POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Anomaly Detection of Angular Travel Curves of Low Voltage ABB Circuit Breakers

TESI DI LAUREA MAGISTRALE IN
Electrical ENGINEERING-INGEGNERIA Eletterica

Author: **REZA MALEK MAHDAVI**

Student ID: 10831176
Advisor: Alessandro Pesatori
Academic Year: 2023-2024

Abstract

An electric switch breaker is a kind of electrical device designed to do the actions including connection or interruption of the current of lines in different electrical circuits and power system networks. Moreover, it would have a protection function during the overload and short-circuit situations of the network by connecting to the various kinds of relays to stop the current flow. As a result, they can protect the power system from whole breakout, otherwise it may get difficult to return the network to its normal condition by remaining the fault for a long time. So, the components and structure of the circuit breakers are highly dependent on electrical and mechanical faults and their effects.

All circuit breakers have operating coils known as tripping coils, whenever these coils are energized by switching pulse, the plunger inside them displaced. This operating coil plunger is typically attached to the operating mechanism of circuit breaker, as a result, the mechanically stored potential energy in circuit breaker is released in forms of kinetic energy, which makes the moving contact to move, and these moving contacts mechanically attached through a gear lever arrangement with the operating mechanism.

The performance of circuit breakers depends on various characteristics consisting of nominal voltage and current, type, voltage class, vibration etc. So, the fault in the operation of circuit breakers affected by its internal circuit could cause occurrence of problem in interruption or connection of current, inability to stop the fault current flow and increasing the time of the breaker operation.

Furthermore, it is notable that by considering the vital role of breakers in different levels of voltage zones of power systems, it is important to increase their reliability which can help to increase the transient stability and degree of reliability of power networks.

One of the significant issues of circuit breakers which must be considered is the large arcing between the contacts during the interruption of current. So, the safety manners and regularities should be regarded in numerous operations of power switch breakers.

The purpose of this research is to program and analyze the methods and algorithms on recorded data received from ABB low-voltage circuit breakers in order to detect early signs of possible malfunction in their behavior with the aim of being able to take the appropriate reactions and considerations before the complete failure of circuit breakers. One of the most significant characteristics of circuit breakers for the perfect performance is their angular travel curve during the closing or opening operations. Therefore, based on data collected by ABB on this feature, this work is mainly considered the evaluation of angular travel curves, calculation of their deviation from normal case based on batches of data collected during several endurance tests through the relevant sensor installed on low voltage circuit breakers by ABB. Moreover, python language in Jupiter-notebook environment has been used to develop the codes based on different ML (machine learning) techniques in order to detection of abnormal tests by calculating the anomaly score of all tests and representing of travel curve figures.

Keywords: low-voltage circuit breakers, machine learning (ML), angular travel curve, anomaly detection, fault current, endurance tests

Abstract in lingua italiana

Un interruttore elettrico è un tipo di dispositivo elettrico progettato per eseguire azioni, tra cui il collegamento o l'interruzione della corrente delle linee in diversi circuiti elettrici e reti di sistemi di alimentazione. Inoltre, avrebbe una funzione di protezione durante le situazioni di sovraccarico e cortocircuito del circuito collegandosi ai vari tipi di relè per interrompere il flusso di corrente. Di conseguenza, possono proteggere il sistema di alimentazione da un'interruzione completa, altrimenti potrebbe essere difficile riportare la rete alla sua condizione normale rimanendo guasto per lungo tempo. Pertanto, i componenti e la struttura degli interruttori automatici dipendono fortemente dai guasti elettrici e meccanici e dai loro effetti.

Tutti gli interruttori automatici hanno bobine di manovra note come bobine di sgancio e chiusura, ogni volta che queste bobine vengono eccitate dall'impulso di commutazione, lo stantuffo al loro interno viene spostato. Questo stantuffo della bobina operativa è tipicamente collegato al meccanismo di azionamento dell'interruttore automatico, di conseguenza, l'energia potenziale immagazzinata meccanicamente nel meccanismo dell'interruttore viene rilasciata sotto forma di energia cinetica, che fa muovere il contatto mobile mentre questi contatti mobili sono fissati meccanicamente attraverso una disposizione della leva del cambio con il meccanismo di azionamento.

Le prestazioni degli interruttori automatici dipendono da varie caratteristiche costituite da tensione e corrente nominali, tipo, classe di tensione, vibrazioni ecc. Pertanto, il guasto nel funzionamento degli interruttori automatici potrebbe essere influenzato dal suo circuito interno, dal verificarsi di problemi durante l'interruzione o il collegamento di corrente, incapacità di arrestare il flusso di corrente di guasto e aumento del tempo di funzionamento dell'interruttore.

Inoltre, è da notare che, considerando il ruolo vitale degli interruttori automatici in diversi livelli di zone di tensione dei sistemi di alimentazione, è importante aumentare la loro affidabilità che può contribuire ad aumentare la stabilità transitoria e il grado di affidabilità delle reti di alimentazione.

Uno dei problemi significativi degli interruttori automatici che deve essere considerato è il grande arco tra i contatti durante l'interruzione della corrente. Pertanto, le modalità e le regolarità di sicurezza dovrebbero essere considerate in numerose operazioni degli interruttori di potenza.

Lo scopo di questa ricerca è quello di programmare e analizzare i metodi e gli algoritmi sui dati registrati provenienti dagli interruttori di bassa tensione ABB al fine di rilevare i primi segni di possibili malfunzionamenti nel loro comportamento con l'obiettivo di poter prendere le reazioni e le considerazioni appropriate prima del guasto completo degli interruttori automatici. Una delle caratteristiche più significative degli interruttori per la perfetta prestazione è la loro curva di corsa angolare durante le manovre di chiusura o apertura. Pertanto, sulla base dei dati raccolti da ABB su questa caratteristica, questo lavoro è principalmente considerato la valutazione delle curve di corsa angolare, il calcolo della loro deviazione dal caso normale sulla base di lotti di dati raccolti durante diversi test di resistenza attraverso il relativo sensore installato sul circuito a bassa tensione interruttori di ABB. Inoltre, il linguaggio Python nell'ambiente Jupiter-notebook è stato utilizzato per sviluppare i codici basati su diverse tecniche di ML (apprendimento automatico) al fine di rilevare test anomali calcolando il punteggio di anomalia di tutti i test e la rappresentazione dei dati della curva di viaggio.

Parole chiave: interruttori di bassa tensione, apprendimento automatico (ML), curva di corsa angolare, rilevamento di anomalie, corrente di guasto, test di resistenza

Contents

Introduction

In recent years, power circuit breakers play a crucial role in improving the reliability and stability of power systems all over the world. Appropriate operation of circuit breakers can increase the efficiency of the whole performance of power network to supply different levels of loads, however, the malfunction of circuit breakers could affect ability of power system facing with a sudden events and faults. In this section, the anomaly score calculation techniques introduced in some background and previous works will be presented.

One of the methods in anomaly detection of electrical devices is known as covariance matrix estimation technique which is discussed in [1]. In this paper, the proposed depth-based covariance estimation has a better performance compared to other estimation approaches like t-distribution and Tyler methods on Hyperspectral (HS) data. In [2] the composite anomaly detection on data-driven power plant condition monitoring has been done. Also, it has been demonstrated that this method provides more reliable conditions and detect anomalies more accurately and effectively.

In [3] the anomaly detection based on data-driven deep learning approach performed by using sensor-level detectors. According to this paper, information based on Sensor-level detectors cause more efficient decisions at the control unit.

Furthermore, Different clustering methods could be used as anomaly detection approaches in numerous problems of data analytics. By regarding smart grid anomaly detection in [4] a hierarchical clustering method has been developed to consider anomaly detection based on three challenges which are large-scale measurements, detection of missing points and selection of variables. In [5] the anomaly detection and fault location are considered by using a new approach based on developing self-adjoint matrix and the results have been improved in comparison with previous works.

The anomaly detection of power quality is done by sequential Bayesian approach in [6]. Also, this approach is effective in detection of noises from power electronic devices used in the network to eliminate them in an appropriate way.

In [7] two variants of PMUNET: DCDL and CDL are proposed for anomaly detection over data drifting synchro phasor data stream. Anomalies represent any event in the power grid resulting in a reasonable impact on the grid and affecting the power grid operation strategies by effective usage of deep learning approach.

In [8] the cloud computing method is considered for anomaly detection on dispatching data of power grid. This approach is based on clustering technique by detecting the isolated points with a far distance from the core of each cluster (calculation of proposed distance) shown in this paper.

The data analytics for power transmission equipment based on random matrix approach has been done in [9]. According to this paper the proposed approach is useful for large random matrices and online monitoring of power grid.

In [10] the SDAE (stacked denoising autoencoder model) is proposed to effectively reconstruct abnormal data and abnormal status representation by training the network model ability. Moreover, in this paper KDE approach based on clustering, has been regarded to define the threshold value of reconstruction errors (anomaly scores).

In the paper [11] the graph-based method known as CONGO develops to detect anomalies on power grids based on unsupervised ML technique and 3-phase short circuit is simulated on IEEE37-node system.

In [12] a data analytics technique to monitor and detect malicious activity in the cyber-physical transmission protection is considered. The proposed method utilizes streaming PMU and cyber data, and breaker status data. Moreover, bad data and missing data in the streaming PMU data streams are detected through LSTM networks.

The DYNWATCH, known as an online algorithm, is proposed in [13], which can accurately detect anomalies using sensor data on a changing graph (grid) during time and it has a good efficiency than baseline approaches.

In [14] An Ensemble-based anomaly detector for PMU data is developed in this paper. The ensemble method is observed to have a good performance in all the tests based on the simulated and the real-world data. The considered case studies demonstrate the efficiency of the ensemble-based anomaly detector.

In [15] the FDI (False Data Injection) attack and defense using reinforcement learning techniques in a case that Automatic Voltage Controllers (AVC) may cause some abnormal effects on the operation of power systems has been evaluated.

In the paper [16] NARX-CUSUM and PLS-CUSUM which are two residual-based machine learning models proposed and the health of power electronic evaluated by anomaly detection process on MOSFETS is considered.

In [17] the DNN-based FALCON system to classify faults and cyber-attacks and determine the location of faulty lines in power distribution systems is considered. Based on the simulation results the proposed model can detect the anomaly type and location with high level of accuracy.

In [18] anomaly Effects and costs on Low-Voltage Mobile Power Systems regarded and some features and conclusions related to aircraft applications at this paper mentioned.

In [19] Anomaly detection of power consumption considering electricity stealing and unexpected power energy which is one of the essential routine works in power system management and maintenance, proposed. A three-stage multi-view stacking ensemble (TMSE) machine learning model based on hierarchical time series feature extraction represented in this paper.

In [20] a model-based anomaly detection approach to detect measurement anomalies due to stealthy cyber-attacks that utilizes information that is essentially independent of traditional SCADA measurements considered.

In the figure below the ABB sample low voltage circuit breakers are represented which are used for mechanical tests in this research. The nominal voltage is around 660 Volts.

Fig A. ABB circuit breakers used for testing.

1. Travel curves of circuit breakers

Introduction

One of the essential characteristics of circuit breakers is angular travel curves of these devices. The travel curve of circuit breakers has considerable impact on their performance so, it is important to always monitor the travel curve by system operators. The performance of a circuit breaker is highly dependent on the correct travel curve provided by the operating mechanism. As a result, a permanently installed travel transducer allows one to record the position of the contacts of circuit breakers vs. time for every device. Various practical types of angular transducers used and installed on low voltage circuit breakers. Their responsibility is mainly to gather data recorded by sensors and sent to scope to have a graphical view for future analysis.

Fig 1.1 presents an analog transducer used in ABB lab to record the data collected by sensors and send them to show on the screen of real scopes.

Fig 1.1. Analog Transducer in lab

Also, Fig 1.2. represents the sample Angular travel curve of low voltage circuit breaker for both closing and opening cases. Various parameters could be shown on this figure:

- Opening speed
- Closing speed
- Over travel and rebound by closing operation
- Over travel and rebound by opening operation

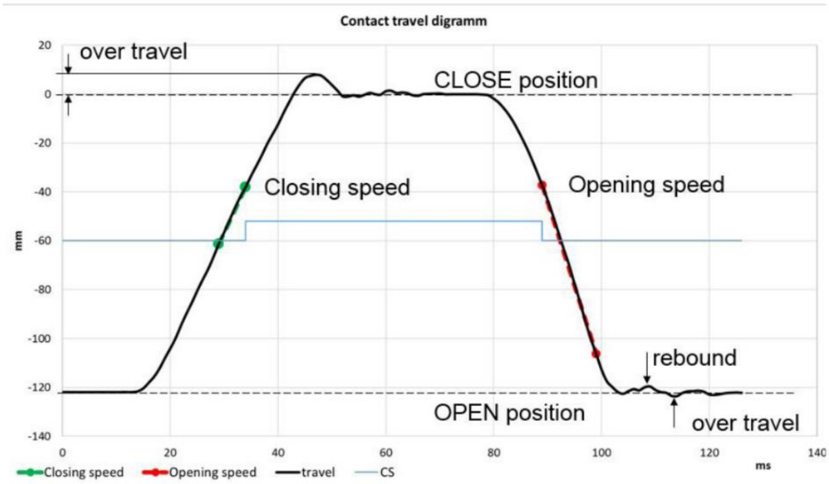


Fig 1.2. Angular curve of circuit breaker

A lower speed by closing operation increases the duration of pre-arc resulting in a heavier thermal stress of the arcing contacts. The time interval between contacts touch and current commutation to main contacts increases as well, exposing the tulip to higher current values and consequent electrodynamic forces. The resulting higher friction between the tulip and plug could prevent the breaker from completing the intended closing operation with possible failure of latching, or in the worst-case scenario, damage the arcing contact system causing an internal catastrophic fault. When the closing speed exceeds the maximum value, depending upon the specific design, the consequences can go from higher mechanical stress to irreparable damage of the nozzle, plug and tulip with fatal consequences on the next opening operation.

A rebound means that the contacts come together after having reached their fully open position representing a risk of re-strike by clearing. Excessive over travel while opening could come from an opening speed which is too high or a possible problem with the damper of the operating mechanism. The consequences of this range from a general higher mechanical stress up to damage of the interrupter due to internal collisions. If the opening speed is correct and the over travel and rebound by opening has the tendency to increase, it is a good indication that the damper in the operating mechanism is less and less efficient.

1.1 Type of study

In this study, the anomaly detection of angular travel curves of two ABB low voltage circuit breakers in the same model analyzed by considering the different ML techniques in python environment and the number of tests which have been detected as normal or anomaly presented. Moreover, diverse angular travel curves by various colors in the relating figures have been represented.

1.2 Objective tests

A number of endurance tests have been done in ABB lab in Bergamo and the data relating to angular travel curves of circuit breakers recorded by relevant sensors installed on circuit breakers. Angular travel curves of low voltage circuit breakers are typically used to monitor and analyze the performance of these devices. The sensors responsible for collecting data from these curves are generally position sensors or angle sensors. They provide information on the movement and position of the circuit breaker's operating mechanism.

Fig 1.3 represents the practical mechanical test in the lab. By using the mechanical handle, the automatic operation can be done for closing and opening tests of circuit breaker. Also, Precision instrument air has been used to move the handle automatically during the mechanical tests.

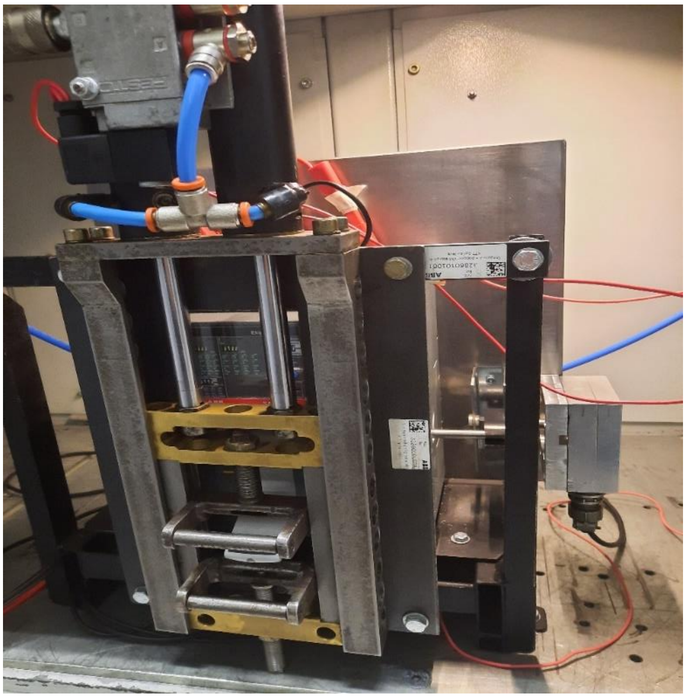


Fig 1.3. Circuit breaker under endurance test

There are different types of sensors that could be used for different purposes, depending on requirements. Here are a few commonly used sensors in breakers:

• Angular sensor: detects the rotation of the switch main shaft (angular travel curve).

• Vibration sensor (mono-axial): detects the acceleration profile at a specific point of the switch.

• Hall sensors: detect the absorption current of the auxiliary accessories.

The travel transducer can provide various essential parameters, including the opening and closing times, contact travel distances, and velocity profiles. This information is utilized for analyzing the breaker's performance, evaluating its reliability, and ensuring that it operates within specified timing and motion requirements.

The data collected by the travel transducer can be recorded and used for several purposes, such as anomaly detection, condition monitoring and predictive maintenance of circuit breakers. It helps in identifying potential issues, abnormalities and it could affect the breaker's operation or performance.

1.3 Number of tests and detection method

The total number of tests done by ABB experts is 11170 and 12076 for each endurance test (each device) respectively, which last for several weeks. The angular sensors installed on the breakers have the responsibility to collect data during the test. These sensors collect data on the angular position of the circuit breaker's mechanism at various points during its closing or opening operation. The collected data can be used for analysis, diagnostics, and condition monitoring to ensure the proper functioning and performance of low voltage circuit breakers.

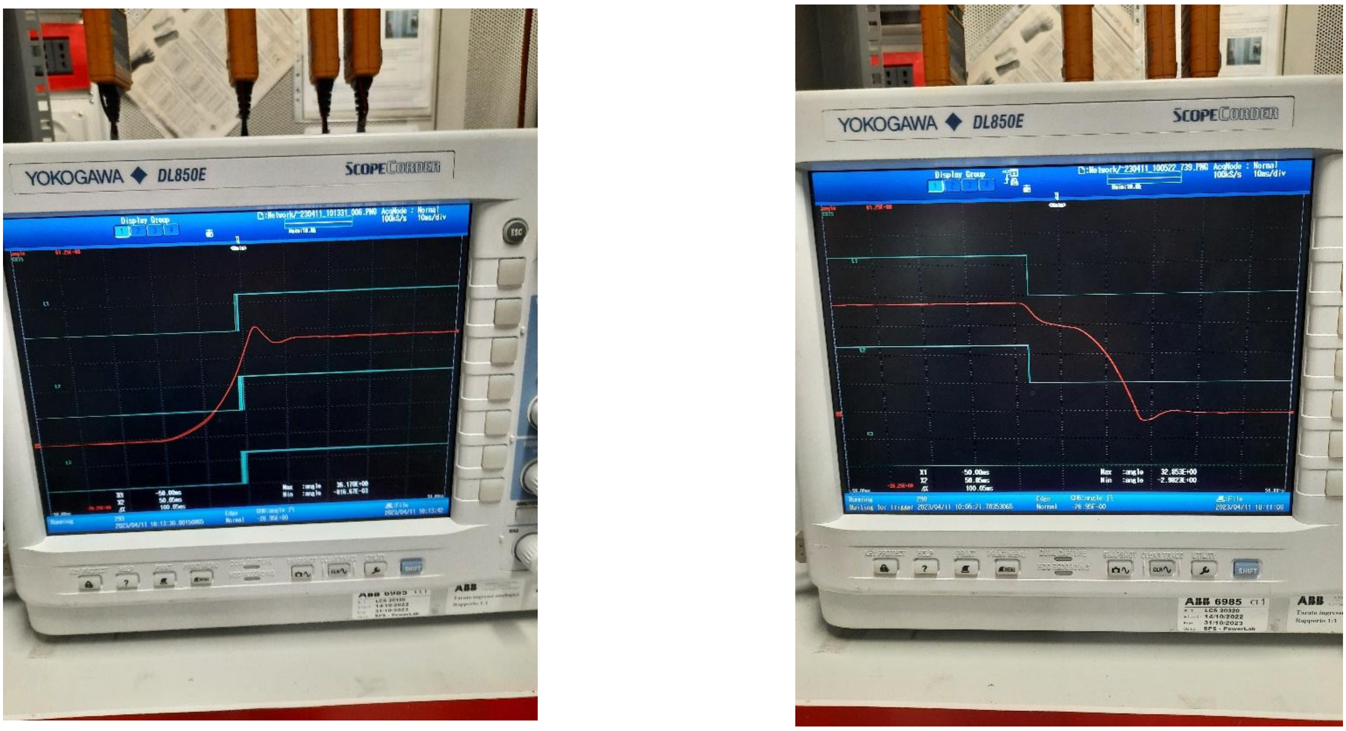


Fig 1.4. Closing and opening practical results.

Fig 1.4 Shows the practical case of figures for closing (left) and opening (right) of endurance tests. The data recorded by sensors have been shown on scopes in the actual position. Each closing and opening are the pairs of one unique circuit breaker test which is done one after another respectively.

2. Models of Analysis

In this section the mathematical formulation of different machine learning techniques and algorithms used in this analysis are discussed. In addition, a new way for threshold selection regardless of predefined value based on statistics and considering appropriate probability density function is introduced in the next chapter.

2.1 Auto-Encoders

The first approach in this research which is considered for evaluation and anomaly detection of angular travel curves of low voltage circuit breakers is Auto-Encoders. Auto-Encoders is based on neural networks with the main concept to copy its input to output. In addition, Auto-Encoders are used to train the input data by compression and normalization of raw data. Autoencoders are based on unsupervised and semi-supervised machine learning that applies the backpropagation technique and sets the target values equal to the inputs. They work based on dimensionality reduction. One of the advantages of using Auto-Encoders is the performance of them to confront with the non-linearity of data. It allows the Model to learn very powerful generalizations. And the main goal is to reconstruct the output back with minimum loss of information than other similar techniques. The main characteristics of Auto-Encoders can be classified as follows:

- It can be used as unsupervised or semi-supervised ML algorithm.
- Minimization of its objective loss function to have a better performance.
- The output of neural network is its output.
- It can learn non-linear activation functions and multiple layers.
- More efficient in model parameters to learn several layers.
- It contains an encoder in input and decoder in output for doing actions on input data.

- data specific means that Auto-Encoders could only compress data similar to what they have been trained on.
- Loss meaning is that the decompressed output data will not the same as input.

Furthermore, Auto-Encoders, in addition to dealing with non-linear data, can provide different applications from Computer vision to time series forecasting.

2.1.1 Components of Auto-Encoders

Autoencoder is constructed with three main parts known as encoder, Bottleneck/Latent Space, and decoder.

Encoders:

The encoder function is to compress the input data into a latent space or bottleneck representation. This layer encodes the input information as a compressed representation in a reduced dimension; so, the compressed data looks like the original input but not the original one. The encoder network typically consists of multiple hidden layers that continuously reduce the dimensionality of the input data. Each hidden layer uses activation functions and weight parameters to extract relevant features from the input and create a compact representation.

Bottleneck/Latent Space:

The bottleneck layer represents the compressed version of the input data generated by the encoder. It includes a low-dimensional representation of the original data, capturing the most significant patterns with related parameters.

The bottleneck of an Auto-Encoders refers to the layer or layers in the middle of the network, also known as the latent space or encoding layer. The purpose of the bottleneck is to capture and represent the most important features of the input data in a compressed form. By limiting the dimensionality of the bottleneck layer, the autoencoder is forced to learn a more compact representation of the data, which can be useful for tasks such as denoising, anomaly detection, and generating new samples.

However, there are a few potential bottlenecks or limitations associated with the hidden layers of Auto-Encoders:

Information loss:

- By compressing the input data into a lower-dimensional representation, some information from the original data may be lost in the bottleneck layer. This loss of information can affect the quality of the reconstructed output.

Underfitting:

- If the bottleneck layer has insufficient capacity or if the dimensionality is too low, the Auto-Encoders may struggle to capture the essential features of the input data which can cause underfitting, where the reconstructed output is a poor approximation of the original input.

Overfitting:

- On the other hand, if the bottleneck layer is too large or if the dimensionality is too high, the autoencoder may memorize the training data and fail to generalize well to new and unseen data. This scenario can result in overfitting.

Lack of interpretability:

- While the bottleneck layer is designed to capture essential features, it may not always provide human-interpretable representations. The encoded features may be highly abstract or incomplete, making it challenging to understand their meaning.

Addressing these bottlenecks often involves careful tuning of the autoencoder architecture, including the size and dimensionality of this hidden layer, regularization techniques, and training strategies. It's important to set a balance between the compression capability of the bottleneck and the ability to reconstruct the input faithfully while preserving the desired properties of the data.

Decoder:

The decoder decodes the encoded data back to the original case of the same input dimension. The decoder takes the data from the lower latent space to the reconstruction phase, where the dimensionality of the output is equal to input. In addition, like the encoder, the decoder consists of multiple hidden layers, but in a mirrored structure each layer gradually increases the dimensionality of the output until it matches the original input size. But in the case of Autoencoders, there is lossy compression, so after the compression, in an ideal case the output is the same as input but in fact there is a loss which is the difference between input and output known as reconstruction loss or reconstruction error.

Reconstruction Loss:

During training, autoencoders minimize a reconstruction loss, which measures the difference between the input data and the reconstructed output. Normally, different loss functions can be used consisting of mean squared error (MSE), mean absolute error (MAE) or binary cross, depending on the type of reconstructed data and other characteristics of the model used for specific problem.

2.1.2 Architecture of Auto-Encoder

The architecture of Auto-Encoders allows us to have a deeper insight into the hidden layers. Therefore, in Autoencoders, there are some layers between input and output, and the sizes of these layers are smaller than the input layer.

A critical part of the Autoencoder is the bottleneck. The bottleneck approach is a beautifully elegant approach to represent learning specifically for deciding which aspects of data are relevant information and which aspects are not useful.

The architecture of Auto-Encoders is shown in Fig 2.1. In this case, the difference between input and output is known as reconstruction error (error between input vector and output vector). One of the predominant use cases of the Autoencoder is anomaly detection. For instance, sensors in CPU, memory devices etc. Different parts of Auto-Encoders including input layer, Encoder, Decoder, and output layer are represented in the figure below.

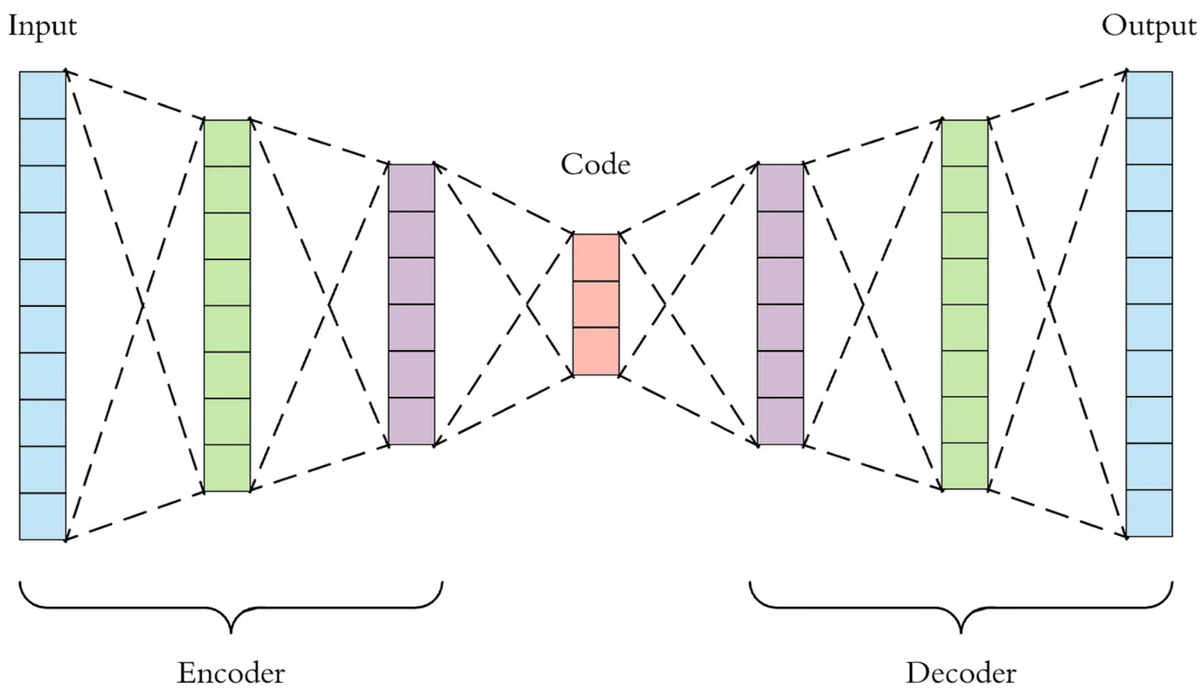


Fig 2.1. Auto-Encoder model

Normal data are used to train Auto-Encoders and the objective is to minimize the reconstruction error. As training progresses, the model weights for the encoder and decoder may need to be updated. The encoder is a down sampler, and the decoder is an up sampler.

The architecture of Autoencoders have several features:

Unsupervised or semi-supervised Learning:

- Autoencoders learn from unlabeled or semi-labeled data without the need for explicit target outputs. The main purpose is to learn and capture patterns and structures within the data.

Dimensionality Reduction:

- The compressed latent space produced by the encoder acts as a low-dimensional representation of the input data. This dimensionality reduction can help in tasks such as data visualization and feature extraction.

Data Reconstruction:

- Autoencoders reconstruct the input data from the compressed representation in the decoder. By minimizing the reconstruction loss, the autoencoder learns to generate outputs that closely resemble the original inputs.

Anomaly Detection:

- Autoencoders can detect anomalies or abnormalities by comparing the input data with the reconstructed output. Unusual or anomalous samples tend to have higher reconstruction errors, indicating deviations from normal patterns.

Regularization and Noise Robustness:

- Autoencoders can act as regularization techniques by forcing the model to learn a compact representation of the data. Furthermore, they can handle noisy input data by learning to reconstruct clean versions, effectively denoising the inputs.

In addition, in Auto-Encoders for normal data the reconstruction error is low but for abnormal (anomaly) data it can be considerable. As a result, by calculation of reconstruction errors and comparing it with threshold value the anomaly detection process can be regarded. Also, Autoencoders have various variations and enhancements, such as denoising autoencoders, variational autoencoders (VAEs), and sparse autoencoders, which apply additional constraints or modifications to the basic architecture to achieve specific objectives or improve performance in practical usages.

2.1.3 Training of Auto-Encoders

Before starting the train of Auto-Encoders it is needed to set 4 major parameters as follows:

1) number of nodes in the middle layer: the smaller size results in more compression.

2) Number of Auto-Encoder Layers: high layers equal to deep Auto-Encoder

3) Loss function: Calculation of the loss to update the weights, which we need to minimize using optimizer and weight updating. Mainly mean squared error, binary cross-entropy, or other techniques.

4) Number of nodes per layer: the number of nodes decreases with each subsequent encoder layer and increases back in the decoder.

2.2 LSTM (Long Short-term memory)

Long Short-Term memory (LSTM) Network is a deep learning, sequential neural network that allows information to persist. It is a special type of Recurrent Neural Network (RNN) which can handle the eliminating gradient descent problem of RNN.

LSTM was designed by Hochreiter and Schmidhuber that resolves the problem caused by traditional RNNs and machine learning algorithms.

Moreover, RNNs can remember the previous information and use it for processing the current input. The shortcoming of RNN is that they are not able to remember long-term dependencies due to vanishing gradient. LSTMs are designed to avoid long-term dependency problems. In the rest of this section some general features of LSTMs will be explained.

Memory Cell:

LSTMs include a memory cell that retains data over time. This memory cell allows LSTMs to learn and remember long-term dependencies in sequential data. It preserves information by using gates, which regulate the flow of information and data sets into and out of the memory cell.

Hidden State:

LSTMs maintain a hidden state, which acts as the output of the LSTM at a given time step. The function of hidden state is to carry information that the LSTM has learned and is used as input for the next time step.

Backpropagation Through Time (BPTT):

The training process of LSTMs is based on BPTT algorithm, which extends backpropagation to handle sequences. BPTT calculates gradients by unfolding the LSTM over time and propagating errors from the output back to the input. This gives a chance for the LSTM to learn from previous information and patterns while considering the temporal context.

Long-Term Dependency Handling:

LSTMs are designed to capture long-term dependencies in sequential data by eliminating the vanishing gradient problem. The forget gate allows the LSTM to keep useful or discard irrelevant information over various sequences, enabling the network to remember target information for extended periods.

Bidirectional LSTMs:

Bidirectional LSTMs (BiLSTMs) can be also used in some applications. A BiLSTM combines two LSTMs, one processing the input sequence in forward order and the other in reverse order. This enables the network to capture information from both past and future contexts, improving its ability to model complex dependencies in sequential data.

Overall, LSTMs could be effectively used to capture long-term dependencies. Their ability to retain information over time, along with the gating mechanisms, makes them particularly appropriate for applications which involve sequential data analysis and prediction.

2.2.1 LSTM Architecture

The LSTM network consists of three main parts, as shown in figure 2.2. Also, each part performs an individual function.

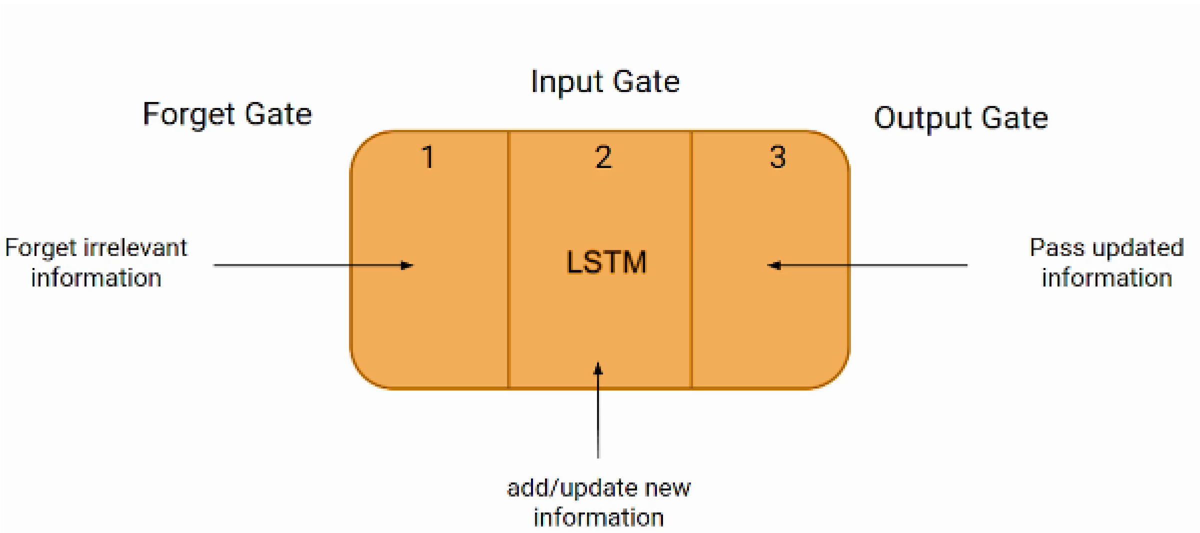


Fig 2.2. LSTM Structure

The first part chooses whether the information coming from the previous timestamp is to be remembered or is irrelevant and can be forgotten which is noted as useful or un useful information. In the second part, the cell tries to learn new information and features from the input to this cell. At last, in the third part, the cell passes the updated information from the current timestamp to the next timestamp. This one cycle of LSTM is considered as a single-time step.

These three parts of an LSTM unit are known as gates. They control the flow of information in and out of the memory cell or LSTM cell. The first gate is called Forget gate, the second gate is known as the Input gate, and the last one is the Output gate. An LSTM unit that consists of these three gates and a memory cell or LSTM cell can be considered as a layer of neurons in traditional feedforward neural network, with each neuron having a hidden layer and a current state.

Moreover, the hidden state is known as short term memory, and the cell state is known as long term memory.

2.2.2 Forget Gate

In a cell of the LSTM neural network, the first step is to decide whether we should keep the information from the previous time step or forget it. In other words, the forget gate determines which information is not useful and needs to be eliminated from the memory cell. It takes input the previous hidden state and the current input, passes them through an introduced activation function, and produces a forget gate vector. This vector is multiplied element-wise with the previous cell state, allowing the LSTM to forget or remember specific information. Here is the equation of forget gate.

$$f_t = \sigma(X_t \times U_f + H_{t-1} \times W_f) \tag{8}$$

the parameters used in this equation can be represented as follows:

X_t: input to the current timestamp.

U_f: weight associated with the input.

H_{t-1}: The hidden state of the previous timestamp

W_f: The weight matrix associated with the hidden state.

2.2.3 Input Gate

The input gate is used to quantify the importance of the new information carried by the input. The input gate adjusts the flow of new information into the memory cell. It consists of two major parts: the input gate and the candidate values. The input gate determines which values must be updated in the cell state, while the candidate values are potential new values that could be added to the cell state. The input gate equation can be presented as follows:

$$i_t = \sigma(X_t \times U_i + H_{t-1} \times W_i) \tag{9}$$

X_t: Input at the current timestamp t

U_i: weight matrix of input

H_{t-1}: A hidden state at the previous timestamp

W_f: Weight matrix of input associated with hidden state.

2.2.4 Output Gate

The output gate controls the output of the LSTM and selects relevant information from the memory cell. It takes the previous hidden state and the current input, passes them through a suitable activation function, and generates an output gate vector. This vector is multiplied element-wise with the cell state, producing the current hidden state. The output gate formula can be presented as follows:

$$O_t = \sigma(X_t \times U_o + H_{t-1} \times W_o) \tag{10}$$

The LSTM architecture allows the network to learn and retain information over long sequences, enabling it to capture dependencies that are further apart in time. The gating mechanisms provide the LSTM with the ability to control the flow of information, allowing it to selectively retain or discard information as needed. This makes LSTMs particularly effective for tasks involving sequential data, such as natural time series analysis and signal processing.

2.3 Isolation Forest

Isolation Forest is a kind of method to detect outliers or abnormal information in data that was first introduced by Fei Tony Liu and Zhi-Hua Zhou in 2008. The approach applies binary trees or isolation trees to identify anomalies, resulting in a linear time complexity and low memory usage that is well-suited for processing and analysis of many datasets.

Moreover, Isolation Forest has been known as one of the fast and reliable algorithms for anomaly detection in various fields such as cybersecurity, finance, and medical research. Although the efficiency of this technique is undeniable, in some cases it is unable to show suitable performance to detect anomalies and abnormal behavior of data.

In an Isolation Forest method, selected sub-sampled data is processed in a tree structure. The samples that travel deeper into the tree are less likely to be anomalies as they require more cuts to isolate them. Similarly, the samples which end up in shorter branches indicate anomalies as it was easier for the tree to separate them from other observations.

The path length in this technique is a crucial feature in Isolation Forest. It measures the number of splits required to isolate an instance. Anomalies are expected to have different average path lengths compared to normal instances. By measuring the path length, the algorithm can quantify the degree of isolation for each data point.

The isolation score is calculated for each data point and serves as an anomaly score. It is derived from the average path length across all the trees in the forest. Anomalies will have different average path lengths, indicating that they are easier to isolate and distinguish from normal data points. Hence, higher isolation scores compared to predefined threshold value indicate a higher likelihood of being an anomaly.

Decision Function and Threshold:

The anomaly scores obtained from the forest can be used to determine whether a data point is an anomaly or not. By defining a threshold on the anomaly scores, data points with scores over the threshold can be classified as anomalies.

Isolation Forest offers a simple yet effective approach for anomaly detection, especially in scenarios where anomalies are isolated and differ from normal data points. Its ability to handle high-dimensional datasets and provide interpretable results makes it a valuable tool for detecting outliers in various domains, such as fraud detection, network intrusion detection, and system monitoring.

The main advantages of Isolation Forest can be categorized as follows:

- Efficiency:

  Isolation Forest has linear time complexity, making it efficient for processing large datasets. The recursive strategy allows it to isolate anomalies faster than many other outlier detection techniques.

- Insensitivity to Normality:

  Isolation Forest does not assume a specific data distribution or require normality assumptions. It can detect both global and local outliers effectively.

- Scalability:

  Isolation Forest is parallelizable, allowing it to take advantage of multi-core or distributed computing environments, further enhancing its scalability.

And the limitations and disadvantages of Isolation Forest:

- Sensitivity to Sample Size:

  Isolation Forest may struggle with very small sample sizes, as it requires a minimum number of instances to construct meaningful partitions.

- Sensitivity to Parameters:

  The performance of Isolation Forest can be influenced by the number of trees in the ensemble and the subsample size used for each tree. Proper parameter tuning is essential for optimal results.

- Difficulty in Handling Clustered Outliers:

  Isolation Forest may have difficulty differentiating clustered outliers from normal instances, as they may require similar partitioning steps.

3. Dynamic Threshold Techniques

In this section the introduction and relating formulas of two different threshold techniques known as kernel function and Kalman filter methods which have been used in this research will be represented.

3.1 Kernel Function

In machine learning and kernel methods, a kernel function, also known as a kernel, is a mathematical function that calculates the similarity or distance between pairs of data points in a higher-dimensional feature space. Kernels are commonly used in support vector machines (SVMs) and other algorithms for tasks such as classification, regression, and clustering.

The main purpose of kernel functions is to enable algorithms to operate in a higher-dimensional feature space without explicitly calculating the coordinates of the transformed data points. They allow algorithms to implicitly capture complex relationships and nonlinear patterns by measuring the similarity or dissimilarity between data points in the original input space.

The Kernel Distribution Function (KDF) is a non-parametric approach used to estimate the probability density function (PDF) of a random variable. It is based on the concept of kernel density estimation (KDE), which approximates the underlying distribution by convolving each data point with a kernel function.

There are several common estimations of the kernel function used in kernel density estimation and the its distribution function. A few widely used kernel functions:

- Gaussian (Normal) Kernel: The Gaussian kernel is one of the most popular choices for kernel density estimation which can be represented as follows:

$$K(x) = (\frac{1}{\sqrt{2\pi\delta^2}}) \times e^{-\frac{1}{2}\left(\frac{x}{\delta}\right)^2} \quad (11)$$

where x is the input value and σ is standard deviation. The Gaussian kernel produces a smooth, bell-shaped curve centered around each data point.

- Epanechnikov Kernel: The Epanechnikov kernel is another common choice for kernel density estimation. The formula regarding this case is as follows:

$$K(x) = \left(\frac{3}{4}\right)(1 - x^2) \qquad \text{for } -1 \leq x \leq 1 \quad (12)$$

- Uniform Kernel: The uniform kernel assigns equal weight to all data points within a fixed bandwidth which will be defined as follows:

$$K(x) = (\frac{1}{2h}) \qquad \text{for } -h \leq x \leq h \quad (13)$$

where h is the bandwidth. The uniform kernel provides a rectangular shape and is less sensitive to individual data points compared to the Gaussian or Epanechnikov kernels. However, it can produce a higher variance estimate.

It must be noted that the selection of kernel function strongly depends on the characteristics of the data and the considered properties of the density estimate, such as smoothness, sharpness, or robustness to outliers. In this research the gaussian kernel will be used to estimate the optimal threshold value as the first technique to compare the output reconstruction errors or anomaly scores of all endurance mechanical tests to identify the normal and anomaly curves in different proposed models. The steps of kernel technique for setting the threshold value could be categorized as follows:

1. create the kernel density estimator based on gaussian function.
2. Generating points for possible threshold values in the range between the minimum and maximum of reconstruction errors in specified model.
3. Calculate the density of possible threshold values mentioned in the second step.
4. Estimate the optimal density value by considering the optimized value as 3.5 % of the maximum generated value for density [40]
5. Obtain the corresponding optimal threshold value regarding the optimal density from previous steps.

3.2 Kalman Filter

The Kalman filter is a recursive algorithm used to estimate the state of a dynamic system from a series of noisy measurements. It combines incoming measurements with predictions from the system's mathematical model to obtain an optimal estimate of the system state. The Kalman filter is widely used in various fields, including control systems, signal processing, robotics, and navigation.

In this research the Kalman filter will be used as the second method to detect the optimal threshold for identifying the anomalies from normal tests. The explanation, concept and key formulas of the Kalman filter will be summarized in this section.

There are two main steps which can be considered for Kalman Filter: the prediction step and the update step.

Prediction Step:

a. State Prediction:

Based on the system's mathematical model, the filter predicts the current state of the system at the next time step. The state prediction is obtained by multiplying the previous state estimate by the state transition matrix.

b. Error Covariance Prediction:

The error covariance matrix predicts the uncertainty associated with the state estimate. It quantifies the expected error in the state prediction and is computed by propagating the error covariance matrix of the previous step through the state transition and process noise matrices.

Update Step:

a. Kalman Gain Calculation:

The Kalman gain determines the weight given to the measurement updates. It is computed by multiplying the error covariance prediction by the observation matrix and the inverse of the sum of the predicted error covariance and measurement noise covariance matrices.

b. State Update:

The state estimate is updated by combining the predicted state with the measurement information. The updated state estimate is obtained by adding the Kalman gain multiplied by the difference between the actual measurement and the predicted measurement.

c. Error Covariance Update:

   The error covariance matrix is updated to reflect the improved estimate obtained from the measurement update. It is computed by multiplying the complement of the Kalman gain by the predicted error covariance.

Formulas of the Kalman Filter:

The Kalman filter involves several matrices and vectors that are updated recursively. The key formulas of the Kalman filter are as follows:

State Prediction:

$$\hat{X}_n = F\,\hat{X}_{n-1} + Bu_n \tag{14}$$

Error Covariance Prediction:

$$P_n = FPF^T + Q \tag{15}$$

Kalman Gain Calculation:

$$K_n = PH^T(HPH^T + R)^{-1} \tag{16}$$

State Update:

$$\hat{X}_n = \hat{X}_n + k_n(z_n - H\,\hat{X}_n) \tag{17}$$

Error Covariance Update:

$$P_n = \left(1 - KH\right)P_n \tag{18}$$

In the above formulas:

$\hat{x}_n$ represents the estimated state at time step n.

F is the state transition matrix that relates the current state to the next state.

B is the control input matrix.

u_n is the control input at time step n.

P_n is the error covariance matrix at time step n.

Q is the process noise covariance matrix, representing the uncertainty in the system dynamics.

K_n is the Kalman gain at time step n.

z_n represents the measurement at time step n.

H is the observation matrix that maps the state space to the measurement space.

R is the measurement noise covariance matrix, representing the uncertainty in the measurements.

I is the identity matrix.

By iteratively applying the prediction and update steps, the Kalman filter provides an optimal estimation of the system state, considering both the system dynamics and the measurement information. It provides a powerful tool for state estimation and tracking in various applications.

In this research the optimal threshold value determination has been done by using the Kalman filter technique as a second threshold setting method.

In the following various steps to obtain the optimal value for threshold to detect anomalies are categorized:

- Define the parameters for the Kalman filter, including the initial state mean and variance, the transition and measurement matrices, and the covariances.

- Initialized the Kalman filter with the initial state mean and variance then iterate through the data by performing the prediction step to estimate the next state, and then the update step to update the state estimate based on the measurement. The residual obtained from the update step is used to estimate the threshold value for anomaly detection.

- As a final step, if a measurement (output reconstruction error or anomaly score for each test) exceeds the threshold, it is considered as anomaly otherwise it can be regarded as normal test.

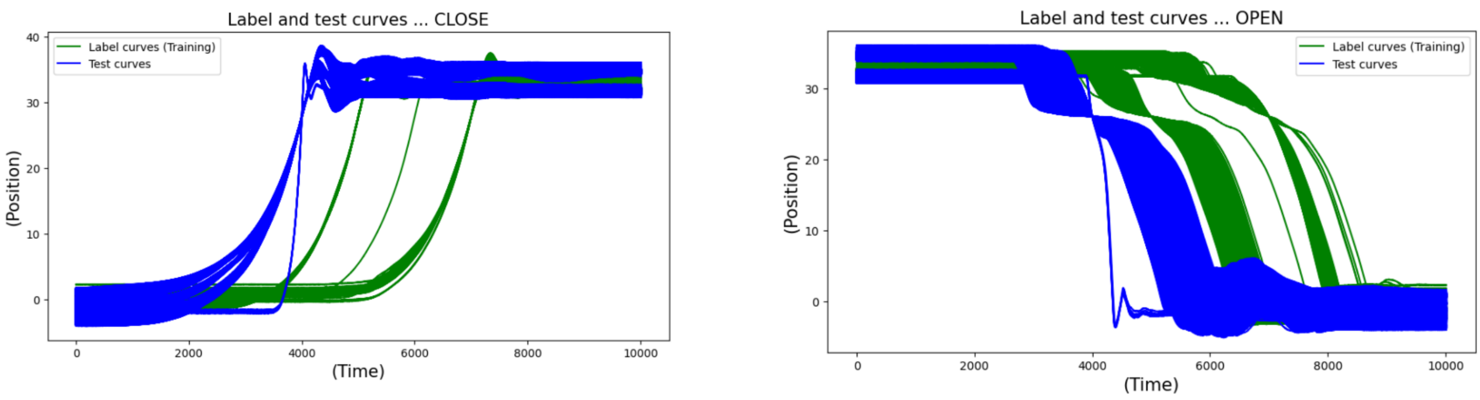
It should be noted that in this research the performance of the previous method (Kernel function) to detect the threshold is more acceptable than using the Kalman filter concept.

4. Results and Comparison

In this thesis the anomaly detection of angular travel curves of two ABB low voltage circuit breakers (with a model known as XT7) during the opening and closing endurance tests has been considered. For this motivation, three different models (Auto-Encoders, LSTM and Isolation Forest) based on two various techniques for setting and finding the optimal and dynamic threshold value instead of setting predefined constant value have been implemented. In this section the results and relevant figures including the comparison and efficiency of each model and threshold method are discussed.

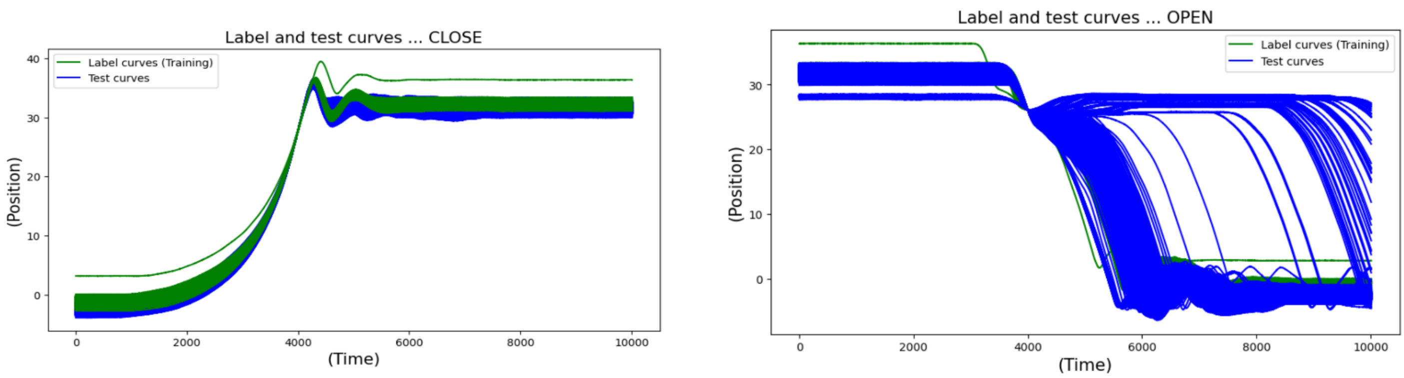
4.1 Training and unknown angular travel curves

In figure 4.1.1 and 4.1.2 the whole angular curves of two devices for both opening and closing tests are presented. In addition, in this analysis the first 1000 tests are labeled as training tests used for training the model represented by green curves in these figures. And other tests with a test number over than 1000 which they are unclear if normal or anomaly are shown by blue curves as follows:



(a) (b)

Fig 4.1.1. Angular curves (first breaker) a: closing b: opening.



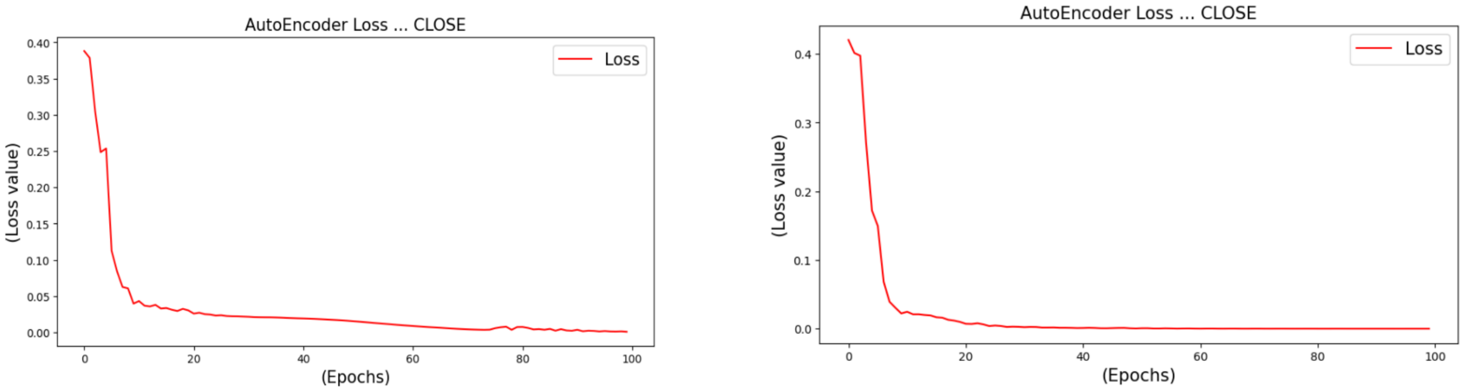
(a) (b)

Fig 4.1.2. Angular curves (second breaker) a: closing b: opening.

In actual situations a permanently installed travel transducer allows one to record the position of the contacts which is a function of time for every circuit breaker closing and opening operations. Additionally, in each operation every deviation from the normal value could have inappropriate effect on the behavior of circuit breakers' function.

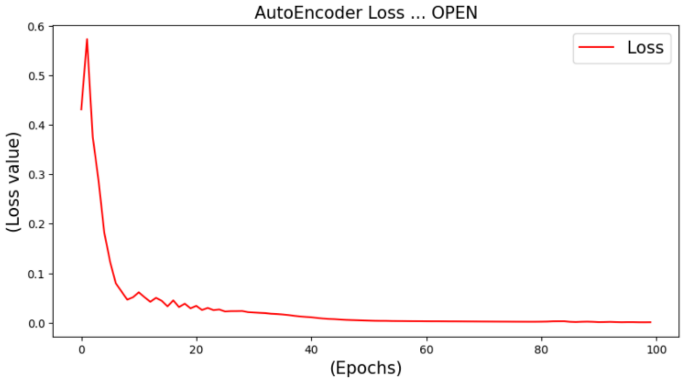
4.2 Auto-Encoder Model Loss

The loss for Auto-Encoder and LSTM models is defined as the difference between the input and output. Furthermore, during the training process the model or algorithm learns how to minimize the loss and reach better performance. In figure 4.2.1 and 4.2.2 the loss regarded to Auto-Encoder model for closing and opening of the first and second tested circuit breakers by considering the 100 epochs (iterations) for training the algorithm is represented respectively:

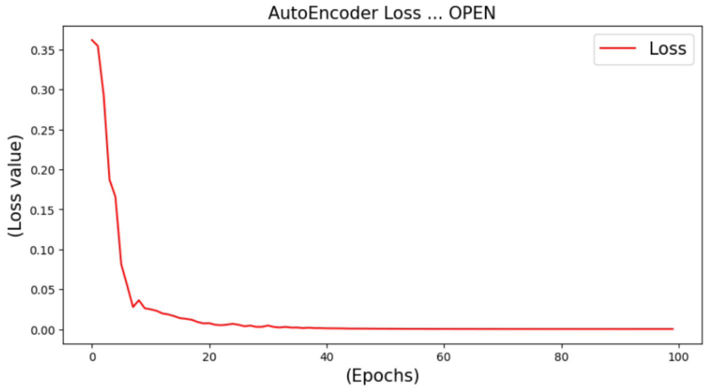


(a) (b)

Fig 4.2.1. closing loss (Auto-Encoder model) a: first breaker b: second breaker



(a)



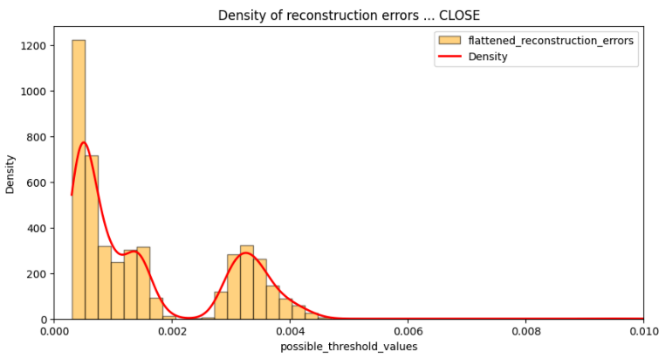
(b)

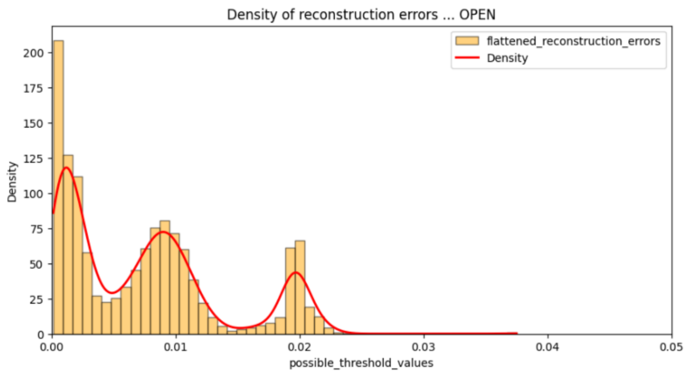
Fig 4.2.2. opening loss (Auto-Encoder model) a: first breaker b: second breaker

During the training process, the autoencoder's parameters are adjusted iteratively to reduce the loss and improve the quality of the reconstructed output. According to these figures the loss in Auto-Encoder model perfectly approaches highly to zero which demonstrates that our model maps the input information to its output with a high accurate and appropriate learning by selected training dataset based on using MSE as a common loss function which explained before.

4.3 Threshold setting method (Kernel Function)

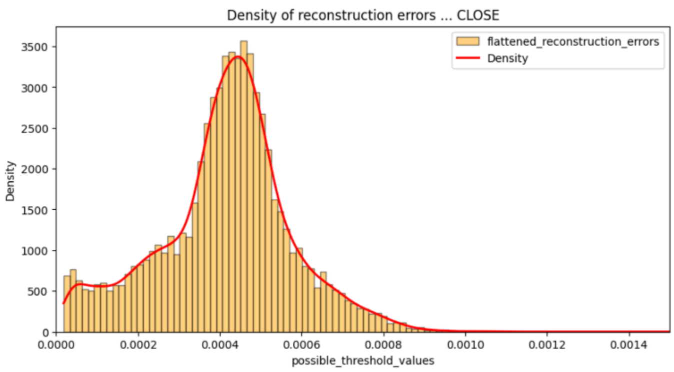
In this section the histogram model of calculated reconstruction errors for Auto-Encoder model has been represented. Moreover, in these figures the red curve shows the value of density for each case. After calculation of reconstruction errors, the different gaussian distribution functions are considered for each area of histogram model and by combination of them it will cause to create a kernel model to estimate the final and optimal threshold value.

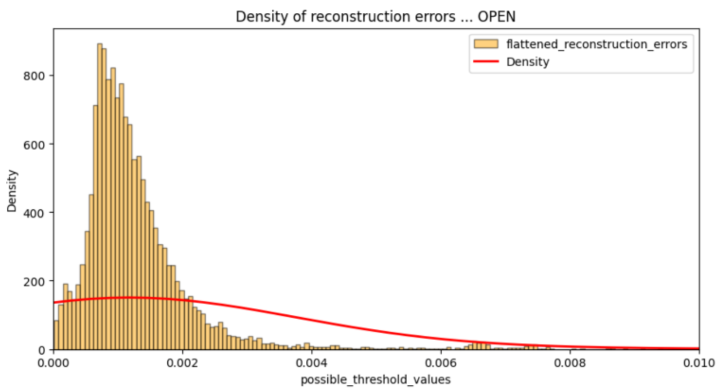




(a) (b)

Fig 4.3.1. Histogram and Density of errors (first breaker) a: closing b: opening.





(a) (b)

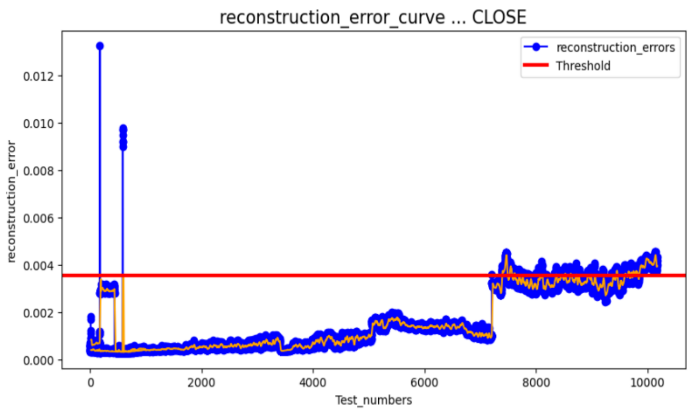
Fig 4.3.2. Histogram and Density of errors (second breaker) a: closing b: opening.

According to the above figures the optimal threshold value is considered as corresponding value of 3.5% of maximum density on the x axis.

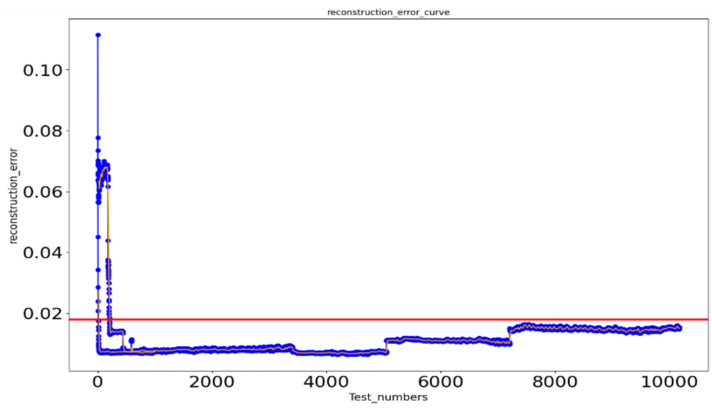
As a result, by using this approach, the dynamic model of system would have ability to estimate its own threshold value based on calculated instant reconstruction errors of model. Also, it is obvious that by setting the constant threshold value for each test an anomaly detection process cannot be done in an optimal way, however, by using this technique we are able to set the optimal threshold value by considering the actual condition of system and its relating introduced state variables which causes to obtain the anomaly scores, or reconstruction errors based on predefined weights in the layers.

4.4 Reconstruction Errors vs Threshold value

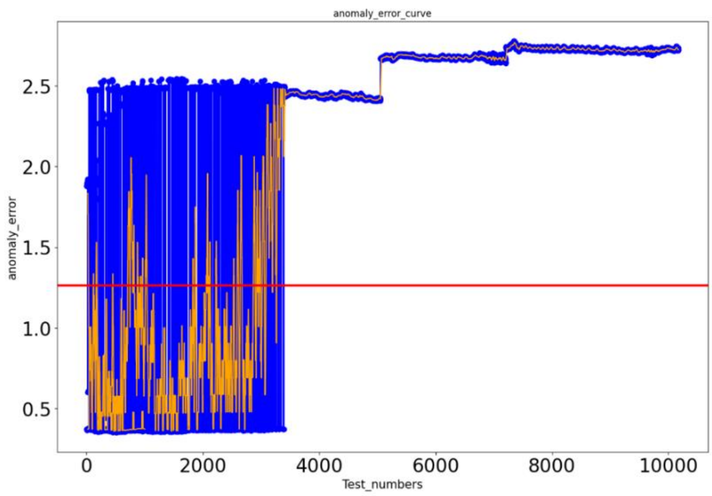
The calculated reconstruction errors of each test to evaluate the accuracy of a model in comparison with threshold value for both breakers in closing and opening positions could be represented as following figures. Based on the obtained figures by python, the results and performance of Auto-Encoder model are acceptable compared to other models such as LSTM and Isolation Forest.



(a)

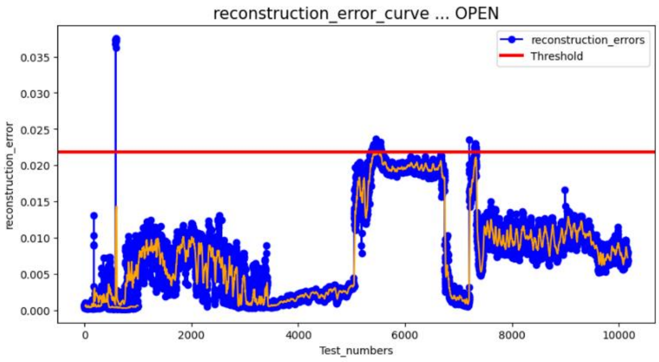


(b)

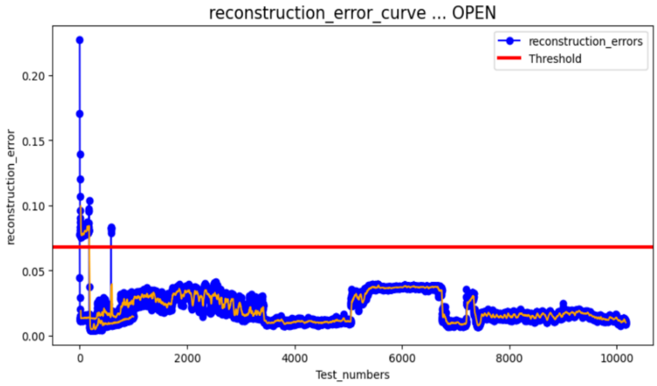


(c)

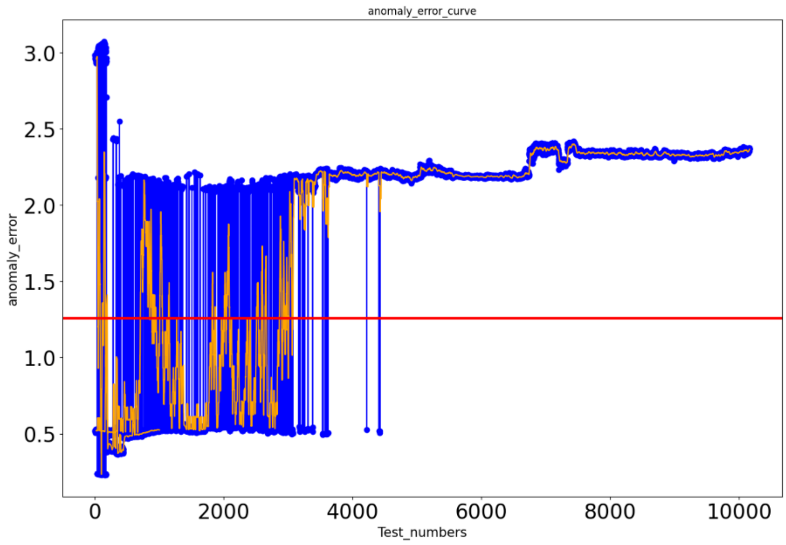
Fig 4.4.1. Reconstruction errors vs threshold (First breaker - closing):
a: AutoEncoder b: LSTM c: Isolation Forest



(a)



(b)

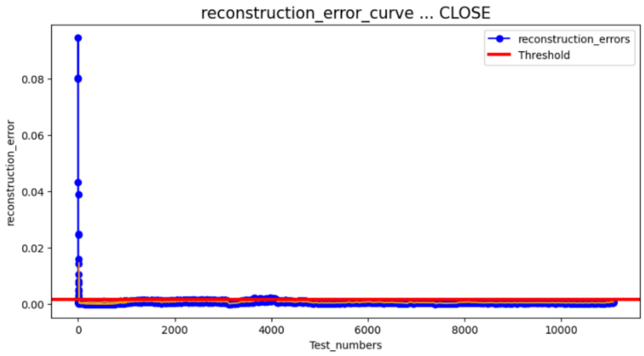


(c)

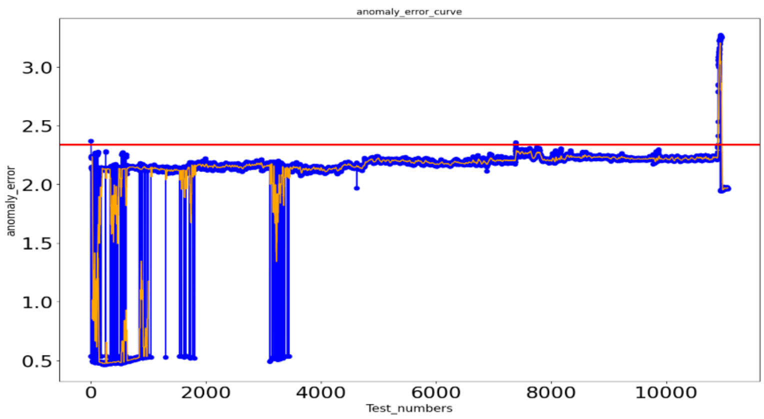
Fig 4.4.2. Reconstruction errors vs threshold (First breaker - opening):
a: AutoEncoder b: LSTM c: Isolation Forest

In the figures 4.4.3 and 4.4.4 the reconstruction errors compared to threshold value are represented for endurance tests of second circuit breaker:

(a)

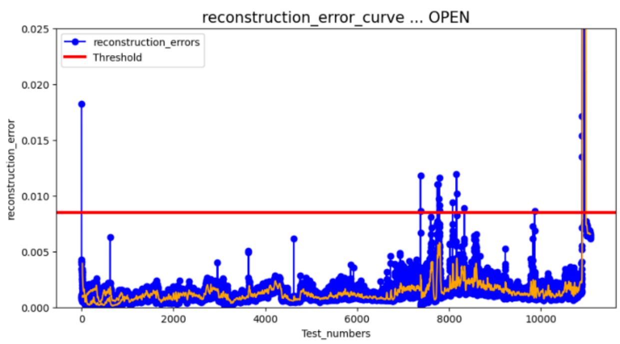


(b)

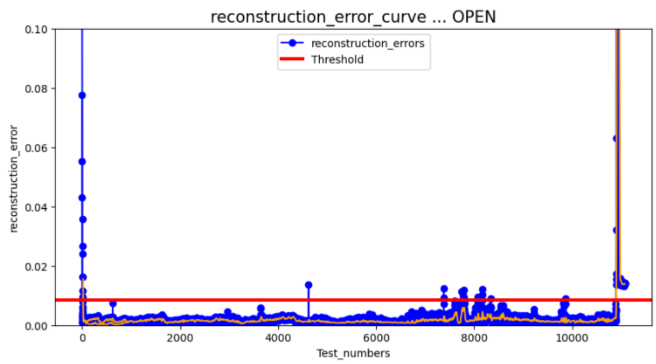


(c)

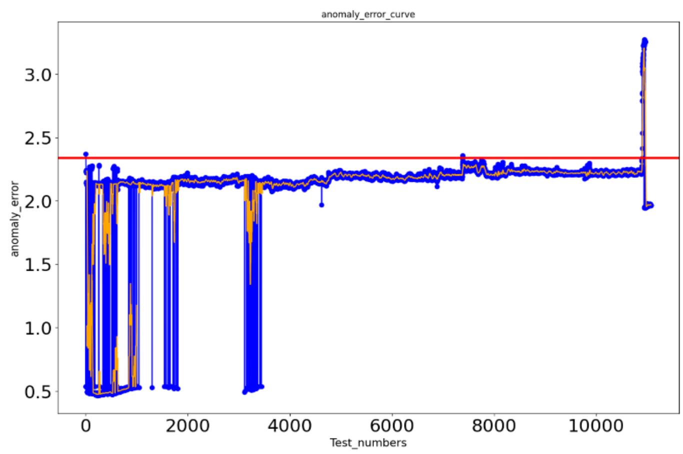
Fig 4.4.3. Reconstruction errors with threshold (Second breaker - closing):
a: AutoEncoder b: LSTM c: Isolation Forest



(a)



(b)



(c)

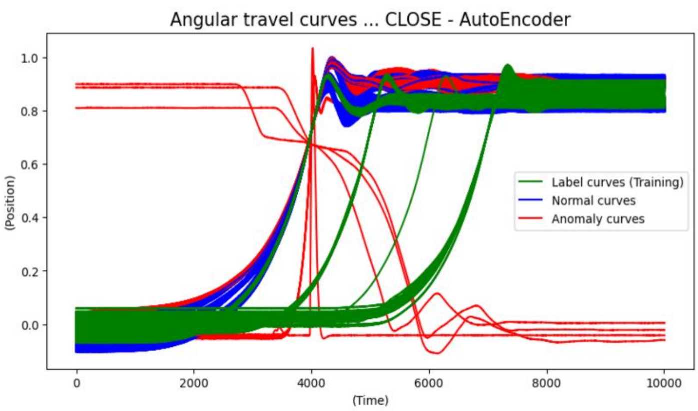
Fig 4.4.4. Reconstruction errors with threshold (Second breaker - opening):
a: AutoEncoder b: LSTM c: Isolation Forest

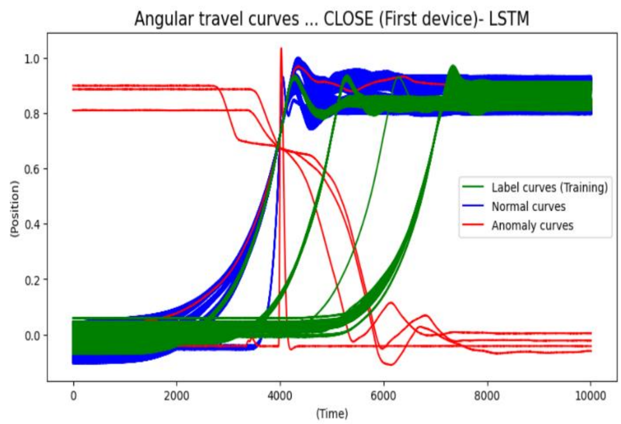
According to these figures and previous analysis, the obtained threshold value by using the kernel technique does not have an acceptable and optimal value in LSTM and Isolation-Forest methods, however, the only acceptable result in this analysis is based on Auto-Encoder model.

It is good to remember that the effectiveness of anomaly detection relies on the quality of the model, the appropriateness of the threshold value, and the representative nature of the training data which has been obtained by using the Auto-Encoder model and kernel distribution for determination of threshold value.

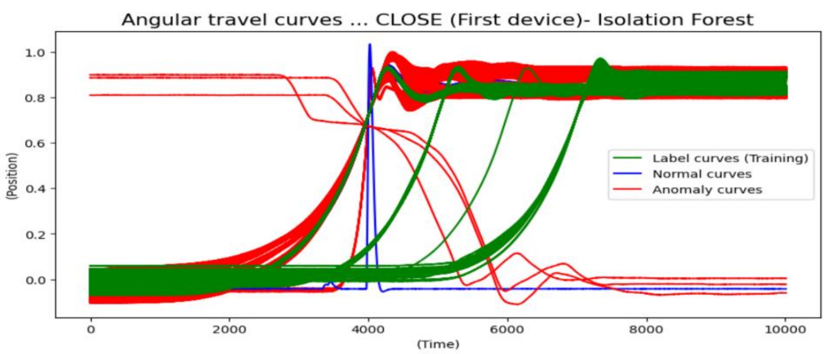
4.5 Classification of Angular travel curves by proposed algorithms

In figures 4.5.1-4 the condition of tests detected by algorithms are presented. In the following figures the green curves are the first 1000 tests used for training the model. The blue curves are the normal tests detected by algorithm in which their reconstruction errors are lower than the optimal threshold value by kernel function. Finally, the red ones are anomaly curves which have the reconstruction error over than the threshold value.



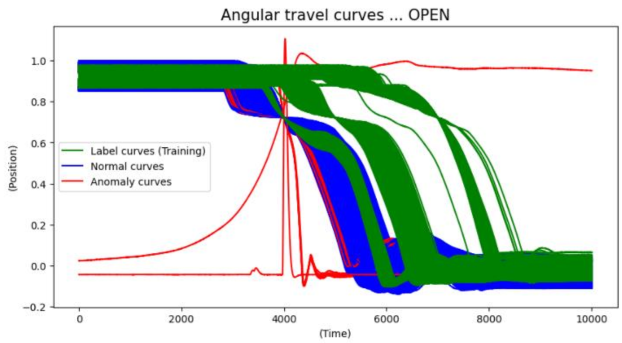


(a) (b)

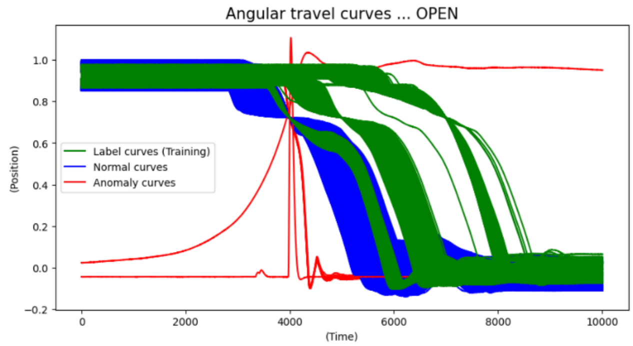


(c)

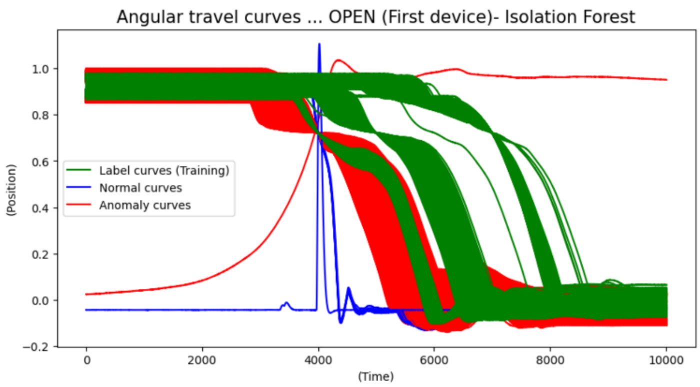
Fig 4.5.1. Angular travel curves (First breaker - closing):
a:AutoEncoder b: LSTM c: Isolation Forest



(a)



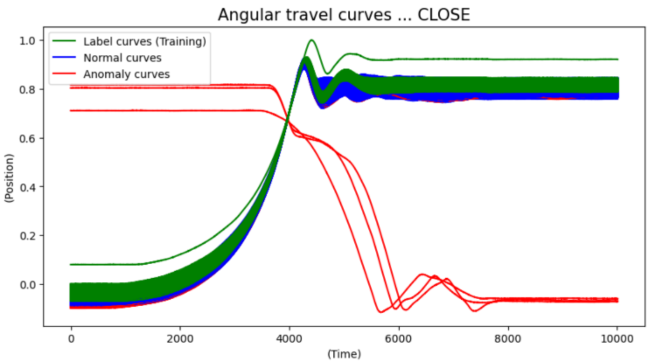
(b)

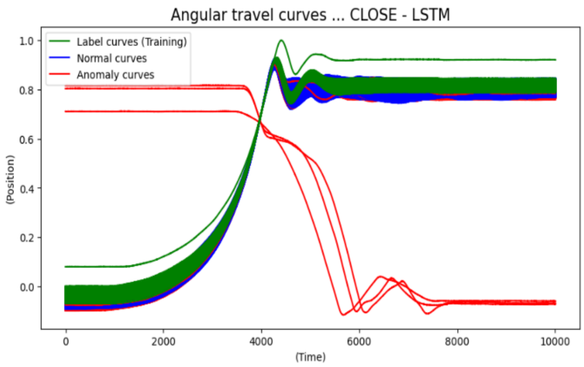


(c)

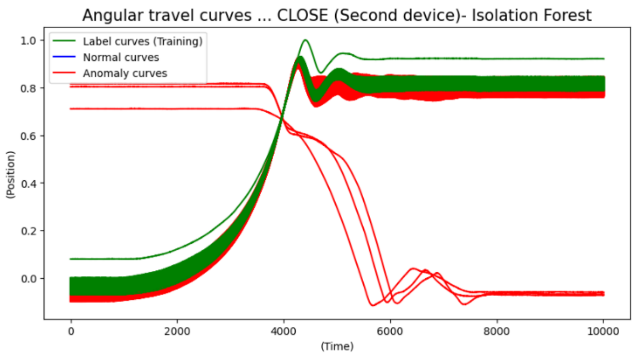
Fig 4.5.2. Angular travel curves (First breaker - opening):
a:Auto-Encoder b: LSTM c: Isolation Forest

Figures 4.5.3 and 4.5.4 show the detected normal and anomaly curves for the second breaker endurance tests. It must be noted that in both cases the kernel function technique considered as an acceptable threshold setting value method.



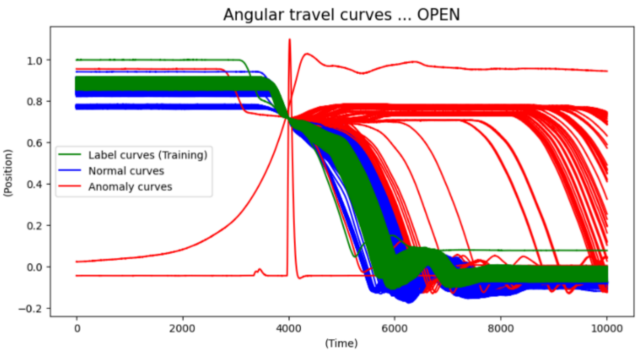


(a) (b)

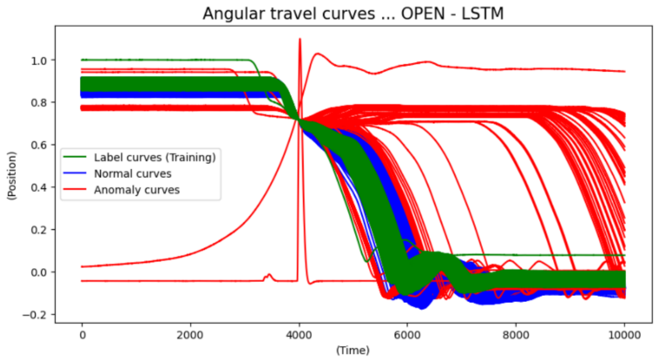


(c)

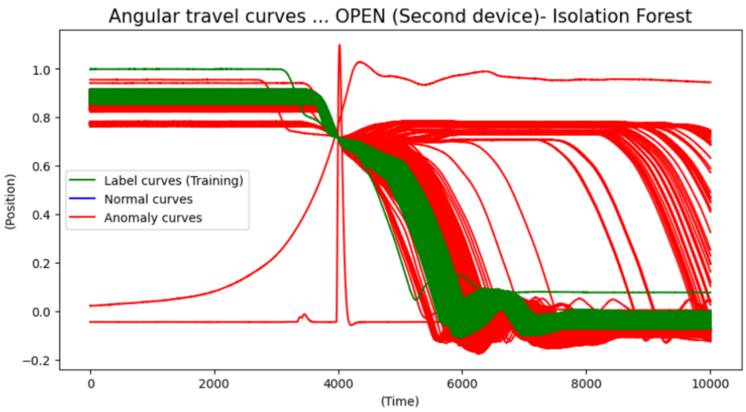
Fig 4.5.3. Angular travel curves (Second breaker - closing):
a:AutoEncoder b: LSTM c: Isolation Forest



(a)



(b)

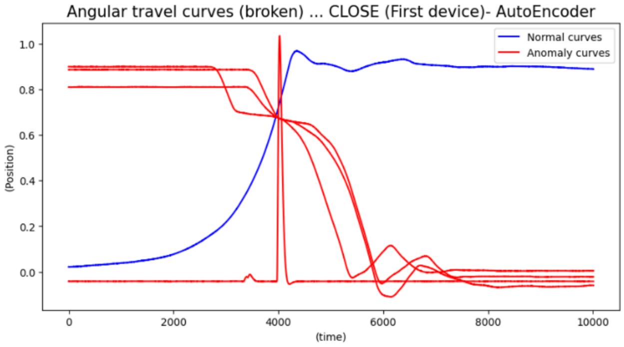


(c)

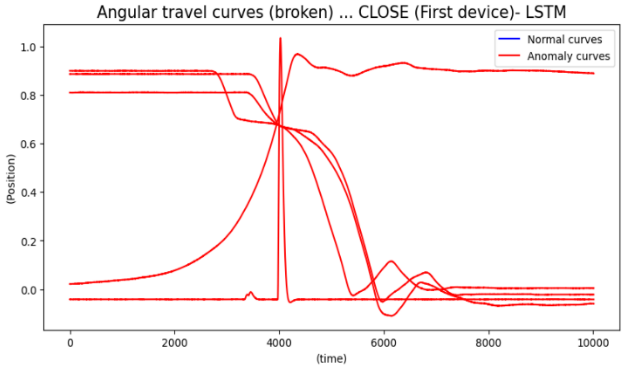
Fig 4.5.4. Angular travel curves (Second breaker - opening):
a:Auto-Encoder b: LSTM c: Isolation Forest

4.6 Problematic curves during endurance tests

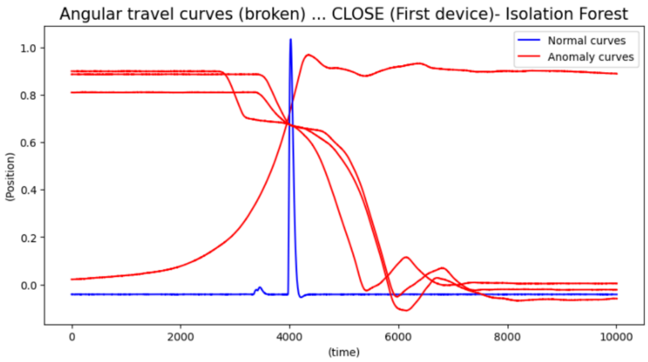
During the testing process, some tests labeled as possible problematic tests by ABB experts which are important to detect their condition if they are anomaly or not. As a result, by using three different models and considering the kernel method as threshold setting technique. Figures 4.6.1-4 represent these curves by proposed algorithms for both circuit breakers.



(a)

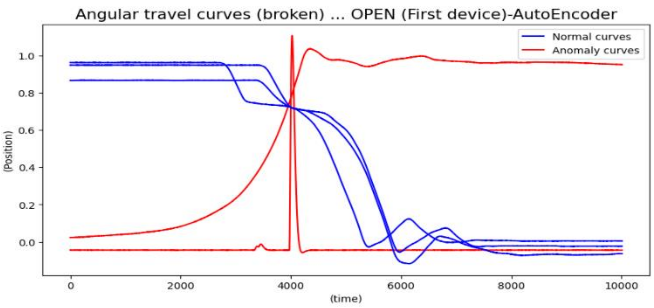


(b)

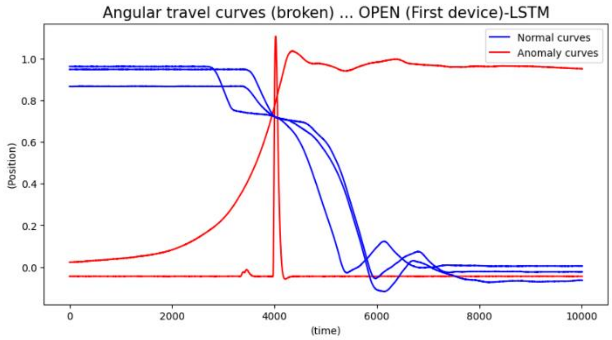


(c)

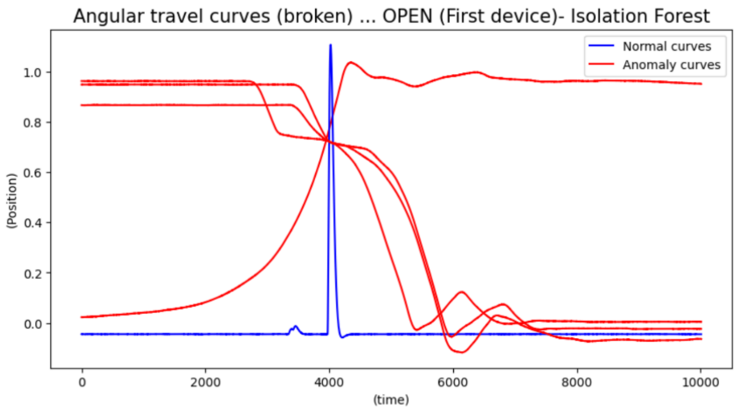
Fig 4.6.1. Problematic curves (First breaker - closing):
a:AutoEncoder b: LSTM c: Isolation Forest



(a)

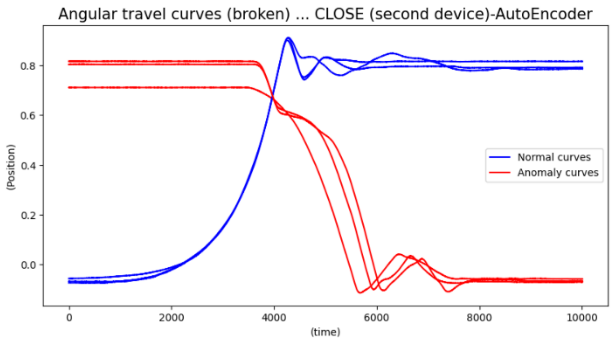


(b)

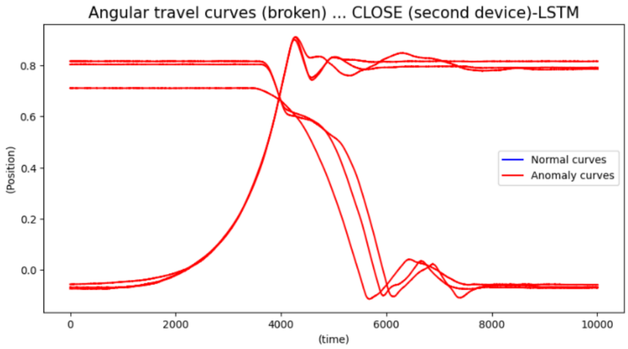


(c)

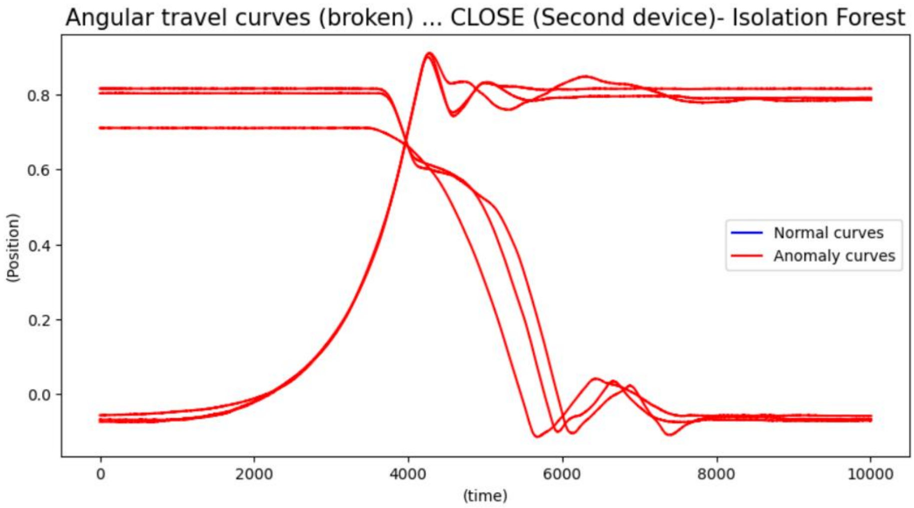
Fig 4.6.2. Problematic curves (First breaker - opening):
a:AutoEncoder b: LSTM c: Isolation Forest



(a)

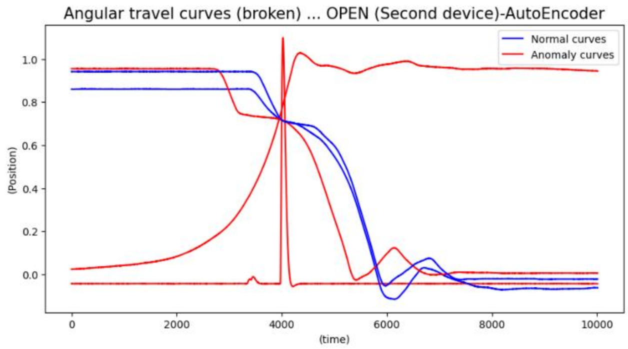


(b)

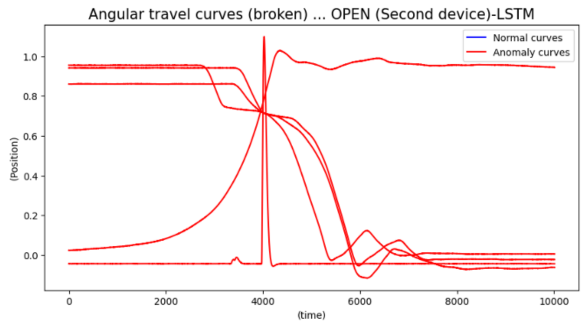


(c)

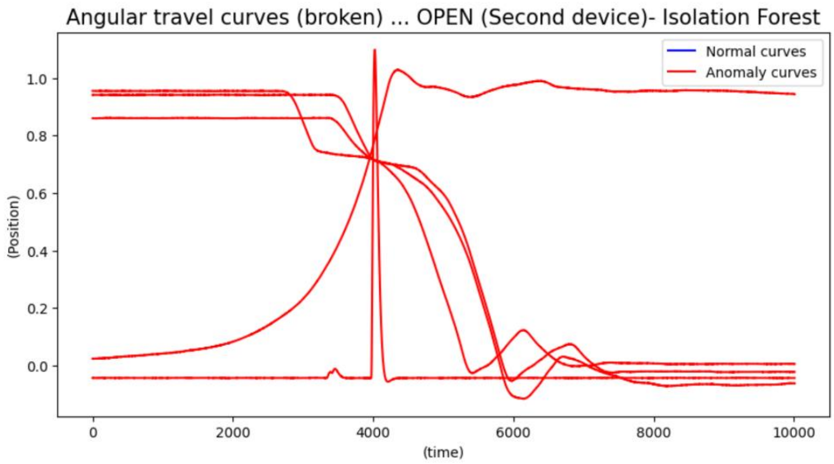
Fig 4.6.3. Problematic curves (Second breaker - closing):
a:AutoEncoder b: LSTM c: Isolation Forest



(a)



(b)



(c)

Fig 4.6.4. Problematic curves (Second breaker - opening):
a:AutoEncoder b: LSTM c: Isolation Forest

In the first breaker all the curves in closing and opening tests detected as normal signals by Auto-Encoder model because they do not have much deviation from the dc (average) value, however, in the opening of second circuit breaker two of three curves detected as normal and one detected as anomaly curve.

According to these results for both closing and opening cases the Auto-Encoder model based on kernel function for detecting the threshold value works fine and has a good performance compared to other methods.

4.7 Total percentage of anomaly and normal curves detected by models

Tables 4.7.1 and 4.7.2 represent the total percentage of normal and anomaly tests based on three different models and two threshold value setting techniques in the first circuit breaker. According to these tables both in closing and opening test of first circuit breaker over 90% of the tests detected as normal test by using the Auto-Encoder model and kernel function as a threshold technique.

Whenever, there is the same result for using the Kalman filter concept for closing but when we come to opening results only around 45% of the tests detected as normal test by using the Kalman filter in this case which could not be an acceptable answer for opening endurance tests.

4.7.1. Percentage of anomaly and normal tests (First breaker - closing)

Closing	Normal (%)		Anomaly (%)	
Threshold Technique	Kernel function	Kalman Filter	Kernel function	Kalman Filter
AutoEncoders	91.93	99.15	8.064	0.841
LSTM	99.92	94.74	0.071	5.25
Isolation Forest	31.08	94.06	68.91	5.93

4.7.2. Percentage of anomaly and normal tests (First breaker - opening)

Opening	Normal (%)		Anomaly (%)	
Threshold Technique	Kernel function	Kalman Filter	Kernel function	Kalman Filter
AutoEncoders	99.36	45.47	0.635	54.52
LSTM	99.92	13.58	0.071	86.41
Isolation Forest	30.26	97.68	69.73	2.31

Tables 4.7.3 and 4.7.4 also represent similar results for the second circuit breaker under test. It is obvious that in this test also the Auto-Encoder model based on kernel function for setting the threshold value works fine and better than Kalman filter which contains heavy and complex mathematics behind it.

4.7.3. Percentage of anomaly and normal tests (Second breaker - closing)

Closing	Normal (%)		Anomaly (%)	
Threshold Technique	Kernel function	Kalman Filter	Kernel function	Kalman Filter
AutoEncoders	99.37	99.85	0.621	0.140
LSTM	99.43	97.71	0.563	2.28
Isolation Forest	8.28	39.21	91.71	60.78

4.7.4. Percentage of anomaly and normal tests (Second breaker - opening)

Opening	Normal (%)		Anomaly (%)	
Threshold Technique	Kernel function	Kalman Filter	Kernel function	Kalman Filter
AutoEncoders	99.40	99.51	0.59	0.488
LSTM	98.36	99.97	1.639	0.0248
Isolation Forest	8.28	99.50	91.71	0.496

5. Conclusion and future development

In this thesis the anomaly detection of angular travel curves of ABB low voltage circuit breakers (XT7) has been considered and results relating to the three different models including Auto-Encoders, LSTM and Isolation Forest with two various techniques to obtain the optimal threshold value were represented. According to the analysis, the Auto-Encoder model based on introduction of kernel distribution function to define the threshold value has a perfect and acceptable result and performance compared to other approaches. Based on previous sections, Kalman filter not only has a large amount of complexity in setting the matrices but also sometimes it generates the threshold value with a high deviation from possible actual threshold values resembling to our reconstruction errors or anomaly scores. In addition, other methods regardless of Auto-Encoder, also have disadvantage to detect the normal signals which are thought of as problematic curves during the test process.

Moreover, it must be noted that due to new circuit breakers used for mechanical endurance tests, the acceptable concept is that most tests regardless of final tests (nearly over 8000) have been expected to be determined as normal tests. However, in LSTM or isolation forest models and Kalman filter technique some abnormal values for total percentage of normal and anomaly curves have appeared in the results.

In previous research and analysis, the threshold value was set a specific and constant value during the whole analysis, however in this project the dynamic threshold value has been considered which is highly practical and it can be set automatically based on the condition of our tests, defined models and reconstruction or anomaly scores calculated after using the model to minimize the final loss.

Auto-Encoders have shown great potential in anomaly detection tasks, and there are several possible future directions for enhancing their capabilities. One of the potential areas of future work for anomaly detection based on Auto-Encoders for low voltage circuit breakers analysis is advanced Autoencoder Architectures which are novel architectures that improve the representation power of Autoencoders. This includes variations such as convolutional, recurrent, or attention-based Auto-Encoders. These architectures can capture spatial or temporal dependencies in data, making them more suitable for anomaly detection in our data set.

In addition, semi-supervised pretraining can be employed to enhance the performance of Autoencoders for anomaly detection. Pretraining the Autoencoder on a large dataset with normal instances can help in learning more robust and generalizable representations, which can then be fine-tuned for anomaly detection on labeled datasets.

Also, Transfer learning techniques can be employed to adapt Auto-Encoders trained from one domain to another related domain with limited labeled anomaly data. By leveraging the learned representations, the Autoencoder can generalize to new domains with few labeled instances, making it more effective in real-world scenarios where labeled anomaly data is scarce.

Ensemble and Adversarial Approaches could be used in multiple Autoencoders with different architectures or training strategies which can enhance anomaly detection performance. Adversarial training methods, such as Generative Adversarial Networks (GANs), can be used to generate challenging anomaly samples, forcing the Autoencoder to learn more robust representations.

Developing Autoencoder-based anomaly detection techniques that can operate in real-time or streaming environments is an important area of future work. This involves designing efficient architectures and algorithms that can process data in an online manner and quickly adapt to changing patterns.

Evaluation Metrics and Benchmark Datasets can help to perform standardized evaluation metrics and benchmark datasets specifically designed for Autoencoder-based anomaly detection. Developing robust evaluation metrics and diverse datasets can facilitate fair comparisons between different models and promote advancements in the field.

Finally other characteristics and parameters of circuit breakers could be considered and analyzed in future works. For instance, in addition of angular curves maybe there would be different opportunities to consider the vibration, absorption current, temperature and other parameters and data set of circuit breakers.

.

Bibliography

[1] S. V. Forero, M. Chen, A. Goh, and K.P. Sze, “A comparative analysis of covariance matrix estimation in anomaly detection,”6th Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS), 24-27 June 2014

[2] Y. Zhang, Z.Y. Dong, W. Kong, and K. Meng, “A Composite Anomaly Detection System for Data-Driven Power Plant Condition Monitoring,” IEEE Trans. Industrial Informatics, vol. 16, no. 7, pp. 4390–4402, July. 2020

[3] K. R. Mestav, X. Wang, L. Tong, “A Deep Learning Approach to Anomaly Sequence Detection for High-Resolution Monitoring of Power Systems,” IEEE Trans. Power Systems, vol. 38, no. 1, pp. 4–13, Jan. 2023

[4] R. Moghaddass, J. Wang, “A Hierarchical Framework for Smart Grid Anomaly Detection Using Large-Scale Smart Meter Data,” IEEE Trans. Smart Grid, vol. 9, no. 6, pp. 5820–5830, Nov. 2018

[5] Z. Ling, R. Qiu, X. He, and L. Chu, “A New Approach of Exploiting Self-Adjoint Matrix Polynomials of Large Random Matrices for Anomaly Detection and Fault Location,” IEEE Trans. Big Data, vol. 7, no. 3, pp. 548–558, Sep. 2021

[6] X. Jiang, B. Stephen, S. McArthur, “A Sequential Bayesian Approach to Online Power Quality Anomaly Segmentation,” IEEE Trans. Industrial Informatics, vol. 17, no. 4, pp. 2675–2685, April. 2021

[7] A. Ahmed, K. S. Sajan, A. Srivastava, and Y. Wu, “Anomaly Detection, Localization and Classification Using Drifting Synchrophasor Data Streams,” IEEE Trans. Smart Grid, vol. 12, no. 4, pp. 3570–3580, July. 2021

[8] L. Jie, L. Haoxiang, L. XiangXiang, “Anomaly detection method of power dispatching data based on cloud computing platform,” International Conference on Big Data & Artificial Intelligence & Software Engineering. Sep. 2020

[9] Y. Yan, G. Sheng, R.C. Qiu, and X. Jiang, “Big Data Modeling and Analysis for Power Transmission Equipment: A Novel Random Matrix Theoretical Approach,” IEEE Access. Smart Grid, vol. 6, pp. 7148–7156, Dec. 2018

[10] J. Dai, H. Song, G. Sheng, and X. Jiang, “Cleaning Method for Status Monitoring Data of Power Equipment Based on Stacked Denoising Autoencoders,” IEEE Access, vol. 5, pp. 22863–22870, Aug. 2017

[11] J. Yu, H. Cheng, J. Zhang, Q. Li, S. Wu, W. Zhong, J. Ye, W. Song, P. Ma "CONGO^2: Scalable Online Anomaly Detection and Localization in Power Electronics Networks," IEEE internet of things journal, vol. 9, no. 15, pp. 13862–13875, Aug. 2022

[12] A. Ahmed, V. Krishnan, S.A. Foroutan, Md. Touhiduzzaman, C.Rublein, A. Srivastava, Y. Wu, A. Han, S. Suresh "Cyber Physical Security Analytics for Anomalies in Transmission Protection Systems," IEEE Trans. Industry Applications, vol. 55, no. 6, pp. 6313–6323, Nov. 2019

[13] S. Li, A. Pandey, B. Hooi, and C. Faloutsos, L. Pileggi "Dynamic Graph-Based Anomaly Detection in the Electrical Grid," IEEE Trans. Power Systems, vol. 37, no. 5, pp. 3408–3422, Sep. 2022

[14] M. Zhou, Y. Wang, A.K. Srivastava, Y. Wu, P. Banerjee "Ensemble-Based Algorithm for Synchrophasor Data Anomaly Detection," IEEE Trans. Smart Grid, vol. 10, no. 3, pp. 2979–2988, May. 2019

[15] Y. Chen, S. Huang, F. Liu, Z. Wang, X. Sun "Evaluation of Reinforcement Learning-Based False Data Injection Attack to Automatic Voltage Control," IEEE Trans. Smart Grid, vol. 10, no. 2, pp. 2158–2169, Mar. 2019

[16] Q. Yang, A. Gultekin, V. Seferian, and K. Pattipati, A.M. Bazzi, F. Palmieri, R. Rajamani, S.N. Joshi, M. Farooq, H. Ukegawa "Incipient Residual-Based Anomaly Detection in Power Electronic Devices," IEEE Trans. Power Electronics, vol. 37, no. 6, pp. 7315–7332, Jun. 2022

[17] M. Ganjkhani, M. Gilanifar, J. Giraldo, and M. Parvania, "Integrated Cyber and Physical Anomaly Location and Classification in Power Distribution Systems," IEEE Trans. Industrial Informatics, vol. 17, no. 10, pp. 7040–7049, Oct. 2021

[18] S. Liu, C.H. Singer, R.A. Dougal, "Power Anomaly Effects and Costs in Low-Voltage Mobile Power Systems," IEEE Trans. Aerospace and electronic systems, vol. 42, no. 2, pp. 612–624, April. 2006

[19] Z. Ouyang, X. Sun, J. Chen, D. Yue, T. Zhang "Multi-View Stacking Ensemble for Power Consumption Anomaly Detection in the Context of Industrial Internet of Things," IEEE Access, vol. 6, pp. 9623–9631, Feb. 2018

[20] A. Ashok, M. Govindarasu, V. Ajjarapu, "Online Detection of Stealthy False Data Injection

Attacks in Power System State Estimation," IEEE Trans. Smart Grid, vol. 9, no. 3, pp. 1636–1646, May. 2018

[21] M. N. Kurt, O. Ogundijo, C. Li, and X. Wang, “Online cyber-attack detection in smart grid: A reinforcement learning approach,” IEEE Trans. Smart Grid, vol. 10, no. 5, pp. 5174–5185, Sep. 2019

[22] K. R. Mestav and L. Tong, “Universal data anomaly detection via in- verse generative adversary network,” IEEE Signal Process. Lett., vol. 27, pp. 511–515, 2020.

[23] H. Salehfar and R. Zhao, “A neural network preestimation filter for bad- data detection and identification in power system state estimation,” Electric Power Syst. Res., vol. 34, no. 2, pp. 127–134, 1995.

[24] M. Zhou, Y. Wang, A. K. Srivastava, Y. Wu, and P. Banerjee, “Ensemble-based algorithm for synchrophasor data anomaly detection,” IEEE Trans. Smart Grid, vol. 10, no. 3, pp. 2979–2988, May 2019.

[25] V. M. Catterson, S. D. J. McArthur, and G. Moss, “Online conditional anomaly detection in multivariate data for transformer monitoring,” IEEE Trans. Power Del., vol. 25, no. 4, pp. 2556–2564, Oct. 2010.

[26] A. J. Brown, V. M. Catterson, M. Fox, D. Long, and S. D. J. McArthur, “Learning models of plant behavior for anomaly detection and condition monitoring,” in Proc. 2007 Int. Conf. Intell. Syst. Appl. Power Syst, pp. 1–6.

[27] T. Ince, S. Kiranyaz, L. Eren, M. Askar, and M. Gabbouj, “Real-time motor fault detection by 1-D convolutional neural networks,” IEEE Trans. Ind. Electron., vol. 63, no. 11, pp. 7067–7075, Nov. 2016.

[28] H. Quan, D. Srinivasan, and A. Khosravi, “Short-term load and wind power forecasting using neural network-based prediction intervals,” IEEE Trans. Neural Netw. Learning. Syst., vol. 25, no. 2, pp. 303–315, Feb. 2014.

[29] M. Raciti and S. Nadjm-Tehrani, “Embedded cyber-physical anomaly detection in smart meters,” in Critical Information Infrastructures Security (LNCS 7722). Berlin, Germany: Springer, pp. 34–45, 2013.

[30] N. Yu et al., “Big data analytics in power distribution systems,” in Proc. IEEE Power Energy Soc. Innov. Smart Grid Technol. Conf., Washington, DC, USA, 2015, pp. 1–5.

[31] K. Zhou, C. Fu, and S. Yang, “Big data driven smart energy management: From big data to big insights,” Renew. Sustain. Energy Rev., vol. 56, pp. 215–225, Apr. 2016.

[32] Loubaton, Philippe, and Vallet, “Almost sure localization of the eigenvalues in a Gaussian information plus noise model. Applications to the spiked models,” Electron. J. Probability, vol. 16, no. 24, pp. 1934–1959, 2011.

[33] A. Pawling, N. V. Chawla, and G. R. Madey, “Anomaly detection in a mobile communication network,” Comput. Math. Organization Theory, vol. 13, no. 4, pp. 407–422, 2007.

[34] L. Xie, Y. Chen, and P. Kumar, “Dimensionality reduction of synchrophasor data for early event detection: Linearized analysis,”IEEE Trans. Power Syst., vol. 29, no. 6, pp. 2784–2794, Nov. 2014.

[35] Y. Chen, L. Xie, and P. Kumar, “Dimensionality reduction and early event detection using online synchrophasor data,” in Proc. IEEE Power Energy Soc. General Meeting, Jul. 2013, pp. 1–5.

[36] S. Belinschi, T. Mai, and R. Speicher, “Analytic subordination theory of operator-valued free additive convolution and the solution of a general random matrix problem,” J. Fr Die Reine Und Angewandte Mathematic, vol. 2017, pp. 21–53, 2015.

[37] Y. Chen, L. Xie, and P. Kumar, “Dimensionality reduction and early event detection using online synchrophasor data,” in Proc. IEEE Power Energy Soc. General Meeting, Jul. 2013, pp. 1–5.

[38] A. Pawling, N. V. Chawla, and G. R. Madey, “Anomaly detection in a mobile communication network,” Comput. Math. Organization Theory, vol. 13, no. 4, pp. 407–422, 2007.

[39] R. Qiu and P. Antonik, Smart Grid Using Big Data Analytics–A Random Matrix Theory Approach. Hoboken, NJ, USA: Wiley, 2016, Art. no. 600.

[40] B. Li, Y. Jing, and W. Xu, “A generic waveform abnormality detection method for utility equipment condition monitoring,” IEEE Trans. Power Del., vol. 32, no. 1, pp. 162–171, Feb. 2017.

[41] K. P. Murphy, “Conjugate Bayesian analysis of the Gaussian distribution,”Def, vol. 1, no. 7, pp. 1–29, 2007.

[42] S. Ahmad, A. Lavin, S. Purdy, and Z. Agha, “Unsupervised real-time anomaly detection for streaming data,” Neurocomputing, vol. 262, pp. 134–147, 2017.

[43] M. Rafferty, P. Brogan, J. Hastings, D. Laverty, X. A. Liu, and R. Khan, “Local anomaly detection by application of regression analysis on PMU data,” in Proc. IEEE Power Energy Soc. Gen. Meeting (PESGM), Portland, OR, USA, 2018, pp. 5.

[44] J. Zhou, Y. Fu, Y. Wu, H. Xia, Y. Fang, and H. Lu, “Anomaly detection over concept drifting data streams,” J. Comput. Inf. Syst., vol. 5, no. 6, pp. 1697–1703, 2009.

[45] K. Yu, W. Shi, and N. Santoro, “Designing a streaming algorithm for outlier detection in data mining—An incrementa approach,” Sensors, vol. 20, no. 5, p. 1261, 2020.

[46] K. Yu, W. Shi, and N. Santoro, “Designing a streaming algorithm for outlier detection in data mining—An incrementa approach,” Sensors, vol. 20, no. 5, p. 1261, 2020.

[47] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, “A density-based algorithm for discovering clusters in large spatial databases with noise,” in Proc. 2nd Int. Conf. Knowl. Discov. Data Min. (KDD), vol. 96, 1996, pp. 226–231.

[48] K. Yu, W. Shi, and N. Santoro, “Designing a streaming algorithm for outlier detection in data mining—An incrementa approach,” Sensors, vol. 20, no. 5, p. 1261, 2020.

[49] Y. Dong and N. Japkowicz, “Threaded ensembles of autoencoders for stream learning,” Comput. Intell., vol. 34, pp. 261–281. Feb. 2018.

[50] R. Yadav, A. K. Pradhan, and I. Kamwa, “Real-time multiple event detection and classification in power system using signal energy transformations,” IEEE Trans. Ind. Informat., vol. 15, no. 3, pp. 1521–1531, Mar. 2019.

[51] F. Pedregosa et al., “Scikit-learn: Machine learning in Python,” J. Mach. Learn. Res., vol. 12, pp. 2825–2830, Nov. 2011.

List of Figures

List of Tables