Active Learning for Incipient Fault Detection

by

Kinjal Ghosh

A Thesis Presented in Partial Fulfillment
of the Requirement for the Degree
Master of Science

Approved April 2023 by the
Graduate Supervisory Committee:

Yang Weng, Chair
Anamitra Pal
Mojdeh Hedman

ARIZONA STATE UNIVERSITY

May 2023

ABSTRACT

Fault detection is an integral part for power systems as without its proper study, analysis and mitigation, people will not be able to power the various appliances and equipment required in all aspects of life. One such type of fault which is very critical in an electrical cable but very difficult to spot is incipient fault. These momentary faults are observed for very short periods however, if it persists, this would lead to consequences like insulation wear-off and even, power outages. With the advent of machine learning in the power systems fraternity, this paper also uses a new and updated Active Learning algorithm to detect incipient fault data from a simulated test case. The objective of the paper is to detect the fault data accurately using this new and precise method. For purposes of data collection and training of the model, MATLAB Simulink and Python programming has been used respectively.

ACKNOWLEDGMENTS

First, I would like to express my deepest gratitude to my advisor, Dr. Yang Weng, for offering great support on this thesis research. I sincerely thank him for his invaluable guidance, patience and encouragement throughout my master studies. He sets a good example of a researcher to our group by his sharp engineering insight and great enthusiasm in research work.

I would also like to thank my committee members, Dr. Anamitra Pal and Dr. Mojdeh Hedman for taking the time to give their invaluable feedback. Finally, I would like to thank my parents and my friends for their continuous support throughout my time in graduate school.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Figure Page

Figure Page

Chapter 1

INTRODUCTION

The recent advancements in technology has engulfed all aspects of human life. Power systems and its various applications has also progressed massively. Some common examples that one can observe in their daily life which they might not have observed 20 years back are renewable usage for electricity generation, electric vehicles and smart meters. There have been advancements in every field possible but the one thing common to every equipment is electricity usage. Consumption of electricity is mostly required for any device to operate. Hence, the power industry has one of the most important role of supplying electricity to people continuously. Although, there are various factors which could harm the power engineers to not meet their expectations and responsibilities.

One of the important thing that power engineers around the world always put effort into is the study of electric faults and its effect on the grid. There are various types of faults that affect the system like short circuit and open circuit faults. Mitigation of all types of faults are very critical to the power network. However, incipient fault is one such example which is really difficult to detect and hence, protection against it too becomes cumbersome. Incipient faults are momentary faults in the electric cables and initially, these look harmless for the grid. However, if such fault persists in the network, it will have detrimental effects to the grid. It could lead to insulation damage and the worst case being massive outages. Currently, there are some numerical approaches for incipient fault detection like Canonical Variate Dissimilarity Analysis (CVDA) method described in Pilario *et al.* (2019) and studying of fault trip events as in Kasztenny *et al.* (2008). Escobet *et al.* (2014) also proposes incipient fault detection methodology like residual generation methods and Xu *et al.* (2018) describes the impact of incipient faults on various power equipment. Kulkarni *et al.* (2014) proposes an algorithm for incipient fault detection in underground cables and research has been carried out in medium voltage level in Zhang *et al.* (2016a).

These methods are effective in their own way but this is not the ideal solution for power engineers. It is not robust and formulated mainly for conventional systems. This is where Machine Learning comes into action. A type of a semi-supervised learning called Active Learning has been one of the recent algorithms that has started gaining popularity in the field of Machine Learning and Artificial Intelligence. The main applications for which Active Learning was deemed to be useful when the database is small, only considering specific instances to be more useful than others and calculating the importance of every data point and then, labeling accordingly. This ability to reduce human effort and error has been mostly tested in the computer science related databases like in Wang *et al.* (2015). However, some recent research has been performed using Active Learning in power systems as in Zhang *et al.* (2022).

As is the case with every popular and trending methods, researchers try to find some error cases in it. Some examples where active learning is not much useful is pointed in Kottke *et al.* (2019). Following up on such drawbacks, this paper used techniques like modular active learning (Danka and Horvath (????)) and active learning using decision trees. These methods overcome the major drawbacks as well as the main formula could be optimized for a power system test case. When compared to the older methods (Bachman *et al.* (2017)), the methods used in this thesis provides better accuracy and faster computation time which means that the power system protection is ensured at the minimum time possible. Due to the tweaking in classification techniques and ensuring that various active learning methodologies work for the fault detection case, the paper presents a creative way to detect incipient fault in the grid. As it is known that incipient fault data is very small in a power systems data, by applying such an updated algorithm, most challenges are overcome when the hyper parameters are tuned well or the learning formula is adjusted according to the particular test system.

Some changes had to be carried out to ensure that the simulated power data with test fault cases could be executed efficiently using the active learning algorithm. Papers like Peng *et al.* (2022) and Jian *et al.* (2021) has worked on usage of active learning for general fault detection. However, this project focuses on incipient fault detection. The first important change was to modify the feedback loop such that the important data points be identified correctly. Then, designing the function which would label these important points. Hence, a repeated loop is run until a satisfied accuracy level of the model is reached. Therefore, the active learning query strategy is important and modified likewise. Apart from solving a real time problem with precision and accuracy, this method also has some added benefits.

One future work in this project which would further elevate the usefulness of this algorithm would be making this work on a real-time level. There has been advancements on the real-time fault detection in power networks using generic machine learning methods as described in Leite *et al.* (2022) and Malkoff (1987). However, incipient fault detection being a difficult and tedious task, using real-time active learning would be very efficient. The current project with minute tweaks for the input of real-time data could easily be deployed. Another benefit to this model is by improving the formula further or by fine-tuning the hyper-parameters, the accuracy from modular active learning could be increased more. Hence, it is evident that the objective of trying to use active learning for incipient fault detection is indeed justified.

As it has been already mentioned, the usefulness of any algorithm is usually judged on the accuracy of whatever task it is required to perform. In this case, it is checked how accurately the algorithm is able to detect incipient fault data amongst a large power database. The data is collected from different simulation cases as described in later sections. This data is then split into train and test data systems. The train data is used to build the initial model and based on the model's accuracy, further data is labelled. After the model is trained completely, the test data is executed and compared with the already labeled data to get the accuracy of the model. By testing the various pools of data previously simulated, the novelty and accuracy of the model could be verified.

The rest of the report is organized as follows: Chapter 2 describes the steps involved in Active Learning algorithm and gives a detailed explanation on how Active Learning is deployed for incipient fault detection. Chapter 3 studies the way to model an incipient fault for simulation purposes on MATLAB Simulink. It also includes the simulation procedure for data collection. Chapter 4 reports all the numerical calculations involved in the modeling along with captured instances of the simulated fault data. Additionally, it contains the algorithm results along with proof of superiority of the method. Chapter 5 concludes the report and then References are noted.

Chapter 2

PROPOSED METHOD

2.1 Active Learning Background

The technique of prioritising the data that has to be labeled in order to have the most influence on training a supervised model is known as active learning. When there is too much data to label and smart labelling needs to be prioritised because there is too much data to label, active learning can be employed.

To apply active learning to an unlabeled data collection, the following steps are followed:

- The very first thing that must be done is the manual labeling of a very tiny subset of the input data.

- The model needs to be trained using a limited amount of labelled data. The model will not be perfect but will provide some insight into which regions of the parameter space should be tagged first to make it better.

- The model is used to forecast the class of each subsequent unlabeled data item after it had been trained.

- Each unlabeled data point is assigned a score depending on the model's prediction.

- This process can be repeated after the optimal method for prioritising the labelling has been selected: a new model can be trained on a new labelled data set that has been tagged using the priority score. The unlabeled data points can be run through the model to update the prioritisation scores and continue labelling after the new model has been trained on the subset of data. In this approach, as the models improve, the labeling strategy may be continually improved.

2.2 Active Learning for Incipient Faults

Using machine learning algorithms to find flaws or anomalies in a system is known as fault detection using active learning algorithm. By choosing the most insightful data points for labeling, active learning aims to reduce the amount of labeled data needed to train the machine learning model. The measures to take when employing the active learning algorithm for defect detection are as follows:

1. Gather and prepare data: Gathering and preparing data is the initial step in active learning fault detection. In order to do this, a test model had been simulated for data collection, cleaned and normalized, and relevant features must be chosen in order to train the machine learning model. This is described in detail in Chapter 3.2.

2. Train the initial model: After the data has been cleaned up, the next step is to use a tiny labeled data set to train an initial machine learning model. A representative sample of the fault and non-fault data points should be contained in this labeled data set.

3. Choose useful data points: The active learning algorithm chooses useful data points for labeling. These should be the most ambiguous or challenging data pieces for the model to classify.

4. Label the chosen data points: A subject-matter expert or domain specialist labels the chosen data points.

5. Retrain the model: With the extended data set, the model is retrained using the newly labeled data points that were added to the training set.

6. Repeat steps 3-5: The model is repeated in stages 3-5 until it reaches an acceptable degree of accuracy.

7. Test the model: After it has been trained, the model can be used to identify faults in the system. It's crucial to test the model on a held-out data set to make sure it applies well to fresh data.

The above points when described numerically, the active learning model is built with a set, $S \equiv \{(x, y)\}$ where x number of data points can request labels y. A similarly defined evaluation set $E \equiv \{(\hat{x}, \hat{y})\}$. Also $S_t^u \equiv \{(x, \cdot)\}$ denotes data set which are still unlabeled after t label queries. Its complementary set, $S_t^k \equiv \{(x, y)\}$, denotes the data points whose labels are determined. Also, S_t denotes the joint set of labeled and unlabeled data after t label queries. A real valued vector h_t is the control state of the model and $R(E, S_t, h_t)$ defines the accuracy of the model while predicting labels after t queries. Considering the model parameters, θ, the prediction reward formula is as follows:

$$R\left(E, S_t, h_t\right) \equiv \sum_{(\hat{x}, \hat{y}) \in E} \log p\left(\hat{y} \mid \hat{x}, h_t, S_t\right) \tag{2.1}$$

This gives the log-likelihood of the predictions $\log p\left(\hat{y} \mid \hat{x}, h_t, S_t\right)$ on the sample set. At each step t of active learning, the model requests label for a data point x from the set S_{t-1}^u. Meanwhile, the control state is updated from h_{t-1} to h_t. Consequently, h_t and S_t helps the model to determine which data point's label is to be requested next. The objective function for training is defined as follows:

$$\underset{\theta}{\text{maximize}} \underset{(S,E)\sim\mathcal{D}}{\mathbb{E}} \left[\underset{\pi(S,T)}{\mathbb{E}} \left[\sum_{t=1}^{T} R\left(E, S_t, h_t\right) \right] \right] \tag{2.2}$$

where T is the maximum number of label queries to take place, (S, E) indicates one query sample from the distribution, $\mathcal{D}$ and $\pi(S, T)$ indicates model's active learning method π for T steps on set S. By using the above steps, the objective function is optimized and ensures maximum accuracy of predictions.

Active learning techniques like query by committee, uncertainty sampling, and density-based sampling can all be used to detect faults in power systems. Each approach has advantages and disadvantages, and the best algorithm will be chosen depending on the application and data set.

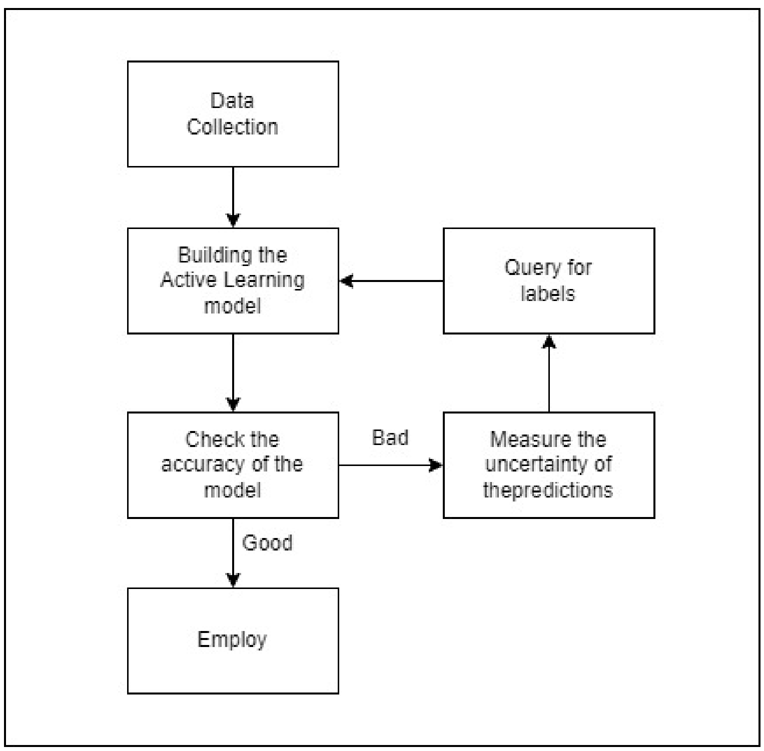

Figure 2.1: Active Learning Algorithm

As observed from 2.1 and described previously, the first step is to collect data. In this case, a test IEEE 13-node system has been simulated with an introduction of fault. Various data sets has been collected with varying phase-to-ground faults. The next step in the procedure is to build the initial active learning model with only limited labels. Once the model is trained, the code checks the accuracy of the model. If the accuracy is not satisfactory, the code tries to determine the uncertainty of the prediction in the testing data set. This is done in order to label some more data. The process then repeats, that is, builds the model and checks the accuracy. Once a satisfactory accuracy level is reached, the model is deployed to carry out the objective. The entire algorithm can be described using a pseudo code.

$f \leftarrow$ initialize classifier

$al \leftarrow$ initialize AL strategy

for all $n \in \{1, \ldots\}$ do

  $(t_n^x, x_n, t_n^y) \leftarrow$ retrieve from data stream S

  $\mathrm{L}_n \leftarrow$ training data set at time t_n for acquisitions A

  $\mathrm{u}_n \leftarrow (\ \mathrm{al},\ (t_n^x, x_n, t_n^y)\ , f, L_n)$

  $\mathrm{a}_n \leftarrow$ Active Learning QUERY (u_n)

  $if\mathrm{a}_n = 1$ then

  $askforlabelof\mathrm{x}_n$ (label y_n will be provided at t_n^y)

end if

end for

Chapter 3

INCIPIENT FAULT SIMULATION

3.1 Establishment of Incipient Fault Model

In power systems, an arc is typically evident when the fault occurs. The fault arc poses a threat to both human life and electrical equipment. Therefore, arc fault current calculation is crucial for minimizing loss.

The arc fault current must be calculated using an arc model. There are three categories into which arc models can be divided: physical models, black box models, and models based on drawings and diagrams. Black box models simply explain how input and output signals are related. Black box models describe how the arc and the electrical circuit interact when there is a fault. One or more differential equations linking the arc conductance, which represents the energy balance of the arc column, are used in black box models to describe arcs Weng *et al.* (2022).

In 1939, Cassie presented the Cassie Arc Model. According to Cassie, the arc's temperature is fixed and is lowered by forced convection. This suggests that the voltage over the arc is constant and that the cross-sectional area of the arc is proportionate to the current. For arcs with strong currents, the Cassie arc model is appropriate. The Cassie Arc Model can be represented by the following differential equation:

$$\frac{1}{g_c}\frac{dg_c}{dt} = \frac{1}{\tau_c}\left(\frac{u_{arc}^2}{u_c^2} - 1\right) \tag{3.1}$$

where,

g_c is the arc conductance,

r_o is the arc time constant and,

u_{ac} is the arc voltage across the breaker.

In 1943, the Mayr Arc Model was introduced. Mayr made the assumption that thermal conduction is what causes power losses and that temperature affects arc conductance. The arc's cross-sectional area is taken to be constant. Currents close to zero are fit by the Mayr arc model. In 1992, a modified Mayr arc model was introduced. Current influences how effective the cooling is in the model. The differential equation for the Mayr model is as follows:

$$\frac{1}{g_m}\frac{dg_m}{dt} = \frac{1}{\tau_m}\left(\frac{u_{arc}i_{arc}}{P_o} - 1\right) \tag{3.2}$$

where,

g_m is the arc conductance,

r_m is the arc time constant,

P_o is the cooling power constant,

u_{aac} is the arc voltage across the breaker and,

i_{ar} is the arc current.

Li and Li (2018) mentions that incipient faults and their intrinsic characteristics can be modeled using Cassie and Mayr models. These models can be treated as black box model which internally consist of that particular differential equation but the application remains the same. In regards to the paper's design, the fault arc is being simulated using the above mentioned models. The results of the simulations are discussed in the subsequent sections but it can be noted that the correctness of the output plots can be verified from the references. Xiong *et al.* (2020) is treated as the base paper for the understanding and working of incipient faults and arc models being exposed to different testing systems. The testing system is similar to this paper, the difference being the priority of this thesis is data collection from the arc fault model. Hence, the fault was introduced only at one node. Wang *et al.* (2008) explicitly uses Cassie and Mayr models for simulating the incipient faults and so, the relevance of use of these models are proved. There are more evidence in support of usage of Cassie and Mayr Arc Models like Samet *et al.* (2021b) mentions that incipient faults are dangerous to the power systems and should be taken care of before it turns into a permanent fault. To have a visual understanding of such faults, these can be modeled using Cassie, Mayr and other universal models. Samet *et al.* (2021a) and Bretas *et al.* (2017) also mentions that Cassie and Mayr models are indeed the correct way to model incipient fault arcs. The papers Kizilcay and La Seta (2005) and Kizilcay and Pniok (1991) also dwells on the fact that fault simulation can be done using the arc models in discussion. However, the application areas of the references are different than what this thesis provides insight on. Yuan *et al.* (2013) acts as an important validation of our obtained output plots. Hence, it can be observed that the simulation methods adopted in this paper is indeed the correct procedure (Li and Li (2018)).

3.2 Construction of the Fault Database

As discussed in the previous section, the test system was the IEEE 13-node system. This is modeled in MATLAB Simulink as the diagrams attached below. The main objective here is to model and predict the fault that can be present in a network. Hence, the fault is introduced at the 633-node of the official IEEE 13-node system. This fault is modeled by two methods: Cassie and Mayr Arc Models.

In Simulink, this arc model is created using the Masked Block model. The main block inside this masked model is the Differential Equation Editor (DEE). This consists of the basic differential equations of individual arc models. Hence, this way, the simulation makes it easier to replicate any arc model according to any requirement. In this case, Cassie and Mayr differential equations as described in equations (3.1) and (3.2) are given as arguments.

This process was repeated for every phase. this means a fault was first introduced at phase A and data collection was done. Then, a fault was introduced only at phase B and the voltage and current data for the entire simulation time period was noted. Finally, a fault was solely introduced at phase C and similar data was stored. This provides a larger data set for further processing and learning. Also, to get a more practical and realistic data, noise was added to these data. A white Gaussian noise was used in this case. This is a basic noise model which mimics the randomness observed in practical scenarios. This was done using MATLAB code. Hence, for both current and voltage waveform, this white, Gaussian noise of Signal-to-Noise Ratio of 20dB, 33dB and 44dB was added. This was done for the above described three cases of testing too, that is, fault being individually introduced at each phase. Hence, for every faulty phase, we get three sets of voltage and current, noise added data. All these are stored in MS Excel again for ease of use.

Another type of black box arc model that could have been utilized is the Kizilcay Arc Fault model like in Zhang *et al.* (2016b). recently, various advanced methods has also been tried to model the incipient fault arcs in power grids like performed in Li *et al.* (2021) and Izadi and Mohsenian-Rad (2021).

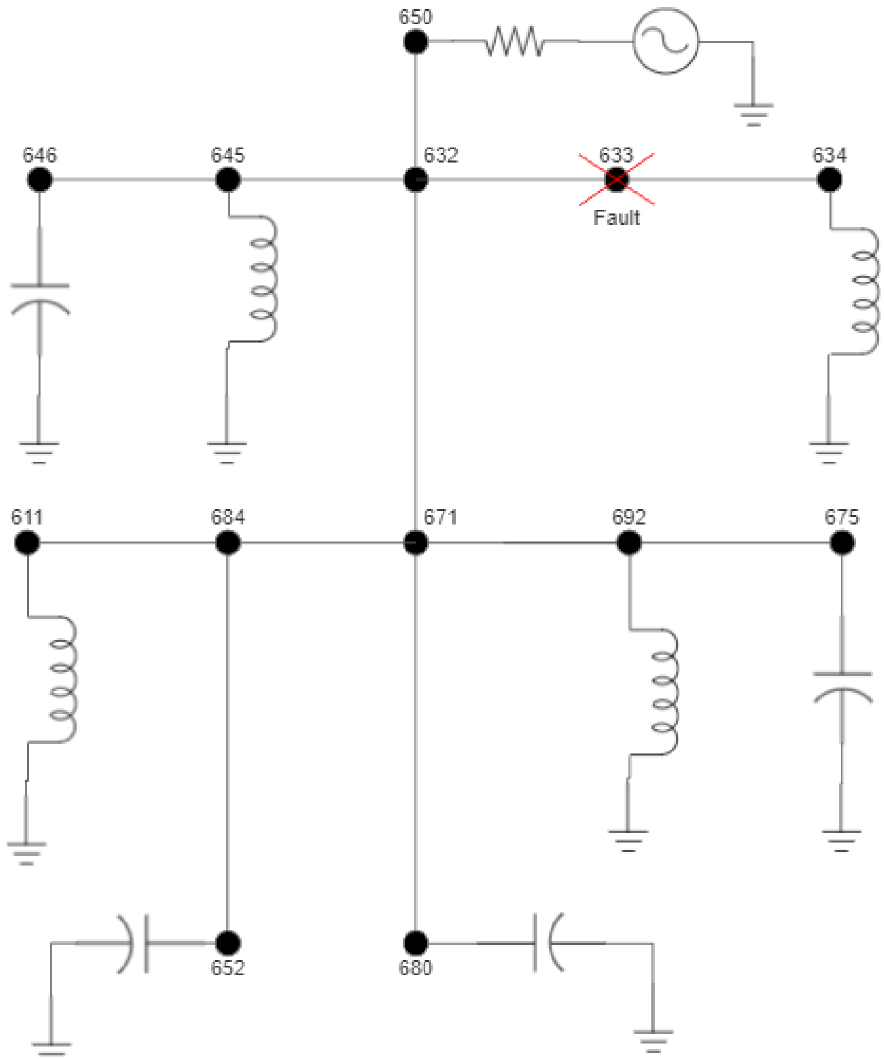

Figure 3.1: Modified IEEE-13 Node System for testing purposes

Chapter 4

NUMERICAL RESULTS

4.1 Parameters of Incipient Fault Arc Models

There are some calculations also involved to get to know the values of passive components. The typical values of inductive and capacitive reactive values were extensively checked and then the particular passive components were computed for our design parameters.

For a Voltage range of 460 V $-$ 33kV, the capacitive reactive power (X_C) is $300-3000$kVAR So, 25kV gives a reactive power of 2272.73kVAR

$$\frac{I^2}{2\pi fC} = 2272.73$$
$$C = \frac{2.6^2}{2\pi \times 60 \times 2272.73 \times 10^3}$$
$$C = 7.938nF$$

Similarly, for a voltage range of 25kV, the inductive reactive power (X_L) is $100-500$kVAR Hence, 25kV gives a reactive power of 500kVAR

$$2\pi fL \times I^2 = 500$$
$$L = \frac{500}{2\pi \times 60 \times 2.6^2}$$
$$L = 0.196H$$

Hence, these are the values that have been used in the simulation. For the single phase model, it can be noticed that a resistive component has been connected directly to the voltage source. This is a very small resistor connected to the voltage source to ensure that there is no direct connection between inductor and the source. The single phase models were run for a simulation period of 0.0004s and the fault was introduced at 0.0002s. The three phase Cassie model was run for 0.06s and the fault was introduced at 0.02s. These time periods were chosen to get the optimal output plots and best data set. This way the output plots are obtained from the Scope block. After the model is run, the data points at the fault location are extracted and collected in MS Excel. Similarly, the three phase Mayr model was run for 0.1s and the fault was triggered at 0.05s.

All the output plots were collected in this section. As discussed in the previous section, the Cassie and Mayr fault models were triggered at the 0.0002s mark. Hence, that particular time range of the entire output plot is displayed. With regards to the three phase models, we can observe that the phase consisting of the fault has a crooked behaviour when compared to non-fault phases which have an expected sine-wave structure.

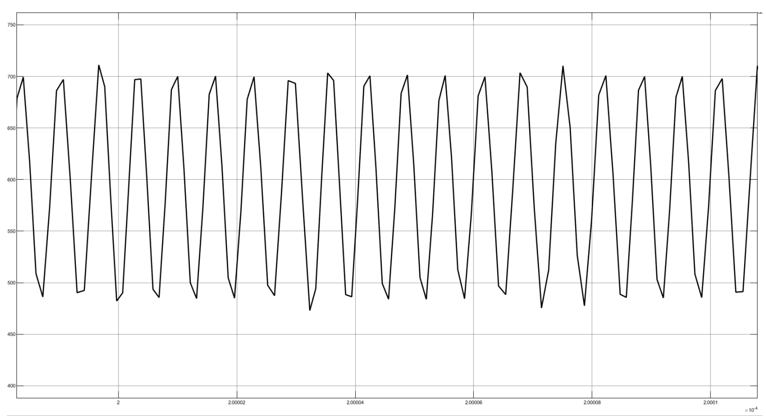

Figure 4.1: Voltage Waveform obtained from Cassie Arc Model in the IEEE-13 Node System

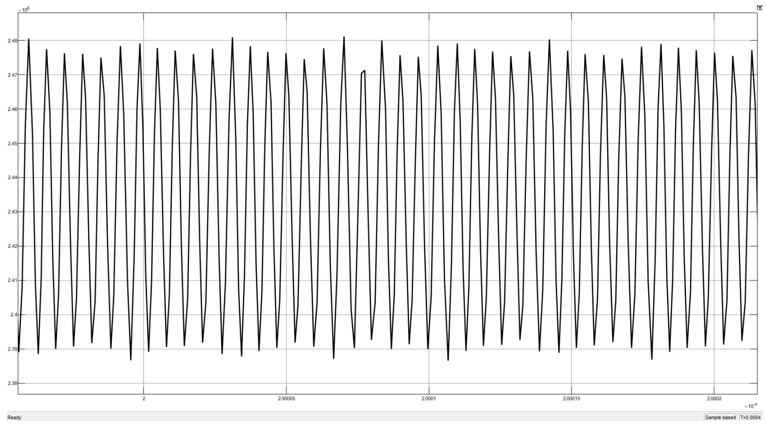

Figure 4.2: Current Waveform obtained from Cassie Arc Model in the IEEE-13 Node System

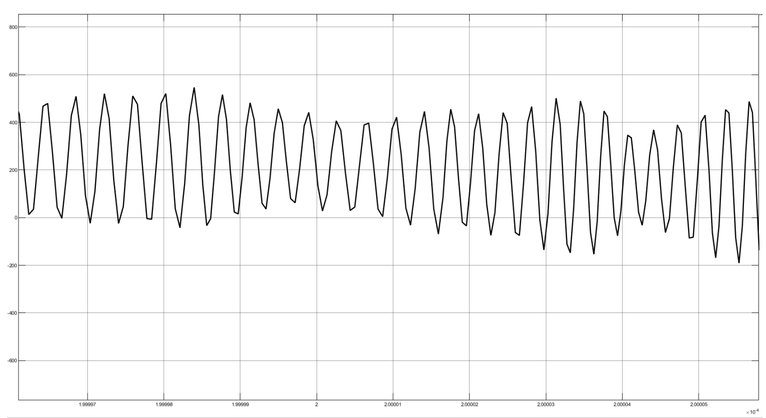

Figure 4.3: Voltage Waveform obtained from Mayr Arc Model in the IEEE-13 Node System

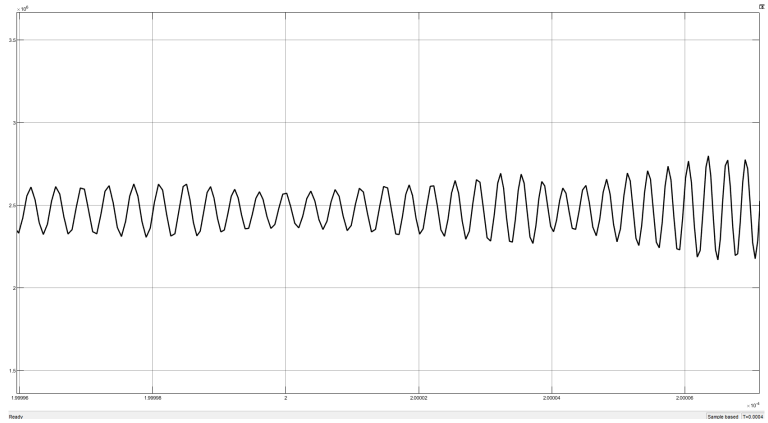

Figure 4.4: Current Waveform obtained from Mayr Arc Model in the IEEE-13 Node System

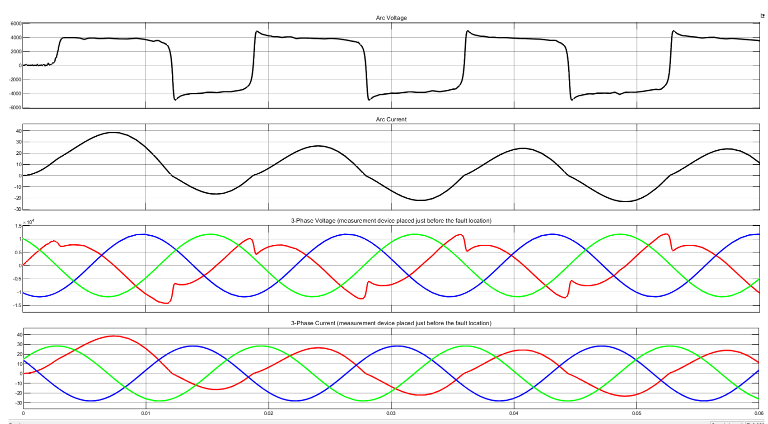

Figure 4.5: Voltage Waveform obtained from Cassie Arc Model when fault applied at phase-A in the 3-Phase IEEE-13 Node System, Current Waveform obtained from Cassie Arc Model when fault applied at phase-A in the 3-Phase IEEE-13 Node System, Voltage Waveform (measurement device placed right before the fault location) of the System, Current Waveform (measurement device placed right before the fault location) of the System,

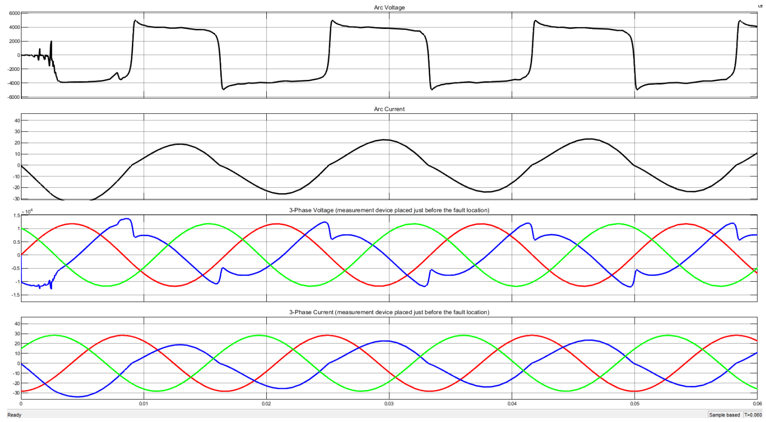

Figure 4.6: Voltage Waveform obtained from Cassie Arc Model when fault applied at phase-B in the 3-Phase IEEE-13 Node System, Current Waveform obtained from Cassie Arc Model when fault applied at phase-B in the 3-Phase IEEE-13 Node System, Voltage Waveform (measurement device placed right before the fault location) of the System, Current Waveform (measurement device placed right before the fault location) of the System,

4.2 Method Superiority Proof

The primary reasons to use Active Learning algorithm in our incipient fault detection case are as follows:

- Data labeling takes less time and cost- As mentioned earlier, the incipient faults are momentary faults and their existence in a data set is very small. Hence, if engineers try to label all data manually, it is a serious waste of time. On the other hand, if it is tried to learn the model without labeling them, the model will learn it wrong. Hence, by using Active Learning algorithm, humans are able to know which are the important data points and label them only. Therefore, this algorithm saves both time and cost.

- Quick determination of model accuracy and feedback- Usually, data is labeled

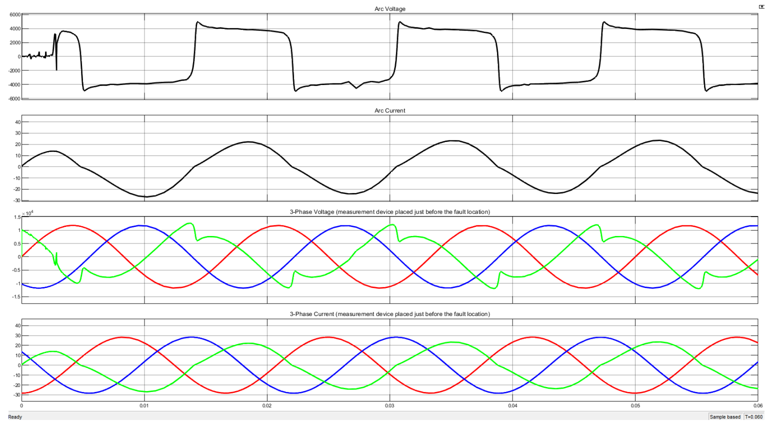

Figure 4.7: Voltage Waveform obtained from Cassie Arc Model when fault applied at phase-C in the 3-Phase IEEE-13 Node System, Current Waveform obtained from Cassie Arc Model when fault applied at phase-C in the 3-Phase IEEE-13 Node System, Voltage Waveform (measurement device placed right before the fault location) of the System, Current Waveform (measurement device placed right before the fault location) of the System,

prior to model training or receiving feedback. Frequently, it takes days or weeks of re-labeling and iterating on annotation criteria before it is understood that the model's performance is grossly deficient or that fresh labels for the data are required. Active Learning makes it possible to train models often, allowing for feedback and error correction that would otherwise need to happen much later.

- Active learning models can train more quickly with less input of data and converge to superior final models. With the widespread belief that more data is better, it is easy to overlook the fact that data quality matters just as much as quantity. If the data set includes examples that are challenging to accurately categorize, the performance of the final model may actually deteriorate. As observed in the testing cases too, this Active Learning algorithm produces better accuracy when compared to general Machine Learning algorithms. Addi-

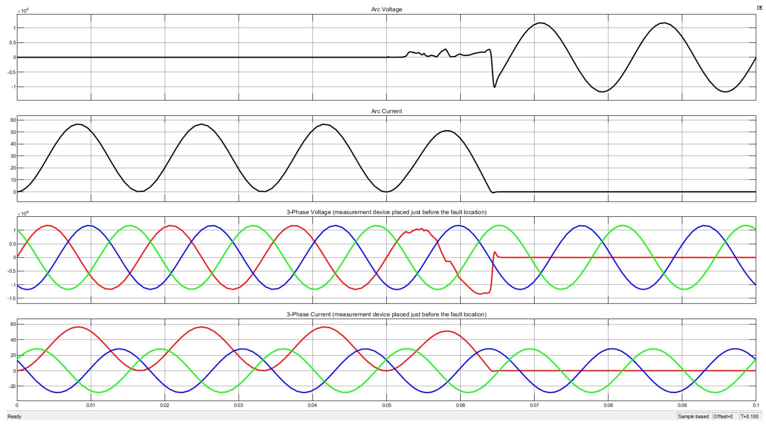

Figure 4.8: Voltage Waveform obtained from Mayr Arc Model when fault applied at phase-A in the 3-Phase IEEE-13 Node System, Current Waveform obtained from Mayr Arc Model when fault applied at phase-A in the 3-Phase IEEE-13 Node System, Voltage Waveform (measurement device placed right before the fault location) of the System, Current Waveform (measurement device placed right before the fault location) of the System,

tionally, for an application like fault detection, we can never underestimate the value of accuracy as any wrong diagnosis of the system would lead to massive outage or blackouts even.

Hence, it is evident that using Active Learning is ideal for an application like incipient fault detection. However, then, the question arises that why this method was not widely used previously. The primary reasons for this are as follows:

- Active Learning or a type of semi-supervised learning was not considered to be a part of generic machine learning and its related algorithms. Hence, not much research was carried out in this direction. Additionally, not knowing the benefits of Active Learning for the most time, power engineers also did not try to use this algorithm in their applications.

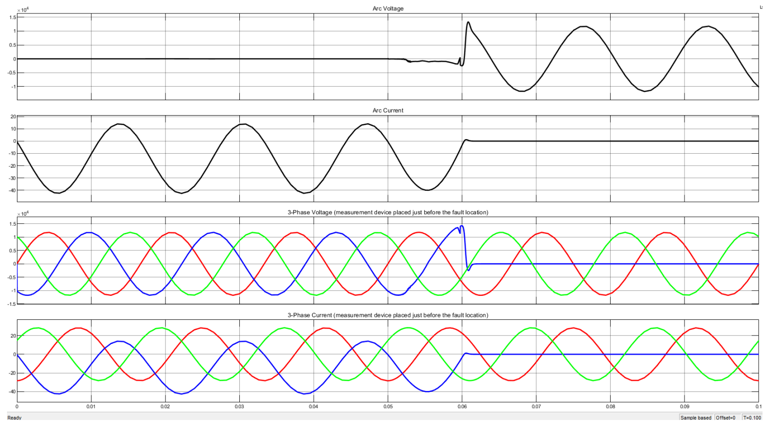

Figure 4.9: Voltage Waveform obtained from Mayr Arc Model when fault applied at phase-B in the 3-Phase IEEE-13 Node System, Current Waveform obtained from Mayr Arc Model when fault applied at phase-B in the 3-Phase IEEE-13 Node System, Voltage Waveform (measurement device placed right before the fault location) of the System, Current Waveform (measurement device placed right before the fault location) of the System,

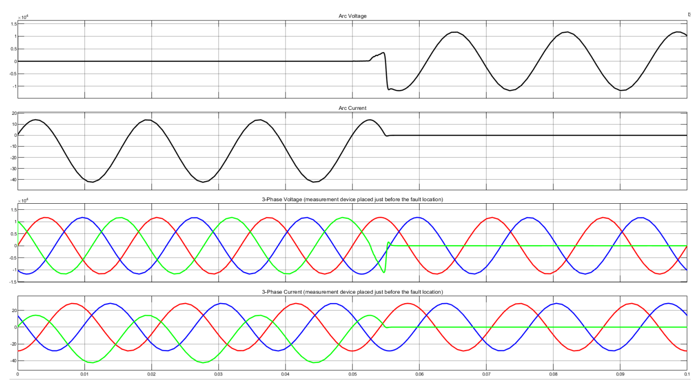

Figure 4.10: Voltage Waveform obtained from Mayr Arc Model when fault applied at phase-C in the 3-Phase IEEE-13 Node System, Current Waveform obtained from Mayr Arc Model when fault applied at phase-C in the 3-Phase IEEE-13 Node System, Voltage Waveform (measurement device placed right before the fault location) of the System, Current Waveform (measurement device placed right before the fault location) of the System,

- Active learning is frequently said to have only one method. However, this is not true. The dynamic sampling of a training data set is the core of active learning, where it is required to "identify" the data set to gradually select the most promising data points. Hence, Active Learning is not just a 'plug data and get result' type algorithm but requires fine tuning for different applications.

- The idea that substantial uncertainty implies importance on the training process is one of the fundamental presumptions of the majority of popular querying algorithms. Adding the record to the training set must help if the model generates a forecast with a high level of uncertainty. Power data generally consists of redundant data and incipient fault data is very hard to find in this large data pool. Earlier, researchers were unable to find a way to identify the high uncertainty cases only.

- Academics frequently forget that at the core of the active learning process lies a fundamental trade-off between the number of labels and the amount of computation. Researchers typically measure their success with active learning by looking at the reduction in the amount of data that required labeling. This is due to the fact that active learning necessitates the model being retrained repeatedly with progressively larger data sets, which frequently results in a quadratic relationship between computation and the number of loops used. With advanced tools and computation power available at hand, active learning is an affordable method for power engineers.

- The concept of labeling the required data points was vague and considered, inefficient earlier. However, understanding the benefits with time has influenced the promotion of Active Learning algorithm.

- There was a fear of building an inaccurate model by identify the wrong data points by this algorithm. However, with more successful test cases been implemented, the accuracy of Active Learning has been identified.

The above points give an overview on why Active Learning is getting more popular in the field of power and energy. Applications like fault detection, in particular, incipient fault where the fault data available is very small, this algorithm turned out to be very helpful and precise. The testing results of different methods with different testing data sets are noted in the following tables. A comparison bar graph is also presented for easy understanding of the obtained results. A line chart showing the times (in seconds and logarithmic scale) for different active learning methods is also visualised.

Table 4.1: Accuracy of Different Active Learning algorithms with Noiseless (Ideal) Data set

Active Learning Method	Accuracy Percentage
Modular	71.01
Decision Trees	99.13
Naive Bayes	97.34
Support Vector Machines	98.27

Modular active learning has been described in previous sections. The next method, which is, decision trees where both the classification and regression issues can be solved. The term itself implies that it displays the predictions that come from a sequence of feature-based splits using a flowchart that resembles a tree structure. The decision is made by the leaves at the end, which follows the root node. Decision

Table 4.2: Accuracy of Different Active Learning algorithms with 20dB Data set

Active Learning Method	Accuracy Percentage
Modular	68.31
Decision Trees	98.39
Naive Bayes	97.02
Support Vector Machines	97.82

Table 4.3: Accuracy of Different Active Learning algorithms with 33dB Data set

Active Learning Method	Accuracy Percentage
Modular	65.37
Decision Trees	98.4
Naive Bayes	96.42
Support Vector Machines	96.92

Table 4.4: Accuracy of Different Active Learning algorithms with 44dB Data set

Active Learning Method	Accuracy Percentage
Modular	62.19
Decision Trees	97.54
Naive Bayes	95.77
Support Vector Machines	96.12

trees basically use a quantity called Entropy which is the measure of impurity at a data node. Its formula is as follows:

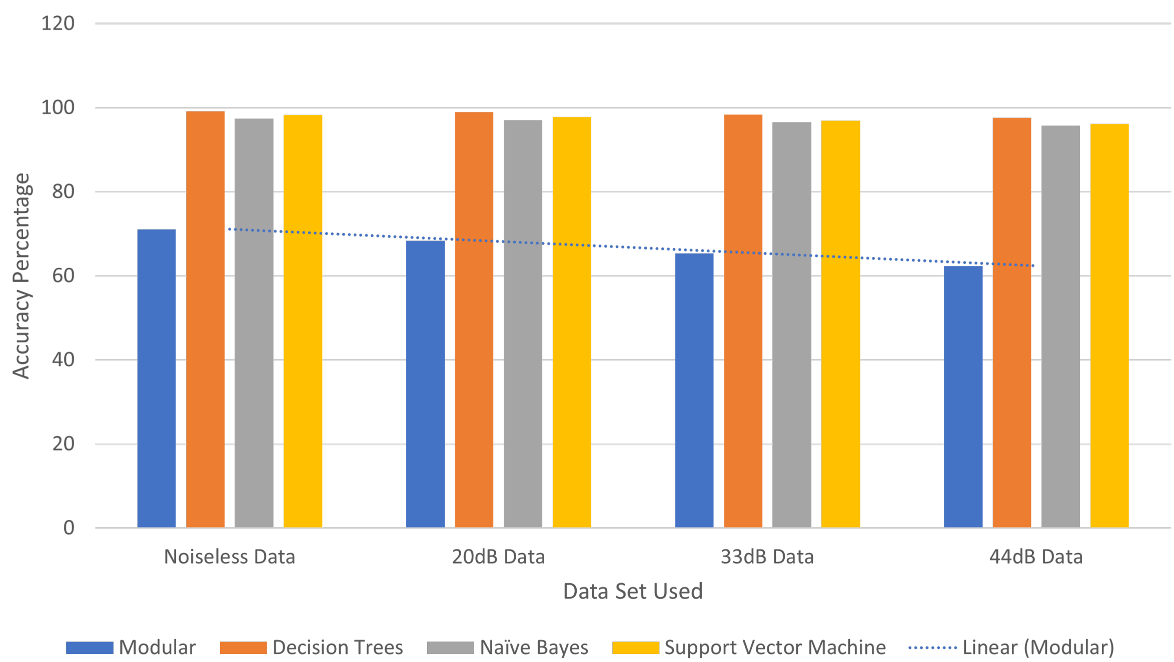

Figure 4.11: Comparison of Accuracy Levels of Different Active Learning Methods with Different Testing Data Sets

$$E(S) = -p_{(+)} \log p_{(+)} - p_{(-)} \log p_{(-)} \tag{4.1}$$

where, $p_{(+)}$ is the probability of positive class,

$p_{(-)}$ is the probability of negative class,

S is the training set.

Naive Bayes is a classification method built on the Bayes Theorem with the assumption of predictor independence. A Naive Bayes classifier, to put it simply, believes that the presence of one feature in a class has nothing to do with the presence of any other feature. The Bayes Theorem for a classification computational problem is as follows:

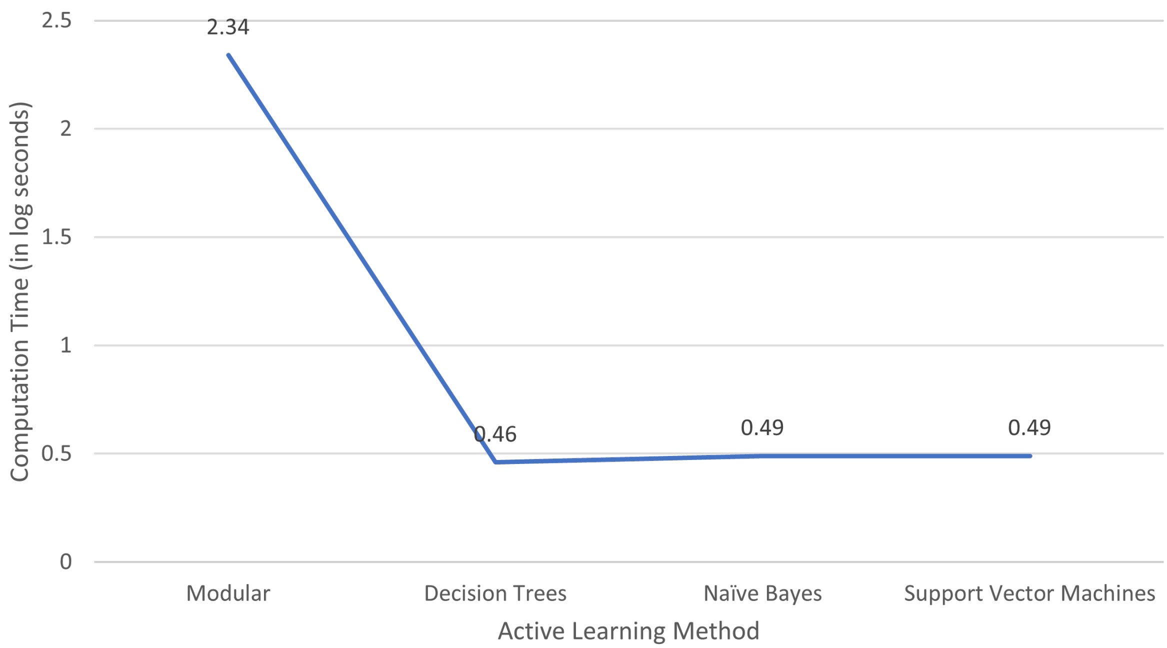

Figure 4.12: Comparison of Computation Time of Different Active Learning Methods

$$P(yi \mid x1, x2, \ldots, xn) = P(x1, x2, \ldots, xn \mid yi)^{*}P(yi)/P(x1, x2, \ldots, xn) \tag{4.2}$$

where, $P(yi)/P(x1, x2, \ldots, xn)$ is the conditional probability for a class label with a given set of input values, $(x1, x2, \ldots, xn)$ and labels y.

Support Vector Machine (SVM) can be applied to classification or regression problems. Each data point is represented as a point in n-dimensional space (where n is the number of features) and each feature's value is represented by the value of a certain coordinate in the SVM method. Then, classification is carried out by locating the ideal hyper-plane that effectively distinguishes the two classes. This hyper-plane is defined as follows:

$$\mathbf{w}^\top \mathbf{x} - b = 0 \tag{4.3}$$

where, there is a n training data set of $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)$ and x data points with y class labels,

w is the normal vector to the hyper-plane,

Parameter $\frac{b}{\|\mathbf{w}\|}$ determines the offset of hyperplane from the origin along the normal vector, w.

All these are generic machine learning algorithms and extensive research has been done individually on these topics. However, this paper use these methods to classify/identify the data points which needs further labelling. The model is actually built using Active Learning only but the classification of important data points has been carried out by the methods mentioned in Table 4.1. It should be noted that by doing further research work on the modular active learning, the accuracy of detection could further be improved.

Further, to find out the algorithm's effectiveness in different test cases. One of the validation technique was to introduce fault at all the nodes individually in the test circuit discussed earlier. It can be observed that more or less every node fault accuracy is in the same range. Hence, it can be inferred that active learning is very useful for incipient fault detection.

Another validation step that was carried out was comparison with generic machine learning algorithms. The following machine learning algorithms were executed to get a fair comparison of the effectiveness of active learning:

- Linear Regression
- Logistic Regression

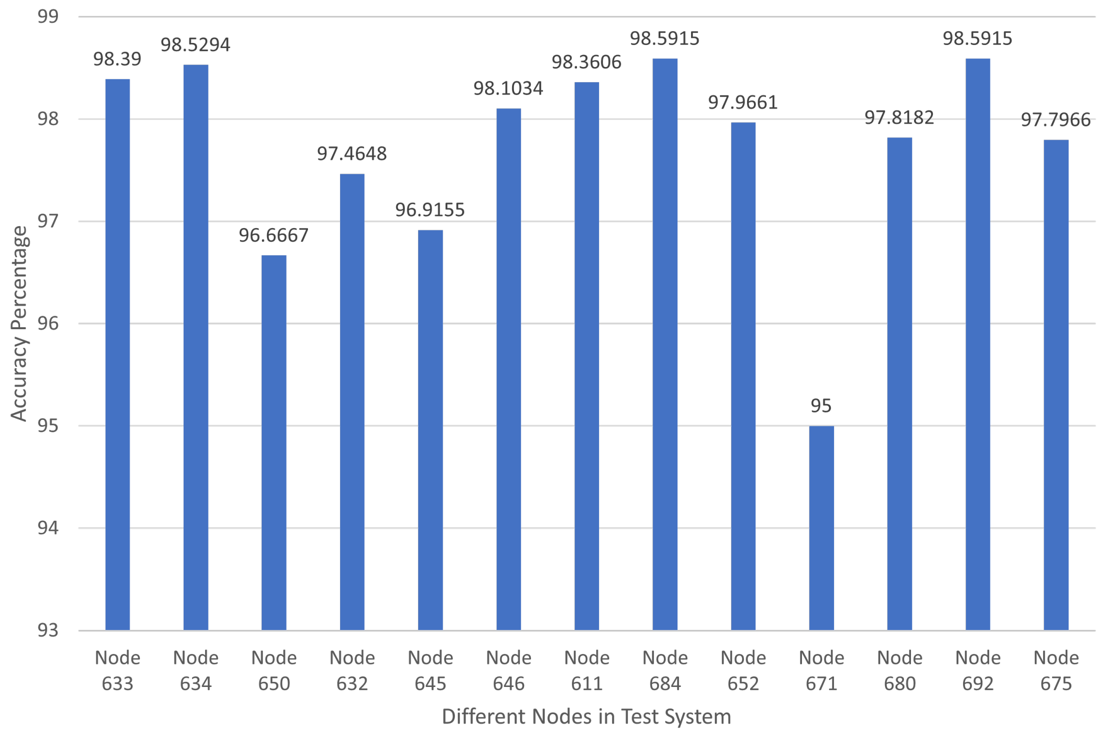

Figure 4.13: Comparison of Accuracy Percentage when Fault is Introduced at the Different Nodes of the Test System

- Decision Trees
- Random Forest
- Support Vector Machine
- Naive Bayes
- K-Nearest Neighbor
- Principal Component Analysis
- Neural Network
- Gradient Boosting

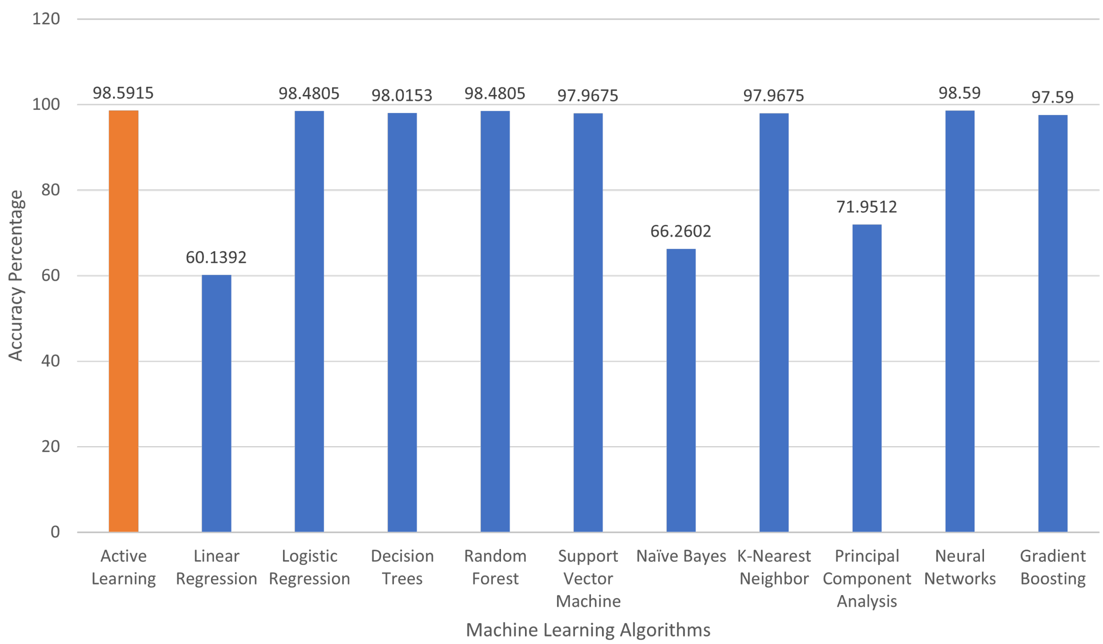

Figure 4.14: Comparison of Accuracy Percentage when Active Learning Algorithm (in orange) with respect to Generic Machine Learning Algorithms (in blue)

It can be noticed that active learning algorithm performs the best amongst all the popular methods that have been already tried out. It should be noted that for a fair comparison, all the algorithms have been executed using the same train and test data set. Similarly, sensitivity analysis was also conducted on the algorithm. The data set was varied such that different ratios of fault data points to non-fault data points in the test data set are obtained. The objective was to find out the usefulness of the algorithm based on the data being fed. It was observed that the default setup gave the best accuracy levels taking into consideration all the constraints and variables. As the number of fault data points keep decreasing, the accuracy level of the model falls. It should be noted that the 'Default Case' is the data set which was used for most testing purposes. Additionally it was observed that from the 15:85 fault data ratio levels, the model was training very poorly and often not giving the best possible output. Finally, some sensitivity analysis was conducted when operating conditions are differing. The test case conducted here was introduction of a solar PV. It was modeled in Simulink using a DC Voltage Source and an inverter to convert from DC to AC. This renewable penetration was introduced at varying nodes to get better validation results. In general, the algorithm had an accuracy level of 98.3996% which is very similar to what was obtained without any renewable penetration. Hence, active learning algorithm is a very effective method in today's date and time as we observe more-and-more renewable penetration into the electric grid.

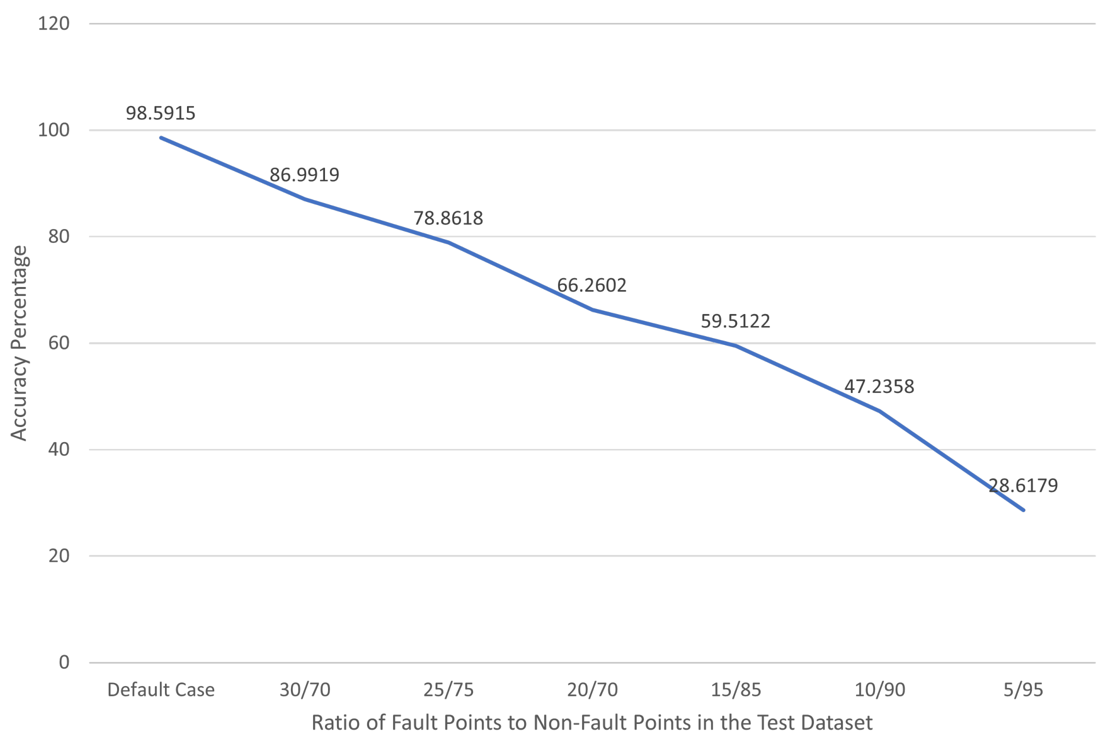

Figure 4.15: Comparison of Accuracy Percentage when the Ratio of Fault Data Points to Non-Fault Data Points is Varied

Finally, another popular method was tried for the application of incipient fault detection. Bayesian Additive Regression Trees or BART is a machine learning approach that models the correlation between a group of predictor factors and a response variable using decision trees. Decision tree modeling's Bayesian approach, or BART, is founded on the ideas of Bayesian statistics. BART operates by fitting a group of regression trees, each of which is a nonlinear function of the input features, to the data. The final prediction is then created by combining the ensemble of trees. BART is superior to other tree-based techniques because it can handle continuous and categorical data and capture intricate relationships between the input attributes. As a typical issue in many real-world data sets, missing data handling is another strength of BART. The posterior distribution of the model parameters, which is acquired by Bayesian inference, can be used to impute missing data. BART is a potent machine learning method that, in general, may be used to solve a variety of prediction issues, particularly when the data contains intricate and nonlinear correlations between the input features and the response variable.

In this paper, BART was tried to implement in Python using the bartpy module. Currently, the best accuracy level obtained for the incipient fault detection case is 61.7886%. However, upon conducting further research, this method can also be fine tuned to get better results.

Chapter 5

CONCLUSION

This paper attempts to solve the problem of incipient fault detection in power cables and the electric grid. Active Learning algorithm is being used to tackle this problem. As discussed in previous sections, incipient fault being momentary faults, the availability of such type of fault data is very small over a large pool of data. This is where Active Learning becomes advantageous as only the important data points are being labeled by the field expert. Once the final model is trained, it is observed that the accuracy of detection using this model is very high. There were some case which did not give favorable results like the Modular Active Learning. Although, several validation cases prove that usage of active learning is indeed beneficial. To be specific, it was observed that no matter the fault node or any varying operating conditions, the accuracy level of the model is very high. Similarly, after executing the many popular machine learning algorithms using the incipient fault data set, it was noticed that active learning gives the most precise model. Hence, we are able to mitigate the bad consequences of incipient faults from the power grid in any condition. A test circuit system was build in MATLAB Simulink for data collection and then Python programming (using Spyder) was used to deploy the Active Learning algorithm. Hence, it can be concluded that using Active Learning model is a very effective method for incipient fault detection.

REFERENCES

Bachman, P., A. Sordoni and A. Trischler, "Learning algorithms for active learning", in "international conference on machine learning", pp. 301–310 (PMLR, 2017).

Bretas, A. S., A. R. Herrera-Orozco, C. A. Orozco-Henao, L. U. Iurinic and J. Mora-Flórez, "Incipient fault location method for distribution networks with underground shielded cables: A system identification approach", International Transactions on Electrical Energy Systems **27**, 12, e2465 (2017).

Danka, T. and P. Horvath, "modAL: A modular active learning framework for Python", URL `https://github.com/modAL-python/modAL`, available on arXiv at `https://arxiv.org/abs/1805.00979` (????).

Escobet, T., V. Puig, J. Quevedo and D. Garcia, "A methodology for incipient fault detection", in "2014 IEEE Conference on Control Applications (CCA)", pp. 104–109 (2014).

Izadi, M. and H. Mohsenian-Rad, "Synchronous waveform measurements to locate transient events and incipient faults in power distribution networks", IEEE Transactions on Smart Grid **12**, 5, 4295–4307 (2021).

Jian, C., K. Yang and Y. Ao, "Industrial fault diagnosis based on active learning and semi-supervised learning using small training set", Engineering Applications of Artificial Intelligence **104**, 104365 (2021).

Kasztenny, B., I. Voloh, C. G. Jones and G. Baroudi, "Detection of incipient faults in underground medium voltage cables", in "2008 61st Annual Conference for Protective Relay Engineers", pp. 349–366 (IEEE, 2008).

Kizilcay, M. and P. La Seta, "Digital simulation of fault arcs in medium-voltage distribution networks", in "15th Power Systems Computation Conference", pp. 22–26 (2005).

Kizilcay, M. and T. Pniok, "Digital simulation of fault arcs in power systems", European Transactions on Electrical Power **1**, 1, 55–60 (1991).

Kottke, D., J. Schellinger, D. Huseljic and B. Sick, "Limitations of assessing active learning performance at runtime", arXiv preprint arXiv:1901.10338 (2019).

Kulkarni, S., S. Santoso and T. A. Short, "Incipient fault location algorithm for underground cables", IEEE transactions on smart grid **5**, 3, 1165–1174 (2014).

Leite, D., A. Martins Jr, D. Rativa, J. F. De Oliveira and A. M. Maciel, "An automated machine learning approach for real-time fault detection and diagnosis", Sensors **22**, 16, 6138 (2022).

Li, T. and H. Li, "Simulation of electric arc characteristics based on matlab/simulink", in "IOP Conference Series: Materials Science and Engineering", vol. 452, p. 042080 (IOP Publishing, 2018).

Li, W., E. Xiang and K. Wang, "Incipient fault identification in power distribution systems based on siamese temporal graph", in "2021 Power System and Green Energy Conference (PSGEC)", pp. 789–794 (IEEE, 2021).

Malkoff, D. B., "A framework for real-time fault detection and diagnosis using temporal data", Artificial Intelligence in Engineering **2**, 2, 97–111 (1987).

Peng, X., X. Jin, S. Duan and C. Sankavaram, "Active learning-assisted semi-supervised learning for fault detection and diagnostics with imbalanced dataset", IISE Transactions pp. 1–15 (2022).

Pilario, K. E. S., Y. Cao and M. Shafiee, "Incipient fault detection, diagnosis, and prognosis using canonical variate dissimilarity analysis", in "Computer Aided Chemical Engineering", vol. 46, pp. 1195–1200 (Elsevier, 2019).

Samet, H., S. Khaleghian, M. Tajdinian, T. Ghanbari and V. Terzija, "A similarity-based framework for incipient fault detection in underground power cables", International Journal of Electrical Power & Energy Systems **133**, 107309 (2021a).

Samet, H., M. Tajdinian, S. Khaleghian and T. Ghanbari, "A statistical-based criterion for incipient fault detection in underground power cables established on voltage waveform characteristics", Electric Power Systems Research **197**, 107303 (2021b).

Wang, L., X. Hu, B. Yuan and J. Lu, "Active learning via query synthesis and nearest neighbour search", Neurocomputing **147**, 426–434 (2015).

Wang, X., M. Sumner and D. W. Thomas, "Application of tlm and cassie-mayr arc model on transformer aging and incipient faults simulation.", Engineering Letters **16**, 1 (2008).

Weng, Y., S. Luo, Q. Cui, R. Trask and H. Wang, "Evaluate dc meter adoption for house-level storage devices", IEEE Transactions on Smart Grid pp. 1–1 (2022).

Xiong, S., Y. Liu, J. Fang, J. Dai, L. Luo and X. Jiang, "Incipient fault identification in power distribution systems via human-level concept learning", IEEE Transactions on Smart Grid **11**, 6, 5239–5248 (2020).

Xu, Z., I. Voloh and L. Torelli, "Impact of incipient faults on sensitive protection", in "2018 71st Annual Conference for Protective Relay Engineers (CPRE)", pp. 1–21 (IEEE, 2018).

Yuan, L., L. Sun, H. Wu *et al.*, "Simulation of fault arc using conventional arc models", Energy and Power Engineering **5**, 04, 833–837 (2013).

Zhang, W., Y. Jing and X. Xiao, "Model-based general arcing fault detection in medium-voltage distribution lines", IEEE Transactions on Power Delivery **31**, 5, 2231–2241 (2016a).

Zhang, W., X. Xiao, K. Zhou, W. Xu and Y. Jing, "Multicycle incipient fault detection and location for medium voltage underground cable", IEEE transactions on power delivery **32**, 3, 1450–1459 (2016b).

Zhang, Y., J. Liu, F. Qiu, T. Hong and R. Yao, "Deep active learning for solvability prediction in power systems", Journal of Modern Power Systems and Clean Energy **10**, 6, 1773–1777 (2022).