AI Powered Power Line Fault Detection

Sudalaiandi Raja Sudalaimuthu, Jayaraman Revathi, Jayasri Raghunathan, Sunil Varghese

October 2019

Abstract

Supplying electricity to cities require High to Medium voltage overhead power lines running for hundreds of miles from the generating stations.These great distances make it expensive to manually inspect the lines for any damage that doesn't immediately lead to a power outage.These damages eventually will lead to a power outage or the start of a fire if not repaired timely. Such kind of damages caused by a tree branch hitting the line or a flaw in the insulator lead to a phenomenon known as 'partial discharge' — an electrical discharge which does not bridge the electrodes between an insulation system completely.

Partial discharges are known as good indicators of degradation of insulation systems. However the background noise distorts the pattern of partial discharges (PD-pattern) and decreases the capability of detection methods to recognize the features of PD-pattern corresponding to the degradation of an insulation system.The objective of the project is to detect correctly the partial discharge patterns in signals acquired from these power lines. Creating an effective classifier using this data will make it possible to continuously monitor power lines for faults and take timely action.

1 Introduction

For long overhead electrical lines insulated conductors are being regularly used nowadays as they are more reliable. However, the disadvantage of the insulated conductors is that now fault detection becomes difficult when the insulation is ruptured due to (i) atmospheric over voltage or (ii) due to destructive degradation of the insulation system when the insulation comes in contact with other dielectrics (tree branch for example). The methods to detect insulation faults is done by evaluating the PD-pattern from the voltage signal.An example of the measured voltage signal for the Partial discharge fault is shown in Figure 1.

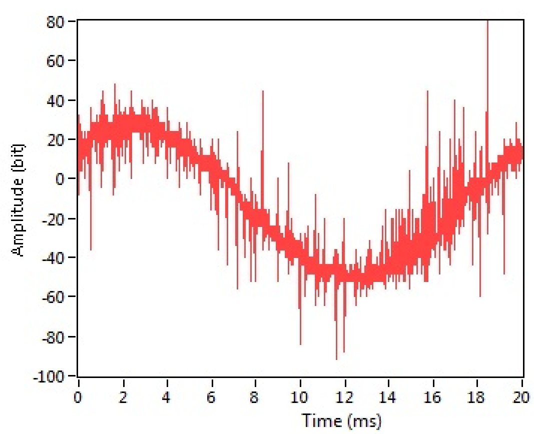

Figure 1: **The measured signal during Partial Discharge**

The issue of PD-pattern detection is a subject of many areas of study, artificial intelligence, signal processing, data analysis, statistic or applied maths. Fourier transformation is a frequently used method of feature extraction and analysis in the application of signal processing. Fourier transformation decomposes the signal's spectra across the frequency domain . Another signal processing method is the wavelet transformation. The wavelet transformation is able to perform a time-frequency analysis.

The models applied for final processing of the extracted features of a PD-pattern are mostly based on machine learning algorithms. It is the presence of external background noise interference which degrades the quality of the observed data. This phenomenon directly implies a lower quality of the detection model.

The background noise is a form of noise pollution or interference of signals generated by another device than analysed, e.g. radio waves. When the analysed PD-pattern is distorted by the background noise disturbing signals, the impulse component consists of relevant and irrelevant (false hits) peaks.

This project handles signal data acquired from the natural environment with high noise interference and low sampling rate,according to the placing of the sensor device . The PD-pattern detection needs to operate online with reasonably low computational power. Therefore, we would like to review and apply only the most relevant approaches for noise suppression, feature extraction and classification. In case of a good outcome, the possible deployment as a PD-pattern detector can increase the reliability of overhead insulated power lines.

2 Related work / References

Detection of Partial discharges (PD) in the Power Transmission is an ongoing world-wide problem. Extensive research is going on currently to address this problem and IEEE paper has been published. IEEE paper - A complex classification approach of partial discharges from covered conductors in real environment To address the above issue, Kaggle has launched the competition. The objective of this project is to develop a low-cost PD detection algorithm using many areas of intelligence such as signal processing, statistics, data analytics and artificial intelligence. The challenge is the noisy data contaminated with radio emission, lightning, switching operations, corona, which may lead to false PD patterns. This entails data flattening and denoising using Fourier transformation, extraction of inherent signal features such as peaks, Fourier coefficients and eminent statistical features. The extracted features are fed into Machine learning algorithms such as linear regression, random forest and Deep learning algorithms such CNN and LSTM

3 Environment Setup

Below are the steps to setup project execution environment. Follow the steps in the same order as specified.

Step1: Download project submission zip file 'AiPowered-power-line-fault-detection.zip' and extract (unzip) the folders and files.

Step2: Create conda environment by running below sciripts in 'Anaconda Prompt' (Anaconda command line interface). Script is available in 'CA3-conda-environment-setup-script.txt' in project base folder.

Step3: Download input datasets from the below URL and copy in to 'aiPowered-power-line-fault-detection
input'.

https://www.kaggle.com/c/vsb-power-line-fault-detection/data

- metadata_train.csv https://www.kaggle.com/c/vsb-power-line-fault-detection/download/metadata_train.csv

- train.parquet https://www.kaggle.com/c/vsb-power-line-fault-detection/download/train.parquet

Step4: Once the datasets have been extracted in datasets folders, run the following python programs using Jupyter notebook : 'aiPowered-power-line-fault-detection/pythonscript/aiPowered-power-line-fault-detection.ipynb'

4 Proposed Approach

The proposed approach can be described as a set of clearly defined steps, where each of the step has its own purpose (see Figure 2).

The above shown steps are described in further detail is the below sections:

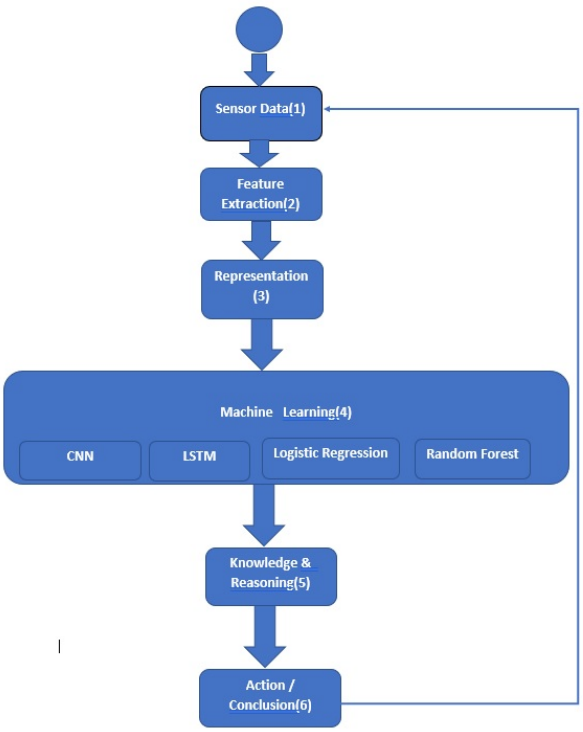

Figure 2: **Steps of the proposed approach**

4.1 Sensor Data

The starting point in the process is the data acquisition (1), which are the raw signals measured in real environment. These signals are already labelled(annotated). Sensor data has total of 8712 records and represented in the parquet file. Each signal has the value of 800000 signals, which is measured in 20ms. Metadata has the meta information such as signal id, id measurement, phase and target.

4.2 Feature Extraction

The next step was to compute a set of features for best possible classification (2). The relevancy of the features was evaluated by their mutual information according to the class labels. An increase in the relevancy of the applied features may lead to an increase in the model's total performance. Below are the analysis of data and feature extraction techniques used in our problem statement.

1. Analysed the meta data and target and concluded as follows.

   (a) Identified the total count of positive and negative partial discharges as below, which represents the dataset has imbalanced class.

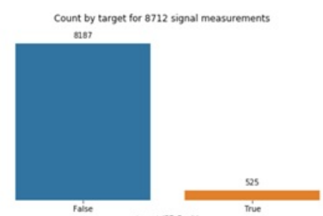

Figure 3: **Count of Positive and Negative PD signals**

   (b) Identified the target is independent of phase, which represents any phase of individual signal can have partial discharge.

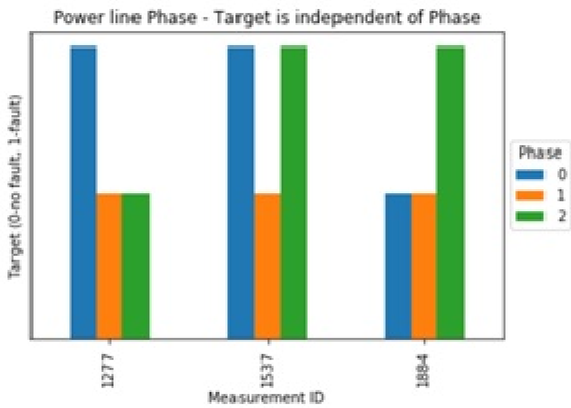

Figure 4: **Target is independent of Phase**

2. Flattening of signals was done to remove the specific noise and anomaly feature and it is calculated as below

$$x = 0.99(x-1) + 0.1(x)$$

where x is the current point

3. De-noising of signal is to remove the background noise caused by noise pollution, interference of signals by other devices like radio waves. The steps for the de-noising are as follow.

   - Apply the Fast Fourier Transformation(FFT) to the signal
   - Compute the frequencies associated with each coefficient
   - Keep only the coefficients which have a low enough frequency (in absolute)
   - Compute the inverse FFT

   FFT is calculated using the Fourier Transformation formula:
   $f_j = \sum_{k=0}^{n-1} x_k e^{\frac{2\pi i}{n} jk}, \quad \forall j = 0, ..., n-1$

4. Peak features:

   (a) Peaks retrieval is performed by picking the threshold by having certain width window to remove the local maxima. Used knee point detection to find the floor noise and sorted from highest to lowest and then looked for the point where the height of the peaks flattened off

   (b) Calculated peak features are as follows

   - Height of the peak
   - RMSE between the peak and sawtooth shaped template
   - Ratio of the peak to the next data point(gradient) (in absolute)
   - Ratio of the peak to the previous data point.
   - The distance to the maximum of opposite of the polarity of the peak.

5. Statistical features:

   (a) Mean
   (b) Median
   (c) 5th Percentile
   (d) 25th Percentile
   (e) 95th Percentile

6. Derived features for shallow learning:

(a) Calculated features of each peak for a signal are grouped by id measurement and quadrants to achieve the dimensionality reduction. Reduced features can be passed to the shallow learning algorithms.

7. Derived features for Deep learning:

    (a) Mean

    (b) Standard deviation

    (c) Mean +Standard deviation

    (d) Mean – Standard deviation

    (e) Percentiles (0, 1, 25, 50, 75, 99, 100)

    (f) Covariance

    (g) Asymmetry (the deviation of 50th percentile from mean)

    (h) Max range

4.3 Representation

1. Peak features for original, flatten, denoise, flatten denoised signal for a sample faulty partial discharge signal and fault free signals are plotted and shown in figures 5 and 6 respectively.

2. Correlation matrix for the derived features of shallow learning are calculated and shown in figure 7.

4.4 Machine Learning

After feature extraction, below machine learning algorithms are identified and created for shallow and deep learning models.

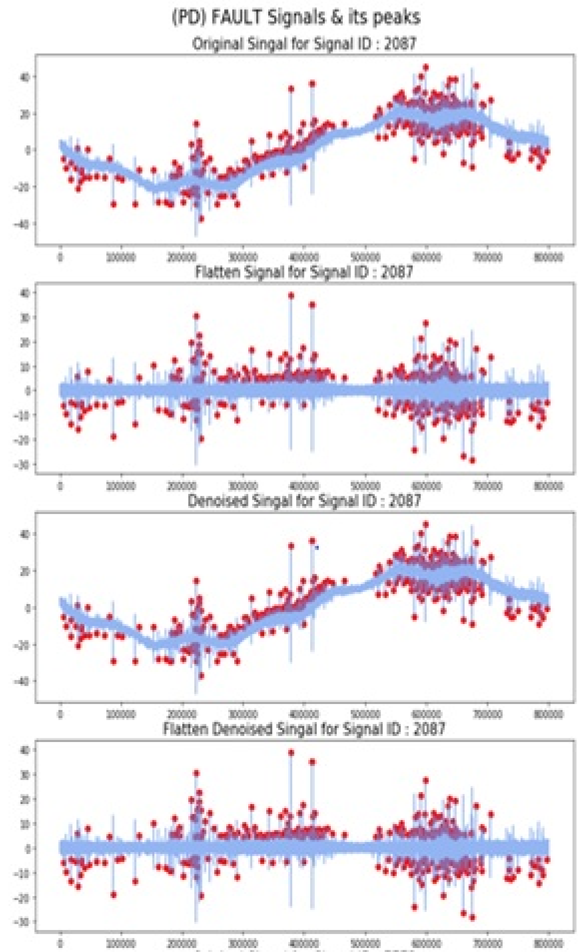

Figure 5: **Faulty Signals and its Peaks**

1. Logistic Regression

2. Random Forest

3. Convolutional Neural Network

4. Long Short-Term Memory

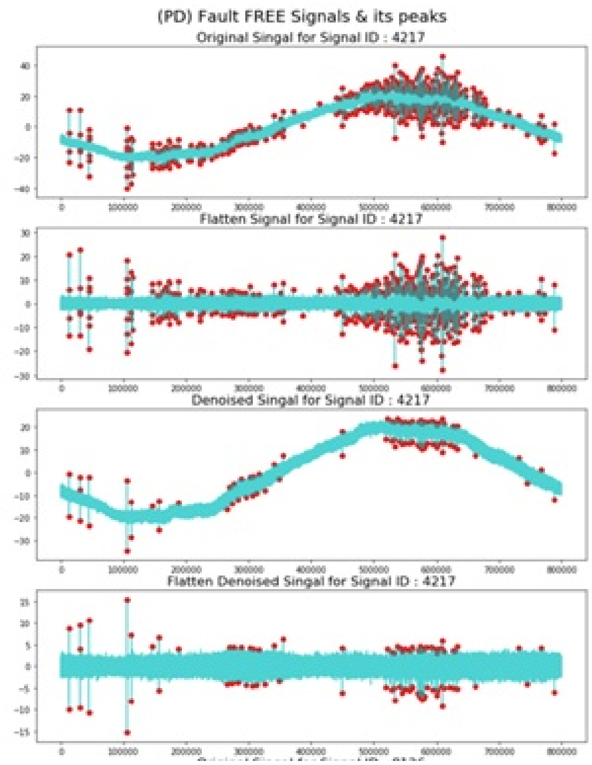

Figure 6: **Fault free Signals and its Peaks**

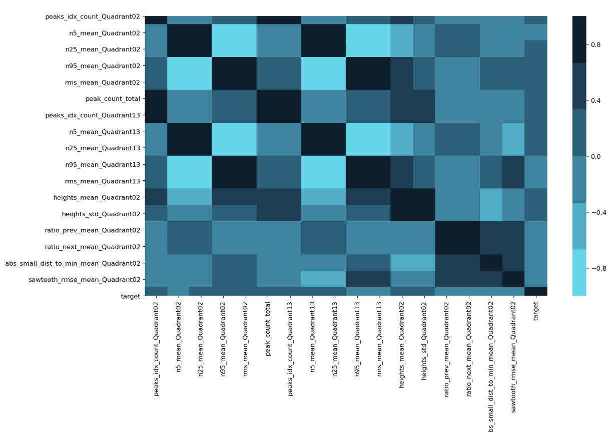

Figure 7: **Correlation Matrix**

4.5 Knowledge and Reasoning

Once the models are created, the metrics were evaluated for all the algorithms and these are described in the Experimental results section.

4.6 Action/Conclusion

Partial discharge is detected by the above algorithmic models. The detection helps to repair the insulation damage on the power lines in early stages before any lasting harm occurs.

5 Experimental results

5.1 Dataset

Dataset has been obtained for this problem from the below URL:

https://www.kaggle.com/c/vsb-power-line-fault-detection/data

Each signal contains 800,000 measurements of a power line's voltage, taken over 20 milliseconds. As the underlying electric grid operates at 50 Hz, this means each signal covers a single complete grid cycle. The grid itself operates on a 3-phase power scheme, and all three phases are measured simultaneously.

5.2 Performance metric

Because of the imbalanced dataset, AUC-ROC score and Mathew Correlation Coefficient can be used as the performance metric. Due to the simplicity in calculations and the popularity, ROC-AUC performance metric is taken and the system should produce results with higher AUC – ROC resulting in lower false-positive rate with higher accuracy of fault detection.

Due to the nature of classification, performance measurement goal to achieve the highest possible ROC-AUC was kept as the primary measurement and Accuracy as the secondary measurement.

5.3 Experimental setup

In order to efficiently train, validate and test the model the sample dataset is split in to 2 distinct datasets. First it is split in to Train-Val dataset of 70% and Test dataset of 25% of sample.

Training dataset is used for training the model and evaluation of the model performance was done using testing dataset. Various hyper parameter tuning techniques were applied to fine tune the models and then evaluated using performance metrics obtained using validation sample dataset.

5.4 Model results

1. Logistic Regression:

   Logistic Regression is one of the basic and popular algorithms to solve a classification problem. Logistic regression model was trained with derived dataset.

   Logistic regression with fine-tuned parameters produced the result with AUC score of 0.9158 for validation dataset with Accuracy of 0.9400 for test dataset.

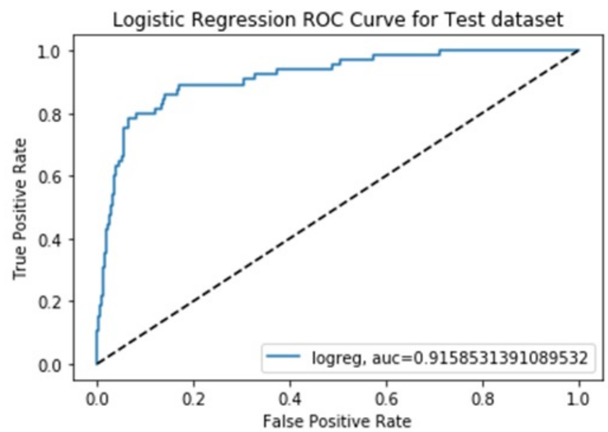

Figure 8: **Logistic Regression**

2. Random Forest Classifier:

   Random forest consists of a large number of individual decision trees that operate as an ensemble. Each individual tree in the random forest spits out a class prediction and the class with the most votes becomes our model's prediction. It was observed that Random Forest with enriched data had higher performance metrics of AUC – 0.95 compared to base data.

   Random Forest Classifier with fine-tuned parameters produced the result with AUC score of 0.9500 with Accuracy of 0.9500 for test dataset.

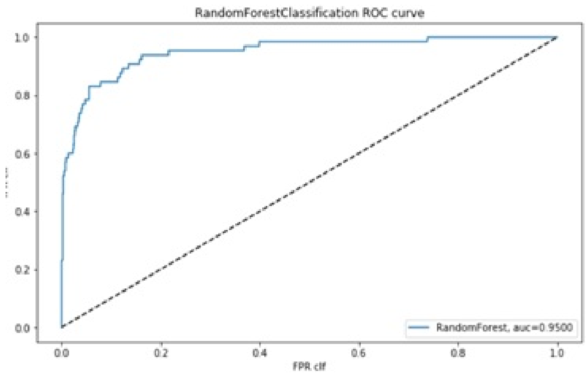

Figure 9: **Random Forest Classifier**

3. Convolutional Neural Network(CNN) Computational models of neural networks have been around for a long time. Neural networks are made up of a number of layers with each layer connected to the other layers forming the network.

   In our problem, there are 8712 signals, each of which comprises of 8,00,000 data points and belong to any of 3 phases 0,1,2. Each signal is divided in to 160 buckets of 5000 data points. Then for each bucket, the 13 statistical features are calculated. Totally there are 39 features including all the 3 phases. This is to create a 2-dimensional input shape for the deep learning algorithms

   This data is fed into CNN deep learning model for prediction a new signal as PD Fault or PD free. There are 3 convolutions filters of sizes (48,64 and 128), followed by Max pooling and dropout to achieve dimensionality reduction and regularization respectively. 2 dense layers are used with output size of 128 and 256 with non-linear activation function Relu. Sigmoid is used for the final layer as it is a binary class prediction

CNN with fine-tuned parameters produced the result with AUC score of 0.9022 with Accuracy of 0.9587 for test dataset.

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	(None, 160, 39)	0
conv1d_1 (Conv1D)	(None, 160, 48)	5664
max_pooling1d_1 (MaxPooling1	(None, 80, 48)	0
dropout_1 (Dropout)	(None, 80, 48)	0
conv1d_2 (Conv1D)	(None, 80, 64)	9280
max_pooling1d_2 (MaxPooling1	(None, 40, 64)	0
dropout_2 (Dropout)	(None, 40, 64)	0
conv1d_3 (Conv1D)	(None, 40, 128)	24704
max_pooling1d_3 (MaxPooling1	(None, 20, 128)	0
dropout_3 (Dropout)	(None, 20, 128)	0
flatten_1 (Flatten)	(None, 2560)	0
dense_1 (Dense)	(None, 128)	327808
dense_2 (Dense)	(None, 256)	33024
dense_3 (Dense)	(None, 1)	257

Total params: 400,737
Trainable params: 400,737
Non-trainable params: 0

Figure 10: **CNN Summary**

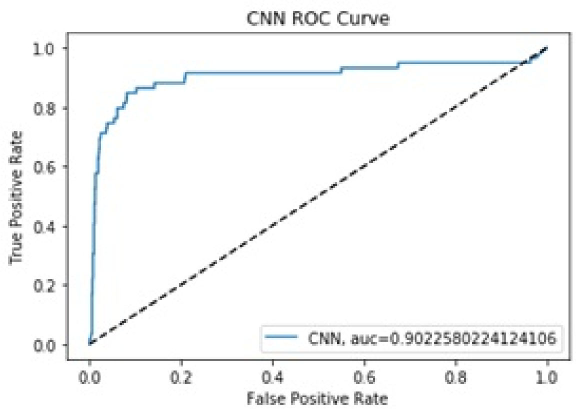

Figure 11: **CNN**

4. Long Short-Term Memory networks (LSTM) LSTM are a special kind of Recurrent Neural Network(RNN), capable of learning long-term dependencies. LSTMs are explicitly designed to avoid the long-term dependency problems similar to the issues found in identifying partial-discharge patterns in a continuous signal measurement. Remembering information for long periods of time is practically their default behaviour, not something they struggle to learn.

   Below LSTM model designed to predict PD-patterns more accurately with higher AUC score. Bidirectional model found to perform better for the 160 segmented features created for each signal measurements.

```
Layer (type)                 Output Shape              Param #
=================================================================
input_1 (InputLayer)         (None, 160, 39)           0
_________________________________________________________________
bidirectional_1 (Bidirection (None, 160, 256)          173056
_________________________________________________________________
bidirectional_2 (Bidirection (None, 160, 128)          164864
_________________________________________________________________
attention_1 (Attention)      (None, 128)               288
_________________________________________________________________
dense_1 (Dense)              (None, 64)                8256
_________________________________________________________________
dense_2 (Dense)              (None, 1)                 65
=================================================================
Total params: 346,529
Trainable params: 346,529
Non-trainable params: 0
_________________________________________________________________
```

Figure 12: **LSTM Summary**

LSTM with hyper-parameter tuned parameters produced the result with AUC score of 0.9598 with Accuracy of 0.9752 for test dataset.

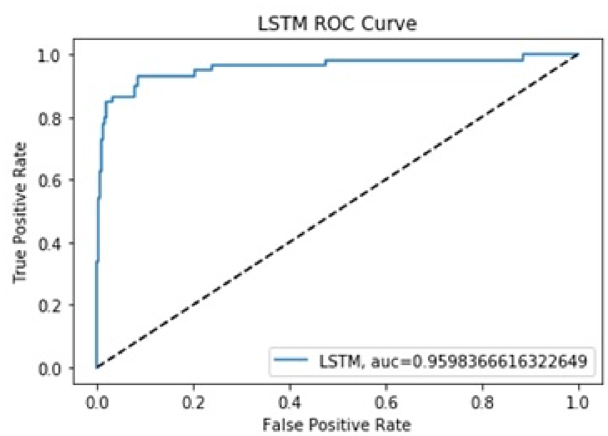

Figure 13: **LSTM**

5.5 Comparison of performance metrics

Best performing, hyper parameter tuned models from each classification are tested with test-dataset (20% of initial sample dataset). Comparison chart is shown below.

LSTM found to perform better compared to 2nd best performing model of Random Forest Classifier. Below is the results for the models comparison and comparison charts.

Model Classification	ROC - AUC	Accuracy
Logistic Regression	0.9158	0.9400
Random Forest Classifier	0.9500	0.9500
CNN	0.9022	0.9587
LSTM	0.9598	0.9752

Figure 14: **Model Comparison Table**

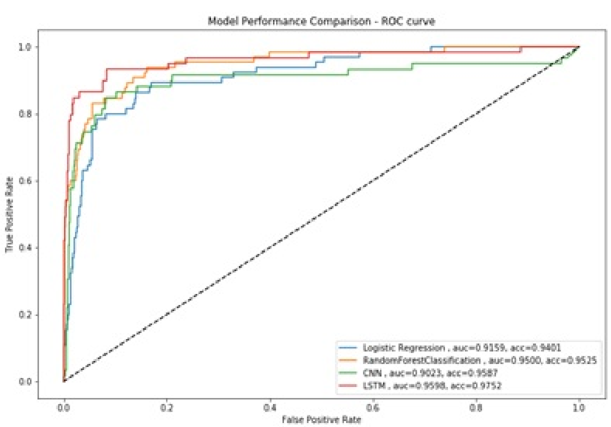

Figure 15: **Model Comparison**

6 Conclusion

6.1 Key Results

For this kind of time series problem, noticed that the LSTM is comparatively gives better performance results than the CNN, Logistic Regression and Random Forest Classifier.

6.2 Limitations

1. Though each signal measurements contains 800000, the number of signal available are only 8712 which may not be sufficient for better learning models

2. As only 6% of data are available for Partial Discharge signals, the imbalanced data can contribute to skewed results.

3. Noticed that the signal pattern for fault free and faulty signal looks similar and difficult to identify because of limited domain knowledge

4. Due to limited domain knowledge in electrical signals processing, it is challenging to extract the suitable features for the building the high-performance machine models.

6.3 Future Work

1. Life become standstill in case of electricity outage. As this problem is wide spread across the globe, the predictive model which are built can be adapted by electricity service providers to prevent the power outage.

2. As partial discharge leads to energy wastage and global warming, the above predictive model helps to rectify the problem without delay and conserve the EARTH. It can also be applied to similar signal patterns like in telecom industry, house hold power wastage.