

Article

Fault Diagnosis in Power Generators: A Comparative Analysis of Machine Learning Models

Quetzalli Amaya-Sanchez ¹, Marco Julio del Moral Argumedo ¹, Alberto Alfonso Aguilar-Lasserre ^{1,*},
Oscar Alfonso Reyes Martinez ² and Gustavo Arroyo-Figueroa ^{2,*}

¹ Tecnológico Nacional de México, Instituto Tecnológico de Orizaba, Orizaba 94320, Mexico

² Instituto Nacional de Electricidad y Energías Limpias; Cuernavaca 62490, Mexico; oareyes@ineel.mx

* Correspondence: alberto.al@orizaba.tecnm.mx (A.A.A.-L.); garroyo@ineel.mx (G.A.-F.)

Abstract: Power generators are one of the critical assets of power grids. The early detection of faults in power generators is essential to prevent cutoffs of the electrical supply in the power grid. This work presents a comparative analysis of machine learning (ML) models for the generator fault diagnosis. The objective is to show the ability of simple and ensemble ML models to diagnose faults using as attributes partial discharges and dissipation factor data. For this purpose, a generator fault database was built, gathering information from operational data curated by power generator experts. The hyper-parameters of the ML models were selected using a grid search (GS) and cross-validation (CV) optimization. ML models were evaluated with class imbalance and multi-classification metrics, a correspondence analysis, and model performance by class (fault type). Furthermore, the selected ML model was validated by experts through a diagnosis system prototype. The results show that the gradient boosting model presented the best performance according to the performance metrics among single and ensemble ML models. Likewise, the model showed a good capacity to detect type 3 and 4 faults, which are the most catastrophic failures for the generator and must be detected in a timely manner for prompt correction. This work gives an insight into the need and effort required to implement an online diagnostic system that provides information about the power generator health index to help engineers reduce the time taken to find and repair incipient faults and avoid loss of power generation and catastrophic failures of power generators.

Keywords: generator fault diagnosis; machine learning; diagnosis system; applied intelligence; power systems



Citation: Amaya-Sanchez, Q.; Argumedo, M.J.d.M.; Aguilar-Lasserre, A.A.; Reyes Martinez, O.A.; Arroyo-Figueroa, G. Fault Diagnosis in Power Generators: A Comparative Analysis of Machine Learning Models. *Big Data Cogn. Comput.* **2024**, *8*, 145. <https://doi.org/10.3390/bdcc8110145>

Academic Editor: Domenico Talia

Received: 19 August 2024

Revised: 10 September 2024

Accepted: 17 October 2024

Published: 25 October 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Power generators are the main component for the generation of the electricity that is transmitted on power grids. A power generator is a rotating electrical machine that generates electrical energy through mechanical energy. These assets were designed for stable and non-oscillating operation. Although power generators are very reliable assets, early fault detection is important to avoid power outages on the grid and prevent asset life reduction [1–3]. The economic consequences of a catastrophic failure of an electric generator are estimated in millions of USD. Three main costs are considered: costs for the lost generation and transmission of electricity, costs for the repair or substitution of the damaged generator, and direct operational costs. The generation loss costs can rise to USD 6177.600 (since the cost for loss of generation is around 2.6 USD/kWh in Mexico). The cost for substitution is more than 6 million USD for a 375 MW/17 KV electric generator, which can vary according to the power level, the topology of the electric unit, and the technical characteristics of the generator (materials and design). The direct operational cost is estimated to be around 1,280,000 USD, including a 72 h affectation window (repair of damaged facilities, remediation operations, reserve generator testing

and commissioning, and performing the startup operation of all the generation unit and systems' prior conditions).

Therefore, it is required to determine the health index of a power generator to ensure its reliability and operational availability [4–6]. Condition-based maintenance (CBM) is defined as a maintenance strategy to determine and quantify the condition or health index of an asset and make the best decision of what maintenance process should be performed and when [7–9]. Condition monitoring provides real-time data on the health status of an asset through electrical testing, online and offline diagnostics, and operation and maintenance parameter monitoring, which can be used to optimize maintenance planning and scheduling supported by appropriate standards and expert knowledge. The most effective condition analysis strategy is the assessment by highly skilled experts, but it is costly and time-consuming. Therefore, the utilities look forward to automated analysis tools to support experts and reduce their intervention [10].

A CBM requires the identification of faults through a fault detection and diagnosis system. Generally, the generator failure mainly occurs in the following three areas: the stator, the rotor, and the bearings. Fault diagnosis in a generator focuses on the generator stator winding subsystem, whose integrity is completely related to the reliability of the generator [11]. The stator winding subsystem is exposed to electrical, mechanical, and thermal stresses. These phenomena are considered normal if they were considered during the generator's design; otherwise, they are considered abnormal. Among the abnormal behaviors are fractures, emergency overloading, arc flashes, transient events, and thermal faults, to mention a few [12]. Further, stator winding tests can provide important information regarding generator degradation and behavior at a very low cost, eliminating the need to carry out expensive offline testing.

Among the classical methodologies to determine the generator's health index are the following: DNV KEMA [13], HAP [14], and INEEL [4]. These methodologies use a linear or non-linear method to determine the health index of the generator, where weights are assigned to each variable that is monitored online and offline, the deterioration trends and the main failure mechanisms of the generator. Each variable is classified into degrees of deterioration based on international criteria and expert knowledge and is associated with the generator failure mechanisms. The information is processed using a linear or non-linear method to obtain the health index of the asset, its criticality, risk level, probability of failure, and remaining life. The generator is divided into six subsystems, stator winding, rotor winding, stator core, mechanical components, cooling system, and excitation system, so it can be determined which subsystem is being affected and which failure type is present. The objective is the optimization of the operation and maintenance actions of the generator according to the CBM and the reduction of its technical-economic impact [15].

However, the problem with these methodologies is that they tend to hide some failure modes in a generator, a fact derived from the large number of tests used for diagnosis. Therefore, they are methodologies focused on certain types of failures. The criteria for determining the health index are established based on the impact of the failure that they seek to identify, and in many cases, many tests (data) are required to adjust the expected result. As the amount of information grows, it is difficult to analyze and determine the correlations between variables and types of failure by an expert; in this case, an intelligent system could be useful.

Owing to the machine learning (ML) algorithms with robust capacity in feature extraction and classification, several artificial intelligence (AI) methods have been proposed for electrical asset condition assessment and fault diagnosis. ML algorithms provide high flexibility: they are able to handle linear and non-linear relations, are robust to noise, and provide high fault diagnosis performance [16].

Several ML algorithms have been used for the transformer fault detection and diagnosis [17], such as artificial neural networks (ANN) [18–20], decision tree (DT) [21], Bayes network [22], extreme learning machine (ELM) [23], K-nearest neighbors (KNN) [24,25], logical analysis of data (LDA) [26], logistic regression (LR) [27], support vector machine

(SVM) [25,28,29], boosting and bagging [30], eXtreme Gradient Boosting (XGBoost) [31], stacking ensemble [32], and even state-of-the-art algorithms such as few-shot learning with belief functions [14].

For the power generator fault detection and diagnosis, ML applications are limited, as they are generally focused on the diagnosis of faults in ship or aircraft generators and rotating machines, such as: prediction of shutdown of generating machines using data mining [33], ship diesel generator diagnosis using fuzzy logic [34], bearing fault diagnosis using KNN [35], generator fault diagnosis and open switch fault diagnosis using SVM [36,37], synchronous generator fault diagnosis using ANN [38–40], fault detector for closed-ring power generators using random forest [41], and diesel generator diagnosis and fault detection in wind turbines using deep learning [42,43]. Therefore, the application of ML models for fault detection and diagnosis in power generators of power plants is novel and has not been reported at the time of this publication.

This work presents the application of ML algorithms (simple and ensemble classifiers) to the detection and diagnosis of faults in a power generator (GFDD). The objective is to compare the performance of the ML algorithms for the GFDD problem under equivalent experimental configurations. For this purpose, a database of partial discharges of 196 generators associated with faults is used. We analyze simple and ensemble ML algorithms to solve the GFDD multiclass classification problem and develop a prototype system for GFDD, the results of which are validated by power generator experts. The analysis will allow us to understand whether ML algorithms will be able to solve the real-world problem of generator fault detection and diagnosis and how much effort we can expect to apply an intelligent diagnostic system, mainly when different faults are considered.

The remainder of this paper is organized as follows: Section 2 presents a detailed definition of the materials and methods used for the comparative analysis of ML algorithms, Section 3 presents and discusses the results obtained from the application of individual and ensemble ML algorithms to power generator fault diagnosis, and, finally, the main conclusions are presented in Section 4.

2. Materials and Methods

The machine learning (ML) methodology proposed for the multi-class diagnosis problem in this paper is shown in Figure 1. The ML methodology is composed of six main stages: problem analysis, data collection, data analysis, pre-processing, ML training, model selection, and system implementation.

1. **Problem analysis:** In this stage, the focus is on understanding the work domain, motivations, and objectives, defining the problem, finding possible solutions, and generating a solution proposal. This task should be performed in conjunction with domain experts.
2. **Data collection:** This step consists of the definition and obtention of the raw process data used for fault diagnosis. The data obtained from the tests carried out to determine their state of health are used to diagnose faults in generators, specifically the partial discharge and the dissipation factor data in the stator winding. An electrical expert verifies all the data and associates generator failures with them.
3. **Data analysis:** This step consists of the analysis of the data to obtain as much information as possible. This is achieved through attribute analysis, the application of statistics, data visualization, and the identification of possible transformations and relationships.
4. **Pre-processing:** At this stage, we preprocess the information from the process's raw data using available methods found in the literature: data cleaning, zero measurement replacement, selection and scaling of attributes (natural logarithm and normalization), and feature engineering (process relationships). The purpose of this step is to extract knowledge and improve our understanding of the data, generating a consolidated learning database.

5. **ML Training and model selection:** This step consists of the training and evaluation of the single and ensemble ML models. First, the ML models are optimized using grid search (GS) and cross-validation (CV) procedures. Second, the performance of the ML models is evaluated using multi-class performance measures: κ score, balanced accuracy, and the micro and macro F1-measure. Finally, the most appropriate and reliable algorithm for the GFDD problem is selected.
6. **System implementation:** The last stage consists of implementing the selected intelligent model in a system capable of interacting with domain experts and users, allowing its application in a system or prototype in a relevant environment. Implementing the intelligent ML model in a computer application (system or web service) is essential to validate its application and retune it in an environment with high uncertainty.

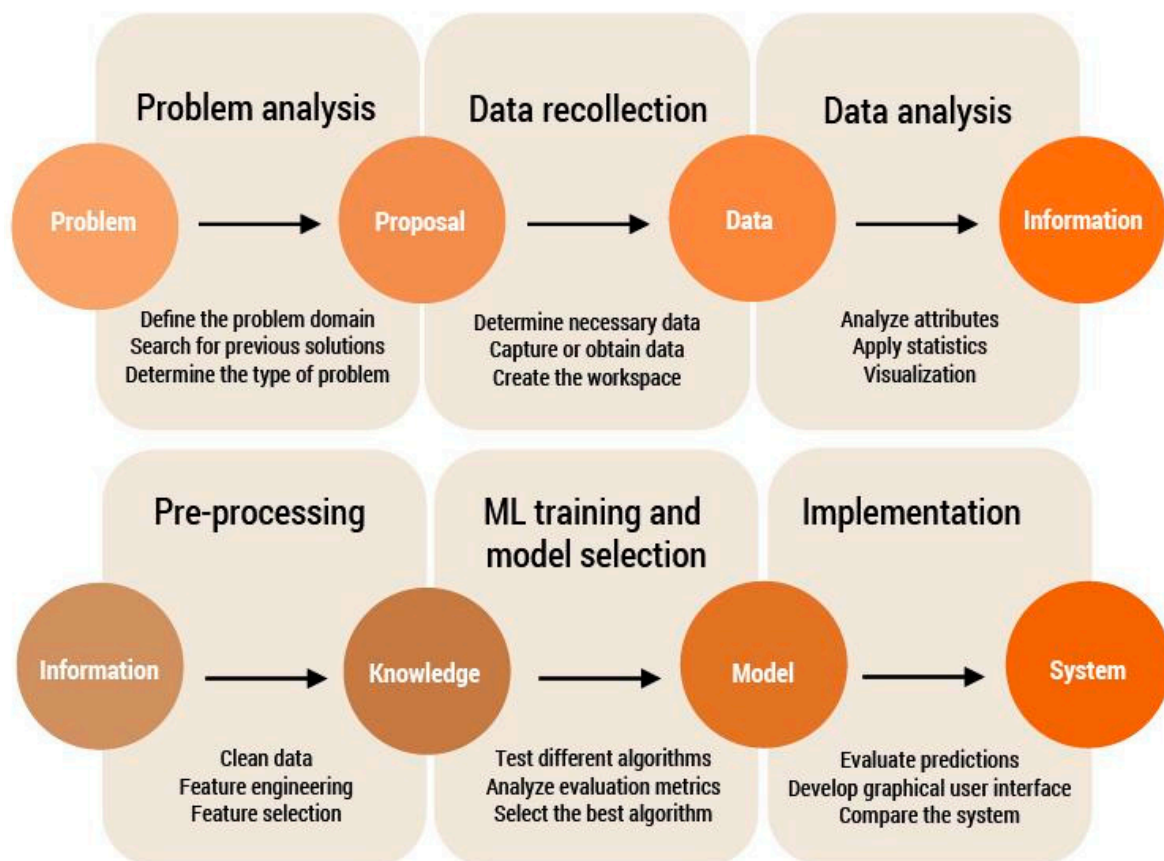


Figure 1. ML methodology proposed for multi-class diagnosis of faults of a power generator.

2.1. Fault Data of the Power Generator

The raw process data for generator fault diagnosis was 196 samples obtained from databases of national utilities and test laboratories [44]. The generator fault database (GFDB) consists of a test of the dissipation factor (DF) and partial electric discharges (PD) of the generator and their corresponding diagnostic. A generator expert validated each sample. The data shows partial discharge activity in the stator windings used to determine the health index of the power generator. The class distribution for the GFDB is shown in Table 1. From the data analysis, the imbalance between the classes stands out, in which the most pre-dominant classes are class 0 and class 1, while the minority classes include class 2, class 3, and class 4. The least severe failures are classes 0 and 1 and occur most frequently.

Table 1. Power generator fault class distribution.

Fault Class	Freq.	%
Class 0. Normal condition	43	22
Class 1. End-winding corona discharges	52	26
Class 2. Surface tracking and phase-to-phase discharges.	33	17
Class 3. Partial discharges	31	16
Class 4. Slot discharges	37	19

2.2. Pre-Processing of Fault Data of the Power Generator

Although experts have curated the GFDB, the database must be initially pre-processed before any fault diagnosis can be carried out with the standard ML framework. The pre-processing stage is composed of two steps: (i) the normalization of values to fit the features to a standard scale and (ii) the removal of any bias in the data to ensure consistency in the learning database. This work's initial GFDB consisted of 18 features and 196 samples, with a label called Diagnostics, which indicates the type of fault. However, not all of these attributes are relevant to the diagnostic task, as they lack relevant information. Therefore, 4 features related to the generator's technical data, such as the name of the plant, unit, test date, and visual observations, were discarded. The final GFDB contains 196 samples and only 14 features, with normalized values and categorical features, effectively converted into numerical representations ready to be used in the training process of the classification algorithms. Table 2 presents the 14 selected characteristics and their definition in terms of nominal voltage (Vn) and phase angle ($\emptyset$). The objective of the data pre-processing stage is to improve the performance of standard ML frameworks for the fault diagnostic problem.

Table 2. Attributes of the generator: dissipation fact (DF) and partial electric discharges (PD).

Feature	Type	Definition
F1	Categorical	Generator cooling system
F2	Categorical	Generator type
F3	Numerical	Dissipation factor (0.2Vn) $\emptyset$ phase A
F4	Numerical	Dissipation factor (0.2Vn) $\emptyset$ phase B
F5	Numerical	Dissipation factor (0.2Vn) $\emptyset$ phase C
F6	Numerical	Dissipation factor (0.6Vn-0.2Vn) $\emptyset$ phase A
F7	Numerical	Dissipation factor (0.6Vn-0.2Vn) $\emptyset$ phase B
F8	Numerical	Dissipation factor (0.6Vn-0.2Vn) $\emptyset$ phase C
F9	Numerical	Partial discharges (nC) $V_n/\sqrt{3}$ $\emptyset$ phase A semicycle (+)
F10	Numerical	Partial discharges (nC) $V_n/\sqrt{3}$ $\emptyset$ phase A semicycle (−)
F11	Numerical	Partial discharges (nC) $V_n/\sqrt{3}$ $\emptyset$ phase B semicycle (+)
F12	Numerical	Partial discharges (nC) $V_n/\sqrt{3}$ $\emptyset$ phase B semicycle (−)
F13	Numerical	Partial discharges (nC) $V_n/\sqrt{3}$ $\emptyset$ phase C semicycle (+)
F14	Numerical	Partial discharges (nC) $V_n/\sqrt{3}$ $\emptyset$ phase C semicycle (−)

2.3. ML Training for Fault Diagnosis of the Power Generator

For training the ML models and selecting the model with the best overall and by-class performance, the GFDB was divided into two databases: training (80% of the data) and test (20% of the data). The proportion of the classes was considered, ensuring that each type of fault kept the same class distribution ratios as in the full DB. The classic methodology used for training and selecting ML models consists of: (i) data pre-processing, (ii) optimization of ML model parameters, (iii) performance evaluation of the ML model, and (iv) selection of the ML model. Ten ML models were selected taking into account the characteristics of the fault diagnosis problem in electrical assets (multi-class classification with unbalanced database) [45] and considering previous work carried out on diagnosis in power transformers [17]: single algorithms (support vector machine (SVM) [46], naive Bayes (NB) [47], linear discriminant analysis (LDA) [48], quadratic discriminant analysis (QDA) [49], decision tree (DT) [50], K-nearest neighbors (KNN) [51], and artificial neural

network (ANN) [52]), and ensemble algorithms (random forest (RF) [53], gradient boosting (GBosst) [54], and extreme gradient boosting (XGBoost) [55]). While single classifiers produce a robust model with good generalization, ensemble classifiers combine multiple machine learning models to produce superior predictive performance compared to a single model, since errors can cancel out when models are combined and robustness is improved because it avoids overfitting noise in the training data. The optimized parameters of the ten ML classification models used are shown in Table 3. A grid search cross-validation procedure with $k = 5$ folds was used.

Table 3. Optimized parameters for the ML classification models.

Model	Parameters
Support vector machines	SVC (probability = True)
Naive Bayes	GaussianNB ()
Linear discriminant analysis	Linear Discriminant Analysis ()
Quadratic discriminant analysis	Quadratic Discriminant Analysis ()
Decision tree	Decision Tree Classifier (random_state = 42, max_depth = 5, min_samples_split = 2)
Random forest	Random Forest Classifier (random_state = 42, max_depth = 5, n_estimators = 50, min_samples_split = 2)
Gradient boosting	Gradient Boosting Classifier (random_state = 42, max_depth = 5, n_estimators = 50, min_samples_split = 2)
XGBoost	XGB Classifier (random_state = 42, max_depth = 5, n_estimators = 50)
K-nearest neighbors	KNeighbors Classifier ()
Artificial neural network	MLP Classifier (alpha = 1, max_iter = 1000)

2.4. Classification Performance Metrics

For the generator fault diagnosis problem, the performance of ML models should be measured using metrics that are sensitive to class imbalance and suitable for analyzing a multi-classification problem [56]. Therefore, the performance metrics used in this work were: a confusion matrix (CM), the balanced accuracy (BA), the F1-score (F1), Cohen's kappa (κ) metric, the Jaccard index (IJ), and Matthews' correlation coefficient (MCC).

MCC, IJ, and κ are metrics robust for assessing the expected performance of classifiers in the presence of class imbalance [57]. CM is a general tool that shows the errors of classifiers in binary, multiclass, and even multilabel scenarios. F1-score is very important for fault generator diagnosis or other electrical assets because a false positive can be crucial for detecting catastrophic failures in these assets [58].

2.5. ML Training and Evaluation Platform

A work environment was used based on Python as a programming language, Jupyter Notebook as a notebook, and Anaconda as a package and framework for the development and evaluation of ML models. For pre-processing, standard Python packages such as numpy [59] and pandas [60] were used. The sklearn [61] package was used for training most ML models, including single and ensemble algorithms. For the case of XGB, the XGBoost [62] package was used.

2.6. System Implementation

Once the ML algorithm has been selected and optimized, the next step is its implementation in a computer system. This is important because the system allows generator experts and users to interact with the system and validate the diagnosis and recommendations obtained. For this purpose, a web service was developed to contain the selected ML algorithm. Web, mobile, and desktop applications can use this web service. This module receives the technical and test data from the generator. It returns the fault diagnosis in text form (according to Table 1), which is interpreted by the interface and shows the diagnosis on the screen and the recommendations according to the diagnosis. Validation in an operation

environment with high uncertainty and noise allows the ML algorithm to be effectively validated, allowing its re-learning process with actual operation conditions.

3. Results

The results of the ML models for the generator fault diagnostic problem are presented in this section. A quantitative comparative analysis was performed to determine the best algorithm, using the performance metrics for class imbalance and multi-classification described in the previous section.

For a more detailed analysis of the performance of the ML models, a comparison of correspondence analyses, the model performance by class (fault type), and an evaluation by the users and experts of the power generator were performed. These evaluations provide a comprehensive view of the ML models' performance in a relevant environment close to the real environment.

3.1. Overall ML Model Performance

Table 4 shows the performance evaluation for ML models using the six performance metrics proposed for the multi-classification problem and class imbalance. The best-performing ML model for each performance metric is highlighted in bold. In general, the best-performing ML model is the gradient boosting model for the six performance metrics. This model outperformed the rest of the ML models, both simple and ensemble. It can be seen that GB presents a slightly better performance than XGB despite the fact that XGB is an improved form of GB. This may occur in multi-classification problems in unbalanced conventional databases, where GB can have a better performance, according to the boosting methods experiments carried out by Jafar Tanha et al. [63], particularly for the MCC, IJ, and κ metrics used for assessing the expected performance of classifiers in the presence of class imbalance, and the F1-macro score metric used for assessing the number of false negatives and false positives in a multiclass classification problem. These results show the ability of the gradient boosting model to handle an imbalanced GFDB. Therefore, the gradient boosting model appears as the model with the best overall performance for fault diagnosis of a power generator.

Table 4. Overall ML classifiers' performance for generator fault detection.

Model	BA	ACC	F1-Score	κ	IJ	MCC
SVM	0.58	0.62	0.57	0.52	0.45	0.56
NB	0.78	0.78	0.78	0.72	0.63	0.73
LDA	0.83	0.82	0.83	0.78	0.70	0.78
QDA	0.78	0.78	0.78	0.72	0.63	0.72
DT	0.81	0.80	0.80	0.75	0.66	0.77
KNN	0.62	0.66	0.62	0.53	0.45	0.55
ANN	0.68	0.65	0.65	0.59	0.61	0.60
RF	0.85	0.85	0.85	0.81	0.74	0.81
GB	0.88	0.88	0.88	0.84	0.78	0.84
XGB	0.86	0.85	0.85	0.81	0.74	0.81

Figure 2a represents the receiver operating characteristic curve (ROC) of the macro averages of the models, and Figure 2b presents the precision-recall curve (PR) of the micro averages of the models. The ROC curve examines the relationship between the sensitivity and specificity of the model. The ROC curve of gradient boosting (0.92) is the curve with the highest performance. The nature of the analysis is crucial for the choice between the ROC curve and the PR Curve. The ROC curve is better when the analysis of false negatives is more relevant than that of false positives, while the PR curve is more appropriate when the analysis of false positives is more relevant. In the case of generator fault diagnosis, the ROC curve was selected because it is necessary to have a high value of true positive rate for good detection of relevant faults.

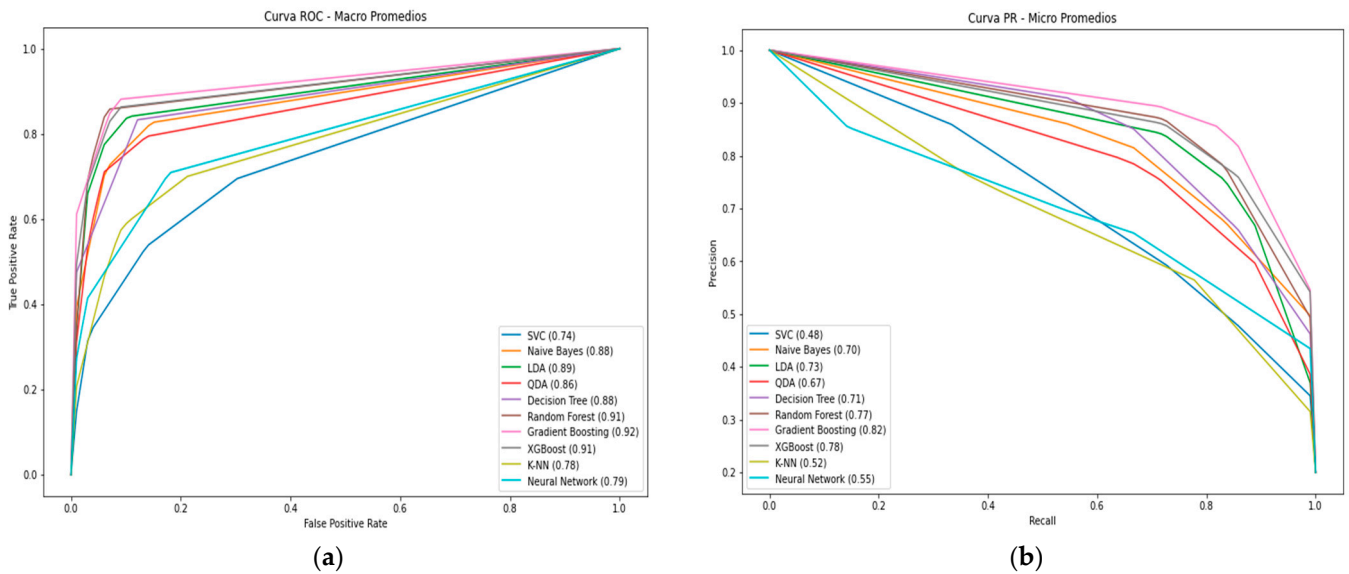


Figure 2. ROC (a) and PR (b) curves for generator fault detection models.

3.2. Correspondence Analysis

The results in Table 4 show that the gradient boosting model has the best performance for identifying faults from the GFDB. However, a comparative analysis of the performance of the other ML models is required. To carry out this comparison, a correspondence analysis is proposed, which is a visualization technique used in data science to compare the relationship of categories, in this case, the performance of ML models [64]. First, the performance metrics to be compared are defined. For this work, the BA and F1-score metrics were used to carry out the analysis because they are the metrics that provide the most outstanding value to the generator fault diagnosis problem. Second, the results of the performance metrics of each ML model are transformed, taking the result of the decision tree model as a baseline, as shown below:

$$N_i(A) = 1 - \frac{M_i(\text{Decision Tree})}{M_i(A)} \quad (1)$$

where $M_i(A)$ is the result of performance metric i for ML model A , M_i is the result of performance metric i for decision tree model, and $N_i(A)$ is the baseline transformed value for the result of performance metric i and ML model A . For instance, for BA and gradient boosting model, the baseline transformed value $BA^{\sim}(GB)$ is obtained, such as:

$$BA^{\sim}(GB) = 1 - \frac{BA(\text{Decision Tree})}{BA(GB)} \quad (2)$$

A transformed value $N_i(A) > 0$ is interpreted as the performance of model A being better than the decision tree model. In contrast, a value $N_i(A) < 0$ is interpreted as the performance of model A being worse than the decision tree model. Figure 3 shows the correspondence analysis with the transformed values of the BA and F1-score metrics with the decision tree model as a baseline.

In Figure 3, the models with better performance than the decision tree model are shown in the upper right quadrant, with blue color, and the worst performing models are shown in the lower left quadrant, with black color. The origin (0, 0) corresponds to the decision tree model. The models that performed better than the decision tree model are GB, XGB, RF, and LDA, with improvements for the BA metric of 3%, 3%, and −10%, respectively, and with improvements for the F1-score metric of 3%, 2%, and 4.5%, respectively. The remaining ML models performed worse than the decision tree model. For instance, the KNN and ANN models performed considerably lower than the decision tree model, with

percentages of transformed values of the BA metric 17% and 14% lower, respectively, and percentages of transformed values of the F1-score metric 5% and 14% lower, respectively.

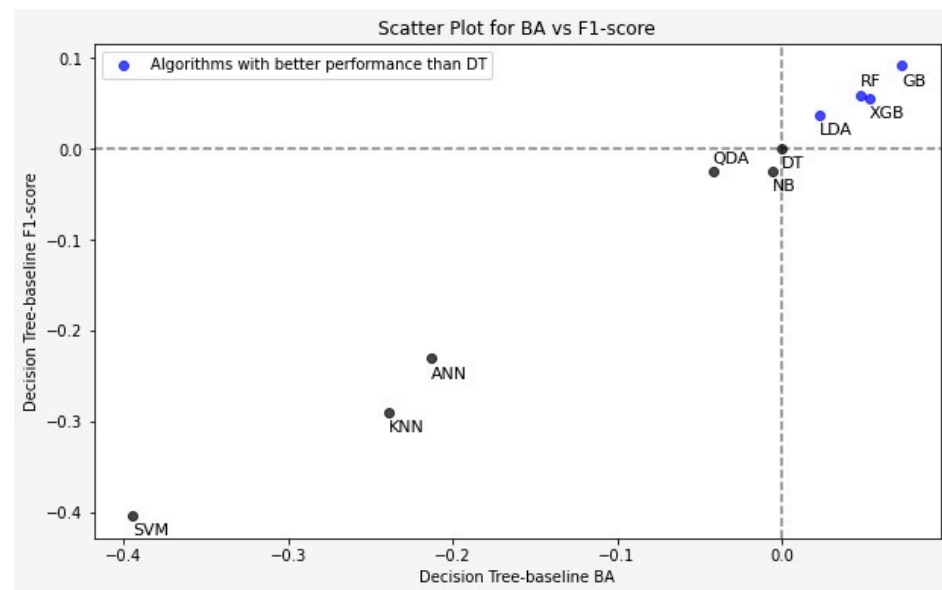


Figure 3. Correspondence analysis using decision tree as a baseline.

3.3. Models Performance by Class (Fault Type)

Based on the analysis of overall performance metrics, the best-performing ML model for generator fault diagnosis is the gradient boosting (GB) algorithm. However, for the generator fault diagnosis problem, it is necessary to know how well it performed for each class (fault type) and analyze how its performance compares to other models, for example, to the best-performing individual model, the DT model. The confusion matrix for the gradient boosting model is presented in Figure 4a,b presents the confusion matrix for the decision tree model. In general, the confusion matrix shows that most fault types were identified with good ($\geq 80\%$) to very good ($\geq 90\%$) accuracy for both models. In contrast, the DT model presents a precision of 55% for type 1 faults and a precision of 67% for type 3 faults. To analyze the performance of an ML model using multiclass CM, it is useful to remember that the rows indicate false negatives (FN) and the columns indicate false positives (FP). Analyzing false negatives (FN), the gradient boosting model shows that 18% of the time, class 1 faults were misclassified as class 2 faults, and 14% of the time, class 2 faults were misclassified as class 1 faults. For the decision tree model, the false negative (FN) analysis shows a higher percentage of misclassification for class 1 faults: 27% of the time as class 2 faults and 18% of the time as class 4 faults; for class 3, it misclassifies them 33% of the time as class 4 faults; and for class 4, it misclassifies them 14% of the time as class 2 fault.

The misclassification of both algorithms may be because the boundaries defined in the feature space for each class require further definition. The feature boundaries can be improved by increasing the sample size of the database, mainly of the imbalanced classes. It is worth noting that the GB model classified with 100% accuracy class 0, 4, and 5 faults. Class 0 (normal condition) and classes 4 and 5 are the most relevant classes (fault types) for the generator diagnosis problem.

Another essential analysis in the generator fault diagnosis is to determine the performance of the algorithms for each class. Table 5 shows the performance of the models for the five classes, using the following as quality parameters: precision, recall, F1-score, and specificity. It can be seen that each algorithm's performance metrics vary by class; for example, an algorithm has a value of 1 for precision and recall for one class and values below 0.8 for other classes. The results show that the GB model does not show the best

performance for all five classes. The relevance of this analysis is in the definition of the most important classes for the diagnosis problem. In this case, the generator experts consider that class 3 and 4 failures are the most important because they are the most catastrophic failures and must be detected promptly. Based on this, the algorithm with the best precision, recall, and specificity performance for classes 3 and 4 is GB. Likewise, for the F1-score, the GB algorithm performs better for classes 3 and 4. The previous analysis is necessary for the generator fault diagnosis and other electrical assets due to the relevance of each type of failure; therefore, not only the global performance of the algorithms must be analyzed.

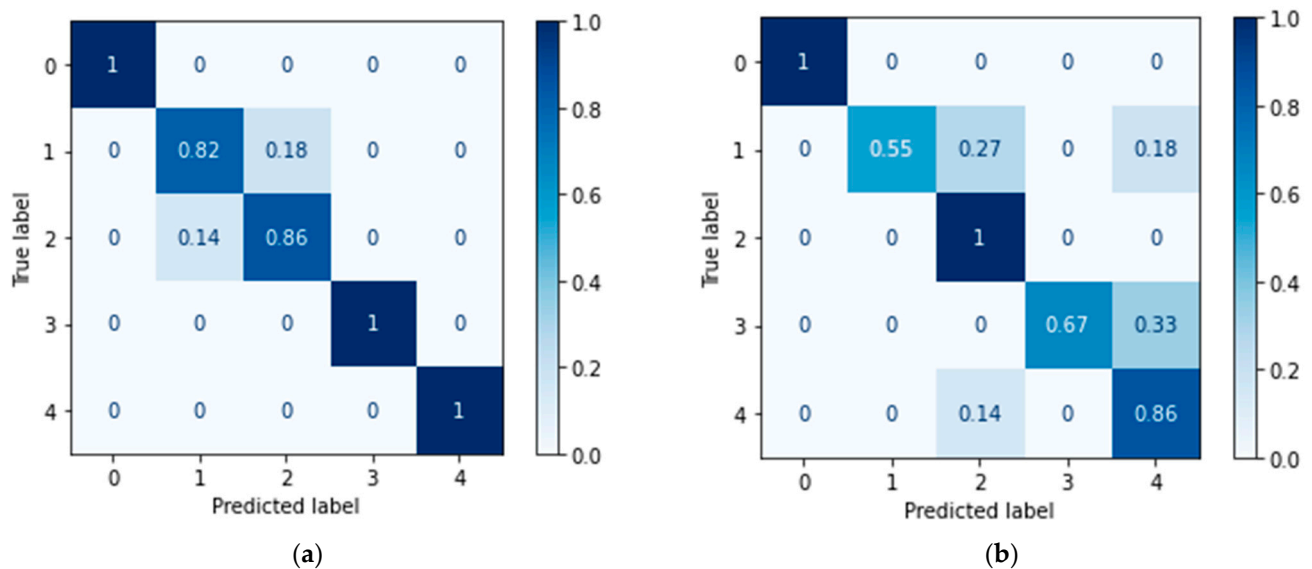


Figure 4. Confusion matrix for (a) gradient boosting and (b) decision tree models.

Table 5. Models' performance by class for generator fault diagnosis.

Model/ Classes	Precision					Recall					F1-score					Specificity				
	0	1	2	3	4	0	1	2	3	4	0	1	2	3	4	0	1	2	3	4
SVM	0.90	0.67	0.00	1.00	0.38	1.00	0.73	0.00	0.33	0.86	0.95	0.70	0.00	0.50	0.52	1.00	0.73	0.00	0.33	0.86
NB	1.00	0.75	0.78	0.50	1.0	0.67	0.55	1.00	0.83	1.00	0.80	0.63	0.88	0.62	1.00	0.66	0.54	1.00	0.84	1.00
LDA	1.00	0.73	0.71	0.83	0.88	0.89	0.73	0.71	0.83	1.00	0.94	0.73	0.71	0.83	0.93	0.88	0.73	0.71	0.84	1.00
QDA	0.89	0.64	0.71	0.67	1.00	0.89	0.64	0.71	0.67	1.00	0.89	0.64	0.71	0.67	1.00	0.88	0.73	0.71	0.84	1.00
DT	1.00	1.00	0.64	1.00	0.60	1.00	0.55	1.00	0.67	0.86	1.00	0.55	1.00	0.67	0.86	1.00	0.54	1.00	0.66	0.86
KNN	1.00	0.57	0.42	0.67	0.60	0.78	0.36	0.71	1.00	0.43	0.88	0.44	0.53	0.80	0.50	0.78	0.36	0.71	1.00	0.43
ANN	1.00	0.55	0.50	0.80	0.54	1.00	0.55	0.14	0.67	1.00	1.00	0.55	0.22	0.73	0.70	1.00	0.54	0.14	0.66	1.00
RF	0.90	0.80	0.71	1.00	0.88	1.00	0.73	0.71	0.83	1.00	0.95	0.76	0.71	0.91	0.93	1.00	0.72	0.71	0.84	1.00
GB	1.00	0.82	0.62	1.00	1.00	1.00	0.82	0.71	1.00	0.86	1.00	0.82	0.86	1.00	1.00	1.00	0.82	0.71	1.00	1.00
XGB	1.00	0.80	0.62	1.00	0.86	1.00	0.73	0.71	1.00	0.86	1.00	0.76	0.67	1.00	0.86	1.00	0.72	0.71	1.00	0.86

3.4. Model Evaluation by Generator Experts

Finally, the performance of the selected ML model was evaluated by users and experts of power generators. The ML model was implemented in a prototype system (web service) capable of capturing data, processing data, determining the fault type, and giving recommendations based on the diagnosis. Users and generator experts used the system to validate the fault diagnosis and its recommendations. Figure 5a shows the data capture interface, and Figure 5b shows the resulting interface: determination of the fault and the suggested recommendation.

Sistema Inteligente para el Diagnóstico de Fallas en Generadores de Potencia (SIDG)

Inicio Usuarios Activos Evaluación Resultados Reportes Administrador de Sistema Cerrar sesión

Enfriamiento

Enfriamiento_AIRE FORZADO

Enfriamiento_AIRE

Enfriamiento_HIDRÓGENO

Enfriamiento_HIDRÓGENO hidrógeno a una presión de 1.2 kg/cm2

Enfriamiento_AIRE FORZADO

FD_tand_A

DP_VnCrT_Ap

DP_VnCrT_An

FD_tand_C

DP_VnCrT_Bp

FD_Dtand_A

DP_VnCrT_Bn

FD_Dtand_B

DP_VnCrT_Cp

FD_Dtand_C

DP_VnCrT_Cn

Enviar

Cancel **Guardar**

Diagnostico

4

Tipo de Falla

Descargas a la ranura (descargas entre la laminación del núcleo y la superficie de bastones)

Recomendaciones

(1) Reemplazo de devanado del estator

(2) Rehabilitar devanado

Acerca de SIDG

(a)

(b)

Figure 5. System interface (a) data capture and (b) diagnosis.

For validation, new data was collected from the process and was not previously validated by experts, which could imply the presence of high levels of noise and uncertainty. A set of 50 samples was used, where a prevalence of data corresponding to fault type 0 and fault type 1 is observed. The 50 new samples were uploaded to the system, and the diagnosis was obtained in each case. On the other hand, the generator experts took each new case and made their diagnostic evaluation. The diagnoses of both cases were compared. The results show a balanced accuracy of 92%, indicating accurate predictions in all categories. Also, it shows a value of 0.89 for Cohen's Kappa index, which suggests substantial agreement between the model predictions and the diagnosis made by generator experts. A Jaccard index of 0.85 indicates a relatively good similarity between the predictions and the experts' diagnosis. On the other hand, a value of 0.89 for the Matthews correlation coefficient shows a close classification between the GB model and the expert diagnosis. Regarding specificity, the model diagnosis shows a specificity of 0.88 in classes 0 and 1, which indicates that it can correctly identify 88% of the negative instances in these classes. Classes 2, 3, and 4 have a high specificity with a value of 1, indicating that the model can avoid misclassifying negative cases as positive. The evaluation of the performance of the generator fault classification model reveals encouraging and promising results. Table 6 presents the cases where the model incorrectly classifies faults. It is essential to carefully examine these instances to understand the discrepancies between the model predictions and the generator expert diagnosis.

Table 6. Classes misclassified by the diagnostic system.

Samples	Diagnosis by Expert	Diagnosis by System
24	Class 1	Class 2
25	Class 1	Class 2
30	Class 0	Class 1
37	Class 0	Class 1

The discrepancies between the diagnoses of the system and the experts, specifically in classifying class 1 as class 2 by the model, are attributed to the similarity in most of their characteristics. The distinction between these classes is more complex, representing an additional challenge for the model. Discrepancies in class 0 instances, where the model

classifies class 0 instances as class 1 instances, reveal the complexity of the diagnosis, especially when critical features share similarities between these classes. It is worth noting the capacity of the model to correctly classify the samples of classes 2, 3, and 4 in a similar way to the experts' diagnosis. The analyses performed provide a comprehensive view of the performance of the ML models, taking into account different metrics, and highlighting their ability to address multiple classes and maintain a balance in prediction accuracy.

4. Conclusions

This paper presents a comparative analysis of ML models, simple and ensemble, applied to the fault detection and diagnosis of power generators under equivalent experimental configurations. The main objective was to compare the performance of ML models and to select the ML algorithm with the best performance, considering the features of the generator fault diagnosis problem. For this purpose, a database of power generator faults associated with partial discharges and the dissipation factor was built. The GFDB was curated by power generator experts. The classic ML methodology used for training and selection of ML models consists of the following steps: (i) data pre-processing, (ii) optimization of ML model parameters using a GS-CV optimization, (iii) performance evaluation of the ML model using the performance metrics for class imbalance and multi-classification, (iv) selection of the ML model with the best performance including correspondence analysis and model performance by class (fault type) to have a detailed look into the types of biases of the models, and (v) validation by power generator experts and users using a computer system.

The results show that the gradient boosting model presented the best performance according to the overall metrics for class imbalance and multi-classification among single and ensemble ML models. In general, single ML models performed worse than ensemble ML models. CM analysis shows that even though the gradient boosting algorithm achieved the highest overall performance metric values, it misclassified some class 1 faults as class 2 faults and some class 2 faults as class 1 faults with percentages above 10%. The misclassification may be because the boundaries defined in the feature space for each class require further definition. The feature boundaries can be improved by increasing the sample size of the database, mainly of the imbalanced classes, and the curing of the samples by the power generator expert through the use of the diagnostic system. The model performance by class shows the importance of this quality metric for the generator fault diagnosis problem, as in the case of the high values in F1-score and BA for classes 3 and 4, the most catastrophic failures for the generator and which must be detected in a timely manner for prompt correction. The development of a prototype system with the ML model selected was essential to achieve validation by users and experts in power generators. This comparative analysis is limited to ML models that require few data for their learning process. Deep learning and other models can be implemented in the future when a larger database of faults is available. However, this analysis has expanded our understanding of the feasibility of applying ML models to solve the generator fault diagnosis problem and has provided insight into how much automation is required to implement an online diagnostic system, especially when it is possible to determine the severity of the fault. In future work, the selected ML model is planned to be incorporated into a condition-based maintenance system to assess the health index of the power generator. The use of an online fault diagnosis system is expected to help engineers reduce the time taken to find and repair incipient faults to avoid catastrophic failures and loss of power generation.

Author Contributions: Conceptualization, A.A.A.-L., M.J.d.M.A. and G.A.-F.; software, Q.A.-S. and M.J.d.M.A.; validation, Q.A.-S., O.A.R.M. and G.A.-F.; investigation, G.A.-F. and O.A.R.M.; data curation, O.A.R.M. and G.A.-F.; writing—original draft, G.A.-F.; writing—review and editing, G.A.-F., A.A.A.-L., M.J.d.M.A., and Q.A.-S.; supervision, G.A.-F. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to being compiled from different sources and validated by power systems experts.

Acknowledgments: The authors would like to thank Carlos Gustavo Azcarraga Ramos from Instituto Nacional de Electricidad y Energías Limpias for his comments and reviews.

Conflicts of Interest: The authors declare no conflicts of interest.

References

- Onu, P.; Mbohwa, C.; Pradhan, A. Machine Learning: A Comprehensive Exploration of Fault Detection and Diagnosis in Smart Grids. In Proceedings of the 2023 International Conference on Electrical, Computer and Energy Technologies (ICECET), Cape Town, South Africa, 16–17 November 2023; pp. 1–6.
- Kumbhar, A.; Dhawale, P.G.; Kumbhar, S.; Patil, U.; Magdum, P. A comprehensive review: Machine learning and its application in integrated power system. *Energy Rep.* **2021**, *7*, 5467–5474.
- Baker, E.; Nese, S.V.; Dursun, E. Hybrid Condition Monitoring System for Power Transformer Fault Diagnosis. *Energies* **2023**, *16*, 1151. [CrossRef]
- Campuzano-Martínez, I.R. Diagnostico de Generadores Electricos de Potencia con Tecnicas de Monitoreo en Linea y Fuera de Linea. *Inf. Tecnol.* **2016**, *27*, 11–20. Available online: <https://scielo.conicyt.cl/pdf/infotec/v27n2/art03.pdf> (accessed on 12 October 2024). (In Spanish). [CrossRef]
- Dashti, R.; Daisy, M.; Mirshekari, H.; Shaker, H.R.; Aliabadi, M.H. A survey of fault prediction and location methods in electrical energy distribution networks. *Measurement* **2021**, *184*, 109947. [CrossRef]
- Yang, B.; Li, F.; Ye, J.; Song, W. Condition Monitoring and Fault Diagnosis of Generators in Power Networks. In Proceedings of the 2019 IEEE Power & Energy Society General Meeting (PESGM), Atlanta, GA, USA, 4–8 August 2019; pp. 1–5.
- Rigatos, G. Condition Monitoring and Fault Diagnosis for Electric Power Generators. In *Intelligent Renewable Energy Systems. Green Energy and Technology*; Springer: Cham, Switzerland, 2016.
- Li, S.; Li, X.; Cui, Y.; Li, H. Review of Transformer Health Index from the Perspective of Survivability and Condition Assessment. *Electronics* **2023**, *12*, 2407. [CrossRef]
- Huang, L. Intelligent Condition Monitoring and Fault Diagnosis of Generator based on Internet of Things and Big Data Technology. In Proceeding of the 2023 IEEE 13th International Conference on Electronics Information and Emergency Communication (ICEIEC), Beijing, China, 14–16 July 2023; pp. 85–89.
- Cocconcilli, M.; Capelli, L.; Cavalaglio Camargo Molano, J.; Borghi, D. Development of a Methodology for Condition-Based Maintenance in a Large-Scale Application Field. *Machines* **2018**, *6*, 17. [CrossRef]
- Brütsch, R.; Tari, M.; Fröhlich, K.; Weiers, T.; Vogelsang, R. Insulation Failure Mechanisms of Power Generators [Feature Article]. *IEEE Electr. Insul. Mag.* **2008**, *24*, 17–25. [CrossRef]
- Krieg-Wezellenburg, M.G.; Krieg, J.H.; Messel, R.; Pant, S.; Montano, O.; Velazquez, G.; Tremblay, R.; Sedding, H.; Shaikh, A.; Delucchi, S.; et al. DDF Measurements on Stator Windings. *CIGRE Technical Brochure*; CIGRE: Paris, France, 2023; p. 918.
- Velasquez, R.M.A.; Lara, J.V.M. Root cause analysis improved with machine learning for failure analysis in power transformers. *Eng. Fail. Anal.* **2020**, *115*, 104684. [CrossRef]
- Xu, Y.; Li, Y.; Wang, Y.; Zhong, D.; Zhang, G. Improved few-shot learning method for transformer fault diagnosis based on approximation space and belief functions. *Expert. Syst. Appl.* **2021**, *167*, 114105. [CrossRef]
- Duque, J.F.P.; Martinez, M.T.V.; Hurtado, A.P.; Carrasco, E.M.; Sancho, B.L.; Krommydas, K.F.; Plakas, K.A.; Karavas, C.G.; Kurashvili, A.S.; Dikaiakos, C.N.; et al. Inter-Area Oscillation Study of the Greek Power System Using an Automatic Toolbox. In Proceedings of the IEEE PES Innovative Smart Grid Technologies Europe (ISGT Europe), Espoo, Finland, 18–21 October 2021; pp. 1–6.
- Arias, R.; Mejia, J. Health index for transformer condition assessment. *IEEE Lat. Am. Trans.* **2018**, *16*, 2843–2849. [CrossRef]
- Santamaria-Bonfil, G.; Arroyo-Figueroa, G.; Zuniga-Garcia, M.A.; Azcarraga Ramos, C.G.; Bassam, A. Power Transformer Fault Detection: A Comparison of Standard Machine Learning and autoML Approaches. *Energies* **2024**, *17*, 77. [CrossRef]
- Nagpal, T.; Brar, Y.S. Artificial neural network approaches for fault classification: Comparison and performance. *Neural Comput. Appl.* **2014**, *25*, 1863–1870. [CrossRef]
- Taha, I.B.; Dessouky, S.S.; Ghoneim, S.S. Transformer fault types and severity class prediction based on neural pattern-recognition techniques. *Electr. Power Syst. Res.* **2020**, *191*, 106899. [CrossRef]
- Mortada, M.A.; Yacout, S.; Lakis, A. Power Transformer Fault Diagnosis Using Neural Network Optimization Techniques. *Mathematics* **2023**, *11*, 4693. [CrossRef]
- Menezes, A.G.C.; Araujo, M.M.; Almeida, O.M.; Barbosa, F.R.; Braga, A.P.S. Induction of Decision Trees to Diagnose Incipient Faults in Power Transformers. *IEEE Trans. Dielectr. Electr. Insul.* **2022**, *29*, 279–286. [CrossRef]
- Carita, A.J.Q.; Leite, L.C.; Medeiros, A.P.P.; Barros, R.; Sauer, L. Bayesian Networks applied to Failure Diagnosis in Power Transformer. *IEEE Lat. Am. Trans.* **2013**, *11*, 1075–1082. [CrossRef]
- Wu, Q.; Zhang, H. A novel expertise-guided machine learning model for internal fault state diagnosis of power transformers. *Sustainability* **2019**, *11*, 1562. [CrossRef]

24. Kumar, A.; Vidya, H.A. Application of k-Nearest Neighbor (kNN) Machine Algorithm for Transformer Fault Classification. *Int. J. Adv. Sci. Technol.* **2020**, *29*, 8441–8448.
25. Sekatane, P.M.; Bokoro, P. Partial Discharge Localization through k-NN and SVM. *Energies* **2023**, *16*, 7430. [\[CrossRef\]](#)
26. Mortada, M.A.; Yacout, S.; Lakis, A. Fault diagnosis in power transformers using multi-class logical analysis of data. *J. Intell. Manuf.* **2014**, *25*, 1429–1439. [\[CrossRef\]](#)
27. Mirowski, P.; Lecun, Y. Statistical machine learning and dissolved gas analysis: A review. *IEEE Trans. Power Deliv.* **2012**, *27*, 1791–1799. [\[CrossRef\]](#)
28. Fei, S.-W.; Zhang, X.-B. Fault diagnosis of power transformer based on support vector machine with genetic algorithm. *Expert Syst. Appl.* **2009**, *36*, 11352–11357. [\[CrossRef\]](#)
29. Zhang, Y.; Wang, Y.; Fan, X.; Zhang, W.; Zhuo, R.; Hao, J.; Shi, Z. An Integrated Model for Transformer Fault Diagnosis to Improve Sample Classification near Decision Boundary of Support Vector Machine. *Energies* **2020**, *13*, 6678. [\[CrossRef\]](#)
30. Khalyasmaa, A.I.; Matrenin, P.V.; Eroshenko, S.A.; Manusov, V.Z.; Bramm, A.M.; Romanov, A.M. Romanov AM. Data Mining Applied to Decision Support Systems for Power Transformers' Health Diagnostics. *Mathematics* **2022**, *10*, 2486. [\[CrossRef\]](#)
31. Wang, T.; Li, Q.; Yang, J.; Xie, T.; Wu, P.; Liang, J. Transformer Fault Diagnosis Method Based on Incomplete Data and TPE-XGBoost. *Appl. Sci.* **2023**, *13*, 7539. [\[CrossRef\]](#)
32. Wang, X.; Han, T. Transformer Fault Diagnosis Based on Stacking Ensemble Learning. *IEEE Trans. Electr. Electron. Eng.* **2020**, *15*, 1734–1739. [\[CrossRef\]](#)
33. González, H.A.; Piedrahita, J.D.; Castrillón, O.D. Predicción de parada de máquinas generadoras en una central hidroeléctrica por medio de minería de datos. *Inf. Tecnol.* **2020**, *31*, 215–222. [\[CrossRef\]](#)
34. Yucesan, M.; Bashan, V.; Demirel, H.; Gul, M. An interval type-2 fuzzy enhanced best–worst method for the evaluation of ship diesel generator failures. *Eng. Fail. Anal.* **2022**, *138*, 106428. [\[CrossRef\]](#)
35. Cui, B.; Weng, Y.; Zhang, N. A feature extraction and machine learning framework for bearing fault diagnosis. *Renew. Energy* **2022**, *191*, 987–997. [\[CrossRef\]](#)
36. Lin, W.-M. Generator Fault Diagnosis with Bit-Coding Support Vector Regression Algorithm. *Energies* **2023**, *16*, 3582. [\[CrossRef\]](#)
37. Ali, M.; Din, Z.; Solomin, E.; Cheema, K.M.; Milyani, A.H.; Che, Z. Open switch fault diagnosis of cascade H-bridge multi-level inverter in distributed power generators by machine learning algorithms. *Energy Rep.* **2021**, *7*, 8929–8942. [\[CrossRef\]](#)
38. Nashee, A.F.; Herez, A.H. Neural Networks for Synchronous Generator Fault Diagnosis. *IOP Conf. Ser. Mater. Sci. Eng.* **2019**, *518*, 042015. [\[CrossRef\]](#)
39. Elfaki, O.E.; Zhong, W.Z.; Hui, L.Q. Condition Monitoring and Faults Diagnosis for Synchronous Generator Using Neural Networks. *Telkomnika Indones. J. Electr. Eng.* **2014**, *12*, 1196–1204.
40. Altaf, S.; Al-Anbuky, A.; GholamHosseini, H. Fault diagnosis in a distributed motor network using Artificial Neural Network. In Proceedings of the International Symposium on Power Electronics, Electrical Drives, Automation and Motion, Ischia, Italy, 18–20 June 2014; pp. 190–197.
41. Quintanilha, I.M.; Elias, V.R.M.; da Silva, F.B.; Fonini, P.A.M.; da Silva, E.A.B.; Netto, S.L.; Apolinário, J.A.; de Campos, M.L.R.; Martins, W.A.; Wold, L.E.; et al. A fault detector/classifier for closed-ring power generators using machine learning. *Reliab. Eng. Syst. Saf.* **2021**, *212*, 107614. [\[CrossRef\]](#)
42. Jo, H.-K.; Kim, S.-H.; Kim, C.-L. Proposal of a new method for learning of diesel generator sounds and detecting abnormal sounds using an unsupervised deep learning algorithm. *Nucl. Eng. Technol.* **2023**, *55*, 506–515. [\[CrossRef\]](#)
43. Helbing, G.; Ritter, M. Deep Learning for fault detection in wind turbines. *Renew. Sustain. Energy Rev.* **2018**, *98*, 189–198. [\[CrossRef\]](#)
44. Sánchez, Q.A. Desarrollo de un Modelo Inteligente Para el Diagnóstico de Fallas en Generadores de Potencia. Master's Thesis, Technological Institute of Mexico Orizaba Campus, Orizaba, Veracruz, Mexico, 2024. (In Spanish).
45. Bacevicius, M.; Paulauskaite-Taraseviciene, A. Machine Learning Algorithms for Raw and Unbalanced Intrusion Detection Data in a Multi-Class Classification Problem. *Appl. Sci.* **2023**, *13*, 7328. [\[CrossRef\]](#)
46. Scholkopf, B.; Smola, A.J.; Bach, F. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*; MIT Press: Cambridge, MA, USA, 2002.
47. Shobha, G.; Rangaswamy, S. Computational Analysis and Understanding of Natural Languages: Principles, Methods and Applications. In *Handbook of Statistics*; Gudivada, V.N., Rao, C., Eds.; Elsevier: Amsterdam, The Netherlands, 2018; Volume 38, pp. 197–228.
48. Xiao, P. *Artificial Intelligence Programming with Python: From Zero to Hero*; Wiley: Hoboken, NJ, USA, 2022.
49. Vapnik, V.N. *Statistical Learning Theory*; Wiley-Interscience: Hoboken, NJ, USA, 1998.
50. Rokach, L.; Maimon, O.Z. *Data Mining with Decision Trees: Theory and Applications*; World Science: Singapore, 2007; p. 69.
51. Mitchell, T.M. *Machine Learning*; McGraw-Hill: New York, NY, USA, 1997.
52. Hastie, T.; Tibshirani, R.; Friedman, J. *The Elements of Statistical Learning, Springer Series in Statistics*; Springer: New York, NY, USA, 2001.
53. Biau, G.; Scornet, E. A random forest guided tour. *Test* **2016**, *25*, 197–227. [\[CrossRef\]](#)
54. Friedman, J.H. Stochastic gradient boosting; Computational statistics & data analysis. *Comput. Stat. Data Anal.* **2002**, *38*, 367–378.
55. Bentéjac, C.; Csörgő, A.; Martínez-Muñoz, G. A comparative analysis of gradient boosting algorithms. *Artif. Intell. Rev.* **2021**, *54*, 1937–1967. [\[CrossRef\]](#)

56. Grandini, M.; Bagli, E.; Visani, G. Metrics for multi-class classification. *arXiv* **2020**, arXiv:2008.05756.
57. Warrens, M.J. Five ways to look at cohen's kappa. *J. Psychol. Psychother.* **2015**, *5*, e197. [[CrossRef](#)]
58. Markoulidakis, I.; Rallis, I.; Georgoulas, I.; Kopsiaftis, G.; Doulamis, A.; Doulamis, N. Multiclass confusion matrix reduction method and its application on net promoter score classification problem. *Technologies* **2021**, *9*, 81. [[CrossRef](#)]
59. Oliphant, T.E. A Guide to NumPy. 2006, Volume 1. Available online: <https://web.mit.edu/dvp/Public/numpybook.pdf> (accessed on 12 October 2024).
60. McKinney, W. Pandas: A foundational python library for data analysis and statistics. *Python High Perform. Sci. Comput.* **2011**, *14*, 1–9.
61. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
62. Chen, T.; Guestrin, C. XGBoost: A scalable tree boosting system. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16, New York, NY, USA, 13–17 August 2016; pp. 785–794.
63. Tanha, J.; Abdi, Y.; Samadi, N.; Razzaghi, N.; Asadpur, M. Boosting methods for multi-class imbalanced data classification: An experimental review. *J. Big Data* **2020**, *7*, 70. [[CrossRef](#)]
64. Freitas, A.A. A critical review of multi-objective optimization in data mining. *ACM SIGKDD Explor. Newsl.* **2004**, *6*, 77–86. [[CrossRef](#)]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.