Optimization Accuracy of 1P-3P Fault Identification System using DT-ML Algorithm

Suresh Aarsey[1], Prof. Ashish Bhargav[2]

[1]*M. Tech. Scholar, Department of Electrical Engineering, Bhabha Engineering Research Institute, Bhopal*

[2]*Assistant Professor, Department of Electrical Engineering, Bhabha Engineering Research Institute, Bhopal*

***Abstract* - Series compensation consists of capacitors in series is used in the transmission lines as a tool to improve the performance after disturbed by a fault. Transmission line needs a protection scheme to protect the lines from faults due to natural disturbances, short circuit and open circuit faults. The fault can happen in any location of transmission line and it is important to know which location has been affected. Therefore, in this paper machine learning (ML) is used to detect and classified the fault happen in single phase (1P) to ground fault and three phase (3P) to ground fault. Two different tests of each types of fault have been tested in order to prove the effectiveness of ML to detect the fault location by using different length and fault resistance. The simulation has been accomplished in Python with ML fitting tool which build and train the network before evaluated its performance using regression analysis. The analysis shows that the decision tree (DT)-ML can accurately detect the different types of faults and classified it into the respective category even the random vectors are put on the system are used.**

***Index Terms* - Single Phase (1P), Three Phase (3P), Fault Detection, Machine Learning.**

I.INTRODUCTION

In recent days, Trustworthy and efficient algorithms and techniques that can provide a correct and accurate analysis of faults on overhead power transmission lines have become necessary in order to implement a modern transmission line protection system. Transmission line protection systems usually operate by identifying the fault and separate only the faulty zone [1, 2]. There are four primary fault types: 1- 3-Phase (Triple line with or without ground) 2- Phase to phase 3- Phase to Ground 4- Double Phase to Ground Three-phase and three-phase to ground faults are similar in terms of electrical quantities (current and short-circuit voltage), which explains the fact that some researchers limit themselves to enumerating 10 types of electrical faults instead of 11 faults. However, in practice, it is important to differentiate them because the short-circuit current is different in both cases and will thus cause different damage to some extent [3, 4]. Restoration or putting into service of an electrical power transmission line after a permanent fault occurrence can only be carried out after maintenance of the electrical power zone which is at fault. The search for faults can be difficult especially over a long transmission line. Thus, it is important to detect and to locate the fault or to evaluate it within minimum error and time. Also quick fault detection can help protect equipment by allowing the disconnection of the power transmission line before any damage occurs. Hence increasing cost savings and power transmission system efficiency and reliability. Due to its simplicity and effectiveness, recent research began to introduce machine learning techniques to fault type and location detection and prediction. Techniques such as multilayer perceptron, random forests, support vector machines were used to locate and predict faults on transmission lines [5, 6]. However, none have introduced the powerful machine learning ensemble methods of Bagging, and Boosting, nor the state of the art the naïve Bayesian classifier and the radial basis function classifier. In addition, a few research was conducted on a fairly long power transmission lines (length > 240km), however none experimented on a length reached 600km [7-10]. In recent years, machine learning algorithms (ML) has been successfully applied in many engineering fields which include computer systems, vision, finance, hospital and medicine, transportation, telecommunications, heuristic classification, aviation, gaming, data mining, speech recognition, and heavy industry [11]. The use of such powerful machine learning techniques in fault prediction could result on enhancing the protection procedures for the power transmission system. In addition it will reduce the time needed to clear the faults, especially for a long transmission line, hence increasing the overall power system reliability and efficiency [9].

II.ELECTRICAL FAULTS

In electrical power systems consisting of generators, transformers, transmission lines and distribution circuits, most of the faults, about two-thirds, are liable to occur in the transmission lines. A fault in a circuit is any failure which interfaces with the normal flow of current. The faults occur in power system due to insulation failure of equipment's, flashover of lines initiated by a lightning stroke, due to permanent damage to conductors and towers or due to accidental faulty operations. The faults can be broadly classified into shunt faults (short circuits) and series faults(open conductors). The shunt fault involves short circuit between conductor and ground or short circuit between two or more conductors. The shunt faults are characterized by increase in current and fall in voltage and frequency. Shunt faults are classified as follows:
1. Single line to Ground fault (LG fault).
2. Line to Line fault (LL fault).
3. Double line to Ground fault (LLG fault).
4. Three phase fault.

The series fault may occur with one or more broken conductors which creates open circuits. It also happens in circuits controlled by fuses or circuit breakers which do not open all phases, i.e., one or two phases of the circuit may open and the other phases may be closed. The series faults are characterized by increase in voltage and frequency and fall in current in the faulted phase. The series faults may be classified as open conductor fault and two open conductor fault. In the faults mentioned above, three phase fault is a symmetrical fault and all other faults are unsymmetrical faults. The symmetrical fault conditions are analyzed on per phase using Thevenin's theorem or by using bus impedance matrix. The unsymmetrical faults are analyzed using symmetrical components. The relative frequency of occurrence of various types of faults in the power systems in the order of decreasing severity is as follows: Three phase faults-5% Double line to Ground fault-10% Line to Line fault-15% Single line to Ground fault-70% Adequate protection has to be afforded to the power system components by incorporating relays and circuit breakers, as the faults may cause interruption in power supply to the consumers; substantial decrease in voltage and frequency, decrease in stability of parallel operation, possibility of drop out of generators and separate generating stations operating in parallel and damage to equipment near the short-circuit points. Circuits should be switched out as warranted by the severity of faults.

Faults in Overhead Transmission Lines

When fault occur, the voltage of the phase on which fault has occur drops and it allows a flow of large current. If this large current left uninterrupted a major damage to the components may occur. The cause of faults is generally, short circuit, mistake in operation, error in equipment's, men error, overload on system or aging of the system.

Nature and Causes of Faults

Either insulation failure or failures of conducting path are the major causes for the occurrence of faults. In addition to this, faults are also caused due to over voltages which are occurring due to switching surges and lightening. Falling of conducting objects on overhead lines, encounter of flying birds, tree branches, direct lightening strokes, ice loading, creepers, storms etc. are the other reasons which can cause different types of faults in overhead lines. Moisture in the soil, heat of earth, ageing of cables may lead to the solid insulation failure in cables, transformers and generators. Types of faults:

- Symmetrical faults
- Unsymmetrical faults

Table 1: Types of faults

Types	Symbol	% of Appearance	Acuteness
Line to Ground	L-G	75-80%	Fall under the category of not so grave
Line to Line	L-L	10-15%	Sever compared to L-G but less than L-L-G
Double Line to Ground	L-L-G	3-10%	Graver
Three phase	3-ϕ	1-5%	Most grave

Line to Ground Fault

When any one of the three lines is in touch with the neutral or ground accidently this type of fault occurs. Hence it is named so. It is the most common type of fault as mentioned already. The possible cause for this fault to occur may be due to the wind flowing with high speed or may be some big object like tree has fall on it or due to the event of lightning.

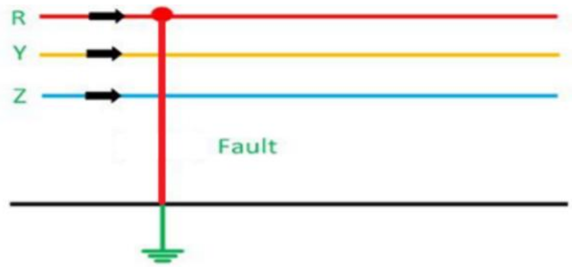

Figure 1: Single Line to Ground Fault

Line to Line Fault

Here two lines are in touch with each other accidently. It can be said that condition is of short circuit. Rest of the details is expressed clearly in figure 2.

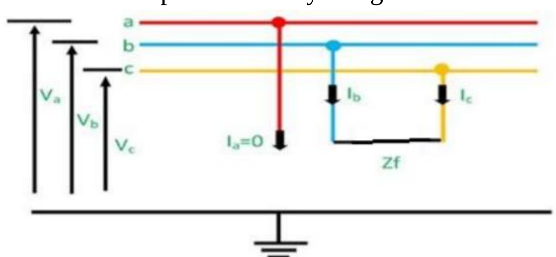

Figure 2: Line to Line Fault

Double Line to Ground Fault

As compared to the L-L fault, here the short circuit condition is between three elements, two of these elements are any two line from given three phases and the third element is ground.

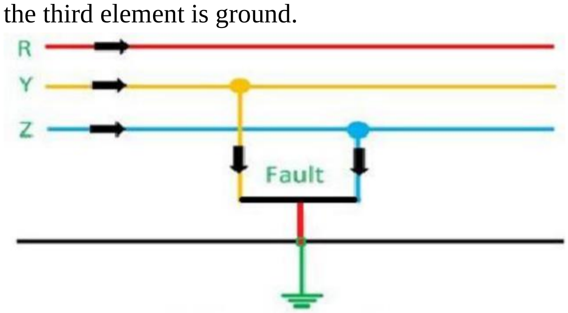

Figure 3: Double Line to Ground Fault

Three Phase to Ground Fault

The most grave but occurrence is most acute, is this fault where all the three phases are touching each other and together in contact with the ground.

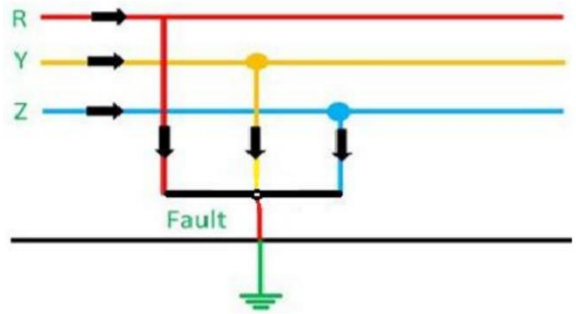

Figure 4: Three Phase Fault

III.PROPOSED METHODOLOGY

Decision tree algorithms divide the dataset into training and testing datasets. It uses a recursive method to construct trees. It recursively partitions a training set to create a trained classifier. The recursion process is completed when the subset at a node has the same prediction value as the target variable. A training dataset consists of a set of instances where each instance is made up of attributes and a class label. An attribute can have ordinal, real, or boolean values. Figure 4.1 illustrates an example of a decision tree. While constructing a decision tree, decision nodes are created which represents a test on selected attributes. For each possible output of the test, one branch from the decision node is created. Hence, the number of branches from the decision node is equal to the possible outputs of the test at that decision node. These decision nodes are called internal nodes (denoted by rectangles). This partitioning of the tree ends at leaf nodes (denoted by ovals). Each leaf in the decision tree represents a predicted class. The decision tree classifier uses a greedy method to construct a tree.

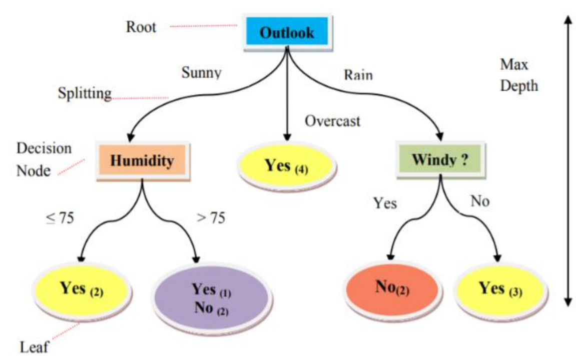

Figure 5: Simple Decision Tree

The greedy approach selects a globally optimal solution at each stage. The selection made by a greedy approach may depend on choices made so far in the construction of a tree. These choices do not depend on future choices or all the solutions of the sub problem. In each iteration, it makes one greedy choice after another. This leads to reducing each specified problem into a smaller one [16]. Choice trees classify examples by navigating the tree from the root to some leaf hub, which gives the order or class expectation of a case. Another case is arranged by beginning at the root hub of the tree, testing the property at the choice hub, then, at that point, crossing down the tree limb relating to the worth of the characteristic as displayed in Figure 5. This course of crossing is then proceeded for the subtree which is established at the new hub.

Information gain is calculated using a probability distribution pi of each nonclass attribute. The training set$Ttrain$ consist of n different classes $C = C1$, $C2$, $C3$, … . Cn . The probability pi that an instance belongs to a class Ci is calculated using Eq. 1

$$p_i = \frac{freq(C_i, T_{train})}{|T_{train}|} \qquad (1)$$

Where, $Ttrain$ is the total number of instances in. The number of instances that belongs to a class Ci is given by , Tt. Information gain of $Ttrain$ is calculated using Eq. 2

$$info(T_{train}) = -\sum_{i=1}^{n} p_i \times \log_2(p_i) \qquad (2)$$

The dataset $Ttrain$ partitioned into s partitions based on domain values of a non-class attribute. The information obtained by this partitioning process is given by Eq. 5.3

$$info(A_i, T_{train}) = -\sum_{j=1}^{s} \frac{|T_{trainj}|}{|T_{train}|} info(T_{trainj}) \qquad (3)$$

The information gain of attribute Ai is calculated using Eq. 4.

$$gain(A_i) = info(T_{train}) - info(A_i, T_{train}) \qquad (4)$$

The $splitinfo\ Ai$ calculates information gained by splitting the training set $Ttrain$ into s subset on test attribute. The $splitinfo\ Ai$ is calculated by Eq. 5.

$$gain\ ratio(A_i) = \frac{gain(A_i)}{splitinfo(A_i)} \qquad (5)$$

The gain ratio of a node is the gain ratio of an attribute tested at that node. The use of the Gain Ratio increases the prediction accuracy of a classifier.

Proposed calculation is a troupe learning method, utilized for order and relapse issues. It can deliver a successful model comprising of powerless students, as a rule choice trees.

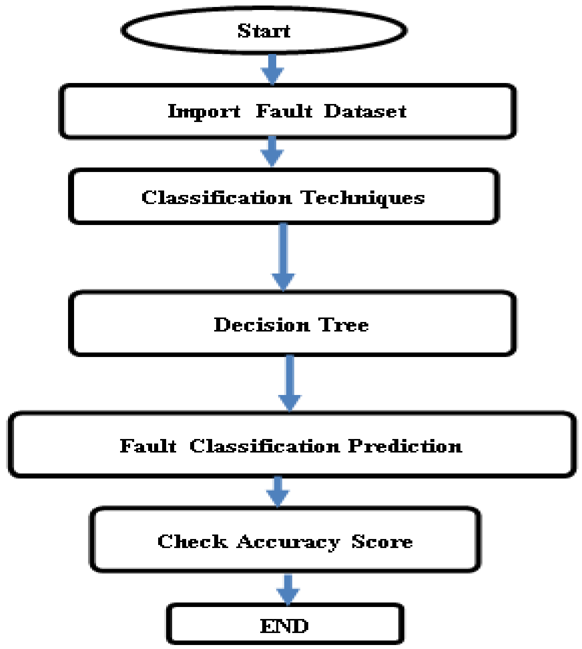

Figure 6: Flow chart of Previous Algorithm

The essential thought of the proposed technique is to assemble and sum up the gathering model in a phase insightful design by improving a true self-assertive misfortune work. The proposed strategy builds its model from the past misfortune capacity of negative angle in an emphasis way. In the ML, limiting the misfortune work is a significant issue and should be enhanced. As such, the misfortune work addresses the contrast between the anticipated result and the objective.

IV.SIMULATION RESULTS

Python is a general programming language and is broadly utilized in a wide range of disciplines like general programming, web improvement, programming advancement, information investigation, M and so on Python is utilized for this venture since it is entirely adaptable and simple to utilize and furthermore documentation and local area support is exceptionally huge.

NumPy is amazingly fantastic group which enables us for coherent enlisting. It goes with refined limits and can perform N-layered display, variable based math, Fourier change, etc NumPy is used where in data examination, picture getting ready and besides exceptional various libraries are worked above NumPy and NumPy goes probably as a base stack for those libraries.

Precision provides a measure of how accurate your model is in predicting the actual positives out of the total positives predicted by your system. Recall provides the number of actual positives captured by our model by classifying these as true positive. F-measure can provide a balance between precision and recall, and it is preferred over accuracy where data is unbalanced.

Therefore, F-measure was utilized in this study as a performance metric to provide a balanced and fair measure using the formula in (6).

$$Precision = \frac{TP}{TP + FP} \quad (6)$$

$$Recall = \frac{TP}{TP + FN} \quad (7)$$

Where,

TP—True Positive, FP—False Positive, FN—False Negative

We should now characterize the most essential terms, which are entire numbers (not rates):

True Positives (TP): These are cases in which we expected yes (they have the sickness), and they do have the disease.

True Negatives (TN): We anticipated no, and they don't have the sickness.

```
df = pd.DataFrame()
for i in ['/content/drive/MyDrive/DATASET_FAULT_EXCEL/svmBGFAULT100.xlsx',
'/content/drive/MyDrive/DATASET_FAULT_EXCEL/svmBGFAULT200.xlsx',
'/content/drive/MyDrive/DATASET_FAULT_EXCEL/svmBGFAULT50.xlsx',
'/content/drive/MyDrive/DATASET_FAULT_EXCEL/svmBGFAULT75.xlsx',
'/content/drive/MyDrive/DATASET_FAULT_EXCEL/svmCAFAULT100.xlsx',
'/content/drive/MyDrive/DATASET_FAULT_EXCEL/svmCAFAULT150.xlsx',
'/content/drive/MyDrive/DATASET_FAULT_EXCEL/svmCAFAULT200.xlsx',
'/content/drive/MyDrive/DATASET_FAULT_EXCEL/svmCAGFAULT100.xlsx',
'/content/drive/MyDrive/DATASET_FAULT_EXCEL/svmCAGFAULT150.xlsx',
'/content/drive/MyDrive/DATASET_FAULT_EXCEL/svmCAGFAULT200.xlsx',
'/content/drive/MyDrive/DATASET_FAULT_EXCEL/svmCGFAULT100.xlsx',
'/content/drive/MyDrive/DATASET_FAULT_EXCEL/svmCGFAULT150.xlsx',
'/content/drive/MyDrive/DATASET_FAULT_EXCEL/svmCGFAULT200.xlsx',
'/content/drive/MyDrive/DATASET_FAULT_EXCEL/svmCGFAULT50.xlsx',
'/content/drive/MyDrive/DATASET_FAULT_EXCEL/svmCGFAULT75.xlsx',
'/content/drive/MyDrive/DATASET_FAULT_EXCEL/svmNOFAULT.xlsx'
]:
```

```
dictt_3 = dict()
dictt_3['BGFAULT50'] = 'BGFAULT'
dictt_3['BGFAULT75'] = 'BGFAULT'
dictt_3['BGFAULT100'] = 'BGFAULT'
dictt_3['BGFAULT200'] = 'BGFAULT'
dictt_3['CAFAULT100'] = 'CAFAULT'
dictt_3['CAFAULT150'] = 'CAFAULT'
dictt_3['CAFAULT200'] = 'CAFAULT'
dictt_3['CAGFAULT100'] = 'CAGFAULT'
dictt_3['CAGFAULT150'] = 'CAGFAULT'
dictt_3['CAGFAULT200'] = 'CAGFAULT'
dictt_3['CGFAULT50'] = 'CGFAULT'
dictt_3['CGFAULT75'] = 'CGFAULT'
dictt_3['CGFAULT100'] = 'CGFAULT'
dictt_3['CGFAULT150'] = 'CGFAULT'
dictt_3['CGFAULT200'] = 'CGFAULT'
dictt_3['NOFAULT'] = 'NOFAULT'
accuracy_score(Y_Test, Y_RF_pred)

    0.8664169787765293
```

Table II: Fault Classification for LG Fault

Fault Type	Resistance	Accuracy
AG ,BG,CG	25Ω	47.17
AG ,BG,CG	25Ω,50 Ω	82.89
AG ,BG,CG	25Ω,50Ω ,75Ω	89.03
AG ,BG,CG	25Ω,50 Ω ,75Ω,100 Ω	92.93
AG ,BG,CG	25Ω,50 Ω ,75Ω,100 Ω ,150Ω	93.88
AG ,BG,CG	25Ω,50 Ω, 75Ω,100 Ω ,150Ω 200Ω	95.15
AG ,BG,CG	25Ω,50 Ω, 75Ω,100 Ω ,150Ω 200Ω,300Ω	94.44

Table III: Fault Classification for Ll Fault

Fault Type	Resistance	Accuracy
AC ,BC,CA	25Ω	49.50
AC ,BC,CA	25Ω,50 Ω	83.55
AC ,BC,CA	25Ω,50Ω ,75Ω	90.25
AC ,BC,CA	25Ω,50 Ω ,75Ω,100 Ω	92.76
AC ,BC,CA	25Ω,50 Ω ,75Ω,100 Ω ,150Ω	93.94
AC ,BC,CA	25Ω,50 Ω, 75Ω,100 Ω ,150Ω 200Ω	94.68
AC ,BC,CA	25Ω,50 Ω, 75Ω,100 Ω ,150Ω 200Ω,300Ω	94.15

Table IV: Fault Classification for LIG Fault

Fault Type	**Resistance**	**Accuracy**
ACG ,BCG,CAG	25Ω	51.16
ACG ,BCG,CAG	25Ω,50 Ω	85.38
ACG ,BCG,CAG	25Ω,50Ω ,75Ω	90.47
ACG ,BCG,CAG	25Ω,50 Ω ,75Ω,100 Ω	92.43
ACG ,BCG,CAG	25Ω,50 Ω ,75Ω,100 Ω ,150Ω	93.35
ACG ,BCG,CAG	25Ω,50 Ω, 75Ω,100 Ω ,150Ω 200Ω	93.40
ACG ,BCG,CAG	25Ω,50 Ω, 75Ω,100 Ω ,150Ω 200Ω,300Ω	93.44

Table V: Fault Classification for LLG and LLLG Fault

Fault Type	Resistance	Accuracy
ABCG ,ABC	25Ω	38.40
ABCG ,ABC	25Ω,50 Ω	74.06
ABCG ,ABC	25Ω,50Ω ,75Ω	71.07
ABCG ,ABC	25Ω,50 Ω ,75Ω,100 Ω	78.55
ABCG ,ABC	25Ω,50 Ω ,75Ω,100 Ω ,150Ω	80.09
ABCG ,ABC	25Ω,50 Ω, 75Ω,100 Ω ,150Ω 200Ω	81.96
ABCG ,ABC	25Ω,50 Ω, 75Ω,100 Ω ,150Ω 200Ω,300Ω	81.36

```
[4] df = pd.DataFrame()
    for i in ['/content/drive/MyDrive/AB Fault/ABFAULT25.xlsx','/content/drive/MyDrive/BC Fault/BCFAULT25.xlsx','/content/drive/MyDrive/CA Fault/CAFAULT25.xlsx']:
      a = pd.read_excel(i)
      a.rename(columns = {'Fault Type': 'Fault_Type'}, inplace = True)
      df = pd.concat([df, a])
```

```
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report

accuracy_score(Y_Test, Y_RF_pred)

0.4950166112956811
```

25 Ω				Accuracy
Fault	AB	BC	CA	49.50

```
[19] df = pd.DataFrame()
    for i in ['/content/drive/MyDrive/AB Fault/ABFAULT50.xlsx','/content/drive/MyDrive/BC Fault/BCFAULT50.xlsx','/content/drive/MyDrive/CA Fault/CAFAULT50.xlsx']:
      a = pd.read_excel(i)
      a.rename(columns = {'Fault Type': 'Fault_Type'}, inplace = True)
      df = pd.concat([df, a])
```

```
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report

accuracy_score(Y_Test, Y_RF_pred)

0.4883720930232558
```

25 Ω				Accuracy
Fault	AB	BC	CA	48.83

```
[34] df = pd.DataFrame()
    for i in ['/content/drive/MyDrive/AB Fault/ABFAULT25.xlsx','/content/drive/MyDrive/BC Fault/BCFAULT25.xlsx','/content/drive/MyDrive/CA Fault/CAFAULT25.xlsx',
              '/content/drive/MyDrive/ABC Fault/ABCFAULT25.xlsx','/content/drive/MyDrive/ABCG Fault/ABCGFAULT25.xlsx','/content/drive/MyDrive/ABG Fault/ABGFAULT25.xlsx',
              '/content/drive/MyDrive/BCG Fault/BCGFAULT25.xlsx','/content/drive/MyDrive/BG Fault/BGFAULT25.xlsx','/content/drive/MyDrive/CAG Fault/CAGFAULT25.xlsx',
              '/content/drive/MyDrive/CG Fault/CGFAULT25.xlsx','/content/drive/MyDrive/AG Fault/AGFAULT25.xlsx']:
      a = pd.read_excel(i)
      a.rename(columns = {'Fault Type': 'Fault_Type'}, inplace = True)
      df = pd.concat([df, a])
```

```
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report

accuracy_score(Y_Test, Y_RF_pred)

0.2148685403445149 7
```

```
[50] df = pd.DataFrame()
    for i in ['/content/drive/MyDrive/AB Fault/ABFAULT25.xlsx','/content/drive/MyDrive/BC Fault/BCFAULT25.xlsx','/content/drive/MyDrive/CA Fault/CAFAULT25.xlsx',
              '/content/drive/MyDrive/ABC Fault/ABCFAULT25.xlsx','/content/drive/MyDrive/ABCG Fault/ABCGFAULT25.xlsx','/content/drive/MyDrive/ABG Fault/ABGFAULT25.xlsx',
              '/content/drive/MyDrive/BCG Fault/BCGFAULT25.xlsx','/content/drive/MyDrive/BG Fault/BGFAULT25.xlsx','/content/drive/MyDrive/CAG Fault/CAGFAULT25.xlsx',
              '/content/drive/MyDrive/CG Fault/CGFAULT25.xlsx','/content/drive/MyDrive/AG Fault/AGFAULT25.xlsx','/content/drive/MyDrive/AB Fault/ABFAULT50.xlsx',
              '/content/drive/MyDrive/BC Fault/BCFAULT50.xlsx','/content/drive/MyDrive/CA Fault/CAFAULT50.xlsx',
              '/content/drive/MyDrive/ABC Fault/ABCFAULT50.xlsx','/content/drive/MyDrive/ABCG Fault/ABCGFAULT50.xlsx','/content/drive/MyDrive/ABG Fault/ABGFAULT50.xlsx',
              '/content/drive/MyDrive/BCG Fault/BCGFAULT50.xlsx','/content/drive/MyDrive/BG Fault/BGFAULT50.xlsx','/content/drive/MyDrive/CAG Fault/CAGFAULT50.xlsx',
              '/content/drive/MyDrive/CG Fault/CGFAULT50.xlsx','/content/drive/MyDrive/AG Fault/AGFAULT50.xlsx']:
      a = pd.read_excel(i)
      a.rename(columns = {'Fault Type': 'Fault_Type'}, inplace = True)
      df = pd.concat([df, a])
```

```
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report

accuracy_score(Y_Test, Y_RF_pred)

0.6699900933816863 1
```

V.CONCLUSION

A three phase series compensated transmission line with ANN has been developed to recognize the fault location. The values of validation performance and error value of 4 tests shows that ML able to run the system effectively due to the values are near to zero. It is proven that the values are closer to zero which means, the system is around the line of best fit. Four tests are run using ML have been proved that the method can accurately detect the different types of faults and classified it into the respective category even the random vectors are put on the system as a fault which can happen in any location on the system. As a conclusion, the ML model is successfully developed to localized and classified the fault happen in the three phase transmission line system.

REFERENCE

[1] C. W. Asbery and Y. Liao, "Electric Transmission System Fault Identification using Modular Artificial Neural Networks for Single Transmission Lines", IEEE 2020.

[2] Avagaddi Prasad, J. Belwin Edward, C. Shashank Roy, G. Divyansh and Abhay Kumar, "Classification of Faults in Power Transmission Lines using Fuzzy-Logic Technique", Indian Journal of Science and Technology, Vol 8(30), 2015.

[3] Ferrero A, Sangiovanni S, Zappitelli E. A fuzzy-set approach to fault-type identification in digital relaying. IEEE Transactions, Power Delivery. 1995 Jan; 10(1):169–75.

[4] Wang H, Keerthipala WWL. Fuzzy-neuro approach to fault classification for transmission

line protection. IEEE Transactions, Power Delivery. 1998 Oct; 13(4):1093–104.

[5] Kumar P, Jarni M, Thomas MS, Moinuddin. Fuzzy approach to fault classification for transmission line protection. Proceedings of the IEEE region 10 Conference. Cheju, Island: IEEE Conference Publications. 1999 Sep 15-17. p. 1046–50.

[6] Youssef OAS. Combined fuzzy-logic wavelet-based fault classification technique for power system relaying. IEEE Transactions, Power Delivery. 2004 Apr; 19(2):582–9.

[7] Sedaghati R, Rouhani A, Habibi A, Rajabi AR. A novel fuzzy-based power system stabilizer for damping power system enhancement. Indian Journal of Science and Technology. 2014 Nov; 7(11):1729–37.

[8] Dash PK, Pradhan AK, Panda G. A novel fuzzy neural network based distance relaying scheme. IEEE Transactions, Power Delivery. 2000 Jul; 15(3):902–7.

[9] Razi K, Hagh TM, Ahrabian GH. High accurate fault classification of power transmission lines using fuzzy logic. Singapore: Power Engineering Conference International, IEEE Conference Publications. 2007 Dec 3-6. p. 42–6.

[10] Costa FB, Silva KM, Souza BA, Dantas KMC, Brito NSD. A method for fault classification in transmission lines based on ANN and wavelet coefficients energy. Vancouver, BC: International Joint Conference on Neural Networks. IEEE Conference Publications. 2006 Jul. p. 3700–5.

[11] k. chen, c. huang, j. he, "fault detection classification and location for transmission lines and distribution systems: a review on the methods", high voltage iet, vol. 1, no. 1, pp. 25-33, april 2016.

[12] a. prasad, j. b. edward, k. ravi, "a review on fault classification methodologies in power transmission systems: part-i", journal of electrical systems and information technology, 2017.

[13] a. prasad, j. b. edward, k. ravi, "a review on fault classification methodologies in power transmission systems: part-ii", journal of electrical systems and information technology, 2016.

[14] k. zimmerman, d. costello, "impedance-based fault location experience", proc. 58th annu. conf. protect. relay eng., pp. 211-226, april 2005.

[15] g. song, j. suonan, y. ge, "an accurate fault location algorithm for parallel transmission lines using one-terminal data", elect. power energy syst., vol. 31, no. 23, pp. 124-129, feb./mar. 2009.

[16] e. e. ngu, k. ramar, "combined impedance and traveling wave based fault location method for multi-terminal transmission lines", elect.power syst. res., vol. 33, no. 10, pp. 1767-1775, dec. 2011.