the entropy value is calculated. The HIF is determined by determining whether the entropy value is 0.

A robust phasor estimation method for fault detection is suggested in [2] by Affijulla et al. The method, which is based on the Hilbert transform, estimates the voltage and current phasors during a fault in order to calculate the fault impedance. After integrating the feature values of the voltage and current signals, they construct a derived feature value and normalize it using impedance. The normalize feature value is an excellent tool for fault detection since it is highly sensitive to the presence of a fault.

Photovoltaic (PV) system failure detection is the focus of the work proposed by Chen et al. [14]. This issue is seen by the authors as a sequential change detection issue. The output signals of the PV system are measured using different meters. Then the time correlation of the faulty signal and the signal correlation among different meters are exploited by a vector auto-regressive (AR) model. Due to the difficulty of obtaining a prior knowledge about the fault, they develop a change algorithm based on the generalized local likelihood ratio (GLLR) test.

The IEC 61850 Merging Unit (MU) is proposed to be updated by Gaouda et al. in [26] so that it can allow two-way communication and be capable of detecting impending faults. MU is upgraded with digital signal processing (DSP) ability that processes grounding currents and reports situation awareness (SA) features to the SCADA system. Self-healing capabilities can make use of SA features to detect and foresee early stages of coming faults.

Pasdar et al. present a method for detecting faulty nodes in [63] that relies on injecting high-frequency current signals. By calculating the difference between the measured and estimated voltage arrays, it is possible to determine changes in the impedance characteristics and locate the problematic node due to the injected current signal's imposition of voltages on the nodes. The information required to determine each node's impedance characteristics is measured using a standard smart meter (SM).

In [69] Saleh et al. propose a hybrid passive-overcurrent relay for fault detection. The proposed relay is outfitted with an inductor and a capacitor in parallel. Under DC fault circumstances, the LC circuit generates a specified frequency. A discrete wavelet transform (DWT) tool can be used to capture this frequency in order to find high-resistance faults.

In their HIF detection algorithm, Wang et al. [84] suggest looking for waveform distortion in the temporal domain. An HIF current waveform will always have some degree of distortion. The quenching and restrike dynamic process of an arc near zero-crossing of the fault current is the primary cause of the nonlinearity of waveform. They use a feature known as the voltage-current characteristic profile (VCCP), which creates a plot with the voltage and current represented by the Y-axis and the X-axis, respectively, to characterize the nonlinear arc resistance. The maximal slope appears at the zero-crossing parts. The variation of the slope can be used as a fault indicator.

Fuzzy-logic and rough-set approaches: it is a family of multivalued logics in which the variables can assume a truth value belonging to the real interval [0, 1], where 0 indicates "completely false", 1 indicates "completely true" and the included values indicate intermediate degrees of truth [16]. Additionally, a rough set is a particular mathematical technique used to cope with ambiguous and imprecise information and data that is strongly related to the fuzzy theory [90].

Kiaei et al. in [41] propose a hybrid fault location method based on fuzzy Petri net (FPN) technique, collecting data from protective devices, fault indicators, smart meters. The suggested method uses an inference system built using fuzzy petri nets (FPN) to predict the failed section utilizing discrete data from protective devices and fault indicators. Due to failures of protection systems and data loss, fuzzy Petri nets may suffer from multiple fault section estimation problems. To reduce the false estimations and detect the exact location of the fault, a consistency index is defined that quantifies the similarity between the measured voltage and current data and the corresponding values calculated using a computer short circuit program.

The system operator has a dilemma since authors in [65] attempt to classify the system's condition and protect it from various problems. To analyze the different combinations of the incoming signals and find patterns for them, the dominance rough set theory (DRS) is proposed to reasonably cope with all the clusters of data.

2.2.3 Other approaches

This section analyze additional fault prediction methods from the literature that do not belong to either data-driven or rule-based approaches. We've incorporated the following techniques: *(i)* sparse representation, *(ii)* electrical circuit modeling, *(iii)* traveling waves, *(iv)* optimization problem, *(v)* power line communication.

sparse representation (SR) is a technique that has gained popularity recently. The term "sparse solution" (also known as "sparse representation") refers to a linear system solution where the majority of the array's items are zero with a small number of non-zero components remaining. Sparse coding (SC), group sparse coding (GSC) [75] and compressed sending (CS) [36, 49] are all techniques which deal with finding a sparse representation. The fundamental idea behind CS is that with prior knowledge about constraints on the signal's frequencies, it can be reconstructed with a small subset of data. This enables the system to consider a compressed version of a problem in order to find a solution. On the other hand, given a feature vector, SC aims to create a smaller matrix, called the dictionary, and to ensure that the original feature vector can be represented as a linear combination of the fewest possible elements of the dictionary, called atoms.

For instance, Majidi et al. in [49] suggest a fault location strategy that builds the sparse current fault vector using compressed sensing (CS) and sparse representation (SR) methods. Non-zero entries in this vector indicate the possible faulted areas. They monitor fault currents from the PMUs and estimate fault voltages to distinguish between healthy and faulty zones, which is utilized to correctly identify the failed line. In the end, they calculate the fault distance in the faulted line using the least-squares method and the substitution theorem.

Shi et al. [75] seek to automatically extract features by taking into account the discriminative properties of sparse representation. Three-phase half-cycle superimposed current signals are measured for the fault classification task by Group Sparse Representation (GSR) because they reach desirable fulfillment even under a low sampling rate. Additionally, for the fault classification task, signals of fault types such as line-to-line, line-to-ground, and three phase-to-ground are learned in an over-complete dictionary.

For a complicated dc distribution network, Jia et al. [36] provide a dc pole-to-pole short-circuit fault location algorithm. First they construct the high-frequency impedance equivalent models of module multilevel converter (MMC) and dc/dc converter. The wavelet transform is then used to extract high-frequency transient voltages from sparse data points. Finally, they pinpoint fault locations using the Bayesian Compressed Sensing (BSC) theory.

Electrical circuit modeling is a common technique used in literature to investigate the behavior of electrical circuits and, consequently, electrical faults. A circuit model is a mathematical representation of an actual circuit or its components, accurately reflecting its behavior. In the literature, this technique has not been used alone but combined with other approaches to understand the behavior of particular faults and validate the proposed methods' efficiency through simulations. For instance Chakraborty et al. [12] and Wang et al. [84] use it to model the behavior of a HIF while Hashemi et al. [32] utilize circuit modeling to investigate three-phase faults during power swing. Moreover, Saleh et al. [69] use a circuit model to test the proposed relay during L-L and L-G faults. Jia et al. [36] simplify the analysis of the impedance of transient processes during faults by using an equivalent impedance model. A methodology based on the Kalman Filter (KF) estimator is suggested by Manandhar et al. [51] to identify faults and attacks on smart-grid systems. The KF estimates the power grid's state variables using data from the sensor network. Then χ^2-detector is used to detect differences between the estimated and measured data. They provide an additional detection method based on the Euclidean distance metric to get over the $chi2$-detector's limitations in identifying false data-injection attacks.

Traveling waves is a technique used for fault location tasks in transmission lines. The idea is that when a fault occurs, faults produce transient currents and voltages that radiate outward from the precise location of the fault. By capturing these waves and measuring the difference in arrival times, it is possible to locate the fault [47]. Since the waves might originate from other locations besides the fault and are, therefore, more difficult to find due to their numerous branches, this technique is more challenging to use in distribution lines. However, various techniques have been proposed to cope with this problem. For example, Shi et al. in [74] propose a method based on reclosure-generating traveling waves to remove traveling waves reflected from branches.

Fault detection and location can be solved through *optimization problems*. An optimization problem aims to find the best solution among all the possible ones, considering a series of constraints that must be verified. Among the optimization techniques, Integer Linear Programming (ILP) is used in [39] to locate faults. In this work, Jiang uses the outage reports from smart meters (SMs) to predict the outage region. Once the outage region is detected, data from Remote Fault Indicators (RFIs) is used by a model based on Mixed Integer Linear Programming (MILP) capable of supporting decision-making and correctly locating the fault.

In the literature, Some methods for fault detection take advantage of the frequencies that *power line communication* (PLC) systems typically employ for communication. For instance, Milioudis et al. in [53] describe an approach for HIF detection and localization using PLC techniques. PLC systems may provide high-speed data transmission and superimpose high-frequency signals on power networks. The HIF detection technique uses signal superposition on the power lines in a specific frequency range. For the chosen frequency range, the difference between input impedances under normal and fault circumstances is found and is utilized to detect the presence of a HIF. A protection scheme for HIF detection and localization in multiconductor distribution networks is also presented by the same authors in [54]. A PLC device installed at the starting point of the monitored line is used for fault detection by calculating differences in input impedance under normal and faulty conditions. Additionally, they determine the precise location of the fault; they use the responses to injected impulses measured from PLC devices.

CHAPTER 3

Data-driven Fault Prediction System Using Handcrafted Features

3.1 Introduction and context

Smart electrical grids, known as smart grids (SGs), are a complex infrastructure of distributed energy resources, appliances, and facilities that allow for the optimal use and asset optimization of resources, consequently lowering power consumption and investment costs [30]. This complicated infrastructure is therefore required to have high reliability, efficiency, and penetration of renewable energy sources [18]. As an illustration, a transmission line breakdown resulted in a cascade effect and a multi-day blackout in the Northeastern United States and Ontario, Canada, in August 2003. More than 50 million people were left without power, losing 4 to 10 billion dollars [10, 49].

Circuit breakers and protection devices have traditionally been employed to monitor faulty lines and locations [49]. According to the power outage examination report [10], The cascading effects and consequent blackout in North America in 2003 were primarily caused by the protection relays and circuit breakers operating improperly. These instances highlight the need for intelligent, quick, precise fault diagnostics and power system security assessment technologies. Gunduz et al. list a number of security requirements that must be met for an SG to be secure, including *confidentiality*, *integrity*, and *availability*, or the CIA triad. Data protection from unauthorized disclosure is referred to as confidentiality. Integrity is the prevention of illegal data tampering and destruction, while availability is the ability of authorized parties in the SG to access data when needed without compromising security.

Several security requirements, in addition to the CIA triad's security goals, must be met to guarantee cyber-security in SG applications. They include authentication, accountability, privacy, dependability, and survivability [30]. We concentrate on *dependability*, *survivability*, and *accountability* attributes and investigate them in a simulation of an actual electrical grid failure. Dependability refers to a system's ability to provide services on schedule, in an accurate manner, and without interruptions due to faults. Ensuring dependability requires fault detection, fault forecasting, and fault prevention. Survivability aims to provide services in the presence of malicious activities and external faults. The essential survival measures are fault localization, maintainability, and security protocols. Accountability operations make it possible to identify the source of a problem by presenting more visible proof of the grid's functionality.

Motivated by this observation, in recent years, some machine-learned techniques have emerged that aim to detect and diagnose the fault in a data-driven manner. Electrical grids must have this self-healing capability to be dependable and intelligent. In a nutshell, self-healing aims to perform fault detection, fault type classification (FTC), and fault location classification (FLC) to automatically restore and recover the interruption of energy in the electrical grid and shorten the interruption length for customers [56]. In this chapter, we put our attention to the following question:

- ***What*** type of fault happened in the electrical grid? known as the *fault type classification (FTC)* problem.
- ***Where*** the fault has occurred within the electrical grid network? known as the *fault location prediction (FLP)* problem.
- ***Why*** the ML system produced specific FLP, or FTC decisions? known

as the *interpretability* problem (This research question will be presented in chapter 6).

We present a systematic and in-depth study of FTC and FLP systems along the following directions:

- we addressee both FTC and FLC tasks.
- Here, we use time, frequency, and wavelet representations. We examine the effects of several statistical aggregation functions for feature representation, such as computing the energy and maximum level of the signals for *time*, *frequency*, and *wavelet*, along with the n-th moment of the probability distribution functions (PDFs) [78] ($n \in [1, 4]$).
- We thoroughly examine the interpretability phase and specify the information it enables us to learn about the effects of the feature classes and statistical aggregation operators used for feature representation (Interpretability part will be explaned in chapter 6).

3.2 Method

In this section, we describe the proposed system that receives a voltage signal as input and outputs two scores related to sub-tasks: FLP (zone 1, 2, 3, and 4) and FTC, line-to-ground (AG, BG, CG), line-to-line (AB, AC, BC), and three-phase fault (ABC). Voltage measurements are taken from a specific phase (A, B, or C) and zones from the IEEE-13 node test feeder simulink environment[1]. The pipeline of processing steps is shown in Figure.

3.2.1 Fault Simulation and Feature Extraction

We used the IEEE-13 node test feeder, a distribution network running at 4.16 kV, for simulation (Complete information about simulation conditions and distribution network will be presented in Chapter 7). We divided the network into four critical zones. All of the identified zones received fault injections. Then, voltage signals from all of those zones

[1] https://it.mathworks.com/help/sps/ug/ieee-13-node-test-feeder.html

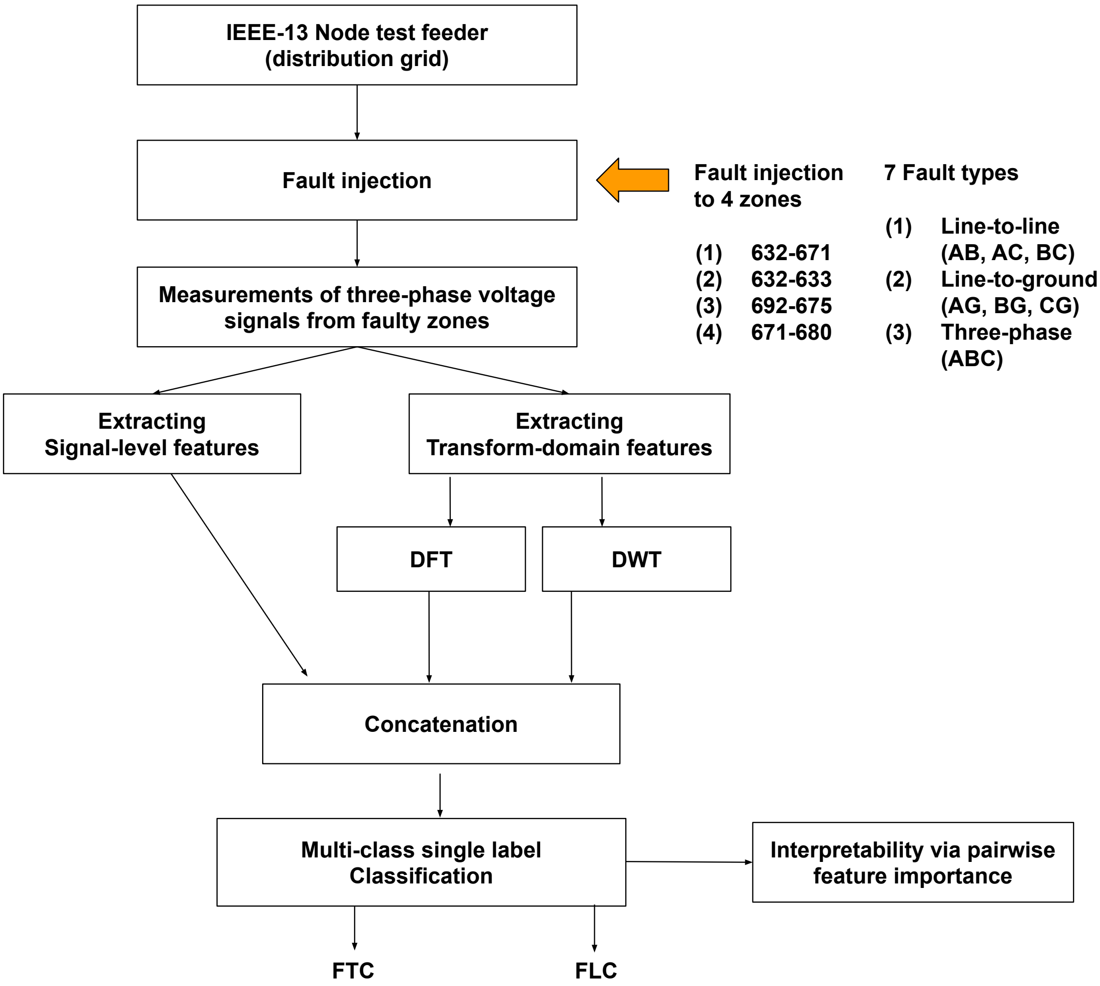

IEEE-13 Node test feeder
(distribution grid)
Fault injection
Fault injection to 4 zones
(1) 632-671
(2) 632-633
(3) 692-675
(4) 671-680
7 Fault types
(1) Line-to-line (AB, AC, BC)
(2) Line-to-ground (AG, BG, CG)
(3) Three-phase (ABC)
Measurements of three-phase voltage signals from faulty zones
Extracting Signal-level features
Extracting Transform-domain features
DFT
DWT
Concatenation
Multi-class single label Classification
Interpretability via pairwise feature importance
FTC
FLC

Figure 3.1: *The pipeline of the proposed system*

were monitored in three-phase mode. Seven different short circuit faults -namely, AG, BG, CG, AB, BC, AC, and ABC-were injected into each zone to serve as inputs for the FTC model. 22 measurements were collected corresponding to 22 resistance values fault resistance R_f values in the range [0.001-2]. For all experimental cases, faults were introduced at a specific start time of $t = 0.01$ and were released at $t = 0.02$; as a result, $t_f = [0.01 - 0.02]$ represents the *faulty period*, and $t_n = [0 - 0.01]$ represents the *normal period*. The overall simulation time was set to $t = [00.022]$ seconds.

All of the features that were taken from the faulty period t_f were normalized by the identical feature that was extracted from the non-faulty (normal) period t_n in order to get relative feature values. Three categories of features from earlier literature were used in this study, each with a specific attention level [1,20,66,82,88]:

- **Time-domain:** it refers to the original data measured in the time domain. Six aggregation functions were applied for the given voltage signal x(t) to produce a feature vector of dimensionality six to represent the time domain feature vector. They include the 1st to 4-th moments: *mean*, *standard deviation*, *skewness*, *kurtosis* together with the *energy* and the *maximum* level of the signal.

- **Discrete Fourier transform (DFT):** Voltage signals were also converted to the frequency domain using discrete DFT under the formula $X(f) = \mathcal{F}(x(t))$, where $\mathcal{F}$ stands for the DFT operation, in order to acquire more detailed information regarding frequency. The computed spectrum was subjected to the same six aggregation functions employed in the time domain, resulting in a feature vector with a dimension of six for the frequency domain signal.

- **Discrete Wavelet transform (DWT):** a digital signal processing method known as DWT applies multi-resolution analysis to signals [50]. The DWT uses a short window at high frequencies and a long window at low frequencies, in contrast to DFT, closely reflecting the properties of the (non-stationary) signals. Approximation A_i and detail D_i wavelet coefficients are present in multi-resolution analysis at decomposition level i. We use five decomposition levels, A_5, $D_{1:5}$, which are motivated by earlier publications [1,88].

There are feature vectors with a dimensionality of 6 for time and DFT domain, respectively. For DWT, We employ 6 (coefficients) $\times$ 6 (aggregation procedures), resulting in a 36-dimensional feature vector for the wavelet domain. In total, 48 (6+6+36) features were collected to represent the features in our labeled training dataset.The following statistics represent features in the DWT domains:

- The maximum value of the coefficients: A_5 and $D_{1:5}$

- The energy of the coefficients: A_5 and $D_{1:5}$

- The first moment (mean) value of the coefficients: A_5 and $D_{1:5}$
- The second moment (standard deviation) value of coefficients: A_5 and $D_{1:5}$
- The third moment (skewness) value coefficients: A_5 and $D_{1:5}$
- The forth moment (kurtosis) value coefficients: A_5 and $D_{1:5}$

3.2.2 Fault Type and Location Classification

Concerning the primary goal, we attempt fault type classification (FTC) and fault location classification (FLC), which are fundamentally multi-class classification issues. We employ a variety of classifiers, including Decision-Tree, SVM, KNN, and Ensemble techniques (Bagged-Tree, subspace k-nearest neighbors).

3.3 Experimental setup

In this section, We thoroughly explain the experimental setup, including the dataset (cf Section 3.1), the training setup, and the classifiers (cf Section 3.2), which are used to verify the effectiveness of the suggested method.

3.3.1 Dataset

As mentioned in Section 3.2.1, The IEEE-13 distribution system is divided into four critical zones for data gathering and constructing the training dataset. The data collection was repeated for 22 different fault resistance values Rf in the range of 0.001 to 2 for each type of fault, as indicated in Table **??**, to augment more data to the training dataset. The total number of training instances developed corresponds to the size of the dataset employed in this study for the empirical evaluation was 4 (zones) $\times$ 7 (faults) $\times$ 3 (phases) $\times$ 22 (resistance values) $= 1848$.

3.3.2 Classifiers and training setup

For FTC and FLP tasks, five different classifiers are applied such as (i) Decision tree (DT), (ii) support vector machine (SVM), (iii) k-nearest neighbors, (iv) ensemble (bagged tree), and (v) ensemble (subspace KNN).

Table 3.1: *characteristic of fault types, locations, and resistances.*

Item	Details
Fault type	phase to ground AG, BG, CG phase to phase AB, AC, BC three phase ABC
Fault location	zone 1 branch 632-671 zone 2 branch 632-633 zone 3 branch 692-675 zone 4 branch 671-680
Fault resistance	0.0010, 0.0273, 0.0535, 0.0798, 0.1061, 0.1323 0.1586, 0.1848, 0.2111, 0.2374, 0.2636, 0.2899, 0.3162, 0.3424, 0.3687, 0.3949, 0.4212, 0.4475, 0.4737, 0.5, 1, 2

For the ensemble (bagged tree) classifier, the learner type was the decision tree, and the number of learners was equal to 30. Likewise, for the ensemble (subspace KNN), the number of learners was set to 30, and the subspace dimension was equal to 18. We employed a hold-out validation (80%-20%) for the training and test sets to expedite the experiments. Table 3.2 demonstrates the precise statistics of the training and test set. We employed MATLAB for feature extraction and classification.

Table 3.2: *IEEE-13 dataset: $|\mathcal{D}|_T$ — total number of data in dataset, $|\mathcal{D}|_{Tr}$ — number of samples in training, $|\mathcal{D}|_{Te}$ — number of samples in testing.*

dataset	$\|\mathcal{D}\|_T$	$\|\mathcal{D}\|_{Tr}$	$\|\mathcal{D}\|_{Te}$
IEEE-13	1848	1478	370

3.4 Results and Discussion

To better understand the merits of the proposed system, we aim to classify the result in two sections (i) classification and (ii) Feature analysis and interpretability. The second part will be analyzed in chapter 6 by answering the following research questions through the course of experiments:

Q1. Which feature classes (domains) impact the prediction task most?

Q2. Which aggregation functions (norm, mean, skewness, etc.) used for the feature representation enhance the classification accuracy the most?

Q3. Which is the most suitable interaction between domains and extracted features?

Classification: Table 3.3 outlines the classification outcomes for the FLC and FTC tasks employing five classifiers. We can point out the outcomes of the three feature classes (time, frequency, and wavelet) for both tasks as follow:

DWT presents the results with the highest classification accuracy on average for all of the experimental instances. The second-ranked method is time-based, and DFT yields the lowest quality. Only with SVM and for the FTC challenge does DFT outperform the time-domain signal in terms of output. This can be explained by the fact that DWT uses a multi-resolution analysis, making it a time-frequency method. Finally, it should be highlighted that combining all of the features results in the classification with the best quality. Thus, the relationship between the quality of various feature descriptors generally holds as follows: ALL > DWT > Time > DFT. Regarding the classifier type, it could be noted that the Ensemble methods typically provide the highest classification quality, followed by SVM, i.e., Ensemble > SVM > Others.

In conclusion, for FLP, (ALL, Ensemble sub-space k-nearest neighbors) yields the best results, with an accuracy of 100%, followed by (DWT, Ensemble) at 99.7%. For FTC, (ALL, SVM) and Ensemble (BT) both achieve the highest accuracy, with 95.4% and 93.5%, respectively.

3.5 Summary

In this chapter, we have addressed the security threats on the electrical grid, representing one of the self-healing features of smart grids. By inserting faults into the IEEE-13 distribution network and gathering data, we first established a large-scale dataset of 1.8K samples. Then, we developed a data-driven methodology to carry out fault location classification (FLC) and fault type classification (FTC) automatically and precisely. Our suggested method is based on a suit of features taken from the time, frequency, and, most crucially, wavelet domains. Additionally, it tests and evaluates the significance of various feature classes by utilizing several cutting-edge classification algorithms. We further investigated how various aggression functions relate to feature representation.

Table 3.3: *Classification accuracy (%) using 48 (6+6+36) features and five classifiers. The first and second most profitable results are shown in Bold and Italic, respectively.*

Domain	**Classifier**	**FLP**	**FTC**
Time	DT	67.5	88.1
	SVM	59.1	90.5
	KNN	58	86.2
	Ensemble (BT)	72.1	88.3
	Ensemble (K)	63.1	88.9
Frequency (DFT)	DT	59.1	82.4
	SVM	59.3	91.6
	KNN	62.3	85.1
	Ensemble (BT)	65	87.3
	Ensemble (K)	59.9	83.2
Wavelet (DWT)	DT	99.2	92.1
	SVM	98.9	93
	KNN	98.6	93
	Ensemble (BT)	*99.7*	*93.5*
	Ensemble (K)	*99.7*	84.8
All	DT	*99.7*	94.3
	SVM	98.6	**95.4**
	KNN	97.3	92.4
	Ensemble (BT)	99.5	94.9
	Ensemble (K)	**100**	84.8

Finally, a unique feature of this research is to present an interpretability analysis for the aforementioned classification problem, in which we highlight how interpretability can illuminate the rationale behind certain decisions made by the ML system. Results are encouraging and demonstrate the benefits of the suggested system to address security vulnerabilities in SGs.

CHAPTER 4

Fault Prediction System Using Visual features based on CNN modeling

Fault diagnosis (fault type and location classification) is crucial in electrical grids due to its significant and economic effects. We suggest using a spectrogram-based representation of the fault signals that can offer higher temporal and spectral resolution by giving 2D space as an input of CNN. Although most of the effort has focused on improving the anticipated accuracy of machine-learning models for fault prediction systems, the interpretability of these systems has gotten less attention than other crucial aspects of this topic. In chapter 6, the visual interpretation of the spectrogram-convolutional neural network will be represented in depth.

4.1 Introduction and Context

The autonomous monitoring of large and complicated electrical power systems has recently drawn much attention from researchers. Correct diagnosis and early detection of developing faults, in particular, have been identified as crucial duties because they help reduce potential severe threats, like extensive power outages across the electrical power grid. As discussed in chapter 1, Cascading failure occurred in the Northeastern

United States and Ontario, Canada, in August 2003, leaving inhabitants without power worldwide for four to seven days. According to reports, the principal cause of the global catastrophe that occurred due to (i) a tree falling on a transmission line, causing a cascade effect, and (ii) the failure of protection systems to maintain the system stability was the short-circuits of a 345kV line.

These illustrations highlight the need for intelligent, fast, and accurate power fault detection systems to assess the grid's condition and perform automatic fault analysis in real time. Over the last decade, the problem of automatic fault monitoring in SGs has been studied from diverse viewing angles, the use of an aggregate of equipment and strategies from computer science, electrical engineering, statistics, and artificial intelligence (AI), mainly using automated data-driven machine learning (ML) algorithms [21, 40, 64, 76].

In PGs, fault monitoring is done to identify and fix one of the following issues:

- Fault detection (FD): determining whether or not a fault occurred within the PG is the goal of fault detection. Relays can isolate the damaged area from the rest of the PG using accurate FD and avoid further damage to the busses in the troubled parts while maintaining power to the healthy sections. FD is a binary classification task.
- Fault zone classification (FZC): where locating the fault's location within the PG network is the goal. The expedition and recovery effort can benefit from this information. FZC is a multi-class classification task.
- Fault type classification (FTC): finding the type of fault that occurred in the network and classifying it appropriately are the objectives of fault type classification. FTC is also a multi-class classification task.

The focus of this chapter is on the FZC and FTC subtasks. This study does not address the topic of FD, which is a binary classification task. Nonetheless, a healthy class is considered one of the classes in the FTC subtask.

The aforementioned fault diagnosis issues have been the subject of extensive investigation. These techniques can be broadly categorized as analytical [27, 69], and data-driven approaches employing ML [13, 71]. Analytical models establish a general framework based on constructing circuit formulation and calculating the circuit parameters. This approach's drawback is that it sometimes requires empirical determination of specific parameters, which may not always be available. However, ML-based methods can speed up this process by quickly accessing and analyzing vast volumes of data about the grid's past and present conditions. The condition for creating precise ML systems is selecting and extracting the target fault classification task's most pertinent properties from faulty signals.

Despite the impressive progress made in the field, the interpretability of current data-driven fault detection systems in PGs is lacking, which is essential for widespread adoption in the energy domain and critical decision-making. In other words, previous "black-box" models were not intended to explain to human operators-who have historically relied on visual awareness-why a particular failure has arisen. It is crucial to create ML models that are more interpretable without compromising prediction accuracy in order to keep people in the control loop. By employing spectrogram-based CNN modeling of fault, which improves prediction performance and allows for the incorporation of prior domain knowledge, we provide a visual explanation of fault detection, which will be considered in detail in chapter 6.

In this chapter, we begin with a representation of the fault signals based on spectrograms, and then we employ two CNN types that have already been trained to extract characteristics pertinent to managing the task at hand automatically. In addition, we overlay the spectrograms with heat maps that visually explain the decisions made by deep learning systems. The contributions of this study include the following:

- **Information representation:** we propose using a spectrogram-based representation of the fault signals since it can offer adequate temporal and spectral resolution than other representations, like a DFT-based model, which tends to provide uniform spectral resolution. According to some studies, because DWT is based on the symmetric kernel [2], it may be less successful at detecting signal asymmetry, which could affect fault types that cause unsymmetrical short-circuits, such as Line to Line (LL), Line to Ground (LG), or

Line to Line to Ground (LLG).

- **Information processing and prediction:** To extract time-frequency properties, spectrogram images were processed using two pre-trained CNN types, *GoogleNet* and *Squeeznet*. As seen in other image classification challenges [35], CNNs can produce excellent outcomes due to the millions of parameters involved in these networks. Before now, these pre-trained models have successfully modeled temporal signals based on spectrograms in fields like music information retrieval [17]. The utilization of Spectrogram-based CNNs allowed us to gain a remarkable classification accuracy for both multi-class fault type classification (FTC) and fault zone classification (FZC) tasks.

- **Simultaneous location and type classification:**we demonstrate the applicability of our proposed approach to suggest joint FTC and FZC, in which the classifier would produce a single score representing both the FTC and FZC, in contrast to most prior work, which focuses on the individual prediction of FTC or FZC, or both but tested in separateness.

- **Visual explanation:** given that machine learning (ML) and deep neural networks are black boxes, we construct an explanation module to shed some light on the system's failure decisions in this study (this topic will be explained in the chapter 6)

4.2 Method

This section illustrates the core contribution of the present work. Figure 4.1 depicts the pipeline of the proposed system. The three following steps make up the main processing actions:

4.2.1 Spectrogram-representation of the data

We employed spectrograms collected from the input voltage signals to generate a time-frequency representation from the measurement signal and utilize the prediction power of deep neural networks. A spectrogram is created by segmenting a time-domain signal into shorter pieces of equal length. Then, each segment is subjected to the fast Fourier transform (FFT). The spectrogram is a graphic representation of the spectrum for each segment. In particular, the computation of the spectrogram entails (1) splitting the signal into segments of length n that overlap equally, (2) windowing each segment, (3) calculating consecutive Fast Fourier-transform (FFT) for each segment, and (4) finally visualizing the power of each segment of the spectrum as an image. Spectrograms can be a valuable tool for illustrating how the signal's non-stationary frequency content changes over time. We primarily use the Spectrograms' visual representation feature as input to deep neural networks for the fault diagnostic task.

The equivalent spectrogram of the original three-phase fault (ABC) voltage signal that is impacted by a three-phase fault is displayed on the left side of Figure 4.1. The visual patterns of the spectrogram before and after fault injection differed, as seen (in the middle of the spectrogram).

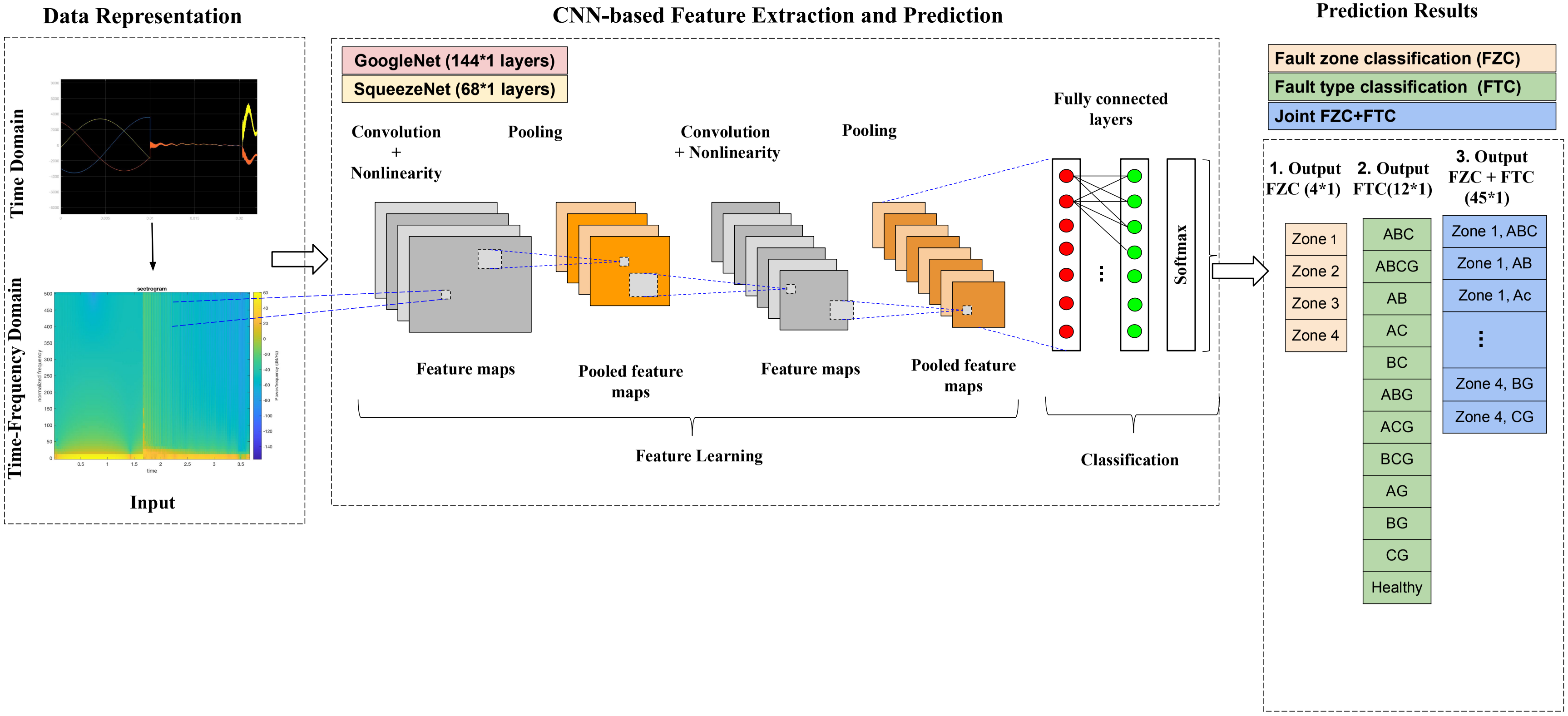


Figure 4.1: *The three main processing steps of our fault diagnosis system.*

4.2.2 CNN-based feature extraction

We utilized two powerful deep pre-trained CNN models, *GoogleNet* and *SqueezeNet*, which have both been used successfully in spectrogram-based modeling of temporal signals, such as music signals in music information retrieval task [17], in order to be able to extract noteworthy features from the visual spectrogram representation (providing both time and frequency information). The following is a description of these networks:

- **GoogleNet**: It is a pre-trained CNN that was developed using the ImageNet database and achieved state-of-the-art performance for classification and detection in the ImageNet Large-Scale Visual Recognition Challenge 2014 (ILSVRC14) [79]. The CNN is based on an inception architecture with 22 deep layers and 144 building blocks.

- **SqueezeNet**: This CNN architecture contains an 18-layer deep network with 68 building blocks that were developed using a smaller CNN architecture [35] and trained on ImageNet. This network has demonstrated the ability to learn characteristics from many photos in different categories. In this network, an input image has a dimension of 227 by 227 (pixel × pixel).

Figure 4.1 illustrates how CNNs deal with I Feature Learning and (ii) Classification tasks. Convolution and pooling are combined for feature learning. Convolutional layers build feature maps that list the presence of particular features (fault features) in the input signal by repeatedly applying learned filters to input spectrogram images. The feature maps' dimensionality is decreased by adding a pooling layer after the convolutional layer. As a result, the network needs to learn a smaller set of parameters. Feature classification uses fully connected layers that essentially act as a classifier and assigns a probability for the input image being one of the fault classes (location or type).

In both deep networks, the Softmax function is applied to the output of Fully connected layers, where the procedure is defined for multi-class classification according to:

$$P(y=j|x) = \frac{e^{x^T w_j}}{\sum_{k=1}^{K} e^{x^T w_k}} \tag{4.1}$$

where $0 \leq P(y = j|x) \leq 1$ is the conditional probability of classifying a given instance input vector x as $y = j$ and w being the weighting vector. The classification layer, which assigns each input to one of the k classes by using the cross-entropy function as below [9], receives the output of the Softmax layer as its input in the final step.

$$loss = -\sum_{i=1}^{N}\sum_{j=1}^{k} t_{ij} ln y_{ij} \tag{4.2}$$

in which k refers to the number of classes ($k = 12$ fault types, and $k = 4$ for fault locations) and N is the number of samples. y_{i_j} indicates the value from the output of the Softmax function for ith sample belonging to the jth class.

4.2.3 Fault diagnosis

We address two significant issues in fault diagnosis in the currently proposed work, namely fault zone classification (FZC) and (iii) fault type classification (FTC), which are both addressed in Section 1. We also added a combined prediction challenge for the two sub-tasks. We use a multi-label method with a one-hot encoding scheme to describe the result. There are 4, 12 (11 + 1 for healthy), and 45 $(11 \times 4 + 1 = 45)$ classes in each of the three subtasks, FZC, FTC, and joint. Note that since no zone could be associated with the healthy class (i.e., zones are only taken into consideration for faults), we count the healthy class as a single class, which brings the total number of classes to $11 \times 4 + 1 = 45$. The algorithm will not check for its location if the model discovers a healthy class.

4.3 Experimental setup

In this section, we explain the experimental setup, including the dataset, data collection, training setup, and baselines used to confirm the effectiveness of the suggested fault zone and type classification system. In our simulation of the IEEE-13 test node feeder using MATLAB, we used the default parameters, a voltage frequency of 60 Hz, and a sample time of 10^{-5}.

Table 4.1: *Specification of fault types, locations and resistance values used in the simulations*

Item	Details
Fault type	phase to ground AG, BG, CG phase to phase AB, AC, BC phase to phase to ground ABG, ACG, BCG three phase ABC three phase to ground ABCG
Fault location	zone 1 branch 632-671 zone 2 branch 632-633 zone 3 branch 692-675 zone 4 branch 671-680
Fault resistance	0.0010, 0.0273, 0.0535, 0.0798 0.1061, 0.1586, 0.2111, 0.2374 0.2899, 0.3162, 0.3424, 0.3949 0.4475, 0.5, 1

4.3.1 Data and Training

We injected 11 fault types with 15 different resistances for each type of fault to these four critical zones adjacent to load flow buses number 671, 633, 675, and 680 (These zones are highlighted in Figure 4.2). This information is summarized in Table 4.1. Chapter 7 has more specifics concerning the condition with Simulink and the collected dataset.

Following data collection, *GoogleNet* and *SqueezeNet*, two separate pre-trained deep CNNs, are employed. As indicated in Table 4.2, several hyperparameters are applied to *GoogleNet*, *SqueezeNet*, and CustomCNN to achieve an ideal CNN structure for the training process. A hold-out validation (70%-30%) was utilized for the training and test sets. MATLAB R2020a was used to implement and validate the system. Codes are available are made available.[1]

Table 4.2: *Hyper-parameters utilized in the deep models (pre-trained CNNs and CustomCNN) for training the classification models analyzed in this research (FZC and FTC).*

Hyperparameter	GoogleNet	SqueezeNet	CustomCNN
Initial learning rate		[1e-3, 1e-4, 3e-3, 3e-4]	[1e-3, 1e-4]
Bach size	[128, 256]	[32, 64]	[64, 128]
Max epochs		[15, 25]	
Number of layers	144	68	4
Number of params.	7M	1.24M	A few K

[1] `https://github.com/atenanaz/FaultClf_SmartGrids`

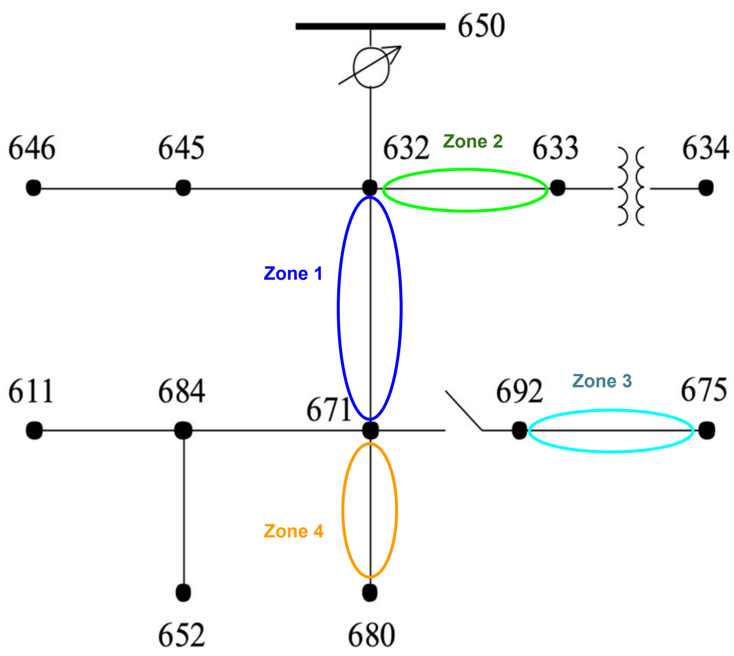


Figure 4.2: *One-line diagram of the IEEE-13 node test feeder with highlighted four selected zones.*

4.3.2 Baseline

By examining several features taken from the time, DFT, and DWT domains and utilizing a collection of well-known classifiers, we evaluated the effectiveness of our system against reliable baselines. Three categories of features, (i) temporal (time-based), (ii) frequency (based on DFT), and (iii) wavelet domain, represent the state-of-the-art in the field, and they were used as the basis for feature extraction.

This study investigated the effects of several statistical aggregation functions on feature extraction, including the probability distribution n-th moment ($n \in [1, 4]$)) as well as the *energy* and *maximum* level of the signals. As a result, the time and frequency domain feature vectors were six dimensions. We collected 6 (coefficients) $\times$ 6 (aggregation procedures) for the DWT, resulting in a 36-dimensional feature vector for the wavelet domain. Therefore, we examined a total of $6 + 6 + 36 = 48$ features that were gathered from the time, DFT, and DWT domains. Finally, for the FTC, FZC, and joint (FTC+FZC) tasks, five different classifiers were employed. They include (i) decision tree (DT), (ii) support vector machine (SVM), (iii) KNN, (iv) ensemble, and (v) Multi-layer perceptron (MLP) besides (vi) CustomCNN with four layers (with 64 and 32 neurons respectively used in the hidden layer). Similar to the splitting used for assessing our system, we employ a hold-out validation (70%-30%) while producing training and test sets.

4.4 Results and Discussion

This section reports the experimental results of three main tasks of fault diagnosis of the proposed system: FTC, FZC, and joint FTC+FZC. Finally, we compared the proposed system with robust baseline methods, as summarized in Table 4.4.

4.4.1 Fault zone classification (FZC)

The first thing that becomes apparent when looking at the data presented in Table 4.3 is that the values obtained for FZC are substantially larger than those for the other sub-tasks, namely FTC and FTC+FZC. This can be explained by the fact that the FZC task has fewer classes (4 classes) than the other sub-tasks, such as FTC (12 classes) and FTC+FZC (45 classes), making the task of the classifier a simpler one. We can see that at epoch 25, *SqueezNet* achieves the maximum level of accuracy, 85.3%, while *GoogleNet* achieves 85.1% (with a longer training time).

4.4.2 Fault type classification (FTC)

The best result for FTC was obtained for *GoogleNet*t at epoch 25, with the classification accuracy equal to 59.4%. However, *SqueezeNet* trained in a remarkably short time and reached an accuracy score of 58.4% at epoch 15.

4.4.3 Joint type-location prediction

In more detail, we can see that *GoogleNet* (57.2%) at epoch 25 achieved the most significant results for the joint type-location classification by looking at the differences between the results of the two-deep networks. Despite this, *SqueezeNet* has managed to achieve its highest level of accuracy (54.4%) at epoch 15 and a training duration that is three times

Table 4.3: *Classification accuracy (%) using spectrogram on GoogleNet, GoogleNet, and CustomCNN along different epochs. The training times were obtained on a regular machine. The could be decreased if performed on a high-speed machine equipped with GPUs. The comparison between the models' training time, however, remains valid.*

		GoogleNet	SqueezeNet	CustomCNN
FZC	best epoch	25	25	15
	best accuracy	85.1	**85.3**	84.2
	training time	10.29 h	3.92 h	8.43 h
FTC	best epoch	25	15	15
	best accuracy	**59.4**	58.4	56.9
	training time	11.26 h	2.64 h	9.01 h
FZC+ FTC	best epoch	25	15	15
	best accuracy	**57.2**	54.4	56.6
	training time	11.12 h	3.82 h	12.93 h

slower than *GoogleNet*.

Summary. in general, *SqueezeNet* looks to be the best choice for the FTC and FZC subtasks when expected accuracy and training time are considered, as it proposes high classification accuracy with a remarkably shorter training period. However, *GoogleNet* can be the best solution if almost precision is necessary. *GoogleNet* and CustomCNN seem to be the top alternatives for joint prediction tasks in this case.

4.4.4 Comparative evaluation under baseline

As shown in Table 4.4, we compared the proposed system to strong baseline techniques representing state of the art in ML-based prediction techniques. The wavelet domain (DWT) produces, on average, features with the highest classification accuracy for all three experimental scenarios, regardless of the classifier type, making it the best domain among the time, frequency, and wavelet domains. The cause may be related to the fact that DWT uses a multi-resolution analysis of the signal, making it appear to be a time-frequency technique. We have also included a CustomCNN with four layers (the hidden layer uses 64 and 32 neurons), whose hyper-parameters are displayed for a fair comparison with Table 4.2.

The data indicate a difference between the best outcomes from the suggested method and the baseline. For instance, *SqueezeNet* outperforms all other baselines with a maximum performance level of 85.3% in the FZC task. Further, both *GoogleNet* and *SqueezeNet* outperform CustomCNN, while *SqueezeNet* acts as the best model from a computational point of view.

On the other hand, *GoogleNet* and *SqueezeNet* both outperformed the baseline for the FTC test, and *GoogleNet* scored the highest accuracy (59.4%) in the proposed system. Likewise, for joint type-location classification, the best result was obtained by *GoogleNet* with an accuracy of 57.2%. Additionally, CustomCNN outperformed *SqueezeNet* in accuracy while taking much longer to train, which is time-consuming.

4.5 Summary

The classification of fault type, fault zone, and joint type-location for power grids are issues that we addressed in this research by proposing a spectrogram-based CNN system (PGs). The proposed approach relies on robust deep convolutional neural networks to extract visual elements from the **spectrogram** images, reflecting time and frequency characteristics of faulty signals, and classify them according to their origins, type, and locations. We displayed the competitive results produced utilizing two cutting-edge pre-trained CNNs, *GoogleNet* and *SqueezeNet*, compared to the pre-existing baseline employing DFT, DWT, and CustomCNN.

Table 4.4: *Classification accuracy (%) using 48 (6+6+36) features and five classifiers in three different domains, and CustomCNN classifier for spectrograms as a baseline for comparison with the most promising results of the proposed technique.*

		FZC	**FTC**	**FZC+FTC**
Input	Classifier & DeepNet	accuracy	accuracy	accuracy
Time	DT	58.5	43.6	25.7
	SVM	59.6	49	35.4
	KNN	70.4	55.9	42.7
	Ensemble	69.7	55.6	41.9
	MLP	57.1	51.1	32.9
Frequency (DFT)	DT	55.5	43.1	25.1
	SVM	55	45.6	31.5
	KNN	67.1	51.6	37.9
	Ensemble	66.7	50.8	37.7
	MLP	53.4	45.6	29.6
Wavelet (DWT)	DT	79.2	42.8	26
	SVM	81.6	49.3	49.2
	KNN	84.2	55.3	53.9
	Ensemble	83.1	48.8	45.6
	MLP	82	50.4	47.3
Spectrogram (proposed)	GoogleNet	85.1	**59.4**	**57.2**
	SqueezeNet	**85.3**	58.4	54.4
	`CustomCNN`	84.2	56.9	56.6

CHAPTER 5

Adversarial Machine-learned Attack in Smart Electrical Grid

In smart electrical grids, fault prediction tasks such as fault detection, fault type, and fault location classifications are vital due to their enormous economic and functional consequences. Several smart grid applications have utilized data-driven methodologies, including fault detection and load forecasting. Nevertheless, the robustness and security of these data-driven algorithms have not been adequately investigated for all power grid applications. This chapter addresses the challenges raised by the security of machine learning applications in the smart grid. First, we demonstrate that adversarial perturbation can damage the smart grid's deep neural network approach. We underline how research on fault localization and type classification accentuates the vulnerabilities of present machine learning algorithms in the smart grid to a variety of adversarial attacks.

5.1 Introduction and context

According to a statement released by the World Health Organization, at least one in every ten patients experiences suffering due to inadequate infrastructure security. Instability or inadequate distribution of electrical energy can directly influence people's lives and societal well-being. The current study is concerned with the "security of power grids", which serve as the country's critical energy infrastructure (CEI) [60].

Under the smart grid (SG), conventionally-operated electrical grids have undergone significant revisions and upgrades regarding dependability, robustness, and efficiency. One of the most critical components of SGs is their application in fault detection, fault classification, and routine examination of the underlying disruptions that trigger the failures. Power grid networks are inherently vulnerable to physical damage. Electrical faults can be caused by natural disasters such as tree or bird contact, lightning, or aging of the equipment [70]. Additionally, due to a lack of insulation around the cables, these faults may frequently occur in High Voltage Transmission Lines. Large-scale cascading consequences from power system breakdowns might have a disastrous effect on the nation's economy and security. As a result, for the Electric Power Supply industry and the overall security of CEI, rapid fault detection and classification with a high degree of fidelity is a vital service.

The classification of faults and the locations where they occur are the main topics of this chapter. While the primary goal of fault type classification (FTC) is to identify the fault's type class, fault zone classification (FZC) seeks to identify the zone (or occasionally the precise location) in which the fault has occurred. In both transmission and distribution systems, voltage sags are the primary cause of faults, which can appear as asymmetric (phase-to-phase (LL), single-phase-to-ground (LG), or two-phase-to-ground (LLG) faults or symmetric (three-phase-to-ground (LLLG or LLL) faults) [1, 75]. Previous literature has utilized a combination of tools and approaches from electrical engineering, signal processing, and artificial intelligence (AI) [21, 71, 75] to solve the above fault classification tasks. Due to the vast volumes of data covering energy networks, machine-learned (ML) models, particularly those based on deep learning, have seen an increase in adoption in the present infrastructure of power systems.

Notwithstanding their great performance, the complexity of existing (deep) inference methods could be their undoing. Adversarial attacks can use their weaknesses to vulnerable the confidentiality, integrity, or

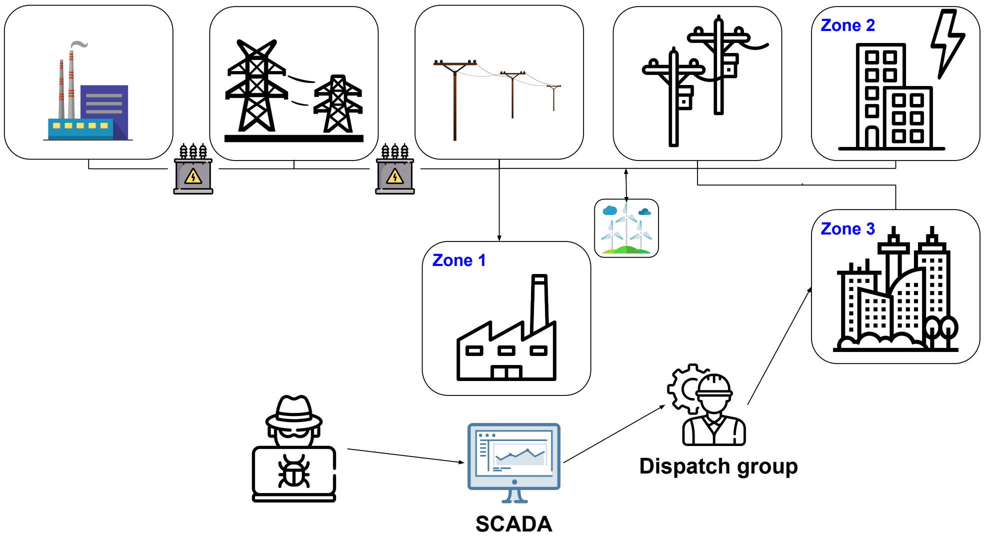


Figure 5.1: *A hypothetical illustration of targeted adversarial attacks against fault zone prediction in smart grids. A fault location prediction system was subjected to an adversarial attack, and as a result, recovery groups were unintentionally sent to zone 3 when they should have been in zone 2.*

accessibility of SGs (aka the CIA triad) [24, 89]. *Adversarial examples*, small but deliberate perturbations intended to make a machine learning model generate incorrect results, are used to operationalize adversarial attacks (e.g., to mis-classify an input sample).

The following is an example of a motivating scenario. As shown in Figure 5.1, Here is an illustration of a motivating scenario. An attacker can attack the fault prediction system used in supervisory control and data acquisition networks (SCADA) [19] by breaching the SG system's communication network. The attacker's objective is to launch a targeted adversarial attack, i.e., to cause the ML model employed in the SCADA's fault classification system to misclassify an input sample into a known but erroneous class. To accomplish this goal, in the FZC scenario, the attacker selects an (illegitimate) target class label that can cause more significant damage and suffering by prolonging the expedition and recovery effort. In other words, the attacker could activate a false positive signal and guide the rescue team to more difficult-to-reach and truly faultless and densely populated regions. Similarly, in the FTC example, the attacker might fool the system into predicting a simpler-to-repair fault type when the fault is more complicated to repair. All these examples and scenarios motivate that adversarial attacks if left unchecked, can potentially cause catastrophic harm to society owing to their often impenetrable nature.

Our key contributions are summarized as follows:

- We investigate the consequence of adversarial attacks against several critical fault classification problems, namely fault type classification (FTC), fault zone classification (FZC), and their combination on a widely used dataset based on the IEEE-13 test node feeder with renewable energies;
- We analyze adversarial attacks by examining multiple experimental situations with different adversarial goals (targeted vs. untargeted), attack models such as FGSM and C&W, and comparing them to random noise and the baseline model (unattacked);
- Empirical experiments on a widely adopted dataset based on the IEEE-13 test node feeder with renewable energies (more explanation in chapter 7) indicate that adversarial attacks can degrade the quality of classification significantly;

5.2 Security in Smart Grid

With the increasing development and widespread use of machine learning applications, it is vital to discover the adversarial vulnerabilities of intelligent systems driven by these models. Recently, interest in adversarial machine learning (AML), a subject that analyzes the security of machine learning models under attack and from a defense viewpoint, has risen.

We can classify the distinguishing characteristics of attacks against machine-learning (ML) systems according to the following dimensions [83]:

- **Attacks Timing.** This concerns the time in the ML pipeline where the attack is applied is considered. A decision-time attack (evasion attack) attempts to alter the output of a model by introducing adversarial samples that are precisely constructed and contain negligible human-imperceptible perturbations. A training-time attack (also known as a poisoning attack) alters the training data by inserting erroneous data points. The terms evasion and poisoning relate to their respective mechanisms of operation, i.e., “evading the classifier decisions” and “adding poisons to training data”, respectively.

- **Attacks Information.** A distinction is made between white-box and black-box attacks. In the first case, the attacker has full knowledge of the target model. In the other case, the attacker only has partial or no knowledge of the target model. A white-box attack is thought to be more potent than a black-box one. Therefore, as a conservative measure, investigating the security of SGs under white-box attacks is deemed more important.

- **Attacks Goals.** Adversarial attacks may be directed at creating a specific misclassification, such as causing a trained function f to predict an erroneous label l on an instance x (targeted attack), or they may be directed at causing a generic misclassification (untargeted attack).

5.2.1 Adversarial examples and attack on SG

[15] concern the vulnerability of machine learning algorithms used in building load forecasting and power quality disturbance investigation against specific adversarial attacks such as [25] developed a generative-adversarial system for partially labeled samples, named semi-supervised generative-adversarial learning (GBSS). Generative-adversarial learning aims to construct a semi-supervised learner resistant to attacks and faults. In [58], adversarial attacks on convolutional neural network-based event causes for three different power grid events (i) line energization, (ii) capacitor bank energization, and (iii) fault prediction were given. The fast gradient sign technique (FGSM) was employed to create false voltage or current data. The level of the opponent of the FGSM is further compared using the Jacobian-based Saliency Map Attack (JSMA). The performance of the CNN classifier against specific threats is enhanced through adversarial training. According to [77], voltage stability is evaluated using adversarial instances produced utilizing methods like FGSM, PGD, DeepFool, Universal Adversarial Network (UAN), as well as Universal Adversarial Perturbation (UAP) (UAP). Adversarial training is used to protect against these adversarial examples.

5.2.2 False data injection (FDI) attacks on SG

False data injection was considered an adversary in particular literature, and a defense mechanism against this attack was originally presented in the smart grid area [3]. For instance, [31], the authors examined the benefit an attacker can obtain by attacking power measurements. Using zero-sum game theory, they quantified the gain received by injecting false data points into the smart grid. [81] proposes a joint attack (attack within an attack). They argued that deep learning technologies presented as detectors are vulnerable to adversarial attacks. In this context, they propose a joint adversarial example and FDIA (AFDIA) by perturbing the neural attack detection employed in FDI.

5.3 Method

We have performed adversarial attacks against two machine-learned fault classification tasks in smart electrical grids, which serve as the core attack target. The attacks are conducted as non-targeted and targeted. This section discusses our strategy in depth.

5.3.1 Problem definition

Adversarial task. Given a training dataset $\mathcal{D}$ of n pairs $(x, y) \in \mathcal{X} \times \mathcal{Y}$, where x is the input sample, and y is its corresponding class label, the classification problem is formulated as finding a target function $f_\theta : \mathcal{X} \to \mathcal{Y}$ that can predict the class label y surroundings the input sample x, where θ is the model parameter. The goal of the adversarial attacks is to find a non-random perturbation δ to produce an adversarial example $x^{adv} = x + \delta$ such that it can induce an inaccurate detection (e.g., misclassification). The methods by which $delta$ is learned are referred to as *adversarial attacks*, and they can be either targeted or untargeted.

Definition 4 (Targeted adversarial attack)**.** *Given a trained classifier $f(\boldsymbol{x}; \theta)$ and a test instance from the dataset $\boldsymbol{x}_0 \in \mathcal{D}$ where $f(\boldsymbol{x}_0; \theta) = y_0$, the goal of a targeted attack is to perturb $\boldsymbol{x}_0$ with a small budget $||\delta|| \leq \epsilon$ such that the perturbed sample would be mis-classified to the target label $y_T \neq y_0$, referred to as the mis-classification label. The problem can be* *represented using an unconstrained optimization problem formulation*

$$\underset{\delta:\|\delta\|\leq\epsilon}{min} \quad \mathcal{L}(f(\boldsymbol{x}_0+\delta;\theta),\ y_T) \tag{5.1}$$

One can note that in this case, here the attacker aims to ***minimize*** *the distance (loss) between the adversarial prediction $f(\boldsymbol{x}_0+\delta)$ and the mis-classification label y_T.*

□

Definition 5 (Untargeted attack)**.** *The goal of the attacker in untargeted attack is to cause any mis-classification to maximize the loss between the adversarial prediction and the legitimate label y_0*

$$\underset{\delta:\|\delta\|\leq\epsilon}{max} \quad \mathcal{L}(f(\boldsymbol{x}_0+\delta;\theta),\ y' \neq y_0) \tag{5.2}$$

as such, it is clear that the attacker's objective in this scenario is to cause any mis-classification y', regardless the of the specific type.

□

5.3.2 Fault Classification in Smart Grids

In this study, we explore different multi-class classification problems pertinent to fault prediction in smart grids with $K \geq 2$ classes in this paper, in which X is the input space and $y = \{1, 2, ..., K\}$ the output space. The two goal labels for the issues at hand in our scenario are (i) fault location and (ii) fault type. Therefore, the main task is split into three sub-tasks:

1. Fault location classification (FLC): with $K = 4$ the task seeks to classify a given signal into its originating zone.
2. Fault type classification (FTC): with $K = 11$ the task aims to classify a given signal into one of the predefined fault types.
3. Joint location and type classification (FLC+FTC) $k = 44$ combining both fault class labels in the preceding cases.

where, (1) and (2) are explicitly contained in the dataset, while (3) is obtained by combing each different possible combination of task 1 and task 2. Therefore, we can state that the joint task is expected to be more complex than the former.

5.3.3 Adversary threat model

Before examining the effects of adversarial attacks, we explain the adversary threat model provided. The adversary's assumption entails:

- **Adversary goal.** The adversary wants to deploy untargeted and targeted assaults to misclassify smart-grid fault classification tasks in each of the three FZC, FTC, and joint sub-tasks. In the targeted situation, the purpose may be to produce more difficult-to-reach or difficult-to-resolve (mis-classification) labels to obstruct or delay the recovery of the task.

- **Adversary knowledge.** We presume that the attacker is operating in a *white-box* environment and is fully aware of all the feature extraction model's input and output parameters and the perturbation they wish to estimate. In addition, the attacker has full access to the input features that would be changed due to the attack. The attacker can also obtain the class labels in targeted attack scenarios.

5.4 Experimental Evaluation

We analyzed adversarial attacks against smart grids on a dataset acquired from IEEE-13 test node feeder with renewable energies. The experimental setup is presented below.

5.4.1 dataset

The MATLAB Simulink environment was used to inject short-circuit faults into an IEEE-13 node test feeder to collect data and create the training dataset for fault classification in smart grids [1, 73, 75] . Renewable energy sources like solar systems and wind turbines were included in the node feeder. We divided the network into four zones, adjacent to four load flow buses (numbered via 671,633, 675, and 680, see [59]), and measured the three-phase voltage signals. We applied 11 short circuit faults to four specified zones in the IEEE-13 network. These faults cover every conceivable short-circuit fault. To ensure having a sufficient number of samples in the training dataset, each fault was generated with 22 different fault resistance values [34, 73]. Our final training dataset contained 4 (zones)× 4 (measurement-zone) × 11 (faults) × 3 (phases) ×

22 (resistance values) $= 11616$ samples. Dataset will be described in depth in the chapter 7. Note that we collected (measured) signals from 4 locations regardless of locations, and after feature extraction (see below), we stacked them together to create a super-vector fed into the neural network ML model.

The time series signals were represented as discrete features retrieved from the time, frequency, and wavelet domains utilizing temporal, Discrete Fourier transform (DFT), and Discrete wavelet transform (DWT) analysis, as previously investigated [8, 69]. After that, we extract six features from each domain related to energy, maximum, and the 4-th moment of their probability distribution functions (PDFs) (e.g., mean, norm, skewness, kurtosis). The overall size of the feature vectors utilized in the learning model is 48, divided into 6 (time) $+ 6$ (DFT) $+ 36$ (DWT), where we employed 6 (coefficients) 6×6 (aggregation operations) for the DWT features, resulting in a 36-dimensional feature vector.

5.4.2 Adversarial Attacks

The implemented attacks consist of the fast gradient sign method (FGSM), basic iterative method (BIM) [43], and Carlini and Wagner (C&W) [11]. FGSM is a white-box attack that utilizes the sign of the loss function's gradient to learn adversarial perturbations, and BIM is the iterative version of the FGSM. In the untargeted scenario, FGSM aspires to generate a perturbation that maximizes the training loss formulated as

$$\delta = \epsilon \cdot \mathrm{sign}(\bigtriangledown_x \ell(f(x;\theta), y)) \tag{5.3}$$

where ϵ (perturbation level) represents the attack strength and $\bigtriangledown_x$ is the gradient of the loss function w.r.t. input sample **x**, y is the legitimate label and $\mathrm{sign}(.)$ is the sign operator. A targeted FGSM attack is, instead, formulated as

$$\delta = -\epsilon \cdot \mathrm{sign}(\bigtriangledown_x \ell(f(x;\theta), y_T)) \tag{5.4}$$

in which the goal of the attacker is maximize the conditional probability $p(y_T|x)$ for a given input x.

The second category of adversarial attacks is Carlini and Wagner. It is a powerful attack model for finding adversarial perturbation under three various distance metrics (ℓ_0, ℓ_2, ℓ_∞). Its key insight is similar to L-BFGS [80] as it transforms the constrained optimization problem into an empirically chosen loss function to form an unconstrained optimization problem as

$$\min_{\delta} \left(\|\delta\|_p^p + c \cdot h(\mathbf{x} + \delta, y_T) \right) \tag{5.5}$$

where $h(\cdot)$ is the candidate loss function. □

The C&W attack has been used with several norm-type constraints on perturbation among which the ℓ_2 and ℓ_∞-bound constraint has been reported to be most effective [11].

5.4.3 Fault Classification

Model and training details. For the three classification tasks listed in Section 5.3.2, we trained a multi-layer perceptron (MLP), a type of deep neural network. An input layer, two dense layers, and an output layer make up the model. As its number of neurons must match the number of output classes in each task, the latter is the only layer that varies throughout the three tasks. Separate training stages are required for each task, and they are all conducted using the same settings: 500 epochs, the Adam Optimizer, a fixed learning rate of 10e-3, and a batch size of 20. The hyper-parameters were obtained after fine-tuning.
Implementation of the attacks. We employed the IBM Adversarial Robustness Toolbox to accomplish the adversarial attacks due to its compatibility with Keras and wide offer of suitable attacks for a deep learning model. The performed attacks consist of FGSM, multi-step (BIM), and C&W attacks. These attacks were conducted in both untargeted and targeted scenarios.

5.5 Results and Discussion

Through the course of experiments, we want to answer the following evaluation questions to understand better the efficacy of the researched adversarial attacks against the fault classification system in SGs.

RQ 1: Compared to random noise, how successful are adversarial perturbations produced by various adversarial attack methods (FGSM, BIM, and C&W) against the three fault classification tasks in SGs provided in Section 5.4.3?

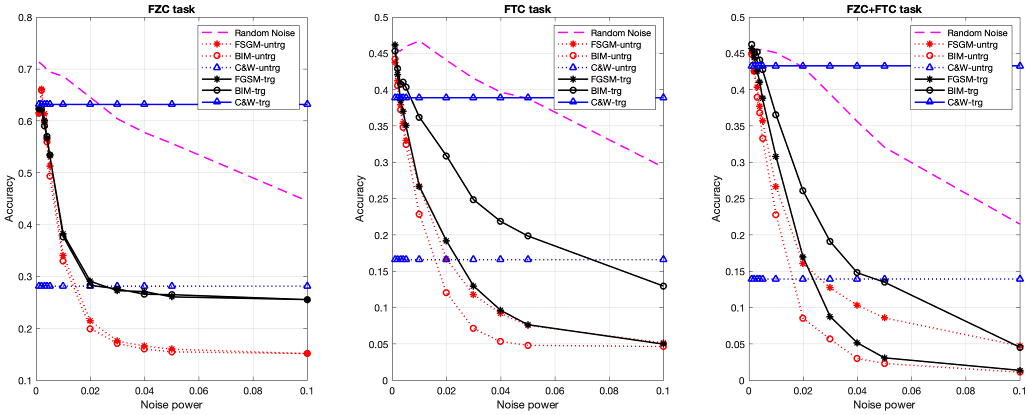


Figure 5.2: *Three tasks under targeted and untargeted adversarial attacks. Classification accuracy for* $FZC = 0.7134$*,* $FTC = 0.4569$*, and* $FZC + FTC = 0.4543$*. Best results for C&W were obtained under* ℓ_∞ *for untargeted attacks and* ℓ_2 *for targeted attacks. Note that the starting point of noise power for all attacks and random noise is* 0.001.

RQ 2: How does the performance of attacks change when we alternate between the **attack targets**?

Discussion. We begin our experimental study by addressing the evaluation questions mentioned above.

Answer to RQ 1. This research question examines whether using adversarial attacks on fault classification systems (FZC, FTC, and joint) affects how the ML models behave. Figure 5.2 demonstrates that, across three tasks and under various noise levels (ϵ), all analyzed adversarial attacks - FGSM, BIM, and C&W - have a significantly more prominent effect than random perturbation, with the impact expanding as the perturbation budget grows. Comparing the strength of the three adversarial attack models, BIM is the strongest in all tasks. For instance, in the situation of (untargeted, FTC) with an attack budget (noise level) equal to $epsilon = 0.04$, BIM untargeted adversarial attack accuracy reaches 0.05, while FGSM and C&W reach 0.09 and 0.16, respectively, under the same scenario. BIM and C&W are more affected by attack targets (targeted vs. untargeted) than FSGM. For example, for the FTC ($\epsilon = 0.04$), the classification accuracy is 0.21 vs. 0.05 (BIM-untargeted vs. BIM-targeted), however for FGSM the corresponding difference is only 0.1 vs. 0.09 (FGSM-untargeted vs. FGSM-targeted).

In summary, the strengths of the assaults can be contrasted using

BIM>C&W>FGSM *(the first being the strongest). Only textbfC&W-targeted deviates from the pattern and performs poorly, while* ***C&W-untargeted*** *performs sufficiently in all of the scenarios that were investigated.*

Answer to RQ 2. This research question concerns how the effectiveness of various adversarial attacks differs across smart grid fault prediction tasks and explores whether task complexity affects the results.

Starting with three tasks, we evaluate the attacks' overall strength. At $\epsilon = 0.04$ the power of attacks FGSM-untargeted, BIM-untargeted, C&W-untargeted, FGSM-targeted, BIM-targeted, C&W-targeted is equal to 0.166, 0.160, 0.281, 0.271, 0.265, and 0.631 respectively. Thus, w.r.t the base ML model (0.713), we may remark a relative degradation of 329% , 345% , 153% , 163%, 168%, and 13%. The equivalent relative degrading power of attacks for FTC task are 396%, 756%, 175%, 374%, 108%, 17% and for the joint FZC+FTC task include 339%, 1408%, 226%, 779%, 206%, 4.9%. As a result, FZC = (275.6%, 114.6%), FTC = (442.3%, 166.3%), and FTC = (657.6%, 329.9%) are the average degradation powers for (untargeted, targeted) goals. It is possible to observe that when a task becomes more challenging, both untargeted and targeted attack models perform better (are stronger).

In summary, the result of empirical evaluation shows that the difficulty of the fault prediction tasks (in SGs) impacts the effectiveness of the investigated adversarial attacks, which means that the attacks are better able to manipulate the decision outcomes following ***FZC+FTC>FTC>FZC****.*

5.6 Summary

In this chapter, we analyzed the security and vulnerability of deep-learning-powered machine learning (ML) models applied for fault predictions systems such as FTC, FZC, and FTC+FZC under adversarial attacks. We attacked the fault prediction systems using three distinct adversarial assaults, FGSM, BIM, and C&W, with various attack targets, including targeted and untargeted attacks. We can conclude that adding a small amount of noise to the data can have a significant impact on the quality of fault classification systems, particularly the complicated model. Future research should examine these vital systems in depth against further adversarial threats and defend them with adversarial training and detection techniques.

CHAPTER 6

Explanation in Fault Prediction System

6.1 Introduction and context

Why Explainable AI?

Machine learning (ML) is increasingly being utilized in critical infrastructures (CI), such as criminal justice and healthcare, for prediction applications that have a substantial impact on the lives of individuals. Many machine learning (ML) models are black boxes with insufficient justification for their decisions [68]. Interpretable and explainable ML techniques are necessary to design comprehensible machine learning systems, i.e., systems that can be comprehended by a human mind, as well as to comprehend and explain predictions generated by sophisticated models, such as deep neural networks [52]. According to [52], explainable ML attempts to provide post-hoc explanations for already-existing black box models or proprietary models that are incomprehensible to humans, while interpretable ML focuses on constructing models that are intrinsically interpretable.

In the context of smart grids, the literature is aligned by the tendency of empirical experiments to concentrate on predicting the *accuracy* of fault prediction, seeking an answer to questions such as "Is it possible to identify a fault using ML techniques accurately?" or "Which classification method can more accurately predict a fault class type?" Sadly, these trends toward total automation of the SG self-healing capability are not meant to warn human operators, who typically rely on manual/visual awareness. To keep humans in the control loop, it is critical to develop *Explainable* ML models that can replace these black-box prediction models and provide rules that can be understood with a little investigation.

Another area of interest that this thesis looks at is the *interpretability* of existing ML models that are usually black-box. Novel techniques have been developed and existing ones refined to keep human operators informed about why certain decisions were made. To the best of our knowledge, the latter aspects have been rarely considered in the previous literature, and thus this chapter pushes the current literature one step toward a transparent, accountable, and suitable platform, which is in line with the recent General Data Protection Regulation (GDPR). Methods for ML interpretability and explanations applied in this chapter are classified in Figure 6.1. It is noted that this chapter goes through details of interpretability and other parts of the work described in a nutshell or represented in the mentioned chapter.

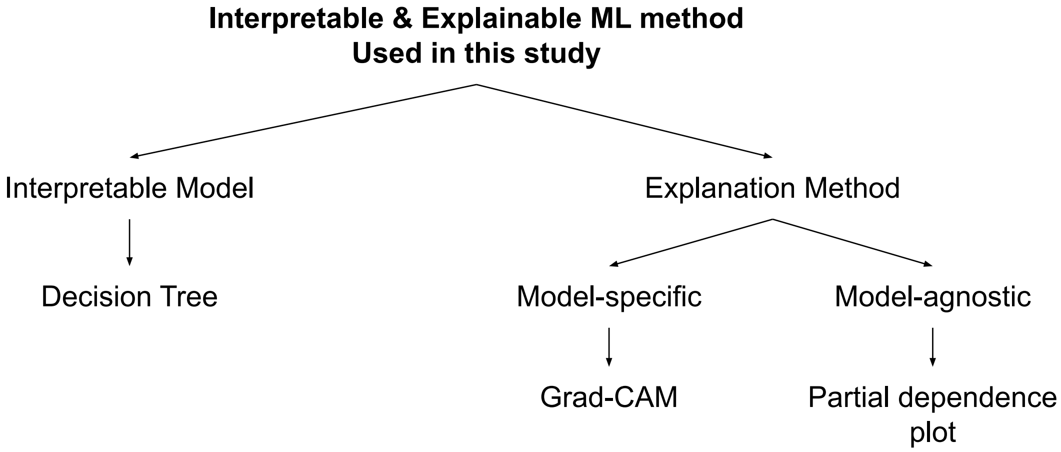


Figure 6.1: *Interpretable and explainable method used to keep human-operatores informed in SG.*

6.2 Feature-learned Interpretability

6.2.1 Explainability with partial dependence plots (PDP)

In place of covering the topic of recommending alternative classification approach for fault prediction, the focus of this section is on the core question,"*Given popular classification techniques already identified by the community, is it possible to exploit the results of predictions in order to obtain more interpretable outcomes?*". A **feature-based model explanation** indicates the contribution of each input feature to a model's output for a particular data point.

Experiments. Voltage signals gathered from the IEEE-13 node test feeder are the input to the system, while the output is one of seven fault kinds, including line-to-ground (AG, BG, CG), line-to-line (AB, AC, BC), and three-phase fault (ABC). Faults were induced into a randomly selected zone, zone 4 in this example, as depicted in Figure 4.2, and then characteristics were retrieved from the zone's three-phase voltage signals. This section describes one approach we adopted for a feature-based explanation of the fault type classification method in SGs based on *partial dependence plots* (PDPs).

Approach. The considered steps include two primary phases:

1. *Feature representation.* We used features derived from three-phase voltage signals represented in the time and frequency domains (DFT). To define characteristics, we compute the energy and maximum of the signals on both time-domain and frequency-domain signals, as well as the n-th moment of the probability distribution functions (PDFs) [78] ($n \in [1, 4]$). We collected a total of 12 (6+6) features to represent the characteristics in our labeled training dataset [23, 28].

2. *Interpretability and explaination.* To examine explainability, we propose employing visual analytic techniques such as *partial dependence plots* (PDPs) [57] and *feature importance measurement* using an interpretable model based on decision trees [86]. Using these two complementary visual analysis methodologies, which examine and illustrate the individual influence of features and their

pairwise relationship, the user may interpret the classification model's outcomes with a high degree of accuracy;

Two primary classifiers were used to classify fault types: decision tree and k-nearest neighbors. We represent the classification task as a *multi-class signal-label classification*. We used two classifiers as shown in Table 6.1 and considered DT for feature-level explanation.

Note that in a classification task involving supervised learning, finding significant variables (features) helps in identifying the key drivers. This method does not, however, explain the connection between the input variables and how this relationship influences the outcome of the ML model. To address this issue, a partial dependence plot (PDP) is used to understand the relationship between input variables and predictions. In our scenario, a PDP can for example indicate whether the chance of a certain fault increases with signal energy and frequency signal kurtosis, a question that does not appear to have a simple answer. In addition, PDP can detect whether two features have a monotonic, linear, or no connection. These are crucial indicators that allow the human operator to completely study and grasp the black-box fault predictions.

Results and Discussions. The discussion of the results is divided into two sections. We begin with the classification results. The impact of two feature analysis approaches on the interpretability of classification predictions is discussed below.

Table 6.1 provides a summary of the classification results using two classifiers, namely decision tree and k-nearest neighbors, using a hold-out setting (80%-20%) for the training and test sets. We can observe that the average classification accuracy across all experimental conditions tested is greater than 92%, confirming the discriminative power of the chosen features. The decision tree had the highest classification accuracy at 96.42%. Thus, during the subsequent phase, a decision tree is utilized.

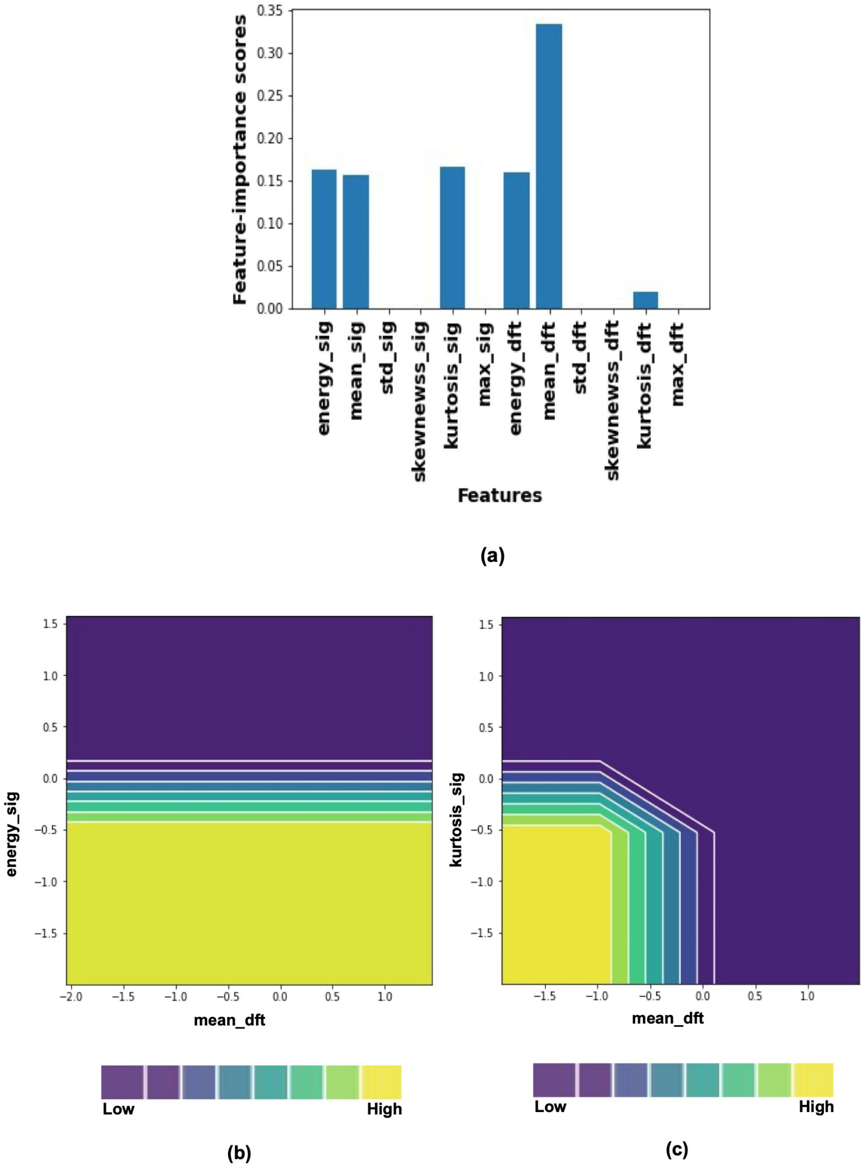
Figure 6.2 illustrates the results of the feature explainability assessment. Particularly, Figure 6.2-a displays the impact of various features on fault type classification predictions. The findings show that the *signal-level features*: energy, mean, and kurtosis, as well as the *frequency-level*

Table 6.1: *Classification accuracy (%) using 12 features and two classifiers. For the k-nearest neighbors, $k = 5$ was used.*

Classifier	**decision tree**	**k-nearest neighbors**
Accuracy	96.42	92.85

features: energy and mean, provide the most valuable features. Therefore, we can note that features at the frequency and signal levels can both impact classification predictions. Figure 6.2-b and Figure 6.2-c offer a more in-depth analysis of the findings. These plots show the effects of mutual feature interactions on the classification outcome and were produced using the PDP technique. We can observe that the two characteristics chosen (as an example) in Figure 6.2-b, namely *mean_dft* and *energy_sig*, are NOT mutually informative, meaning that a change in the values of either one of these features does not affect the classification conclusion in either a positive or negative way. This is equivalent to saying that *mean_dft* has all the necessary information encoded in the set {*mean_dft*, *energy_sig*}. As a result, we may use *mean_dft* for the classification task with confidence and anticipate getting accurate classification results. A different relation is established for the interaction between the features {*mean_dft*, *kurtosis_sig*}, as shown in Figure 6.2-c. We can see that both features have a monotonic effect on the categorization predictions in this situation. When feature values are in the bottom-left corner of the figure, the best classification is made.

The information offered by the PDP analysis for the SG fault type classification task offers additional insights that could not be gained using the conventional feature importance analysis technique, as illustrated in Figure 6.2-a. For instance, while Figure 6.2-a details the influence of the 12 features utilized as a group, it does not provide specific insights on whether the same results could be obtained when a smaller set of features are used. We can see that while some feature pairs, like *mean_dft* and *energy_sig*, are mutually complementary, other feature pairings are correlated. The system designer may eventually use this information to determine (1) which feature(s) to concentrate on for the extraction phase from the SG signals, (2) how to represent the feature to obtain more informative features (e.g., the n-th PDF moment we used), and (3) by the system human operator to understand the cause of particular system faults.



(a)

(b)

(c)

Figure 6.2: *Results of feature analysis (a) feature importance scores for 12 features by the decision tree (b-c) PDP interaction plots utilizing two dominant features in part (a).*

6.2.2 Explainability using Pairwise Feature Selection Analysis

A second approach to providing explanations utilizing pairs of characteristics is considering *pair feature selection* and studying the combined effect in an ML prediction model.

Experiments. This system is fully described in chapter 3. For this section, we focus on fault zone prediction (FZP), which is a multi-class classification task. We use a variety of classifiers, including Decision-Tree, SVM, KNN, and Ensemble approaches (Bagged-Tree, sub-space k-nearest neighbors). For data collection and to generate the training dataset, the IEEE-13 system is divided into four essential zones. Then, seven distinct faults were introduced into each zone. The data collection was repeated for 22 fault resistance values R_f ranging from 0.001 to 2 for each type, with a range of 0.001 to 2. This dataset is larger than the one utilized in the preceding section. Refer to chapter 3 for a discussion of the properties and outcomes of classification tasks in further detail.

Approach. In this study, explainability was achieved by the employment of a decision-model-informed strategy that involved the visualization of the impact of feature pairings that significantly impact the FZP task. In other words, we switched to one of the best classifiers that had won the initial classification task and then searched for the best feature pairs (combination) that would yield the highest classification accuracy. We counted all potential pairs, classified the outcomes, and displayed the results using a heatmap as shown in Figure 6.3.

Results and Discussion. Two sections comprise the outcomes discussion. First, we have classification findings, which are described in section 3.4. Following, we consider the impact of two feature analysis approaches on the interpretability of classification predictions.

The heatmap in Figure 6.3 displays the feature importance analysis visualization. The heatmap illustrates the impact of both feature classes and 48 features and their pairwise relationship on fault location prediction. We depict the interpretability analysis FLP with decision tree classifier results. With the interpretability analysis, we respond to the following experimental inquiries:

- *The impact of the domain (temporal, DFT, or DWT)?* It can be seen from Figure 6.3 that DWT is a significant factor in the majority of the orange and yellow zones that correspond to highly accurate FZP outcomes. It is noteworthy to see that the DWT feature class has more discriminative information (especially for d1 and d2) than the mostly blue-colored time and frequency domain (regardless of the

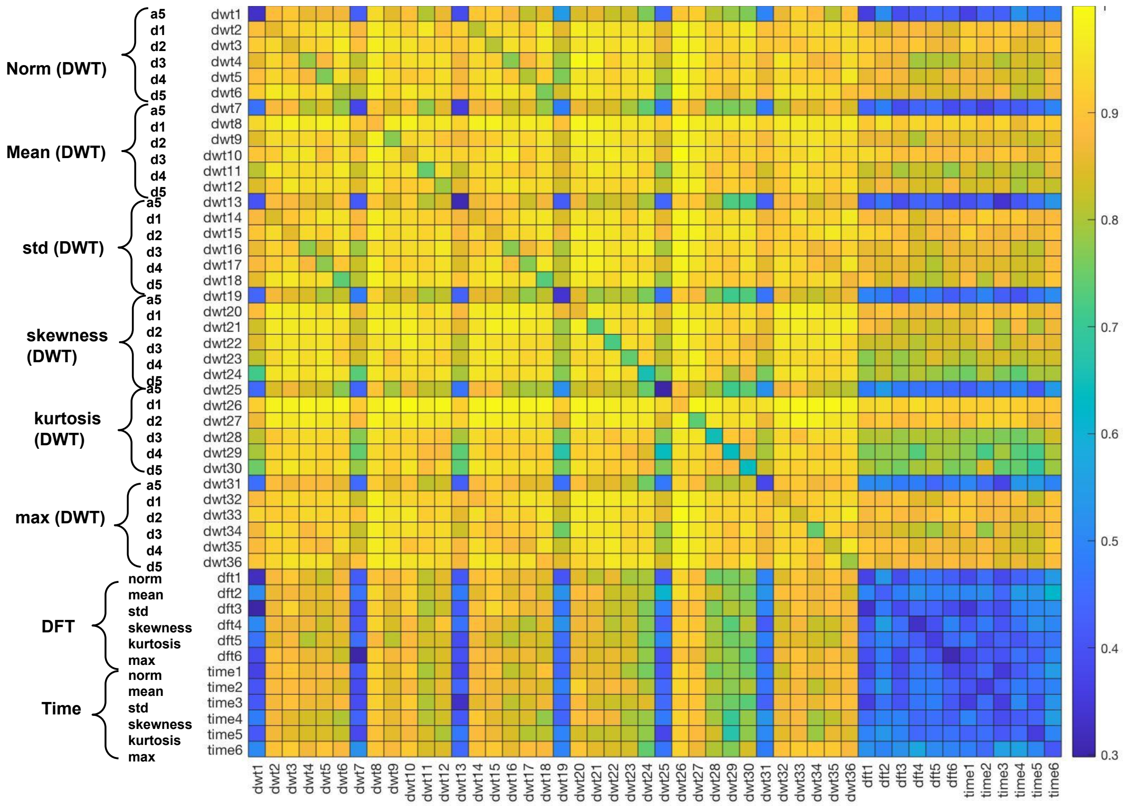


Figure 6.3: *Visualization of the effects of several features on the accuracy of the FLP classifidcation results. This image displays the overall effects of the pairwise interactions of 48 characteristics.*

feature aggregation method). Due to its multi-resolution analysis and filter bank, DWT contains more valuable information than DFT and time signal, and these results provide more clear information on the ML prediction's particulars;

- *The impact of the aggregation function (norm, mean, skewness, etc.)* When examining the DWT results, the majority of yellow patches correspond to $Kurtois > skewness > Mean$. These results are illuminating and demonstrate the significance of n-th moment PDF statistics;

- *The impact of the interaction between extracted features and domains.* Generally, the interaction of DWT with other classes (DWT, DFT, and time) improves classification accuracy. However, the combination of DFT and time provides little classification-relevant information. The most satisfactory results are obtained for (DWT, skewness, d1, and d2) and DWT (DWT, kurtosis, d1, and d2).

6.3 Visual Explanation

We also examined employing a two-dimensional *visual representation* (e.g., an image capturing both time-frequency information) instead of a one-dimensional feature vector for explanation in this research. This representation is significant because it enables the deployment of deep neural networks (DNNs), particularly convolutional neural networks (CNNs), which have demonstrated promising results in various visual recognition applications [44]. The majority of previous research, with the exception of Chen et al. [13], focus on a non-visual representation of input data. Chen et al. only evaluate temporal information visually, omitting frequency information from fault modeling, in contrast to our recommended method, which employs a time-frequency spectrogram representation.

The majority of prior literature on ML fault diagnosis focuses on black-box prediction models that emphasize boosting the system's forecast accuracy. Designing interpretable ML models that can replace traditional black-box prediction models and generate rules that can be comprehended with minimal inspection is vital if we want to keep humans in the control loop who have relied on visual awareness for a long time. The current work at hand introduces a Grad-CAM technique that can *visually* analyze and emphasize the parts of the spectrogram image that are most essential to the classification objective. Detailed information regarding the classification problem is provided in section 4.4 and a spectrogram is used here to provide a *visual explanation*.

6.3.1 Visual explanation using Grad-CAM

Even while it can assist with fault diagnosis with an outstanding level of accuracy, deep learning has one major drawback: it is unable to understand the model or explain why it made particular predictions.

As stated by Zhou et al. [91] , suitable explanation methods must meet two criteria: (i) reliability and (ii) ease of human interpretation. The essential realization is that it is difficult to interpret complicated approaches such as deep neural networks. We utilized the Grad-CAM method [72] to graphically represent the factors that influenced network prediction. Grad-CAM generates a heatmap and visualizes crucial regions in the input image (here spectrograms) that were crucial for the model's decision-making for a specific class label. Comparing it to the FTC and FZC tasks investigated in this work, we find that the similarity in the important regions highlighted in the 2D space for a given class label (e.g., fault type or zone) can provide clearer indications as to why the system made certain decisions for specific classification tasks.

Grad-CAM illustrates how the various regions of the spectrum images associated with (i) various fault types, (ii) various fault zones, or (iii) combinations thereof, affect the system's ultimate judgment. Using the Grad-CAM technique emphasizes two crucial objectives in particular:

- *Intra-class similarity:* it attempts to respond to the question, "*how much the highlighted regions by Grad-CAM are similar for predicting faults within a certain class, e.g., fault AB for the FTC subtask?*".

- *Inter-class variation:* It attempts to answer the question, "How different are the regions highlighted by Grad-CAM for two distinct classes, such as fault AB and fault ABC?"

We created the three visual summaries displayed in several Figures to determine the extent to which Grad-CAM aids in class separation in the context of fault diagnosis in the Power grid. The three primary sub-tasks, FZC, FTC, and joint FTP+FZC, are shown in Figures 6.4, 6.5, and 6.6, respectively.

Visual explanation of FZC. Figure 6.4 displays the visual explanation outcomes obtained using the Grad-CAM technique for the FZC sub-task. In this sub-task, the classifier takes as input the spectrogram images labeled with the zone ids and returns as output the unknown zone id of test images. The results show that samples within each zone have high similarity (high intra-list similarity) and high differences across classes (high inter-list difference). For instance, we can see that the emphasized areas are centered at the top (Zone 1), near to top (Zone 2), middle (Zone 3), and bottom (Zone 4). Regardless of the fault type, we can also observe that the characteristics of activated feature maps (i.e., the size and shape of highlighted regions) inside Zone 1 are substantially more comparable than in Zone 3. These results are interesting and reveal that faults that occurred in different zones produce frequency and time responses that are similar within a zone (intra-similarity) and different when we move across various zones (inter-difference), and this is a valuable piece of information that CNNs can leverage to make accurate predictions.

Visual explanation of FTC. Figure 6.5 displays Grad-CAM images of diverse fault types located in different zones. Here we can also notice similarities in the size and positioning of the red-highlighted portions between the locations in the images on which the network has focused. This signifies that different fault types (e.g., line-to-line vs. line-to-ground) have different characteristics, i.e., intra-class difference, while they look similar for faults within the system.

Visual explanation of joint FTC+FZC We also visualize in Figure 6.6 the results of fault diagnosis for the joint FZC+FTC tasks. It could be noted that the networks' concentration for each <fault type, fault zone> differs from the other scenario. For instance, while there exists a high level of intra-class similarity for Grad-CAM images of fault ABC (e.g., they are all centered in the lower part of the figure), they look different, i.e., inter-class difference when we move from Zone 1 to Zone 4 (e.g., their shape is different).

Overall, the Grad-CAM graphical representation of the results suggests that distinct fault classes (types or zones) generate temporal and frequency responses that are remarkably similar within a class (intra-class similarity) but dramatically different between classes (inter-class difference). Compared to prior, less transparent models, this is a useful piece of information that can be easily interpreted and evaluated by a human operator or field worker.

6.4 Summary

In this chapter, we studied three approaches to the explainability of complex machine learning (ML) models in the context of faults prediction in smart grids: (i) partial dependence plots (PDPs), (ii) visual explanation (Grad-CAM), and (iii) interpretable model (DT). We propose a visual explanation technique based on Grad-CAM for the three tasks FZC, FTC, and joint FZC+FTC to provide insights on why the DNN reached a cer-

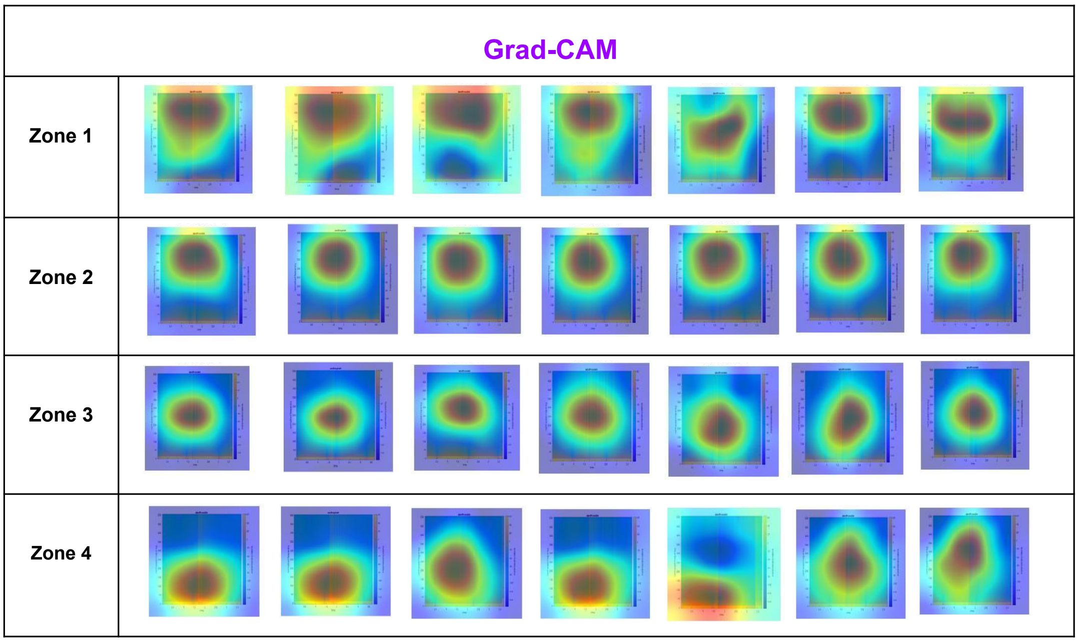


Figure 6.4: *Grad-CAM for four fault zone classification by utilizing seven types of faults (ABC, AB, AC, BC, AG, BG and CG)*

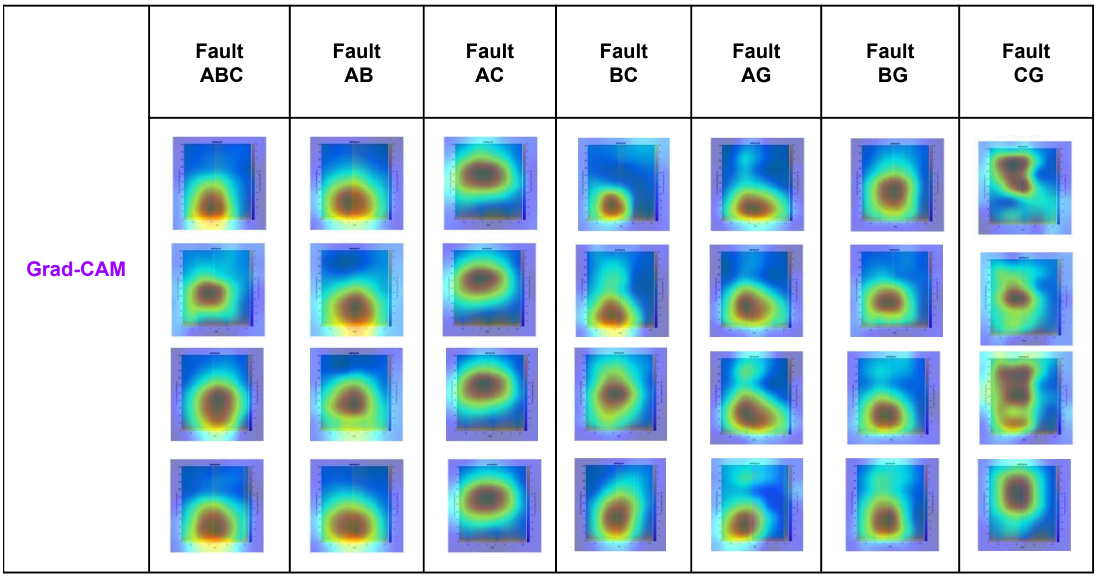


Figure 6.5: *Grad-CAM of fault type classification, seven fault types in different four zones*

tain decision. To understand the results of the FTC task in a reasonable manner, we additionally investigated feature importance measurement

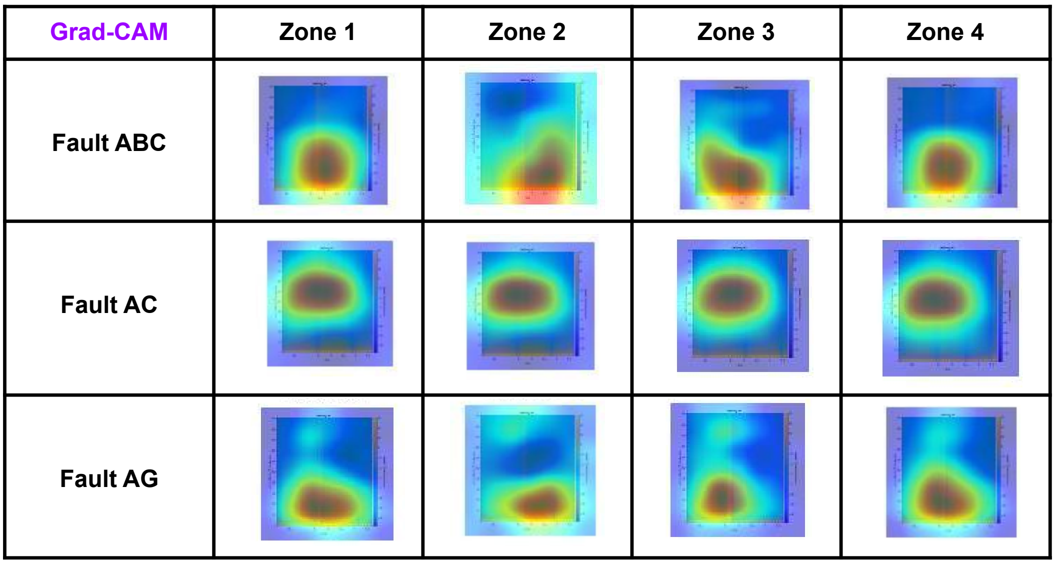


Figure 6.6: *Grad-CAM of joint type-location classification (four zones seven types of faults)*

using an interpretable model based on a decision tree and partial dependence plots. Finally, using the Decision tree, we displayed the influence of pairs of attributes significantly influencing the classification task.

CHAPTER 7

Datasets and Evaluations

7.1 Introduction

Conventionally regulated electrical grids have undergone significant adjustments and upgrades over the years in terms of reliability, robustness, and efficiency. Power grid networks are inherently prone to physical damage and electrical failures can be caused by natural phenomena like a tree falling on a power line, a bird hitting the wire, lightning, or aging of the equipment. One of the most important characteristics of SGs is their application in fault detection, fault classification, fault location prediction, and routine evaluation of the underlying disturbances that cause failures.

Numerous smart grid (SG) applications, such as fault detection and load forecasting, have used data-driven methodologies; nevertheless, the resilience and security of these data-driven algorithms have not been well investigated. One of the largest obstacles in the examination of the security of smart grids is the *lack of publicly available datasets* that can be used to assess the system's resistance to different types of faults and fault prediction systems under adversarial attacks.

This chapter contributes to the distribution of the datasets that we have collected and that might be utilized in a variety of machine learning systems. Specifically, the data may be examined in relation to:

- Input feature characteristics
- The type of the ML system

where for the latter, we address two primary ML systems: (i) fault type and location classification systems, and (ii) the analysis of adversarial machine-learning attacks aimed at the former fault prediction systems (fault type and location classification). For the former aspects, we examine (i) numerical features represented by temporal and DFT features (cf. Section 7.4) versus (ii) visual spectrogram representations (cf. Section 7.3), where the outline of this chapter is based on the feature representations utilized for fault-related tasks.

7.2 Related Datasets

For the purpose of evaluating different three-phase grid algorithms, test grids have been established under SGs that simulate the behavior of a genuine distribution feeder. Incorporating renewable energy sources is another feature of today's modern test grids. The following is a list of the most often seen distribution network test grids found in the literature.

- **IEEE-13.** This exceptionally small circuit model is intended to evaluate typical aspects of 4.16 kV distribution analysis software. It is short, somewhat heavily loaded, characterized by a single voltage regulator at the substation, overhead and subterranean lines, shunt capacitors, an in-line transformer, and unbalanced loading;
- **IEEE-14.** It is a simplified representation of the American electrical grid. The facility contains 11 loads, 5 generators, and 14 buses;
- **IEEE-33.** It encompasses both balanced and unbalanced three-phase power systems, as well as additional information on integrating distributed and renewable generating units, reactive power compensation assets, reconfiguration infrastructures, and load and renewable generation profile statistics for various case studies;

- **IEEE-34.** This feeder is already in place in Arizona, and it has a nominal voltage of 24.9 KV. It has two in-line regulators, an in-line transformer for a short 4.16 KV section, a total of 24 unbalanced loads, and two shunt capacitors, and its overhead transmission lines are lengthy and lightly laden;

- **IEEE-37.**: This feeder is a genuine California feeder with an operational voltage of 4,8 kV, delta-configured. All line segments are underground; substation voltage regulation consists of two single-phase open-delta regulators, spot loads, and extremely unbalanced loads;

- **IEEE-123.** It operates at a nominal voltage of 4.16 kV. This circuit is described by overhead and underground lines, unbalanced loading with constant current, impedance, power, four voltage regulators, shunt capacitor banks, and switches.

Many different simulation programs, such as MATLAB, are used in the electrical sector to handle the problem of fault prediction, and their use in a recent research is discussed here. They include MATLAB Simulink [85], PSCAD [12,37,38], RSCAD [73], PSS/Sincal [2], Opal-RT [29], PST [40, 46], DIgSILENT [32, 49], MATPOWER [33].

Despite the extensive usage of simulation tools for failure prediction systems in SGs, none of the existing research papers have, to our knowledge, attempted to make simulation data from IEEE test node feeds publicly available. This is a significant obstacle to the development of machine-learned adversarial attacks and failure prediction systems.

To address this shortcoming, we provide two exhaustive datasets on smart electrical grids (based on the IEEE-13 test node feeder) that provide both a thorough catalog and a set of distinguishing features for electrical networks.

7.3 IEEE13-AdvAttack: A Novel Dataset for Benchmarking the Power of Adversarial Attacks in SGs

7.3.1 General information

This section describes the first dataset for benchmarking **adversarial machine-learned attacks** against a fault prediction system; the data obtained for this study is based on the IEEE-13 test node feeder. Although other types of node feeders as stated in Section 7.2 may be used, for the sake of simplicity we used IEEE-13 and left the exploration of other node feeders for the future. The IEEE-13 node test feeder consists of a 4.16 KV voltage generator, 13 fault simulation buses, and three-phase signal measuring equipment. This distribution system can be divided into four critical zones: zone 1: 632-671, zone 2: 632-633, zone 3: 692-675, and zone 4: 671-680.

7.3.2 Feature extraction and fault Simulation

In this section, we describe the dataset and data collection used to test the effectiveness of adversarial attacks on fault zone and type classification systems. For the IEEE-13 test node feeder simulation which we performed in MATLAB, we used the default parameters, which comprised a voltage frequency of 60 Hz and a sampling time of $10e-5$. We generated data by injecting faults into the IEEE-13 node test feeder in the Simulink environment of MATLAB. In the four critical zones next to load flow buses 671, 633, 675, and 680, we injected 11 unique fault types with 22 unique resistances per fault type (lines within the red boxes in Figure IEEE-13). Table 7.1 provides a summary of this information, a dataset composed of 11616 faulty samples was created in which 4 (zones)$\times$ 4 (measurement-zone) $\times$ 11 (faults) $\times$ 3 (phases) $\times$ 22 (resistance values) $=$ 11616. For healthy data, we obtained raw healthy signals for 88 different line lengths by measuring from specified four zones and for three phases 88 (Line-length)$\times 4$ (measurement-zone)$\times$ 3 (phases) $= 1056$.

The entire period for fault simulation was $t = [0.0 - 0.02]$, and each fault was added at $t = 0.01$ and removed at $t = 0.02$, resulting in $t_f = [0.01 - 0.02]$. $t_f = [0 - 0.01]$ represents the fault duration, while $t_h = [0.01 - 0.02]$ represents the healthy (non-faulty) period of time. We present on Github [1] a graphical representation of the number of simulation-generated samples for both faulty and healthy signals. In this study, we selected three types of features exploited in prior research [1,20,66,82,88]:

[1] https://bit.ly/3NT5jxG

Table 7.1: *The characteristic of the dataset used for classification and training the machine-learned adversarial attacks in this work.*

Item	Details
Fault type	phase to ground AG, BG, CG phase to phase AB, AC, BC phase to phase to ground ABG, ACG, BCG three phase ABC three phase to ground ABCG
Fault location	zone 1 branch 632-671 zone 2 branch 632-633 zone 3 branch 692-675 zone 4 branch 671-680
Fault resistance	0.0010, 0.0273, 0.0535, 0.0798 0.1061, 0.1323 0.1586, 0.1848 0.2111, 0.2374, 0.2636, 0.2899 0.3162, 0.3424, 0.3687, 0.3949 0.4212, 0.4475, 0.4737, 0.5, 1, 2

- **Time-domain features.** It refers to the original, time-domain-measured data. When six aggregation functions were applied to the voltage signal $x(t)$, a six-dimensional time domain feature vector was generated. They contain (mean, standard deviation, skewness, and kurtosis) as well as the signal's energy and maximum level;

- **Frequency-domain features.** The discrete Fourier transform (DFT) is used to translate voltage signals to the frequency domain. Using the same six aggregation functions employed in the time domain, a six-dimensional frequency-domain feature vector was generated from the calculated spectrum;

- **Discrete Wavelet transform (DWT)**. DWT examines digital signals at multiple resolutions. Multi-resolution analysis employs wavelet coefficients of approximation A_i and detail D_i. Motivated by previous works [1, 88], we employed a large number (five) of level-decompositions according to A_5, $D_{1:5}$;

As a whole, 48 features are obtained, consisting of 6 time-domain features, 6 DFT features, and 36 DWT features (five stages of decomposition plus one level of approximation).

7.3.3 Dataset Structure

The files comprising the dataset are arranged in a predetermined structure to facilitate retrieval. The Github repository provides details about the format of the data set. We have a “DataSet-IEEE13-withRE” folder that contains two additional directories: (1) Voltage readings for all three phases, faulty bad and healthy signals, are included in the “RawTimeSeriesData” file; (2) A “FeatureData” folder containing features extracted using Discrete wavelet transform (DWT), Discrete Fourier transform (DFT), and Time domain.

7.3.4 Benchmarking

In order to accomplish the three classification tasks outlined in section 7.3.5, we trained a Multi-layer Perceptron (MLP) neural network. The model consists of an input layer, a dense layer, and an output layer. Unlike the other layers, the number of neurons in this layer varies among the three tasks, to account for the different numbers of output classes required by each. Each job has its own training phase, but they all follow the same parameters: 500 epochs, Adam Optimizer, a fixed learning rate of $10e-3$, and a batch size of 20. The hyper-parameters were derived from the results of the tuning process.

For adversarial attacks, we used the IBM Adversarial Robustness Toolbox because of its seamless interoperability with Keras and comprehensive collection of attacks well-suited for deep learning models. In both non-targeted and targeted situations, attackers use FGSM and multistep attacks (BIM, PGD).

7.3.5 Fault Classification in Smart Grids

In this study, we focus on many multi-class classification issues relevant to failure prediction in smart grids, where X is the input space and $y = \{1, 2, ..., K\}$ is the output space. Our problem demonstrates the use of two different target labels for the problems at hand (i) fault location and (ii) fault type. As a result, the primary endeavor is broken down into three parts. Therefore, the main task is split into three sub-tasks:

1. Fault location classification (FLC): with $K = 4$ the task aims to classify a given signal into its originating zone as shown in Table

Table 7.2: *Result of application of adversarial attacks against fault classification tasks on the presented IEEE-13 dataset.*

		Base	Random Noise	Adversarial Attack		
				FGSM	BIM	C&W
Attack goal			$\epsilon = 0.05$	$\epsilon = 0.05$	$\epsilon = 0.05$	
UnTargeted	FZC	0.71	0.556	0.160	0.154	0.281 (ℓ_∞)
UnTargeted	FTC	0.46	0.388	0.075	0.048	0.166 (ℓ_∞)
UnTargeted	Joint	0.45	0.320	0.086	0.023	0.139 (ℓ_∞)
Targeted	FZC	0.71	0.556	0.260	0.265	0.631 (ℓ_2)
Targeted	FTC	0.46	0.388	0.076	0.198	0.388 (ℓ_2)
Targeted	Joint	0.45	0.320	0.030	0.135	0.432 (ℓ_2)

7.1;

2. Fault type classification (FTC): the objective of the task with $K = 11$ is to classify a given signal into one of the fault classes shown in Table 7.1;

3. Joint location and type classification (FLC+FTC): $K = 44$ combining both fault class labels in the previous cases;

where, (1) and (2) are explicit in the dataset, whereas (3) is obtained by combining all feasible combinations of tasks 1 and 2. Consequently, we can expect the joint task to be more difficult than the previous one.

7.3.6 Adversarial Attacks against Fault Classification

The adversary seeks to misclassify smart-grid fault classification tasks in each of the three FZC, FTC, and joint subtasks through the use of untargeted versus targeted attacks.
Adversary knowledge. Our assumption is a white-box scenario in which the attacker is aware of all the parameters of the feature extraction model used to estimate the perturbation he or she wishes to estimate. Moreover, the attacker has complete access to the input features that would be modified as a result of the assault. In a targeted attack scenario, an attacker can also collect class labels.
Explored Attacks. The executed attacks include the fast gradient sign technique (FGSM), the basic iterative method (BIM), and Carlini and Wagner (C&W), with FGSM belonging to the ℓ_∞-norm attack type and C&W to the ℓ_∞-norm and ℓ_2-norm attack types, respectively. BIM is

the iterative variant of the FGSM, which is a white-box approach that exploits the sign of the loss function's gradient to learn adversarial perturbations. Formally, in the untargeted case, FGSM seeks to generate a perturbation that maximizes the training loss defined as

$$\delta = \epsilon \cdot \mathrm{sign}(\bigtriangledown_x \ell(f(x;\theta), y)) \tag{7.1}$$

where ϵ (perturbation level) is the attack strength, $\bigtriangledown_x$ is the gradient of the loss function with respect to the input sample **x**, y is the legitimate label, and $sign(.)$ is the sign operator.

Other adversarial attack categories investigated on this dataset were BIM and C&W, which are described in detail in Chapter 5.

Discussion. In Table 7.2, we present the outcomes of benchmarking two security-related scenarios, (i) fault classification and (ii) adversarial attack against a fault classification system, using the proposed dataset. **Base** illustrates the result of the pure classification system prior to an assault (FZC, FTC, or combined). As the complexity of the work develops from FZC to FTC to joint, it can be demonstrated that classification accuracy decreases. We compare the efficacy of adversarial perturbations generated by FGSM, BIM, and C&W to that of random noise. In addition, we consider the performance of attacks to vary as we switch between **attack targets**. As demonstrated in Table 7.2 the investigated adversarial attacks FGSM, BIM, and C&W have a significantly bigger impact in untargeted settings.

For instance, comparing the strength of the three adversarial attack models, BIM is the strongest in all tasks. In the case of (untargeted, Joint), BIM untargeted adversarial attack accuracy reaches 0.023, whilst FGSM and C&W reach 0.086 and 0.139, respectively, under the same condition. The effect of attack target (targeted vs. untargeted) is stronger on BIM and C&W than on FSGM. For example, for the (FTC), the classification accuracy of 0.048 vs. 0.198 (BIM-untrg vs. BIM-trg), however for FGSM the corresponding difference is only 0.075 vs. 0.076 (FGSM-untrg vs. FGSM-trg).

7.3.7 Conclusion

By recreating IEEE-13 test feeders with renewable energy and generating relevant data, we investigated the security and vulnerability of fault classification systems in the context of smart electrical grids. First released was IEEE13-AdvAttack, a large-scale simulated dataset based on the IEEE-13 test node feeder that is appropriate for supervised fault classification tasks under SG. In the dataset, both traditional and renewable energy sources are represented. We investigate the resilience of fault-type classification and fault zone classification systems in the face of adversarial attacks. To defend these systems against alternative adversarial training and detection strategies will necessitate more nuanced and in-depth research, which we intend to conduct in the future.

7.4 A Dataset for Electrical Grid Using Spectrogram-Based CNN Modeling

In addition to IEEE13-AdvAttack described in the preceding section, we have contributed to the creation of a new class of fault prediction-applicable features based on CNN representation of spectrograms. This chapter explains the characteristics of the features and attributes that accompany this dataset. We note that the dataset presented in this section was collected from an IEEE-13 node test feeder *without renewable energy* (RE) and simulated in MATLAB Simulink. The dataset has been divided into three folders as shown in Figure 7.1:

1. *Raw time-series data*: This subfolder contains the raw time-series data, one containing raw false data and the other having raw healthy information.

   - *FaultySignal-withoutRE.* There are two folders containing raw time-series data; one contains raw faulty data and the other contains raw healthy data. For the fault simulation, as shown in Figure 7.1, We partitioned the network into four zones. For the entire simulation time, $t = [0.0 - 0.022]$, we injected 11 fault types (AG, BG, CG, AB, AC, BC, ABG, ACG, BCG, ABC, ABCG) with 22 different resistances for each type of faults into these four critical zones next to load flow buses numbers 671, 633, 675, and 680. As each fault with each resistance was applied at a specific start time of $t = 0.01$ and was removed at $t = 0.02$, the faulty duration is represented

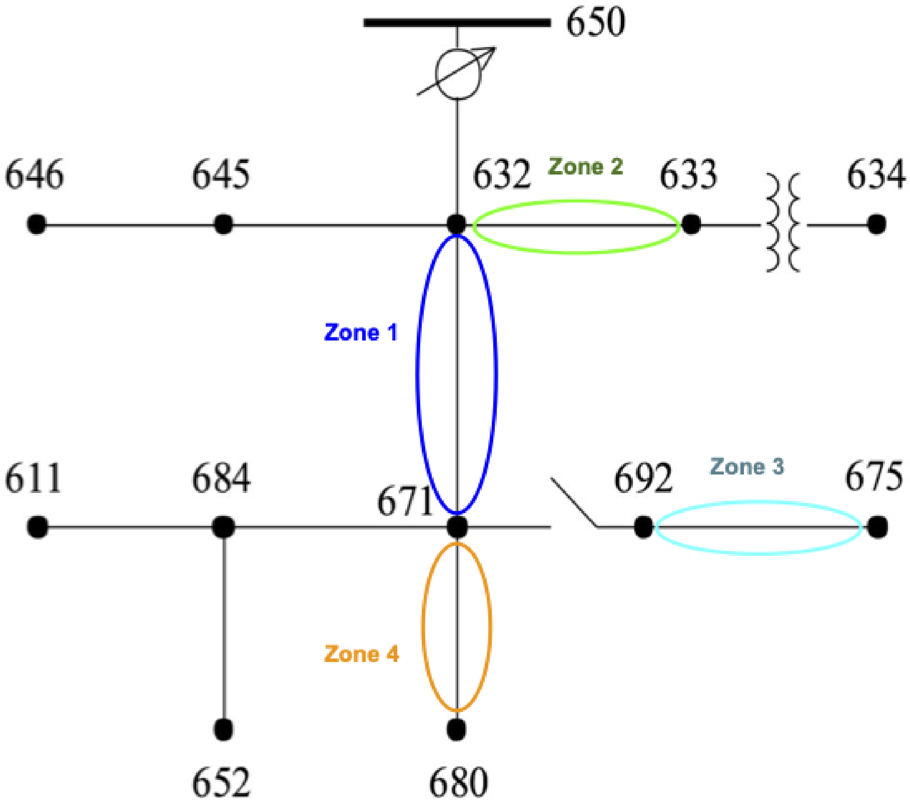


Figure 7.1: *The structure of the dataset in our SpectCNN benchmark.*

by $t_f = [0.01 - 0.02]$ and the healthy (non-faulty) duration is represented by $t_h = [0 - 0.01]$. Three.csv files in this folder include voltage measurements for phases A, B, and C;

- *HealthySignal-withRE.* We obtained raw healthy signals for 88 line lengths from four locations and three stages for Healthy data. This folder contains three.xlsx files containing voltage measurements for phases A, B, and C. The whole duration of the simulation was $t = [0, 0 - 0.022]$. The chart below depicts the number of samples generated by simulation for both faulty and healthy signals;

2. *FeatureData*: Consideration is given to three domains for the baseline: time, frequency (discrete Fourier transform DFT), and discrete wavelet transform (DWT). As inputs to classifiers, we examine the influence of several statistical aggregation functions that compute the n-th moment of probability distribution functions (PDFs) ($n \in [1, 4]$) together with the energy and maximum level of the signals for three domains. In all, 48 features are retrieved, including six time-domain features, six DFT features, and thirty-six DWT features (five stages of decomposition plus one level of approximation). In this sense, six attributes are multiplied by six for DWT. In this folder, we find two folders containing information on features for broken and healthy signals.

   - *Features faultySignal.* There are three.csv files for flawed data, each containing 48 features and labels and 2640 samples for phases A, B, and C, respectively. In addition, there are two labels, "locLabel" and "faultLabel", as well as resistance and measurement location information in columns with the headers "resistance" and "measloc";
   - *Features healthySignal.* There are three.csv files for healthy data, each including 48 features and labels and 240 samples for phases A, B, and C, respectively. In addition, columns with the titles "lineLength" and "measloc" include information regarding line length and the location of measurements. There are three.csv files for healthy data, each including 48 features and labels and 240 samples for phases A, B, and C, respectively. In

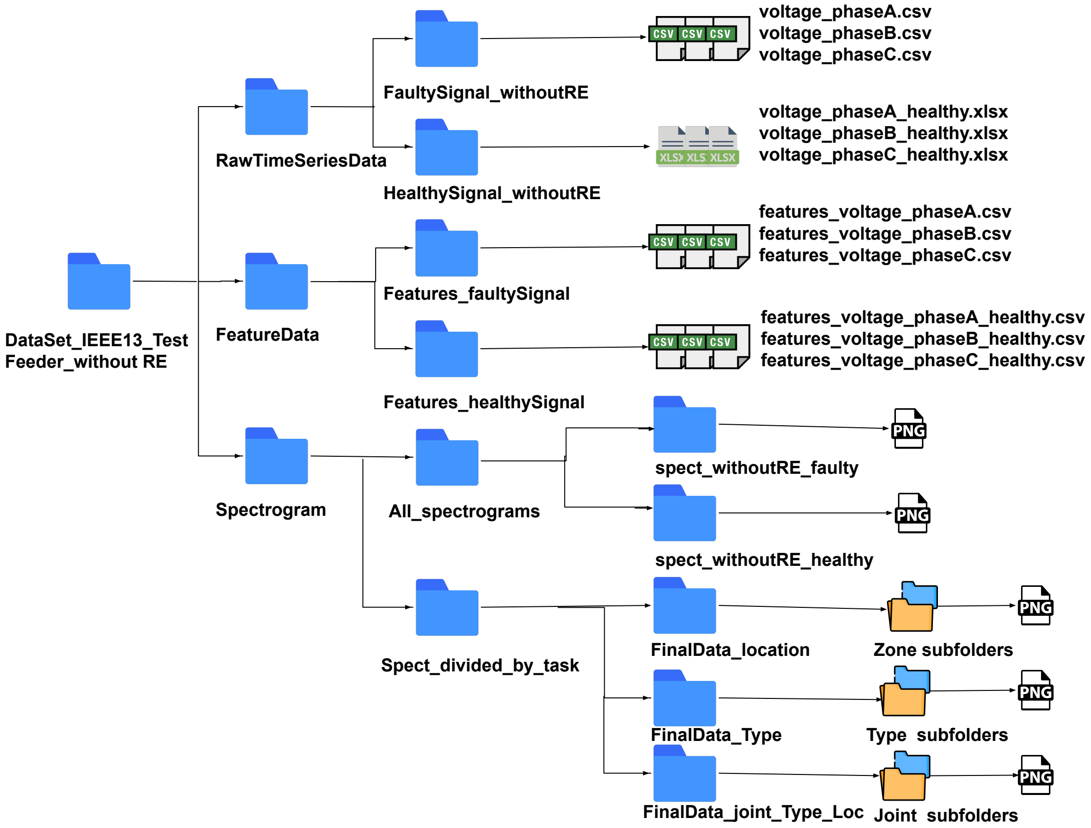


Figure 7.2: *The graphic illustration of how our proposed CNN-based Spectrogram represents the features for fault and healthy signals.*

addition, columns with the titles “lineLength” and “measloc” include information regarding line length and the location of measurements.

3. *Spectrogram.* Two primary folders exist for spectrograms. (1) All spectrograms and (2) Spect by task. The "Spect divided by task" folder is created because the MATLAB datastore class requires pictures to be grouped in subfolders where each subfolder has a class label.

    - *All spectrograms.* This folder is home to two files: (i) spect withoutRE faulty, which contains spectrograms with errors. For example, “IEEE13-locLabel-1-mesloc-1-resistance-0.001-faultLabel-AB-voltage-phaseC.png” comprises information regarding the location label, the location at which the signal is measured, the resistance, the type of the fault, and the phase; (ii) spect-withoutRE-healthy file which includes healthy spectrograms. For example, “IEEE13-mesloc-1-lineLength-0.60554-voltage-phaseB-healthy.png” is the name of an image that contains all information regarding the location where the signal is measured, the line length, and the phase.

- *Spect by task.* This folder contains three directories that have been formatted as MATLAB datastore classes, with each subfolder serving as a class label. These three folders correspond to the three classification tasks.
  - *FinalData-location*. This folder has four subfolders labeled with CNN-related images Zone-1, Zone-2, Zone-3, and Zone-4;
  - *FinalData-Type*. This folder contains 12 subfolders containing CNN-related image labels (Fault-AB, Fault-ABC, Fault-ABCG, Fault-ABG, Fault-AC, Fault-ACG, Fault-BC, Fault-BCG, Fault-BG, CG, AG, and Healthy);
  - *FinalData-joint-Type-Loc*. This folder contains 5 sub-folders (Zone-1, Zone-2, Zone-3, Zone-4, Zone-h) and each of these sub-folders contain 11 sub-sub-folders for instance (z1-Fault-AB, z1-Fault-ABC, z1-Fault-ABCG, z1-Fault-ABG, z1-Fault-AC, z1-Fault-ACG, z1-Fault-BC, z1-Fault-BCG, z1-Fault-BG, z1-Fault-CG, z1-Fault-AG) except sub-folder Zone-h which contain one sub-sub-folder *Healthy* that are labels with corresponding images for CNN.

Several fault classification strategies are compared in Table 7.3, all of which make use of a convolutional neural network (CNN) to represent electrical data via spectrograms. Learn more about the experiments in Chapter 4.

7.5 Summary

In this chapter, we presented the distribution of the datasets that we have gathered and that may be exploited in a range of machine learning systems thanks to the information provided in this chapter. To be more specific, the data can be analyzed in connection to:

Table 7.3: *Classification accuracy (%) using spectrogram on GoogleNet, GoogleNet, and CustomCNN along different epochs. The training times were obtained on a regular machine. The could be decreased if performed on a high-speed machine equipped with GPUs. The comparison between the models' training time, however, remains valid.*

		GoogleNet	SqueezeNet	`CustomCNN`
FZC	best accuracy	85.1	**85.3**	84.2
FTC	best accuracy	**59.4**	58.4	56.9
FZC+ FTC	best accuracy	**57.2**	54.4	56.6

- Input feature characteristics
- The type of the ML system

where for the latter, we address two primary ML systems: (i) fault type and location classification systems, and (ii) the analysis of adversarial machine-learning attacks aimed at the former fault prediction systems (fault type and location classification). For the former aspects, we examine (i) numerical features represented by temporal, frequency, and wavelet features (cf. Section 7.4) versus (ii) visual spectrogram representations (cf. Section 7.3), where the outline of this chapter is based on the feature representations utilized for fault-related tasks.

In order to further assist researchers in using this dataset and comparing their results with those of other papers and experiments, we attempted to provide some baseline results through experiments. This was done in order to make it easier for the adoption of the datasets in fault prediction tasks that are performed in SGs. We demonstrated in detail the performance of various numerical features, notably those represented by time, frequency, and wavelet (see Chapter 5) when used to fault prediction tasks and adversarial attacks against such systems (cf. Section 7.3). Following that, we demonstrated how well SoA CNNs performed on an entirely new category of data, which was distinguished by time-frequency spectrograms as opposed to the frequency or temporal properties that are more traditionally used. We offered experimental examples of how this visual data can be utilized in the fault prediction tasks (cf. Section 7.4) that were devised and obtained competitive performance, see also chapter 4 for more detailed information.

CHAPTER 8

Conclusion

8.1 Summary of Thesis

This Ph.D. study topic was sponsored by the e-distribution smart grid lab in Milan, Italy. Notably, we used a Human-Centred Design approach, initially visiting the e-distribution Smart Grid Lab in Milan to interview electrical engineers and see their work, systems, and artifacts. Then, because of the restrictions imposed by the COVID-19 outbreak, we held monthly video conferences with the Lab team to review the preliminary research findings.

In this Ph.D. dissertation, the problem of fault prediction in smart electrical grids is elaborated, and approaches for leveraging trustworthy AI in SGs are investigated and presented. The presented methods tackle the issue of enhancing SGs' self-healing capabilities by accounting for the robustness (security) aspect of trustworthiness, enhancing the transparency of the presented systems through explanation, and evaluating the proposed methods by developing a dataset, which is one of the field's gaps. In this chapter, we first summarize and provide the conclusion of the works conducted in this thesis. Afterward, we present the future research directions for extending this work.

8.2 Main Contributions.

Several self-healing systems for smart grids based on machine learning approaches have been presented. This method often aims to predict the nature and location of a fault in its earliest stages. Multiple novel strategies, in addition to the refinement of previously existing ones, have been presented to extract meaningful information from the electrical signal and incorporate it into a machine-learning fault prediction system for developing fault prediction systems, such as type and location classification tasks. These methods include 2D CNN-based visual spectrogram methods and hand-crafted temporal, frequency, and wavelet inputs for ML algorithms.

In addition, as a second contribution field of research, we investigated adversarial attacks on fault prediction systems. Illustrated is the capability of cutting-edge adversarial techniques, such as FGSM and BIM, to learn perturbations that can trick ML models, for example, by misclassifying the fault type or location and delaying the rescue team's recovery time.

As a third research field of this thesis, we also explored the explainability of the many integrated technologies, including the use of visual explanation, to make the systems more understandable to a larger audience (operators, consumers).

Last but not least, we exampled the distribution of the datasets that we have introduced in the context of this Ph.D. thesis and that, as a result of the information offered in the present thesis, can be utilized to train various types of ML models for the intended fault prediction tasks.

8.3 Outlook and Future Work

In the following, we lay forward some open research directions and challenges in fault prediction task in SGs, for more exploration:

- *Defenses against adversarial agents.* In recent years, the fragility of the smart grid as critical infrastructure has become a major concern. In [15], the vulnerability of various adversarial attacks against machine learning algorithms used in building load forecasting and power quality disturbances is investigated. Relevant studies on various elements of SGs, such as fault classification and fault zone prediction, have, to the best of our knowledge, been investigated as infrequently as we did in this thesis. Recent advancements in adversarial machine learning (AML) technology, specifically *defense* against these adversarial attacks, should be studied in more-depth, notably for fault detection, location identification, and type categorization;

- *Privacy in SG*. The collected information from individuals is a valuable resource. Before using this information, consensus must be established, and no action should be made if approval is lacking. Private information falls into two broad categories: (1) *identifiers* (e.g., a person's name and social security number) and *quasi-identifiers* (information such as a person's postal code) and (2) *sensitive qualities* (properties that people do not want revealed such as health status, voting history, income, and location data). Privacy in ML and SGs frequently addresses the second method and is one of the first steps in building trustworthy systems. Unreliable parties may steal or attacks data collected by devices (sensors) on transmission and distribution networks, as well as data supplied to SCADA via communication networks. Federated learning and differential privacy are two of the most prominent methods for protecting privacy in machine learning, and merit higher consideration in smart grids;

Overall, the taxonomy presented in Chapter 2 of this thesis provides an actionable catalog that practitioners and scholars can use to identify specific predictive activities and countermeasures, along with a list of sources where additional information about the proposed techniques can be obtained. The remaining chapters, however, give and demonstrate actual fault prediction models and adversarial attacks against such systems.

Bibliography

[1] Tamer S. Abdelgayed, Walid G. Morsi, and Tarlochan S. Sidhu. A new harmony search approach for optimal wavelets applied to fault classification. *IEEE Trans. Smart Grid*, 9(2):521–529, 2018.

[2] Shaik Affijulla and Praveen Tripathy. A robust fault detection and discrimination technique for transmission lines. *IEEE Trans. Smart Grid*, 9(6):6348–6358, 2018.

[3] Mohiuddin Ahmed and Al-Sakib Khan Pathan. False data injection attack (FDIA): an overview and new metrics for fair evaluation of its countermeasure. *Complex Adapt. Syst. Model.*, 8:4, 2020.

[4] Sajjad Amini, Fabio Pasqualetti, Masoud Abbaszadeh, and Hamed Mohsenian Rad. Hierarchical location identification of destabilizing faults and attacks in power systems: A frequency-domain approach. *IEEE Trans. Smart Grid*, 10(2):2036–2045, 2019.

[5] Carmelo Ardito, Yashar Deldjoo, Tommaso Di Noia, Eugenio Di Sciascio, and Fatemeh Nazary. Ieee13-advattack a novel dataset for benchmarking the power of adversarial attacks against fault prediction systems in smart electrical grid. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 3817–3821, 2022.

[6] Carmelo Ardito, Yashar Deldjoo, Tommaso Di Noia, Eugenio Di Sciascio, and Fatemeh Nazary. Visual inspection of fault type and zone prediction in electrical grids using interpretable spectrogram-based CNN modeling. *Expert Syst. Appl.*, 210:118368, 2022.

[7] Carmelo Ardito, Yashar Deldjoo, Eugenio Di Sciascio, and Fatemeh Nazary. Interacting with features: Visual inspection of black-box fault type classification systems in electrical grids. In Cataldo Musto, Daniele Magazzeni, Salvatore Ruggieri, and Giovanni Semeraro, editors, *Proc. of the Italian Workshop on Explainable Artificial Intelligence co-located with 19th International Conference of the Italian Association for Artificial Intelligence, XAI.it@AIxIA 2020, Online Event, November 25-26, 2020*, volume 2742 of *CEUR Workshop Proceedings*, pages 135–141. CEUR-WS.org, 2020.

[8] Carmelo Ardito, Yashar Deldjoo, Eugenio Di Sciascio, and Fatemeh Nazary. Revisiting security threat on smart grids: Accurate and interpretable fault location prediction and type classification. In Alessandro Armando and Michele Colajanni, editors, *Proceedings of the Italian Conference on Cybersecurity, ITASEC 2021, All Digital Event, April 7-9, 2021*, volume 2940 of *CEUR Workshop Proceedings*, pages 523–533. CEUR-WS.org, 2021.

[9] Christopher M Bishop. *Pattern recognition and machine learning*. springer, 2006.

[10] D Burpee, H Dabaghi, L Jackson, F Kwamena, J Richter, T Rusnov, K Friedman, L Mansueti, and D Meyer. U.s. - canada power system outage task force : final report on the implementation of task force recommendations, 2006.

[11] Nicholas Carlini and David A. Wagner. Defensive distillation is not robust to adversarial examples. *CoRR*, abs/1607.04311, 2016.

[12] Soham Chakraborty and Sarasij Das. Application of smart meters in high impedance fault detection on distribution systems. *IEEE Trans. Smart Grid*, 10(3):3465–3473, 2019.

[13] Kunjin Chen, Jun Hu, and Jinliang He. Detection and classification of transmission line faults based on unsupervised feature learning and convolutional sparse autoencoder. *IEEE Trans. Smart Grid*, 9(3):1748–1758, 2018.

[14] Leian Chen, Shang Li, and Xiaodong Wang. Quickest fault detection in photovoltaic systems. *IEEE Trans. Smart Grid*, 9(3):1835–1847, 2018.

[15] Yize Chen, Yushi Tan, and Deepjyoti Deka. Is machine learning in power systems vulnerable? In *2018 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids, SmartGridComm 2018, Aalborg, Denmark, October 29-31, 2018*, pages 1–6. IEEE, 2018.

[16] Petr Cintula, Christian G. FermÃ¼ller, and Carles Noguera. Fuzzy Logic. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, Winter 2021 edition, 2021.

[17] Yandre MG Costa, Luiz S Oliveira, and Carlos N Silla Jr. An evaluation of convolutional neural networks for music classification using spectrograms. *Applied soft computing*, 52:28–38, 2017.

[18] Jochen L Cremer, Ioannis Konstantelos, and Goran Strbac. From optimization-based machine learning to interpretable security rules for operation. *IEEE Transactions on Power Systems*, 34(5):3826–3836, 2019.

[19] Lei Cui, Youyang Qu, Longxiang Gao, Gang Xie, and Shui Yu. Detecting false data attacks using machine learning techniques in smart grid: A survey. *J. Netw. Comput. Appl.*, 170:102808, 2020.

[20] Qiushi Cui and Yang Weng. Enhance high impedance fault detection and location accuracy via μ-pmus. *IEEE Trans. Smart Grid*, 11(1):797–809, 2020.

[21] Swagata Das, Sundaravaradan Navalpakkam Ananthan, and Surya Santoso. Estimating zero-sequence line impedance and fault resistance using relay data. *IEEE Trans. Smart Grid*, 10(2):1637–1645, 2019.

[22] Enrico De Santis, Antonello Rizzi, Alireza Sadeghian, and FM Frattale Mascioli. A learning intelligent system for fault detection in smart grid by a one-class classification approach. In *2015 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2015.

[23] Yashar Deldjoo, Markus Schedl, Paolo Cremonesi, and Gabriella Pasi. Content-based multimedia recommendation systems: Definition and application domains. In *Proceedings of the 9th Italian Information Retrieval Workshop, Rome, Italy, May, 28-30, 2018*, volume 2140 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2018.

[24] Quang Do, Ben Martini, and Kim-Kwang Raymond Choo. A data exfiltration and remote exploitation attack on consumer 3d printers. *IEEE Trans. Inf. Forensics Secur.*, 11(10):2174–2186, 2016.

[25] Maryam Farajzadeh-Zanjani, Ehsan Hallaji, Roozbeh Razavi-Far, Mehrdad Saif, and Masood Parvania. Adversarial semi-supervised learning for diagnosing faults and attacks in power grids. *IEEE Trans. Smart Grid*, 12(4):3468–3478, 2021.

[26] Ahmed M. Gaouda, Atef Abdrabou, Khaled Bashir Shaban, Mutaz Khairalla, Ahmed M. Abdrabou, Ramadan El Shatshat, and M. M. A. Salama. A smart IEC 61850 merging unit for impending fault detection in transformers. *IEEE Trans. Smart Grid*, 9(3):1812–1821, 2018.

[27] Mohammad Gholami, Ali Abbaspour, Moein Moeini-Aghtaie, Mahmud Fotuhi-Firuzabad, and Matti Lehtonen. Detecting the location of short-circuit faults in active distribution network using pmu-based state estimation. *IEEE Trans. Smart Grid*, 11(2):1396–1406, 2020.

[28] Mostafa Gilanifar, Jose Cordova, Hui Wang, Matthias Stifter, Eren E Ozguven, Thomas I Strasser, and Reza Arghandeh. Multi-task logistic low-ranked dirty model for fault detection in power distribution system. *IEEE Transactions on Smart Grid*, 11(1):786–796, 2019.

[29] Mostafa Gilanifar, Jose Cordova, Hui Wang, Matthias Stifter, Eren Erman Ozguven, Thomas I. Strasser, and Reza Arghandeh. Multi-task logistic low-ranked dirty model for fault detection in power distribution system. *IEEE Trans. Smart Grid*, 11(1):786–796, 2020.

[30] Muhammed Zekeriya Gunduz and Resul Das. Cyber-security on smart grid: Threats and potential solutions. *Computer networks*, 169:107094, 2020.

[31] Kian Hamedani, Lingjia Liu, Jithin Jagannath, and Yang Cindy Yi. Adversarial classification of the attacks on smart grids using game theory and deep learning. In Christina Pöpper and Mathy Vanhoef, editors, *WiseML@WiSec 2021: Proceedings of the 3rd ACM Workshop on Wireless Security and Machine Learning, Abu Dhabi, United Arab Emirates, July 2, 2021*, pages 13–18. ACM, 2021.

[32] Sayyed Mohammad Hashemi, Majid Sanaye-Pasand, and Mohammad Shahidehpour. Fault detection during power swings using the properties of fundamental frequency phasors. *IEEE Trans. Smart Grid*, 10(2):1385–1394, 2019.

[33] Miao He and Junshan Zhang. A dependency graph approach for fault detection and localization towards secure smart grid. *IEEE Trans. Smart Grid*, 2(2):342–351, 2011.

[34] Md Shakawat Hossan and Badrul H. Chowdhury. Data-driven fault location scheme for advanced distribution management systems. *IEEE Trans. Smart Grid*, 10(5):5386–5396, 2019.

[35] Forrest N Iandola, Song Han, Matthew W Moskewicz, Khalid Ashraf, William J Dally, and Kurt Keutzer. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and< 0.5 mb model size, 2016.

[36] Ke Jia, Tao Feng, Qijuan Zhao, Congbo Wang, and Tianshu Bi. High frequency transient sparse measurement-based fault location for complex DC distribution networks. *IEEE Trans. Smart Grid*, 11(1):312–322, 2020.

[37] Huaiguang Jiang, Xiaoxiao Dai, David Wenzhong Gao, Jun Jason Zhang, Yingchen Zhang, and Eduard Muljadi. Spatial-temporal synchrophasor data characterization and analytics in smart grid fault detection, identification, and impact causal analysis. *IEEE Trans. Smart Grid*, 7(5):2525–2536, 2016.

[38] Huaiguang Jiang, Jun Jason Zhang, David Wenzhong Gao, and Ziping Wu. Fault detection, identification, and location in smart grid based on data-driven computational methods. *IEEE Trans. Smart Grid*, 5(6):2947–2956, 2014.

[39] Yazhou Jiang. Data-driven fault location of electric power distribution systems with distributed generation. *IEEE Trans. Smart Grid*, 11(1):129–137, 2020.

[40] Iman Kiaei and Saeed Lotfifard. A two-stage fault location identification method in multiarea power grids using heterogeneous types of data. *IEEE Trans. Ind. Informatics*, 15(7):4010–4020, 2019.

[41] Iman Kiaei and Saeed Lotfifard. Fault section identification in smart distribution systems using multi-source data based on fuzzy petri nets. *IEEE Trans. Smart Grid*, 11(1):74–83, 2020.

[42] Michal Krátký, Stanislav Misák, Petr Gajdos, Petr Lukas, Radim Baca, and Peter Chovanec. A novel method for detection of covered conductor faults in medium voltage overhead line systems. *IEEE Trans. Ind. Electron.*, 65(1):543–552, 2018.

[43] Alexey Kurakin, Ian J. Goodfellow, and Samy Bengio. Adversarial examples in the physical world. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Workshop Track Proceedings*, 2017.

[44] Jongpil Lee, Jiyoung Park, Keunhyoung Luke Kim, and Juhan Nam. Samplecnn: End-to-end deep convolutional neural networks using very small filters for music classification. *Applied Sciences*, 8(1):150, 2018.

[45] Weilin Li, Antonello Monti, and Ferdinanda Ponci. Fault detection and classification in medium voltage DC shipboard power systems with wavelets and artificial neural networks. *IEEE Trans. Instrum. Meas.*, 63(11):2651–2665, 2014.

[46] Wenting Li, Deepjyoti Deka, Michael Chertkov, and Meng Wang. Real-time faulted line localization and pmu placement in power systems through convolutional neural networks. *IEEE Transactions on Power Systems*, 34(6):4640–4651, 2019.

[47] Hanif Livani and C. Yaman Evrenosoglu. A machine learning and wavelet-based fault location method for hybrid transmission lines. *IEEE Trans. Smart Grid*, 5(1):51–59, 2014.

[48] Javier Lopez, Juan E Rubio, and Cristina Alcaraz. A resilient architecture for the smart grid. *IEEE Transactions on Industrial Informatics*, 14(8):3745–3753, 2018.

[49] Mehrdad Majidi, Mehdi Etezadi-Amoli, and Mohammed Sami Fadali. A sparse-data-driven approach for fault location in transmission networks. *IEEE Trans. Smart Grid*, 8(2):548–556, 2017.

[50] Stéphane Mallat. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 11(7):674–693, 1989.

[51] Kebina Manandhar, Xiaojun Cao, Fei Hu, and Yao Liu. Detection of faults and attacks including false data injection attack in smart grid using kalman filter. *IEEE Trans. Control. Netw. Syst.*, 1(4):370–379, 2014.

[52] Ricards Marcinkevics and Julia E. Vogt. Interpretability and explainability: A machine learning zoo mini-tour. *CoRR*, abs/2012.01805, 2020.

[53] Apostolos N. Milioudis, Georgios T. Andreou, and Dimitris P. Labridis. Enhanced protection scheme for smart grids using power line communications techniques - part I: detection of high impedance fault occurrence. *IEEE Trans. Smart Grid*, 3(4):1621–1630, 2012.

[54] Apostolos N. Milioudis, Georgios T. Andreou, and Dimitris P. Labridis. Detection and location of high impedance faults in multiconductor overhead distribution lines using power line communication devices. *IEEE Trans. Smart Grid*, 6(2):894–902, 2015.

[55] Ramin Moghaddass and Jianhui Wang. A hierarchical framework for smart grid anomaly detection using large-scale smart meter data. *IEEE Trans. Smart Grid*, 9(6):5820–5830, 2018.

[56] Seyed Mohsen Mohammadi-Hosseininejad, Alireza Fereidunian, Alireza Shahsavari, and Hamid Lesani. A healer reinforcement approach to self-healing in smart grid by phevs parking lot allocation. *IEEE Transactions on Industrial Informatics*, 12(6):2020–2030, 2016.

[57] Christoph Molnar. *Interpretable Machine Learning*. Lulu. com, 2020.

[58] Iman Niazazari and Hanif Livani. Attack on grid event cause analysis: An adversarial machine learning approach. In *IEEE Power & Energy Society Innovative Smart Grid Technologies Conference, ISGT 2020, Washington, DC, USA, February 17-20, 2020*, pages 1–5. IEEE, 2020.

[59] Adeniyi Kehinde Onaolapo, Kayode Timothy Akindeji, and Emmanuel Adetiba. Simulation experiments for faults location in smart distribution networks using ieee 13 node test feeder and artificial neural network. In *Journal of Physics: Conference Series*, volume 1378, page 032021. IOP Publishing, 2019.

[60] Ijeoma Onyeji, Morgan Bazilian, and Chris Bronk. Cyber security and critical energy infrastructure. *The Electricity Journal*, 27(2):52–60, 2014.

[61] Yuval Oren and Saharon Rosset. Semi-supervised empirical risk minimization: When can unlabeled data improve prediction. *CoRR*, abs/2009.00606, 2020.

[62] Yeshwant G Paithankar and SR Bhide. *Fundamentals of power system protection*. PHI Learning Pvt. Ltd., 2011.

[63] Amir Mehdi Pasdar, Yilmaz Sozer, and Iqbal Husain. Detecting and locating faulty nodes in smart grids based on high frequency signal injection. *IEEE Trans. Smart Grid*, 4(2):1067–1075, 2013.

[64] Sanjeev Raja and Ernest Fokoué. Multi-stage fault warning for large electric grids using anomaly detection and machine learning. *CoRR*, abs/1903.06700, 2019.

[65] Sarvesh Rawat, Ahmed Patel, Joaquim Celestino Jr., and André Luiz Moura dos Santos. A dominance based rough set classification system for fault diagnosis in electrical smart grid environments. *Artif. Intell. Rev.*, 46(3):389–411, 2016.

[66] Evandro Agostinho Reche, Jeovane Vicente de Sousa, Denis Vinicius Coury, and Ricardo A. S. Fernandes. Data mining-based method to reduce multiple estimation for fault location in radial distribution systems. *IEEE Trans. Smart Grid*, 10(4):3612–3619, 2019.

[67] M. Jayabharata Reddy, D. Venkata Rajesh, Pathirikkat Gopakumar, and Dusmanta Kumar Mohanta. Smart fault location for smart grid operation using rtus and computational intelligence techniques. *IEEE Syst. J.*, 8(4):1260–1271, 2014.

[68] Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nat. Mach. Intell.*, 1(5):206–215, 2019.

[69] Khaled A. Saleh, Ali Hooshyar, and Ehab F. El-Saadany. Hybrid passive-overcurrent relay for detection of faults in low-voltage DC grids. *IEEE Trans. Smart Grid*, 8(3):1129–1138, 2017.

[70] Enrico De Santis, Antonello Rizzi, and Alireza Sadeghian. A cluster-based dissimilarity learning approach for localized fault classification in smart grids. *Swarm Evol. Comput.*, 39:267–278, 2018.

[71] Nikolaos Sapountzoglou, Jesus Lago, Bart De Schutter, and Bertrand Raison. A generalizable and sensor-independent deep learning method for fault detection and location in low-voltage distribution grids. *Applied Energy*, 276:115299, 2020.

[72] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.

[73] Md Shafiullah and M. A. Abido. S-transform based FFNN approach for distribution grids fault detection and classification. *IEEE Access*, 6:8080–8088, 2018.

[74] Shenxing Shi, Beier Zhu, Aoyu Lei, and Xinzhou Dong. Fault location for radial distribution network via topology and reclosure-generating traveling waves. *IEEE Trans. Smart Grid*, 10(6):6404–6413, 2019.

[75] Shenxing Shi, Beier Zhu, Sohrab Mirsaeidi, and Xinzhou Dong. Fault classification for transmission lines based on group sparse representation. *IEEE Trans. Smart Grid*, 10(4):4673–4682, 2019.

[76] Elham Shirazi and Shahram Jadid. Autonomous self-healing in smart distribution grids using agent systems. *IEEE Trans. Ind. Informatics*, 15(12):6291–6301, 2019.

[77] Qun Song, Rui Tan, Chao Ren, and Yan Xu. Understanding credibility of adversarial examples against smart grid: A case study for voltage stability assessment. In Herman de Meer and Michela Meo, editors, *e-Energy '21: The Twelfth ACM International Conference on Future Energy Systems, Virtual Event, Torino, Italy, 28 June - 2 July, 2021*, pages 95–106. ACM, 2021.

[78] Aris Spanos. *Probability Theory and Statistical Inference: Empirical Modeling with Observational Data*. Cambridge University Press, 2019.

[79] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[80] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian J. Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In Yoshua Bengio and Yann LeCun, editors, *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

[81] Jiwei Tian, Buhong Wang, Zhen Wang, Kunrui Cao, Jing Li, and Mete Ozay. Joint adversarial example and false data injection attacks for state estimation in power systems. *IEEE Transactions on Cybernetics*, 2021.

[82] Veerapandiyan Veerasamy, Noor Izzri Abdul Wahab, Rajeswari Ramachandran, Mariammal Thirumeni, Chitra Subramanian, Mohammad Lutfi Othman, and Hashim Hizam. High-impedance fault detection in medium-voltage distribution network using computational intelligence-based classifiers. *Neural Comput. Appl.*, 31(12):9127–9143, 2019.

[83] Y. Vorobeychik, M. Kantarcioglu, R. Brachman, P. Stone, and F. Rossi. *Adversarial Machine Learning*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers, 2018.

[84] Bin Wang, Jianzhao Geng, and Xinzhou Dong. High-impedance fault detection based on nonlinear voltage-current characteristic profile identification. *IEEE Trans. Smart Grid*, 9(4):3783–3791, 2018.

[85] Xiaowei Wang, Jie Gao, Xiangxiang Wei, Guobing Song, Lei Wu, Jingwei Liu, Zhihui Zeng, and Mostafa Kheshti. High impedance fault detection method based on variational mode decomposition and teager-kaiser energy operators for distribution network. *IEEE Trans. Smart Grid*, 10(6):6041–6054, 2019.

[86] Xindong Wu, Vipin Kumar, J Ross Quinlan, Joydeep Ghosh, Qiang Yang, Hiroshi Motoda, Geoffrey J McLachlan, Angus Ng, Bing Liu, S Yu Philip, et al. Top 10 algorithms in data mining. *Knowledge and information systems*, 14(1):1–37, 2008.

[87] Min Xia, Haidong Shao, Xiandong Ma, and Clarence W de Silva. A stacked gru-rnn-based approach for predicting renewable energy and electricity load for smart grid operation. *IEEE Transactions on Industrial Informatics*, 17(10):7050–7059, 2021.

[88] James Jian Qiao Yu, Yunhe Hou, Albert Y. S. Lam, and Victor O. K. Li. Intelligent fault detection scheme for microgrids with wavelet-based deep neural networks. *IEEE Trans. Smart Grid*, 10(2):1694–1703, 2019.

[89] Alireza Zarreh, HungDa Wan, Yooneun Lee, Can Saygin, and Rafid Al Janahi. Risk assessment for cyber security of manufacturing systems: A game theory approach. *Procedia Manufacturing*, 38:605–612, 2019.

[90] Qinghua Zhang, Qin Xie, and Guoyin Wang. A survey on rough set theory and its applications. *CAAI Trans. Intell. Technol.*, 1(4):323–333, 2016.

[91] Bolei Zhou, Yiyou Sun, David Bau, and Antonio Torralba. Interpretable basis decomposition for visual explanation. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part VIII*, volume 11212 of *Lecture Notes in Computer Science*, pages 122–138. Springer, 2018.