



Comparison of Different Machine Learning Algorithms and Neural Network for Detecting and Classifying Electric Faults

¹Maqsood Hussain Rather, ²Vivek Gupta

¹Department of Electrical Engineering,

Rayat Bahra University Chandigarh Mohali India .

Author Email: rmaqsoodali557@gmail.com

²Assistant professor department of EE/ECE

Rayat Bahra University Chandigarh Mohali India

Corresponding Author: Vivek.gupta@rayatbahrauniversity.edu.in

Abstract:

Faults in power systems are common, and they may lead to the devastation of costly parts like motors, generators, and transformers. Additionally, these defects have the potential to result in deadly events including explosions, overvoltages, excessive currents that are chosen outages, and even deaths. A power protection system is required to quickly identify, classify, and locate the defects, reducing their effect, in order to remedy these problems. For the purpose of providing constant power supply, minimizing interruptions, and avoiding equipment damage, it is essential to analyze power system failures. This technical report provides a thorough technique for locating, classifying, and identifying power system issues. For accurate fault identification, classification, and fault location determination, the suggested method makes application of machine learning techniques like Convolutional Neural Networks (CNN) and K-Nearest Neighbors (KNN), Random Forest (RF), and Decision Trees. models. Additionally, the technique divides power quality into five different groups.. The goals of identifying defects in power systems, classifying them, pinpointing their locations, and grading power quality are all achieved by this thorough technical explanation. This study employs the CNN, K-Nearest Neighbors, RF, and decision tree algorithms developed in MATLAB/SIMULINK to identify, classify, and localize power system failures on the Test Network

Keywords: Line fault, Ground fault, SVM, Linear Regression, CNN, Machine learning

I. INTRODUCTION

. The number of operating lines and the total length have increased significantly as a result of the electric power infrastructure's recent, fast expansion. These lines are prone to problems because of lightning, short circuits, damaged equipment, improper usage, human error, overloading, and aging. [1]. Any problems that occur on electrical transmission lines must be recognized, In order to do the required repairs and quickly restore electricity, everything must be sorted and located. How well the electricity is supplied will be determined by how long it takes to find the line's trouble point. Therefore, a permanent defect must have an advanced fault detection technique and be precisely located on the wire. It is especially helpful for transient faults, which might indicate a weak spot and be brought on by an insulator that is slightly contaminated or a tree that is swinging or growing below the line [2-3].



The necessity to provide a sophisticated approach for fault analysis in electrical power systems serves as the driving force behind this study. The goal is to precisely identify and categorize all fault types that can occur at different fault sites and fault resistances. The study intends to quickly restore power, understand the reasons of disruptions, and reduce recurrences by using machine learning techniques. The analysis also aims to improve comprehension of protection system elements to enable the implementation of preventative actions and decrease the possibility of service interruptions and equipment damage..

Machine learning techniques including CNN, K-Nearest Neighbors (KNN), Random Forest, Decision Tree, and Support Vector Machine (SVM) have demonstrated promising results in a variety of fields when used for defect diagnosis. These methods are included in the research's suggested solution to increase the precision of defect detection, classification, and placement determination. Additionally, the concept incorporates the division of power quality into various classes, allowing for a thorough comprehension of system behavior.

This study attempts to determine the best method for fault analysis in electrical power systems by evaluating the effectiveness of various machine learning techniques. neural networks with convolutions (CNN) will be used in the assessment and comparison of algorithms, which will give important insights into their advantages and disadvantages in the context of power system failure investigation.

In general, this research aims to overcome the difficulties in fault analysis in power systems and give a thorough approach for reliably identifying, categorizing, and locating fault sites. The findings of this research will help to maintain a steady supply of power, reduce interruptions and equipment damage, and improve the toughness of electrical power systems.

II. LITERATURE REVIEW

The early localisation of problems also enables utility operators to carry out maintenance to resume the usual operation of the electrical system. Successful fault finding and diagnosis guarantee not only quick line reconnection and uninterrupted power supply, but also economic advantages by minimizing revenue loss from unsupplied power and minimizing labor and motor vehicle costs related to manual pinpointing of faults searches (Tzelepis et al., 2020). [9].

Numerous research Publications have been made for enhancing machine learning-based automated defect detection, categorization, and localization. approaches in response to the growing concerns over fault diagnosis. Reputable authors have already published a few review papers. We draw attention to the projects completed in this region. The use of neural networks (NNs) for load prediction, fault diagnostics, economics dispatch, and security in electrical power systems, to name a few closely related problems. evaluation, is highlighted by Haque and Kashtiban (2005) [10].

III. METHODOLOGY.

Analysis of power system faults is crucial for assuring a constant supply of electricity and reducing interruptions and equipment damage. A thorough approach for locating, categorizing, and detecting power system faults is presented in this technical description. The main goals of this study are to identify and categorize all forms of faults occurring at various fault sites and fault resistances, to shed light on the reasons behind interruptions, to quickly restore power, and to reduce recurrences. The analysis also seeks to deepen comprehension of



the protection system's components in order to put preventative measures in place and lessen the possibility of service interruptions and equipment damage.. The suggested approach makes precise use of machine learning methods for defect identification, classification, and placement determination, including K-Nearest Neighbors (KNN), Random Forest, and Decision Tree. In addition, the technique divides power quality into five different categories. The goals of fault detection, classification, fault site determination, and power quality classification in power systems are achieved by this thorough technical explanation.

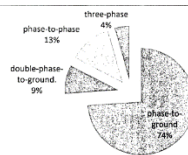


Figure 1 Percentage of fault across the systems

Analysis of power system faults is essential for assuring a steady supply of electricity. This study's objective is to offer a comprehensive framework. for identifying, categorizing, and pinpointing the position of distinct fault types that can occur at various fault sites and with variable fault resistances. The study attempts to reduce further occurrences, identify the root causes of outages, and swiftly restore power. Additionally, knowing the parts of the protective system makes it possible to take preventative action to lessen service interruptions and equipment damage.



Figure . 2 Logical representation of the use of symmetrical structures to produce fault signs

A. Data Collection:

The dataset gathering procedure is the first step in the system architecture for electrical system failure detection and categorization. The electrical system's data is gathered, including crucial elements like G, C, B, A, Ia, Ib, Ic, Va, Vb, Vc, and fault_types. These variables offer vital details regarding the operation of the system and the incidence of faults.

The dataset is preprocessed once it is gathered to make sure it is appropriate for analysis. Data cleansing, addressing missing values, and normalizing are just a few of the preprocessing chores. These procedures aid in removing noise and consistency issues from the dataset, improving its dependability for further research..

Among the machine learning algorithms are Convolutional Neural Networks (CNN), the Random Forest (RF), the k-nearest-neighbor algorithm (KNN), Decision Tree, among others, and Support Vector Machine (SVM). learning techniques used in the system design. Utilizing their skills to examine the information and provide correct predictions, these algorithms are used for defect identification and categorization..



The machine learning algorithms get the information that has been labeled as input. during training, and their internal settings are changed to improve performance. The algorithms can properly identify and categorize various defect kinds because they learn from the data patterns and correlations. To guarantee the generality and resilience of the models, cross-validation methods may be applied.

The assessment procedure utilizing the evaluation dataset starts once the models have been trained. Based on the recognized patterns, the trained models examine the data in the evaluation set and forecast the various failure kinds. To find the best-performing model, each algorithm's performance and accuracy are compared..

Overall, the processes of dataset collection, preprocessing, training, assessment, and fault location detection are integrated into the system architecture. It makes use of several machine learning algorithms to accurately identify, classify, and determine the location of faults in electrical systems, thereby maintaining a constant supply of power, reducing interruptions, and avoiding equipment damage.

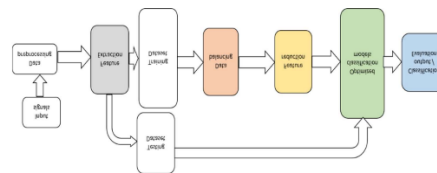


Figure 3 Machine learning approach for fault detection and classification.

Sensors are often put in the power system to monitor several electrical characteristics like voltage, current, and frequency in order to collect the data for problem detection and categorization. After processing the data, properties including phase angle, harmonic distortion, and root mean square are extracted and utilized as input variables in algorithms for problem identification and classification.

The input variables utilized in the dataset for KNN, Random Forest, and Decision Tree algorithms may comprise elements like voltage magnitude, current magnitude, phase angle, frequency, and time-domain characteristics like maximum and lowest values, mean, variance, and skewness. The target variable in the dataset may additionally include phase to ground, phase A to phase B, etc.) and the fault category.

Installing sensors in key areas of the power system as part of the data collecting process requires making sure the data is accurate and trustworthy. Data is pre-processed to reduce noise and identify pertinent information after it has been obtained. The algorithms for defect detection and categorization are then tested. and tested using these characteristics as input variables.

Preprocessing the data will assure its quality before machine learning techniques are applied. The data cleaning, normalization, and feature selection processes we used as part of the preparation for identifying and categorizing defects..

First, we cleansed the data by getting rid of any extraneous or pointless details, addressing missing numbers, and handling outliers. This guarantees the consistency and dependability of the data.



To guarantee that all characteristics are scaled equally, we then normalized the data. This is significant because the magnitude of the input characteristics can affect the performance of various machine learning methods.

Finally, feature selection was used to determine which properties were crucial for identifying and categorizing errors. This entails limiting the amount of features to increase computational efficiency while choosing the features that are most essential to the model's correctness.

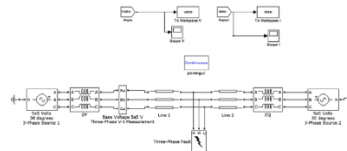


Figure 4 Model overview

Overall, these preprocessing methods aid in enhancing the accuracy & data's accuracy, which enhances the effectiveness of the machine learning algorithms used to identify and categorize problems.. Reduce dimensionality and increase computing effectiveness by using feature selection algorithms. It is necessary to separate this data set into training and testing sections. after preprocessing the data and choosing or engineering the features.. The training dataset is used to develop models based on machine learning, while the testing dataset is used to evaluate their performance. The ratio used to split the data should be carefully chosen, taking into account the necessity for both sufficient training data and a sufficient test set for objective evaluation.

Following the selection the chosen machine learning methods, such as K-Nearest Neighbors (KNN), Random Forest, Decision Tree, or Support Vector Machine (SVM), are trained using the data set being used for training. Using relevant evaluation metrics as precision, recall, precision, the F1-score, or the area under the receiver characteristic curve (ROC) line, the trained models are then assessed on the testing dataset. Other validation methods, such as cross-validation, which is a useful tool for performance evaluation.

The full documentation of the whole data collecting

B. Fault Types and Classification:

The technique takes into account all types of faults, including phase-to-ground faults, defects within phases (which might include phase A and phase B to ground), faults within all three periods, and three-phase asymmetric faults.

Utilizing machine learning techniques like Random Forest and Decision Tree to generate the developed solution, and KNN, CNN to effectively forecast and identify problems. These algorithms were selected because of how well they handle multi-class classification jobs..

1. Power Quality Classification:

Class Definition: According to industry standards or particular specifications, power quality is divided into five different classes. These classes include factors that are pertinent to power quality, such as harmonics, transients, and voltage variations.

The power quality classification algorithm uses the Random Forest, Decision Tree, and KNN algorithms due to their applicability for multi-class classification jobs.



Performance Metrics: Reliability of the power source is evaluated using metrics such as accuracy, precision, recall, F1-score, and confusion tensor. classification models. An evaluation of the models' precision in dividing power quality into the designated classes is made, and a comparison analysis is done..

The method that is being provided offers a thorough way for Energy systems fault location, fault site identity, categorization, and reliability of electricity classification. Accurate defect detection and classification are accomplished by using machine learning techniques like KNN, Random Forest, and Decision Tree. Additionally, the technique divides power quality into five different classifications. The thorough technical explanation satisfies the goals of comprehending fault causes, expeditious power restoration, and putting preventative measures in place to reduce recurrences and equipment damage. For scholars and practitioners in the fields of power systems and machine learning, this technical description is an invaluable resource..

IV. SIMULATION AND RESULTS.

Table 1 Results recorded for CNN

	Output (S)	Ia	Ib	Ic	Va	Vb	Vc
0	0	-170.472196	9.219613	161.252583	0.054490	-0.659921	0.605431
1	0	-122.235754	6.168667	116.067087	0.102000	-0.628612	0.526202
2	0	-90.161474	3.813632	86.347841	0.141026	-0.605277	0.464251
3	0	-79.904916	2.398803	77.506112	0.156272	-0.602235	0.445963
4	0	-63.885255	0.590667	63.294587	0.180451	-0.591501	0.411050

The provided code implements a binary classification model using a neural network. The dataset is divided into two sets: a training collection that contains 9600 samples and a testing set that has 2401 items. An input layer, a hidden layer, and an output layer make up the neural network structure. The binary cross-entropy loss function and the Adam optimizer are used to construct the model, which is trained across five iterations. The training process is evaluated using accuracy and loss metrics. The model summary provides an overview of the architecture and parameters. The training history is visualized with accuracy and loss plots, showing the progress of the model during training.

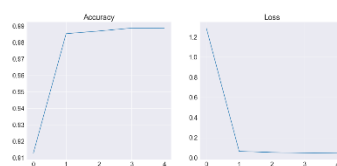


Figure 5. Accuracy and loss for CNN



Predictions: For the testing data (X_{test}), the framework anticipates the objective parameter (y_{pred}), resulting in an array of shape (2401, 1). The shape of the actual target variable (y_{test}) is (2401,).

Threshold Adjustment: The predicted values are thresholded using a cutoff of 0.5. Values less or equal to 0.5 are labeled with a label of 1, as well as beliefs over 0.5 to 0.5 are assigned a label of 0.

Evaluation Metrics: The accuracy, precision, and recall the predicted labels (y_{pred}) and the real ones (y_{test}) are used to determine the scores. The degree of accuracy evaluates how well the predictions were made overall, whereas the accuracy and recall ratings concentrate on how well the students in the favorable group performed (1).. The results are displayed as percentages.

Confusion Matrix: The The predicted labels (y_{pred}) and the real ones (y_{test}) are used to calculate the perplexity matrix. True negatives (TN), positives that are false (FP), false negatives (FN), and true promotions (TP) are all represented by counts.

Report on Classification refers to At each class (0 and 1), a classification report offers accuracy, recall, an F1-score, and support. It also includes the accuracy, macro average, and weighted average metrics.

Overall, the model achieves high accuracy (99.042%) and demonstrates excellent precision (99.816%) and recall (98.098%) for the positive class (1). The confusion matrix further confirms the model's performance, with very few misclassifications. The classification report provides a comprehensive summary of the model's performance across different metrics and classes.

A. Exploratory Data Analysis - Multiclass Classification



Figure 6 Confusion Matrix

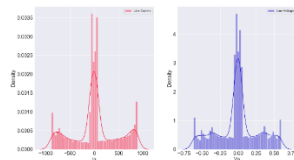


Figure 7 CNN line a voltage and current

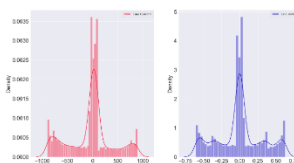


Figure 8 CCNN Line B voltage and current

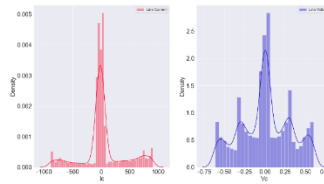


Figure 9 CNN line C voltage and current

B. Multiclass Classification Neural Network Model

Results: [G C B A] [0 0 0 0] No Fault. [1 0 0 1] - LG fault (0 0 1 1) connecting Phase A and Earth - Between Phases A and B, the LL fault [1 0 1 1] - LLG Fault (0 1 1 1) (Between Phases A,B, and Ground) - Between all three stages, LLL Fault [1 1 1 1] Three phase symmetrical fault, LLLG fault

Since output is of four different types, we put them together in one column and there permutations become different output classes

The code snippet creates a new column called 'faultType' in the multi_data dataset by concatenating the values of the 'G', 'C', 'B', and 'A' columns as strings. The resulting column represents different fault types based on the combinations of these four variables.

Table 2 Parameters (G, C, B, A, Ia, Ib, Ic, Va, Vb, Vc) and their corresponding fault types (faultType) for fault detection analysis.

	G	C	B	A	Ia	Ib	Ic	Va	Vb	Vc	faultType
0	1	0	0	1	- 151.29 1812	- 9.6774 52	85.800162	0.400750	-0.132935	-0.267815	1001
1	1	0	0	1	- 336.18 6183	- 76.283 262	18.328897	0.312732	-0.123633	-0.189099	1001
2	1	0	0	1	- 502.89 1583	- 174.64 8023	-80.924663	0.265728	-0.114301	-0.151428	1001
3	1	0	0	1	- 593.94 1905	- 217.70 3359	-124.891924	0.235511	-0.104940	-0.130570	1001
4	1	0	0	1	- 643.66 3617	- 224.15 9427	-132.282815	0.209537	-0.095554	-0.113983	1001

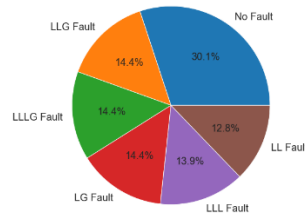


Figure 10 Fault classification

The provided segment performs the following tasks: First, the data is prepared by splitting the 'multi_data' dataframe into input features (X) and the target variable (y). The 'faultType' column is encoded using LabelEncoder and further transformed into categorical one-hot encoded vectors. The data is then split into training and testing sets. A sequential model, 'model2', is defined with specific architecture, and it is trained on the training data for 50 epochs. The training history is stored for analysis. Additionally, two other data files, 'detect_dataset.csv' and 'classData.csv', are loaded into separate dataframes. The head of the 'ddtr' dataframe is displayed.

In [3]:

```
clts.head()
```

Out[3]:

Table 3 Multiple instances or observations. Each row corresponds to a specific measurement instance, and the columns represent the values o

	G	C	B	A	Ia	Ib	Ic	Va	Vb	Vc
0	1	0	0	1	-15.291812	-9.677452	85.800162	0.400750	-0.139393	-0.267881
1	1	0	0	1	-33.6186183	-76.283262	18.328897	0.312732	-0.122363	-0.118999



	G	C	B	A	Ia	Ib	Ic	Va	Vb	Vc
2	1	0	0	1	- 50 2. 89 15 83	- 17 4. 64 80 23	- 80 .9 24 66 3	0. 2 6 5 7 2 8	- 0. 1 1 4 3 0 1	- 0. 1 5 1 4 2 8
3	1	0	0	1	- 59 3. 94 19 05	- 21 7. 70 33 59	- 12 4. 89 19 24	0. 2 3 5 5 1 1	- 0. 1 0 4 9 4 0	- 0. 1 3 0 5 7 0
4	1	0	0	1	- 64 3. 66 36 17	- 22 4. 15 94 27	- 13 2. 28 28 15	0. 2 0 9 5 3 7	- 0. 0 9 5 5 5 4	- 0. 1 1 3 9 8 3

In this case, we have two files (datasets), ddtr and clts:

Both ddtr and clts are used to classify flow faults and train the model to identify any form of malfunction.

The code snippet provides information about the datasets used for fault classification. The 'ddtr' dataset contains 12,001 rows and 9 columns, representing the input features and output classes for fault detection. The 'clts' dataset, which is the dataset to be classified, has 7,861 rows and 10 columns.

To prepare the 'ddtr' dataset, the last two columns ('Unnamed: 7' and 'Unnamed: 8') are dropped using the drop() function. The head of the modified 'ddtr' dataframe is displayed to show the remaining columns and their corresponding values.



Table 4 dataset with measurements of electrical system parameters

	O u t p u t (S)	Ia	Ib	Ic	Va	Vb	Vc
0	0	- 170 .47 219 6	9.2 19 61 3	161 .25 258 3	0.0 54 49 0	- 0.6 59 92 1	0.6 05 43 1
1	0	- 122 .23 575 4	6.1 68 66 7	116 .06 708 7	0.1 02 00 0	- 0.6 28 61 2	0.5 26 20 2
2	0	- 90. 161 474	3.8 13 63 2	86. 347 841	0.1 41 02 6	- 0.6 05 27 7	0.4 64 25 1
3	0	- 79. 904 916	2.3 98 80 3	77. 506 112	0.1 56 27 2	- 0.6 02 23 5	0.4 45 96 3
4	0	- 63. 885 255	0.5 90 66 7	63. 294 587	0.1 80 45 1	- 0.5 91 50 1	0.4 11 05 0

C. data types and values of dataset

In [6]:

```
ddtr.info ()
```

```
<class 'pandas.core.frame.DataFrame'>
```



Received: 14-10-2023

Revised: 12-12-2023

Accepted: 20-12-2023

RangeIndex: 12001 entries, 0 to 12000

Data columns (total 7 columns):

#	Column	Non-Null Count	Dtype
---	--------	----------------	-------

0	Output (S)	12001 non-null	int64
1	Ia	12001 non-null	float64
2	Ib	12001 non-null	float64
3	Ic	12001 non-null	float64
4	Va	12001 non-null	float64
5	Vb	12001 non-null	float64
6	Vc	12001 non-null	float64

dtypes: float64(6), int64(1)

memory usage: 656.4 KB

In [9]:

clts.describe()

Out[9]:

If there is any confusion regarding the values of the Line voltages, then let me clarify it that they are most probably in p.u.i.e.

We can convert by multiplying the four 11 10 3 V generator that make up the power system by 1100011000 Volts, given that they used 11k as their basis.

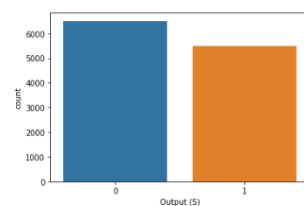


Figure 11 Count vs output

We have a balanced dataset.

For Current Ia

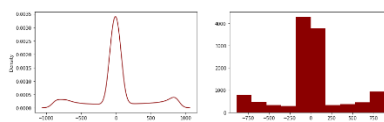


Figure 12 Current of phase A



Received: 14-10-2023

Revised: 12-12-2023

Accepted: 20-12-2023

For Current Ib

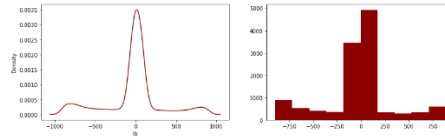


Figure 13 Current of Phase B

For Current Ic

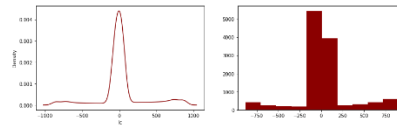


Figure 14 Current of phase C

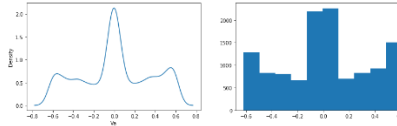


Figure 15 Voltage of Phase A

For Voltage Vb

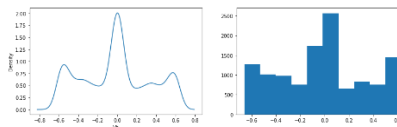


Figure 16 Voltage of Phase B

For Voltage Vc

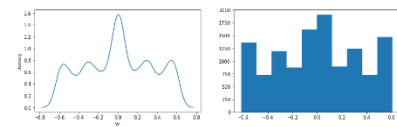


Figure 17 Voltage of phase C

For Line A

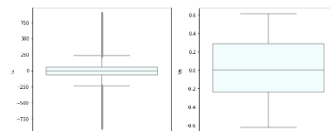


Figure 18 Line A

For Line B

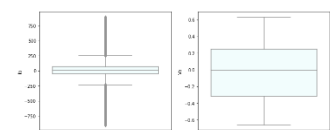


Figure 19 Line B



For Line C

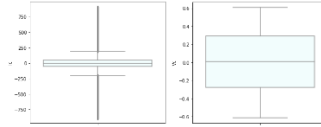


Figure 20 Line C

All the data is somewhat normally distributed.

D. Classification Dataset

Table 5 dataset with measurements and fault types of an electrical system.

	G	C	B	A	Ia	Ib	Ic	Va	Vb	Vc	fault_types
0	1	0	0	1	-151.2911812	-9.67452	85.8001701413279	0.400077535	-0.132278915	-0.2667815	1001
1	1	0	0	1	-336.1861183	-7.6283262183262	18.328897	0.33123633	-0.1123633	-0.11899	1001
2	1	0	0	1	-502.74	-17.04	-80.9	0.261	-0.11	-0.11	1001



	G	C	B	A	Ia	Ib	Ic	Va	Vb	Vc	fault_types
					8 9 1 5 8 3	6 4 8 0 2 3	2 4 6 6 3 3	5 7 2 8	1 4 3 0 1	5 1 4 2 8	
3	1	0	0	1	- 5 9 3. 9 4 1 9 0 5	- 2 1 7. 7 0 3 3 5 9	- 1 2 4. 8 9 1 2 4	0 . 2 3 5 5 1	- 0 . 1 0 4 9 4 0	- 0 . 1 3 0 5 7 0	1 0 0 1
4	1	0	0	1	- 6 4 3. 6 6 3 6 1 7	- 2 2 4. 1 5 9 4 2 7	- 1 3 2. 2 8 2 8 1 5	0 . 2 0 9 5 3 7	- 0 . 0 9 5 5 4	- 0 . 1 1 3 9 8 3	1 0 0 1

E. SVM model

In the provided code, a Support Vector Classifier (SVC) is created with a specific hyperparameter (C=1000). The classifier is trained on the training data (X_train, y_train), and cross-validation scores are computed using the testing data (X_test, y_test). The average score across 10 folds is printed. Furthermore, the confusion matrix for the Support Vector Classifier is plotted using the metrics.plot_confusion_matrix() function, with X_test and y_test as inputs. The resulting confusion matrix is then displayed using plt.show(). Score: 0.9969999999999999

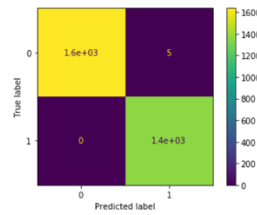


Figure 21 Confusion Matrix 1

F. Decision Tree Model

In the provided code, a Decision Tree Classifier (dct) is created with specific hyperparameters (criterion='gini', ccp_alpha=0.0012). The classifier is trained on the training data (X_train, y_train), and cross-validation scores are computed using the testing data (X_test, y_test). The average score across 10 folds is printed. Additionally, the confusion matrix for the Decision Tree Classifier is plotted using the metrics.plot_confusion_matrix() function, with X_test and y_test as inputs. The resulting confusion matrix is then displayed using plt.show().

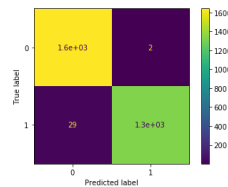


Figure 22 Confusion matrix 2

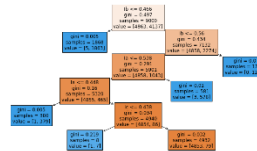


Figure 23 Decision tree pruning

G. KNN Model

In the provided code, a k-nearest neighbors classifier (knn) is created with specific hyperparameters (leaf_size=1, n_neighbors=5, p=1). The classifier is trained on the training data (X_train, y_train), and cross-validation scores are computed using the testing data (X_test, y_test). The average score across 10 folds is printed.

Additionally, the confusion matrix for the k-nearest neighbors classifier is plotted using the metrics.plot_confusion_matrix() function, with X_test and y_test as inputs. The resulting confusion matrix is then displayed using plt.show(). Score: 0.9956666666666667

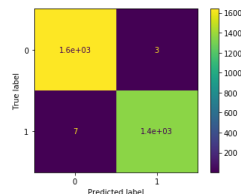


Figure 24 Knn Confusion Matrix



H. Random Forest Classifier

The commented code section suggests the use of grid search with random forest classifier (rf) to find the best hyperparameters. However, the results of the grid search are not provided in the given code.

In the subsequent code, a random forest classifier (rfclf) is created with specific hyperparameters (`max_depth=6`, `min_samples_leaf=10`, `min_samples_split=15`, `n_estimators=10`). The classifier is trained on the training data (`X_train`, `y_train`), and the cross-validation scores are computed using the testing data (`X_test`, `y_test`). The average score across 10 folds is printed.

Furthermore, the confusion matrix for the random forest classifier is plotted using the `metrics.plot_confusion_matrix()` function, with `X_test` and `y_test` as inputs. The resulting confusion matrix is then displayed using `plt.show()`.

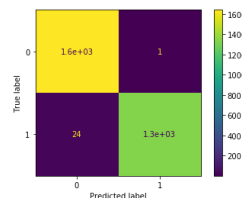


Figure 25 Random Forest Confusion Matrix

It appears that SVM (Support Vector Machine) is performing well in fault detection compared to other models. SVM has shown its capability to efficiently predict signals and identify faults accurately. On the other hand, other models seem to have instances where faults are present but are not being correctly identified. This highlights the strength of SVM in fault detection tasks, showcasing its ability to effectively classify and predict fault conditions..

V. CONCLUSION

..This work used machine learning methods to identify and categorize various failure types in power systems. Faults connecting all three periods, phase to ground faults, the first phase to phase II faults, component I and II to ground faults, and third-phase symmetrical faults are among the faults taken into consideration. Five grades of electricity quality were also established. Several machine learning techniques, including Random Forest, Decision Tree, K-Nearest Neighbors (KNN), and Support Vector Machine (SVM), were used throughout the investigation. The objective was to evaluate how well these algorithms performed in terms of defect identification and categorization. The findings imply that SVM is a solid and trustworthy technique for power system failure identification. It is an effective instrument for assuring power system dependability and reducing downtime due to its capacity to forecast failures of all kinds with efficiency. Effective problem identification enables quick preventative maintenance and remedial action, which improves power system performance and lowers the risk of damage. It's crucial to understand, though, that the effectiveness of machine learning algorithms may change based on the details of the power system being examined. To confirm the results on other power system setups and datasets, more study is required



REFERENCES

- [1]. S. El Safty and A. El-Zonkoly, "Applying Wavelet Entropy Principle in Fault Classification," World Academy of Science, Engineering and Technology, 2018.
- [2]. -V. Malathi, N. Marimuthu, "Multi-class Support Vector Machine approach for fault classification in power transmission line," Energy Technologies, 2008. ICSET 2018. IEEE International Conference on, Nov. 2018.
- [3]. -J. Upendar, C. P. Gupta and G. K. Singh, "Discrete Wavelet Transform and Genetic Algorithm based Fault Classification of Transmission Systems," National Power Systems Conference (NPSC-2008), Indian Institute of Technology, Bombay, India, December 16-18, 2018. 112
- [4]. D. Biswarup and V. REDDY, "Fuzzy-logic-based fault classification scheme for digital distance protection," IEEE transactions on power delivery , vol. 20, 2022.
- [5]. K. Razi, M Hagh, G Ahrabian, " High accurate fault classification of power transmission lines using fuzzy logic," Power Engineering Conference, 2017. IPEC 2017. International,2017.
- [6]. P. K. Dash, S. Mishra, M. M. A. Salama, and A. C. Liew, "Classification of Power System Disturbances Using a Fuzzy Expert System and a Fourier Linear Combiner," IEEE transactions on power delivery, Vol. 15, No. 2, 2020.
- [7]. S. Vasilic; M. kezunovic, "Fuzzy ART neural network algorithm for classifying the power system faults," IEEE transactions on power delivery, Vol. 20, No.1, 2015.
- [8]. -Nan Zhang, M. Kezonovic, "Coordinating fuzzy ART neural networks to improve transmission line fault detection and classification," IEEE PES General Meeting, San Francisco, June 2015.
- [9]. A. K. Pradhan, A. Routray, S. Pati, and D. K. Pradhan, "Wavelet fuzzy combined approach for fault classification of a series-compensated transmission line," IEEE Trans. Power Delivery, vol. 19, no. 4, 2004.
- [10]. -S. R. Samantaray, P. K. Dash and G.Panda, "Transmission Line Fault Detection Using Time-Frequency Analysis," INDICON, 2005 Annual IEEE, Volume, Issue, 11-13 Dec. 2015.
- [11]. -T. Adu, "An Accurate Fault Classification Technique for Power System Monitoring Devices," IEEE transactions on power delivery, Vol. 17, No. 3, 2012.
- [12]. Bretas A.S. and Phadke A.G. (2003): *Artificial neural networks in power system restoration*---- IEEE Trans. Power Delivery, Vol. 18, No. 4, pp. 1181-- 1186.
- [13]. Kezonovic M. And Mrkic J. (1994): *An accurate fault location algorithm using synchronized sampling*. --- Electric Power Syst, Res., Vol. 29, No. 3, pp. 161-- 169.
- [14]. Lian B. And Salama M.M.A. (1994): *An overview of digital fault location algorithms for power transmission lines using transient waveforms*. -- Electric Power Syst. Res., 29., No. 1 pp. 17-25.
- [15]. Sheng L.B. and Elangovan S. (1998): *A fault location algorithm for transmission lines*. -- Electric Machine & Power Syst., Vol. 26, No. 10, pp. 991-1005.



Received: 14-10-2023

Revised: 12-12-2023

Accepted: 20-12-2023

- [16]. Zhang Q., Zhang Y., Song W. And Yu Y. (1999): *Transmission line fault for phase-to-earth fault using one-terminal data.* -- IEE Proc. Trans. Distribution., Vol. 146, No. 2, pp. 121-124.
- [17]. Zaman M.R. and Rahman M.A. (1998): *Experimental testing of artificial neural network based protection of power transformer.* -- IEEE Trans. Power Delivery, Vol. 13, No. 2, pp. 510-515.
- [18]. Purushothama G.K, Narendranath A.U., Thukaram D. And Parthasarathy K. (2001): *ANN applications in fault locators.* -- Electrical Power & Energy Syst., Vol. 23, No. 6, pp. 491-506.
- [19]. Mohamed E.A and Rao N.D. (1995): *Artificial neural network based fault diagnostic system for electric power distribution feeders.* -- Electric Power Syst. Res., Vol. 35, No. 1, pp. 1-10.
- [20]. Novosel D., Hart D.G., Urden E. And Garitty J. (1996): *Unsynchronized-two-terminal fault location estimation.* -- IEEE Trans. Power Delivery, Vol. 11, No. 1. pp.130-138.
- [21]. Osowski S and Salat R. (2002): *Fault location in transmission line using hybrid neural network.* --Compel, Vol 21, No. 1, pp. 18-30.
- [22]. Bouthiba T. And Maun J.C. (2003): *Relis a base de reseaux de neurones pour la protection des lignes de transport a THT.* -- Revue Internationalade de Genie Electricque, Vol. 6, No. 3-4, pp. 413-428.