ENHANCING POWER GRID RELIABILITY WITH MACHINE LEARNING ALGORITHM FOR FAULT DETECTION AND CLASSIFICATION

by

Md. Sadi Mobassir (190021117)
Safoat Saima Arpi (190021231)
Farhan-Uz-Zaman Fahad (190021324)

A Thesis Submitted in Accordance with the Requirements of the Degree of

BACHELOR OF SCIENCE
IN
ELECTRICAL AND ELECTRONIC ENGINEERING

Islamic University of Technology (IUT)
Board Bazar, Gazipur-1704, Bangladesh

Department of Electrical and Electronic Engineering
June 2024

ENHANCING POWER GRID RELIABILITY WITH MACHINE LEARNING ALGORITHM FOR FAULT DETECTION AND CLASSIFICATION

Submitted by

Md. Sadi Mobassir (190021117)
Safoat Saima Arpi (190021231)
Farhan-Uz-Zaman Fahad (190021324)

has been approved on 25 June, 2024

____

Md. Arefin Rabbi Emon
Supervisor and Lecturer,
Department of Electrical and Electronic Engineering,
Islamic University of Technology (IUT),
Boardbazar, Gazipur-1704.

Declaration of Authorship

This is to certify that the research work presented in this thesis, entitled **“Enhancing Power Grid Reliability with Machine Learning Algorithms for Fault Detection and Classification”**, is the result of thorough and dedicated research conducted under the supervision of **Md. Arefin Rabbi Emon**. It is further declared that this thesis, in whole or in part, has not been submitted elsewhere for the award of any degree or diploma.

Sadi

Md. Sadi Mobassir (190021117)

Safoat Saima Arpi

Safoat Saima Arpi (190021231)

Fahad

Farhan-Uz-Zaman Fahad (190021324)

Acknowledgement

"In the Name of Allah, the Most Beneficent, the Most Merciful. All praise and thanks be to Allah, the Lord of the 'Alamin (mankind, jinns, and all that exists)."

First and foremost, we are deeply grateful to Allah (SWT) for granting us the health, strength, and ability to undertake and complete this dissertation.

We extend our heartfelt gratitude to our supervisor, **Md. Arefin Rabbi Emon** sir, whose exceptional guidance, meticulous instructions, and unwavering support were indispensable throughout this journey. His experience, mentorship, and valuable insights significantly enriched the quality and depth of our thesis. His dedication to fostering our academic growth has been truly invaluable.

We would like to express sincere appreciation to our families and friends, especially our parents, whose boundless love, encouragement, and unwavering belief in our abilities have been the cornerstone of our achievements. Their steadfast support and sacrifices have been a constant source of strength and motivation.

To all those who have contributed in various ways, whether through discussions, feedback, or moral support, we offer our heartfelt thanks. Your encouragement and assistance have played a crucial role in shaping this thesis.

Contents

List of Tables

List of Figures

List of Acronyms

AI	Artificial Intelligence
ANN	Artificial Neural Network
AUC	Area Under the Curve
BPTT	Backpropagation Through Time
CNN	Convolutional Neural Network
CO	Course Outcome
CSV	Comma-Separated Values
DT	Decision Tree
DBN	Deep Belief Network
DWT	Discrete Wavelet Transform
EEE	Electrical and Electronic Engineering
FN	False Negative
FP	False Positive
HPF	High Pass Filter
L-G	Line-to-Ground Fault
L-L	Line-to-Line Fault
L-L-G	Double Line-to-Ground Fault
L-L-L	Three-Phase Fault
L-L-L-G	Three-Phase Line-to-Ground Fault
LSTM	Long Short-Term Memory
LPF	Low Pass Filter
ML	Machine Learning
MRA	Multiresolution Analysis
OBE	Outcome Based Education
OvA	One-vs-All
OvO	One-vs-One
PNN	Probabilistic Neural Network
PO	Program Outcome
RBF	Radial Basis Function
RNN	Recurrent Neural Network
ROC	Receiver Operating Characteristic
SVM	Support Vector Machine
TN	True Negative
TP	True Positive
WMRA	Wavelet Multi Resolution Analysis
WT	Wavelet Transform

Abstract

In the complex and expansive networks of modern electric power systems, the occurrence of faults is an inevitable challenge that can significantly affect grid reliability and stability. This thesis presents a comprehensive study on the enhancement of power grid reliability through the application of machine learning algorithms for fault detection and classification. The primary focus is on the development and implementation of a hybrid model combining Long Short-Term Memory (LSTM) networks and Support Vector Machines (SVM) to accurately identify and classify various types of power system faults.

The research begins with a detailed analysis of power system faults, including their causes, characteristics, and impacts on the stability of the grid. A significant portion of the study is dedicated to the classification of symmetrical and asymmetrical faults, with an emphasis on the most common and disruptive types such as line-to-ground and three-phase faults. The hybrid LSTM-SVM model is then introduced, highlighting its design, training, and validation processes.

Empirical results demonstrate that the proposed model achieves high precision and recall rates across all fault types, with an overall accuracy of 96.7%. This high level of performance indicates the model's robustness and effectiveness in real-time fault detection and classification, making it a viable solution for practical deployment in power systems.

Furthermore, the thesis integrates principles of Outcome-Based Education (OBE) to align the research with specific educational and professional outcomes. This approach ensures that the project not only addresses technical challenges but also enhances the competencies and skills of engineering students, preparing them for real-world applications and professional practices.

The findings of this research contribute significantly to the field of electrical engineering by providing a robust methodology for improving power grid reliability. The successful application of machine learning techniques in fault detection and classification paves the way for further advancements in smart grid technologies and proactive fault management strategies.

CHAPTER 1

Introduction and Background

In the intricate and vast network of an electric power system, faults are inevitable. A fault, characterized by an abnormal electric current, signifies a defect in the electrical circuit that diverts the current from its intended path. These faults, typically arising from the breakdown of conductors or failure of insulation, pose significant challenges to the stability and reliability of power grids. They can be triggered by various factors including mechanical failures, accidents, excessive internal and external stress, and environmental conditions. Consequently, fault currents, which are notably large due to low impedance in the fault path, can cause severe damage to electrical apparatus and disrupt the power supply to neighboring zones.

The unbalanced voltages resulting from faults, coupled with the diversion of power flow towards the fault, necessitate effective fault management strategies. Power systems, comprising generators, transformers, switchgear, and extensive transmission and distribution circuits, are particularly susceptible to faults. Transmission lines, due to their considerable length and exposure to atmospheric conditions, are especially prone to such occurrences. Although the elimination of faults is unattainable, their incidence can be minimized through improved system design, high-quality equipment, and regular maintenance.

Faults in power systems can be broadly classified based on their causes and characteristics. These include single line-to-ground faults, double line-to-ground faults, and three-phase faults, each leading to significant unbalance in current and voltage, over-voltages, power swings, and other system instabilities. Among these, line-to-ground faults are the most common, particularly in overhead lines, and many of these faults are transitory, persisting for only a short duration.

The transient nature of faults, such as those caused by temporary contacts like a twig falling across a line, highlights the dynamic behavior of fault currents, which vary significantly over time. The initial phase, known as the 'sub-transient' state, is marked by very high fault currents that decrease rapidly. This is followed by the 'transient' state where the rate of current decrease slows down, and eventually, the system reaches a 'steady state' with a constant RMS value of short-circuit current.

Effective protection against faults requires robust and reliable protective systems, capable of timely and accurate fault detection and classification. The line-to-ground faults, given their considerable magnitude, necessitate precise relay operations to ensure the reliability of the power grid. On the other hand, three-phase symmetrical faults, often caused by operational errors, can have catastrophic consequences if not promptly addressed, leading to extensive damage, system instability, and potential complete shutdown of the power grid.

The increasing complexity and scale of modern power systems demand advanced methodologies for fault detection and classification. In this context, machine learning algorithms offer promising solutions to enhance the reliability of power grids. By leveraging the vast amounts of data generated within power systems, machine learning techniques can provide precise, real-time fault detection and classification, thus enabling proactive management and mitigation of faults.

1.1 Types of Faults in Power System

Faults in power systems can lead to significant damage to equipment and interruptions in power supply if not promptly detected and addressed. Understanding the types of faults that can occur is crucial for developing effective fault detection and classification methods. Faults in power systems can generally be categorized into two main types:

i. Symmetrical faults
ii. Asymmetrical faults [1], [2]

1.1.1. Symmetrical Faults

Symmetrical faults, also known as balanced faults, involve all three phases of the power system equally. These faults are rare but severe, resulting in balanced conditions among the phases. The primary characteristic of symmetrical faults is their equal impact on all phases, leading to significant power disruptions. The main types of symmetrical faults are:

i. **Three-Phase Fault (L-L-L):** A three-phase fault occurs when all three phases (A, B, and C) are short-circuited together. This type of fault is also referred to as a three-phase short circuit. Three-phase faults are the most severe because they involve maximum fault current, causing extensive damage to electrical equipment and significant disruptions in power supply. [3] Due to their severity, three-phase faults are usually detected quickly by protection systems, which isolate the affected section to prevent further damage.

ii. **Three-Phase Line-to-Ground Fault (L-L-L-G):** A three-phase line-to-ground fault occurs when all three phases come into contact with the ground simultaneously. This type of fault is extremely severe as it involves all three phases and the ground, resulting in very high fault currents. The consequences can be catastrophic, including extensive equipment damage and widespread power outages.

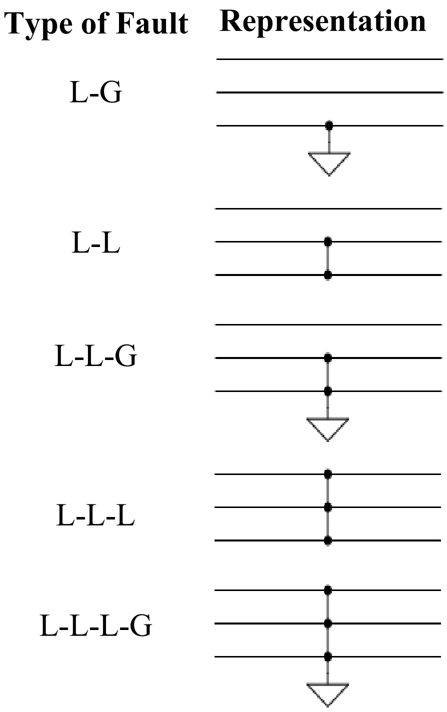

Figure 1.1: Types of Faults

1.1.2. Asymmetrical Faults

Asymmetrical faults, also known as unbalanced faults, do not affect all phases equally. These faults are more common than symmetrical faults and can be further divided into several types based on the phases involved. The main types of asymmetrical faults include:

i. **Single Line-to-Ground Fault (L-G):** This fault occurs when one phase comes into contact with the ground. It is the most common type of fault in power systems. Single line-to-ground faults cause unbalanced currents and can lead to potential damage to insulation and other electrical components if not cleared promptly.[3] These faults are typically detected by protective relays that sense the increase in ground current and isolate the affected line.[1]

ii. **Line-to-Line Fault (L-L):** A line-to-line fault occurs when two phases come into contact with each other, bypassing the neutral or ground. Line-to-line faults result in an unbalanced condition that can cause overloading and damage to the electrical equipment connected to the affected phases.[1], [3] These faults are detected by relays that measure the phase-to-phase current differences and initiate isolation procedures.

iii. **Double Line-to-Ground Fault (L-L-G):** This fault occurs when two phases come into contact with the ground simultaneously. Double line-to-ground faults lead to significant unbalanced currents and can cause severe damage to equipment and interruptions in power supply. [2] These faults are detected by protective systems that monitor ground and phase currents and initiate isolation to protect the system.[1]

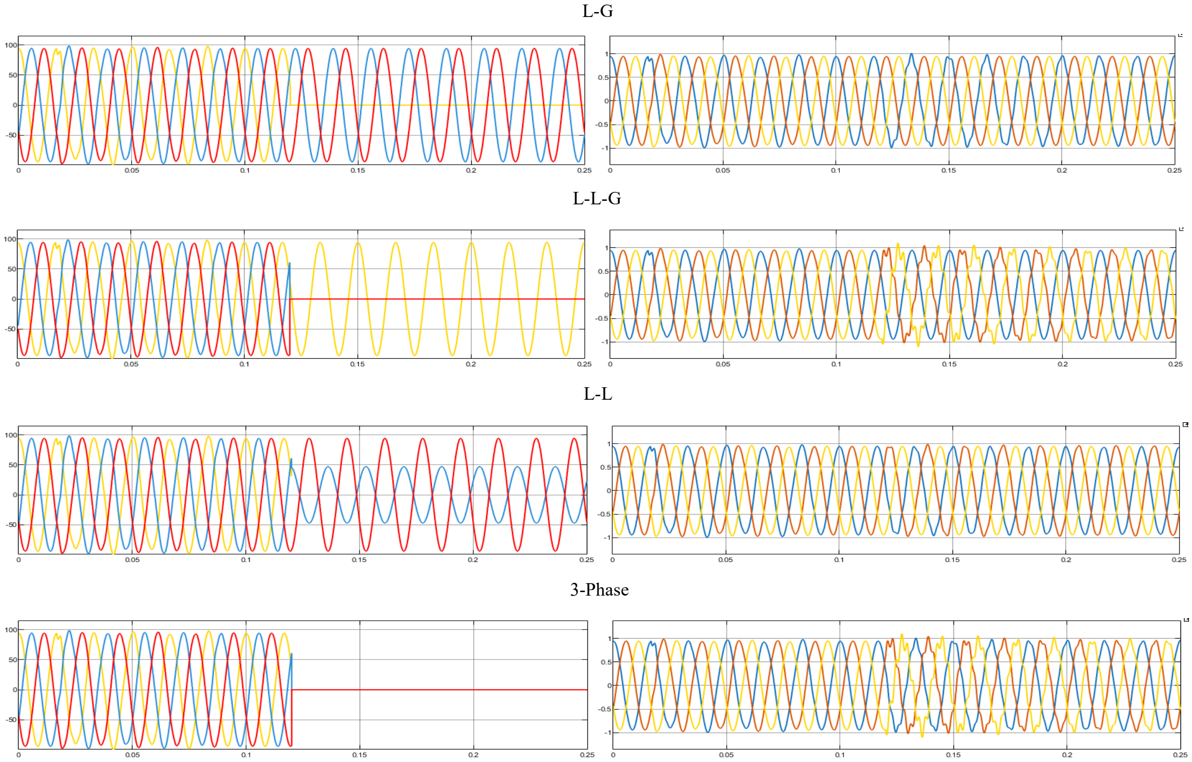

Figure 1.2: Voltage & Current Characteristics During Different Types of Faults

1.2 Machine Learning for Fault Detection

The rapid advancement of technology and the increasing complexity of power systems have necessitated the development of more sophisticated fault detection and classification methods. Traditional methods, relying on predefined rules and thresholds, often fall short in managing the diverse and dynamic nature of modern power grids. Machine learning (ML) offers a powerful alternative, leveraging vast amounts of data to improve the accuracy and efficiency of fault detection and classification.[3]

Machine learning algorithms can analyze large datasets generated by power systems to identify patterns and anomalies indicative of faults. These algorithms can be trained using historical data, enabling them to recognize the signatures of various types of faults. Commonly used ML techniques in fault detection include decision trees, support vector machines (SVM), neural networks, and ensemble methods.[3], [4], [5], [6]

1.2.1 Decision Trees and Random Forests

Decision trees create models that predict the type of fault based on input features such as voltage, current, and impedance. A decision tree algorithm recursively splits the data based on feature values to build a tree structure, where each node represents a decision point. The model learns to classify faults by selecting the most informative features at each split.

$$Gini_{Index} = 1 - \sum_{i=1}^{n} (p_i)^2$$

Random forests aggregate multiple decision trees to improve prediction accuracy and robustness by reducing overfitting. Each tree in the forest is trained on a random subset of the data and features, and the final prediction is made by averaging the predictions of all the trees.[7]

$$\hat{y} = \frac{1}{N} \sum_{i=1}^{N} T_i(x)$$

where $\hat{y}$ is the final prediction, NNN is the number of trees, and $T_i(x)$ is the prediction from the i-th tree[7], [8].

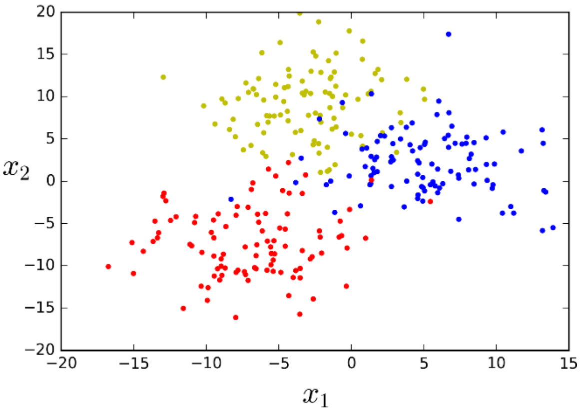

Figure 1.3: Sample Data in RF Classifier

1.2.2 Support Vector Machines (SVM)

SVMs classify faults by finding the optimal hyperplane that separates different fault types in the feature space. They are particularly effective in high-dimensional spaces and can handle non-linear relationships through kernel functions.[9], [10]

$$f(x) = sign(w \cdot x + b)$$

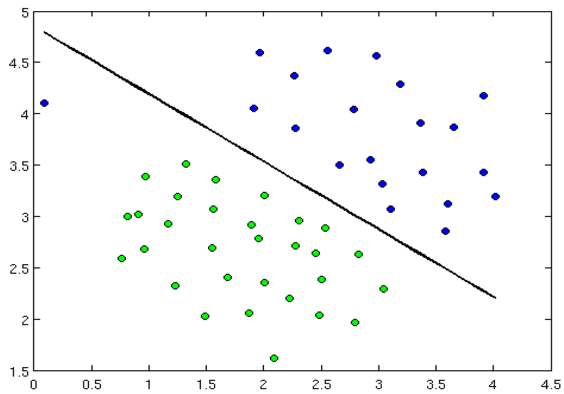

Figure 1.4: Sample Data in SVM Classifier

where w is the weight vector and b is the bias term. The kernel function $K(x_i, x_j)$ allows SVMs to operate in a transformed feature space:

$$K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j)$$

Common kernels include the linear kernel, polynomial kernel, and radial basis function (RBF) kernel.[6]

1.2.3 Neural Networks and Deep Learning

Neural networks, particularly deep learning models, are adept at handling complex patterns in large datasets. Convolutional neural networks (CNNs) and recurrent neural networks (RNNs), including Long Short-Term Memory (LSTM) networks, have been successfully applied to fault detection, benefiting from their ability to capture spatial and temporal dependencies in data.[11]

A simple feedforward neural network can be represented as:

$$y = f(Wx + b)$$

where W is the weight matrix, x is the input vector, b is the bias vector, and f is the activation function.

In CNNs, the convolution operation is defined as:

$$s_{ij}^{(k)} = \left(x * w^{(k)}\right)_{ij} = \sum_m \sum_n \left(x_{i+mj+n}\right) w_{mn}^{(k)}$$

where x is the input, w is the filter, and s is the feature map.

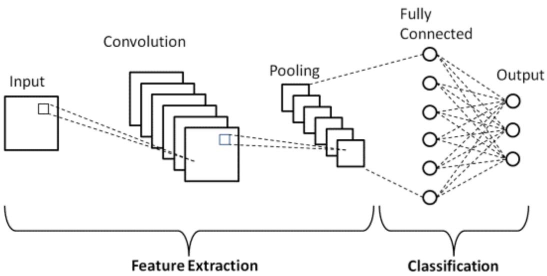

Figure 1.5: Feature Extraction in CNN

RNNs, and specifically LSTMs, are designed to handle sequential data.[12], [13] The LSTM cell includes gates to control the flow of information:

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right)$$
$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$$
$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$
$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$
$$h_t = o_t * tanh\,(C_t)$$

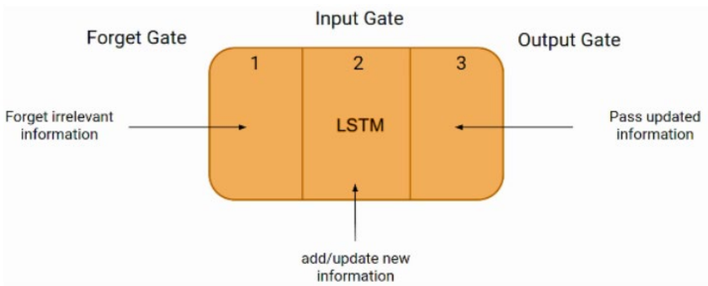

Figure 1.6: LSTM Model Architecture

where f_t, i_t and o_t are the forget, input, and output gates, respectively.

1.2.4 Ensemble Methods

Ensemble methods, such as boosting and bagging, combine multiple ML models to improve overall performance. These methods enhance fault detection accuracy by leveraging the strengths of different models and mitigating their individual weaknesses.[14]

In bagging, models are trained on different subsets of the data, and their predictions are averaged:

$$\hat{y} = \frac{1}{N}\sum_{i=1}^{N} T_i(x)$$

Boosting, on the other hand, sequentially trains models, with each new model focusing on the errors made by the previous ones:

$$F_m(x) = F_{m-1}(x) + \alpha_m h_m(x)$$

where F_m is the boosted model, h_m is the m-th weak learner, and α_m is the learning rate. [14], [15]

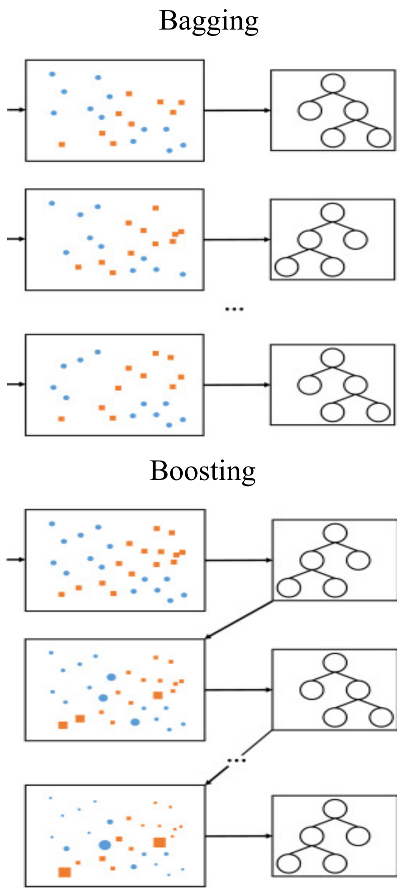

Figure 1.7: Bagging and Boosting

1.3 Wavelet Transform

Signal processing is a crucial part of fault detection in power systems. Traditional methods such as Fourier analysis and Kalman filtering have been widely used. However, these techniques have limitations, particularly in handling non-stationary signals where the frequency content changes over time. Wavelet Transform (WT), a relatively new mathematical tool, offers significant advantages in this context due to its ability to analyze both frequency and time-domain characteristics of signals simultaneously.[2]

Wavelet Transform decomposes signals into a series of wavelet components, each corresponding to a specific octave frequency band. This approach provides more detailed information, making wavelets particularly useful for detecting and classifying the sources of surges, faults, and other transient disturbances in power systems.[2], [16]

The Wavelet Transform is based on two fundamental equations: the scaling function $\phi(t)$ and the wavelet function $\psi(t)$. These functions are defined as:

$$\phi(t) = \sum_{k} h_k\, \phi(2t - k) \dots\dots\dots\dots\dots (1)$$

$$\psi(t) = \sum_{k} g_k\, \psi(2t - k) \dots\dots\dots\dots\dots (2)$$

The scaling and wavelet functions must satisfy specific conditions for orthonormal basis functions:

$$\sum_{k=-\infty}^{\infty} h_k = \sqrt{2} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots..(3)$$

$$\sum_{k=-\infty}^{\infty} h_k h_{k+2l} = \delta_{l0} \ldots\ldots\ldots\ldots\ldots\ldots...(4)$$

Here, h_k and g_k are discrete sequences representing the filters used to solve these equations, where $g_k = (-1)^k h_{1-k}$. These sequences form the basis for constructing the scaling and wavelet functions.[16], [17], [18]

Once a wavelet system is created, it can expand a function f(t) in terms of these basis functions:

$$w(t) = \sum_{l\in\mathbb{Z}} c(l)\,\phi(t) + \sum_{j=0}^{J-1}\sum_{k\in\mathbb{Z}} d(j,k)\,\psi_{j,k}(t) \ldots..(6)$$

Where the coefficients $c(l)$ and (j, k) are calculated by inner product:

$$c(l) = \int f(t)\,\phi_l(t)\,dt \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots.(7)$$

$$d(j,k) = \int f(t)\,\psi_{j,k}(t)dt \ldots\ldots\ldots\ldots\ldots\ldots..(8)$$

1.3.1 Multiresolution Analysis

Wavelet Transform is largely due to the use of Multiresolution Analysis (MRA), which can be efficiently implemented with two filters: a high pass filter (HPF) and a low pass filter (LPF). The high pass filter derives from the wavelet function and retains the details in a certain input signal, while the low pass filter delivers a smoothed version derived from the scaling function associated with the mother wavelet.[3], [16]

In wavelet MRA, results are carried out using the db4 mother wavelet for signal analysis. The wavelet energy is the sum of squared detailed wavelet transform coefficients. The energy of wavelet coefficient varies over different scales depending on the input signals. Energy of signal is concentrated mostly in the approximation part and little in the detail part.[16]

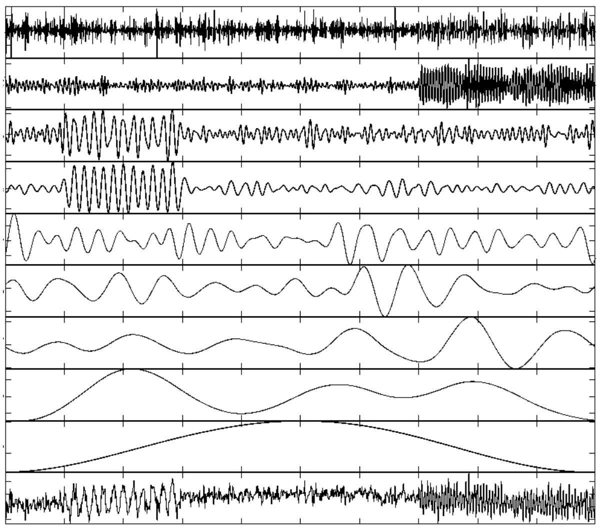

Figure 1.8: Wavelet Multiresolution Analysis

1.3.2 Application in Power System

Wavelet Transform can effectively analyze transient signals in power systems. For example, when a fault occurs within the power network, the transient voltage and current signal in the fault section contain predominantly high frequency components due to the superimposed reflection of the fault signals at the fault point. The energy of these high frequency signals is used as an indicator of fault occurrence.[1]

The fault detection rules are established by analyzing the current waveforms in the time domain and in the first decomposition level of the DWT. The energy of the wavelet coefficients is calculated using the following equation: [4] [5]

$$E_w = \sum_{l=1}^{N} \left[d_{A,l}(k)\right]^2$$

where$d_{A,l}(k)$ is the k^{th} wavelet coefficient within the l^{th} window, and N_w is the window length:[1], [19]

$$N_w = \frac{N_s}{2}$$

Here, N_s is the number of samples within one cycle of the fundamental frequency of 50 Hz.

Figure 1.4 demonstrates a simple power system where such analysis can be applied:

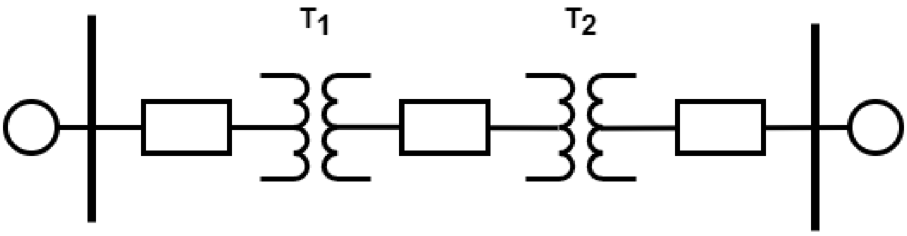

Figure 1.9: Single Line Diagram of a Power System

1.3.3 Advantages and Challenges

Wavelet Transform offers several advantages over traditional Fourier analysis:

Time-Frequency Localization: WT provides better time-frequency localization, allowing for the detection of transients and short-duration events.[20]

Multiresolution: WT can analyze signals at multiple resolutions, providing a detailed view of signal characteristics across different scales.[3], [19]

Non-stationary Signal Analysis: WT is particularly effective in analyzing non-stationary signals where the frequency content changes over time.[3]

However, there are also challenges; The effectiveness of WT depends significantly on the choice of the mother wavelet. WT can be computationally intensive, especially for large datasets and real-time applications.[18]

CHAPTER 2

Literature Review

The application of machine learning techniques for fault detection and classification in power systems has garnered significant attention in recent years. Extensive research has been conducted to explore various machine learning algorithms such as decision trees, support vector machines (SVM), ensemble methods, and neural networks for their efficacy in identifying and classifying different types of faults.

The power grid is a complex network of interconnected transmission lines that transport electricity from generation sources to distribution systems. Faults in power systems are typically caused by a variety of factors including mechanical failures, environmental conditions, insulation breakdown, and operational errors. They can be broadly classified into symmetrical and asymmetrical faults. Symmetrical faults, which involve all three phases of the power system equally, are rare but severe. Asymmetrical faults, which affect one or two phases, are more common. These faults lead to unbalanced currents and voltages, potentially causing significant damage to equipment and interruptions in power supply if not promptly detected and addressed.

Machine learning algorithms offer a powerful alternative to traditional fault detection methods, leveraging vast amounts of data to improve the accuracy and efficiency of fault detection and classification. Decision trees and random forests, for example, are widely used due to their simplicity and effectiveness in handling categorical data. They provide clear insights into the decision-making process, making them advantageous in scenarios where interpretability is crucial. However, their performance can be limited when dealing with highly complex or noisy datasets.

Current methods for detecting and classifying transmission line faults are often slow and inaccurate. This is because these traditional methods rely on human operators to evaluate sensor and measurement data. This can be a laborious and prone-to-error process, particularly in difficult circumstances. These traditional techniques, such as relay protection and fault indicators, are not always effective in identifying all types of faults. For faster, more accurate, and timely fault detection and classification, the suggested system would take a more advanced approach that makes use of machine learning and artificial intelligence approaches. There is ongoing research based on the electrical fault detection technique. All this research tries to improve the performance and increase the efficiency of the system. Wavelet transforms are widely used for detecting electrical faults. Fourier transform is more common for analysis purposes than wavelet transform. However, wavelet transform has some features that make it more suitable for being used in transmission line fault detection. While comparing the Wavelet Transform and Fourier Transform, it is seen that the Fourier transform (FT) provides frequency components of a signal, averaged over the entire signal duration. But Wavelet transform (WT) gives information about the frequency components as well as able to indicate what time these components occur. Wavelet transform uses a short window for high frequency and a long window for low frequency.[21] There are several types of wavelets, such as - Daubechies, Morlet, haar etc. Three ways wavelet transforms can be used. They are - continuous Wavelet Transform (CWT), Discrete wavelet transform (DWT) and multiresolution wavelet transform. For signal processing purposes, DWT is widely seen. DWT offers a denoising facility and compresses the signal keeping the features preserved.[17], [18], [22]A paper combined wavelet multiresolution analysis with ANN to support better results for fault detection and classification. In this case, two ANN models were used, one for the ground fault and another for the phase fault. From Discrete Wavelet transform (DWT) coefficients fault is detected. Then the ANN models help to classify the fault.[19] Another paper combined two modules for fault detection and localization: the classification and detection module. From the discrete wavelet transform (DWT) coefficients, energy is calculated in this model. The energy value leads to fault detection. If a fault is detected, then the model approaches the classification module. This module classifies the fault [2]. In another research, the use of wavelet transform has been combined with a Support vector machine. In this case, wavelet transform and Support vector machine are used (SVM) to detect, classify and localize the transmission line fault. Similar to previous cases, here the Discrete wavelet transform (DWT) coefficients help to calculate the spectral energy. The threshold value of that spectral energy leads to fault detection. Using the optimal hyperplane from the SVM model, fault location detection.[6], [23] Using the Artificial Neural Network (ANN) model, fault classification has been practised for a while now. When any fault occurs, it impacts the frequency characteristics of the voltage and current signals.[24], [25] Recent studies have evaluated the performance of the SVM classifiers[6]. ANN is helpful for the accurate detection of fault locations. Backpropagation Neural Network Architecture is seen to provide quite reliable results. Other neural network-based approaches have been employed as well.[18], [24] Another research shows the comparative performance of the fault detection purpose using various models. In this paper, a comparative analysis between ANN and RNN (LSTM and LSTM-WR) is shown. LSTM shows an accuracy of 99.98%. Meanwhile, ANN showed around 42% accuracy. This analysis has used six feature variables and four output variables for the fault classification model[25]. Recurrent Neural network (RNN) has a backpropagation facility. As a result, the weights can be fixed and the performance becomes higher than ANN. RNN provides an edge in sequential Data handling, Real-Time analysis, and depicting Non-linear relationships. The transmission line fault detection data has these characteristics. Long Strong Time Memory (LSTM) is a type of RNN mode. LSTM is designed to - Capture long-term dependency. LSTM can deal with the vanishing gradient problem as well. In RNN, the vanishing gradient problem fails to capture abrupt changes in the data. LSTM also provides better memory management as it has a gated facility for the irrelevant and unnecessary portion which is called forget gate[12], [26]. The use of a probabilistic neural network (PNN) with wavelet transform is also evident. In this case, the extracted features in wavelet transform are used in PNN. Using the varying window of WT, this method shows high accuracy. [26]Another aspect of Fault detection involves locating the fault. For this purpose, time–frequency domain impedance can be helpful in calculations. A combination of supervised and unsupervised learning is used for this location detection of fault. Deep Belief Network (DBN) −based cable fault type and location recognition model can come in handy for this goal. [27]

Support Vector Machines (SVM) are another popular choice for fault detection and classification. SVMs are effective in high-dimensional spaces and can handle non-linear relationships through the use of kernel functions. Research indicates that SVMs provide robust performance in fault classification tasks, particularly when combined with other techniques for feature extraction and preprocessing. [9] The SVM's ability to find the optimal hyperplane that separates different classes makes it a reliable choice for both binary and multiclass classification problems in power systems.[6]

Neural networks and deep learning models, such as Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks, have shown significant promise in fault detection due to their capability to learn from large volumes of data and capture complex patterns. CNNs are particularly effective for spatial data and image-like representations of signal data, while LSTMs are well-suited for time-series data and sequential patterns, which are common in power system measurements.[26] Studies have demonstrated that LSTM networks can effectively model the temporal dependencies in power system data, leading to improved fault detection accuracy.

Ensemble methods, which combine multiple learning algorithms to obtain better predictive performance, have also been explored. Techniques such as bagging, boosting, and stacking have been applied to improve the robustness and accuracy of fault detection systems. [14]These methods help mitigate the weaknesses of individual algorithms by leveraging their strengths, resulting in more reliable fault detection and classification.[15]

Table 2.1: Different ML Model with Their Accuracy

Model	Accuracy
ANN	42% [28]
LSTM	98% [12]
SVM	93.2% [6]
Ensemble	92.5% [15]
Decision Tree	87.6% [7]

Wavelet transform is another powerful tool used in conjunction with machine learning for fault detection. The wavelet transform's ability to analyze signals at multiple resolutions makes it particularly useful for detecting and localizing transient events in power systems. [20]By decomposing the signal into different frequency components, wavelet transform provides valuable features that can be used by machine learning models for accurate fault detection and classification.[17], [18]

The integration of machine learning with wavelet transform has been shown to enhance fault detection capabilities significantly. Research indicates that wavelet-based feature extraction followed by machine learning classification results in higher accuracy and robustness compared to traditional methods. [17], [22]For instance, using wavelet transform to preprocess the data before feeding it into an SVM or neural network model has been found to improve classification performance by capturing relevant features more effectively.[18]

Recent advancements in the field also include the development of hybrid models that combine the strengths of different algorithms. These hybrid models address the limitations of individual algorithms and provide a more comprehensive approach to fault detection.[29], [30]

Comparative studies of these machine learning models reveal significant differences in their accuracy, efficiency, and applicability to various fault types. For instance, LSTM models have shown high accuracy in detecting temporal patterns and can handle sequential data effectively. SVMs, on the other hand, are more efficient in classifying non-linear data and are particularly useful when the fault features are well-separated in the feature space. [13]Decision trees and random forests, while simpler, provide interpretable results that are crucial for understanding the decision process in fault detection. [8]Ensemble methods and hybrid models generally outperform individual algorithms by combining their strengths and mitigating their weaknesses.[14]

The high level of performance demonstrates the model's robustness and effectiveness in real-time fault detection and classification, making it a viable solution for practical deployment in power systems.

Machine learning applications in power system fault detection and classification reveals a trend towards the use of advanced algorithms and hybrid models. These approaches offer significant improvements in accuracy, robustness, and interpretability, making them valuable tools for maintaining the reliability and stability of power systems.[23] The ongoing research and development in this field continue to push the boundaries of what is possible, promising even more effective solutions for the challenges faced in power system fault detection and classification.[3], [23]

CHAPTER 3

Study of Machine Learning Model

Machine learning, a subset of artificial intelligence, has revolutionized numerous fields by enabling systems to learn from data and improve their performance over time without explicit programming. This chapter delves into the LSTM-SVM hybrid model. This model combines the strengths of Long Short-Term Memory (LSTM) networks and Support Vector Machines (SVM), offering a powerful approach for tasks involving sequential data and classification problems. Understanding the theoretical foundations, practical implementations, and real-world applications of the LSTM-SVM hybrid model is crucial for leveraging its full potential in solving complex problems across various domains.

3.1 LSTM Model

Long Short-Term Memory (LSTM) networks are a specialized type of recurrent neural network (RNN) designed to effectively capture long-term dependencies in sequential data. They address the limitations of standard RNNs, particularly the vanishing gradient problem, by incorporating a sophisticated cell structure. LSTMs are well-suited for tasks such as time-series prediction, natural language processing, and anomaly detection.[5], [13]

3.1.1 LSTM Architecture

The fundamental unit of an LSTM network is the LSTM cell, which contains three types of gates that control the flow of information:

i. **Forget Gate** (f_t): This gate decides what information from the previous cell state should be discarded. The forget gate equation is given by:

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right)$$

where σ is the sigmoid function, W_f is the weight matrix, h_{t-1} is the previous hidden state, x_t is the current input, and b_f is the bias term.[13]

ii. **Input Gate** (i_t): This gate determines which new information should be added to the cell state. The input gate equation is:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

Alongside the input gate, a candidate cell state ($\tilde{C}_t$) is generated by:

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$$

where tanh is the hyperbolic tangent function, and W_C and b_C are the weights and biases for the candidate state.[13]

iii. **Cell State Update** (C_t): The cell state is updated using the forget gate and the input gate:

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

where $*$ denotes element-wise multiplication.

iv. **Output Gate** (o_t): This gate decides the output of the LSTM cell, which also serves as the hidden state for the next time step:

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$

The hidden state (h_t) is then calculated as:

$$h_t = o_t * tanh\ (C_t)$$

3.1.2 LSTM for Transmission Line Fault Detection

Transmission lines are critical components of power systems, and their reliable operation is essential for maintaining power quality and stability. Faults in transmission lines can lead to significant disruptions and must be detected promptly to initiate corrective measures. [26]LSTMs are particularly effective for this application due to their ability to model temporal dependencies in sequential data.

In a binary classification context for transmission line fault detection, the LSTM network processes sequences of electrical measurements (such as voltage, current, and impedance) to predict the presence of a fault. The classification task involves distinguishing between two classes: fault and no fault.[12], [13]

3.1.3 Data Preprocessing

To train an LSTM model for fault detection, the input data must be pre-processed:

i. **Normalization**: Electrical measurements are normalized to ensure the input features have a consistent scale, which helps improve the training efficiency and convergence of the model.
ii. **Segmentation**: The time-series data is segmented into fixed-length sequences. Each sequence serves as an input sample for the LSTM network.

3.1.4 Model Training and Evaluation

The LSTM model is trained using a labelled dataset containing sequences of electrical measurements and their corresponding fault status. The training process involves minimizing a binary cross-entropy loss function using gradient descent and backpropagation through time (BPTT).

The model's performance is evaluated using metrics such as accuracy, precision, recall, and the F1-score. These metrics provide insights into the model's ability to correctly detect faults while minimizing false positives and false negatives.[5]

3.1.5 Implementation Example

Consider a practical implementation of an LSTM model for transmission line fault detection. The input data consists of sequences of voltage and current measurements collected from sensors placed along the transmission line. The LSTM model is structured with multiple LSTM layers followed by a dense output layer with a sigmoid activation function for binary classification.

Table 3.1: Python code snippet for training a LSTM model

```
import numpy as np
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import LSTM, Dense

# Generate synthetic data for demonstration
def generate_data(num_samples, sequence_length, num_features):
X = np.random.rand(num_samples, sequence_length, num_features)
y = np.random.randint(0, 2, num_samples)
return X, y

# Parameters
num_samples = 1000
sequence_length = 50
num_features = 2

# Generate data
X, y = generate_data(num_samples, sequence_length, num_features)

# Define LSTM model
model = Sequential()
model.add(LSTM(64, input_shape=(sequence_length, num_features)))
model.add(Dense(1, activation='sigmoid'))

# Compile model
model.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accuracy'])

# Train model
model.fit(X, y, epochs=10, batch_size=32, validation_split=0.2)
```

In this example, the synthetic dataset consists of sequences with two features, representing voltage and current measurements. The LSTM model is trained to classify each sequence as either a fault or no fault, demonstrating its capability for fault detection in transmission lines.

3.1.6 Performance Optimization

To improve the performance of the LSTM model, several strategies can be employed:

i. **Hyperparameter Tuning**: Adjusting the model's hyperparameters, such as the number of LSTM units, learning rate, batch size, and number of epochs, can significantly impact performance.
ii. **Regularization Techniques**: Techniques such as dropout can be used to prevent overfitting by randomly setting a fraction of the input units to zero during training.
iii. **Data Augmentation**: Generating additional synthetic data or applying transformations to the existing data can help the model generalize better.

3.1.7 Real World Application and Challenges

In real-world scenarios, deploying an LSTM model for transmission line fault detection involves several challenges, including:

i. **Data Quality**: Ensuring high-quality, labeled data is available for training and validation is crucial. Noise and missing values in the data can affect model performance.[12]
ii. **Scalability**: The model must be capable of handling large volumes of data in real-time, especially in extensive power grid systems.[12]
iii. **Integration**: Integrating the model into existing monitoring and control systems requires careful consideration of system compatibility and latency requirements.[12], [13]

3.2 SVM Model

Support Vector Machines (SVM) are a class of supervised learning algorithms used for classification and regression tasks. SVMs are known for their ability to handle high-dimensional data and find the optimal hyperplane that separates different classes. This makes them particularly effective for classification problems, including fault classification in transmission lines.

3.2.1 SVM Architecture

SVM operates by finding the hyperplane that best divides a dataset into classes. In a binary classification scenario, this involves maximizing the margin between the hyperplane and the nearest data points from each class, known as support vectors. [9]The decision function for an SVM is defined as:

$$f(x) = w \cdot x + b$$

where w is the weight vector, x is the input feature vector, and b is the bias term. The optimal hyperplane is found by solving the following optimization problem:

$$\min_{w,b} \frac{1}{2} \|w\|^2$$

subject to the constraints for all training samples (x_i, y_i):

$$y_i(w \cdot x_i + b) \geq 1$$

For non-linearly separable data, SVM can employ kernel functions to map the input features into a higher-dimensional space where a linear hyperplane can be used to perform the separation. Common kernel functions include linear, polynomial, and radial basis function (RBF) kernels.[4], [10]

3.2.2 SVM for Transmission Line Fault Classification

Transmission line fault classification involves identifying the type of fault that has occurred, which can be a multiclass classification problem. Faults in transmission lines can be classified into different categories such as single line-to-ground faults, line-to-line faults, double line-to-ground faults, and three-phase faults.[4]

In a multiclass classification context, SVMs can be extended using strategies such as one-vs-one (OvO) or one-vs-all (OvA). In the OvO approach, a binary SVM classifier is trained for every possible pair of classes, while in the OvA approach, a single SVM classifier is trained to distinguish between one class and all other classes.[6], [10], [23]

3.2.3 Data Preprocessing

To train an SVM model for fault classification, the input data must be pre-processed:

i. **Feature Extraction**: Relevant features such as voltage, current, and impedance measurements are extracted from the raw data.
ii. **Normalization**: Features are normalized to ensure they are on a consistent scale, improving the performance and convergence of the SVM model.
iii. **Label Encoding**: Fault types are encoded as numerical labels for training the SVM classifier.

3.2.4 Model Training and Evaluation

The SVM model is trained using a labelled dataset containing feature vectors and their corresponding fault types. The training process involves solving the optimization problem to find the optimal hyperplane for each binary classifier in the multiclass setting.

The model's performance is evaluated using metrics such as accuracy, precision, recall, F1-score, and the confusion matrix. [6]These metrics provide insights into the model's ability to correctly classify different types of faults while minimizing misclassification.[4], [6]

3.2.5 Implementation Example

Consider a practical implementation of an SVM model for transmission line fault classification. The input data consists of feature vectors derived from voltage and current measurements collected from sensors along the transmission line. The SVM model is structured to handle multiclass classification using the one-vs-one approach.

Table 3.2: Python code snippet for training a SVM model

```
import numpy as np
from sklearn import svm
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import accuracy_score, classification_report,
confusion_matrix

# Generate synthetic data for demonstration
def generate_data(num_samples, num_features, num_classes):
    X = np.random.rand(num_samples, num_features)
    y = np.random.randint(0, num_classes, num_samples)
    return X, y

# Parameters
num_samples = 1000
num_features = 5
num_classes = 4

# Generate data
X, y = generate_data(num_samples, num_features, num_classes)

# Normalize data
scaler = StandardScaler()
X = scaler.fit_transform(X)

# Define SVM model
model = svm.SVC(kernel='rbf', decision_function_shape='ovo')

# Train model
model.fit(X, y)

# Predict
y_pred = model.predict(X)

# Evaluate model
accuracy = accuracy_score(y, y_pred)
print(f'Accuracy: {accuracy}')
print('Classification Report:')
print(classification_report(y, y_pred))
print('Confusion Matrix:')
print(confusion_matrix(y, y_pred))
```

In this example, the synthetic dataset consists of feature vectors with five features representing electrical measurements. The SVM model is trained to classify each feature vector into one of four fault types, demonstrating its capability for fault classification in transmission lines.

3.2.6 Performance Optimization

To improve the performance of the SVM model, several strategies can be employed:

i. **Hyperparameter Tuning**: Adjusting the model's hyperparameters, such as the kernel type, C (regularization parameter), and gamma (kernel coefficient for RBF), can significantly impact performance.[9], [10]
ii. **Feature Selection**: Identifying and using the most relevant features can improve model accuracy and reduce complexity.[6]
iii. **Data Augmentation**: Generating additional synthetic data or applying transformations to the existing data can help the model generalize better.

3.2.7 Real World Application and Challenges

In real-world scenarios, deploying an SVM model for transmission line fault classification involves several challenges, including:

i. **Data Quality**: Ensuring high-quality, labeled data is available for training and validation is crucial. Noise and missing values in the data can affect model performance.[23]
ii. **Scalability**: The model must be capable of handling large volumes of data in real-time, especially in extensive power grid systems.[23]
iii. **Integration**: Integrating the model into existing monitoring and control systems requires careful consideration of system compatibility and latency requirements.[6]

3.3 LSTM-SVM Hybrid Model

Hybrid models combine the strengths of different machine learning algorithms to enhance performance and tackle complex problems more effectively. In the context of transmission line fault detection and classification, a hybrid model integrating Long Short-Term Memory (LSTM) networks and Support Vector Machines (SVM) leverages the temporal sequence learning capability of LSTMs and the powerful classification ability of SVMs. This section explores the architecture, functioning, and advantages of the LSTM-SVM hybrid model.

3.3.1 Architecture of the LSTM-SVM Hybrid Model

The LSTM-SVM hybrid model involves a two-step process:

i. **Feature Extraction with LSTM**: The LSTM network processes sequences of electrical measurements to capture temporal dependencies and generate high-level features.

ii. **Classification with SVM**: The extracted features are fed into an SVM classifier to perform fault classification.

The architecture of the LSTM-SVM hybrid model can be summarized as follows:

i. **Input Layer**: Sequential data of electrical measurements (e.g., voltage, current).
ii. **LSTM Layer**: Extracts temporal features from the input sequences.
iii. **Feature Extraction**: The hidden states from the LSTM are used as feature vectors.
iv. **SVM Layer**: Classifies the extracted features into fault categories.

3.3.2 Data Preprocessing

To train the LSTM-SVM hybrid model, the input data must be preprocessed:

i. **Normalization**: Electrical measurements are normalized to ensure consistent feature scaling.
ii. **Segmentation**: Time-series data is segmented into fixed-length sequences.
iii. **Label Encoding**: Fault types are encoded as numerical labels for the SVM classifier.

3.3.3 Model Training and Evaluation

The hybrid model is trained in two stages:

1. **LSTM Training**: The LSTM network is trained to capture temporal patterns in the input sequences.
2. **SVM Training**: The extracted features from the trained LSTM network are used to train the SVM classifier for fault classification.

3.3.4 Implementation Example

Table 3.3: Python code snippet for training a LSTM-SVM hybrid model

```
import numpy as np
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import LSTM, Dense
from sklearn import svm
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import accuracy_score, classification_report,
confusion_matrix

# Generate synthetic data for demonstration
def generate_data(num_samples, sequence_length, num_features, num_classes):
    X = np.random.rand(num_samples, sequence_length, num_features)
    y = np.random.randint(0, num_classes, num_samples)
    return X, y
# Parameters
num_samples = 1000
sequence_length = 50
num_features = 2
num_classes = 4
```

```
# Generate data
X, y = generate_data(num_samples, sequence_length, num_features, num_classes)

# Normalize data
scaler = StandardScaler()

# Define LSTM model
lstm_model = Sequential()
lstm_model.add(LSTM(64, input_shape=(sequence_length, num_features)))
lstm_model.add(Dense(32, activation='relu'))
lstm_model.add(Dense(16, activation='relu'))

# Train LSTM model
lstm_model.compile(optimizer='adam', loss='mse')
lstm_model.fit(X, y, epochs=10, batch_size=32, validation_split=0.2)

# Extract features using LSTM
lstm_features = lstm_model.predict(X)
lstm_features = scaler.fit_transform(lstm_features)

# Define SVM model
svm_model = svm.SVC(kernel='rbf', decision_function_shape='ovo')

# Train SVM model
svm_model.fit(lstm_features, y)

# Predict using SVM
y_pred = svm_model.predict(lstm_features)

# Evaluate model
accuracy = accuracy_score(y, y_pred)
print(f'Accuracy: {accuracy}')
print('Classification Report:')
print(classification_report(y, y_pred))
print('Confusion Matrix:')
print(confusion_matrix(y, y_pred))
```

CHAPTER 4

Methodology

The methodology used to develop the LSTM-SVM hybrid model for transmission line fault detection and classification encompasses data preprocessing, feature extraction using wavelet transform, model training and evaluation, and performance optimization. Additionally, a detailed algorithm and flowchart are provided to illustrate the overall process.

4.1 Algorithm for LSTM-SVM Hybrid Model

i. **Data Collection**:
   - Load the detect data and class data from CSV files.

ii. **Data Preprocessing**:
   - Clean the data by removing irrelevant columns and handling missing values.
   - Engineer features by creating new columns and dropping redundant ones.

iii. **Feature Extraction**:
   - Apply wavelet transform (BIOR 2.2 and BIOR 4.4) to the electrical measurements.
   - Normalize the extracted features using StandardScaler.

iv. **Model Training**:
   - Train an LSTM model to extract high-level features from the normalized data.
   - Use the features extracted by the LSTM to train an SVM classifier.

v. **Model Evaluation**:
   - Evaluate the model's performance using metrics such as accuracy, precision, recall, and F1-score.
   - Generate confusion matrices and classification reports.

vi. **Performance Optimization**:
   - Use GridSearchCV to perform hyperparameter tuning for the SVM model.
   - Implement feature selection to improve model accuracy.

4.2 Data Collection and Preprocessing

4.2.1 Data Collection

The datasets used in this study include:

- **Detect Data**: Contains voltage and current measurements for fault detection.
- **Class Data**: Contains voltage and current measurements along with fault type labels for fault classification.

Both datasets were loaded from CSV files.

```
# Load the detect_data dataset
detect_data_path = '/content/detect_dataset.csv'
detect_data = pd.read_csv(detect_data_path)
```

```
# Load the class_data dataset
class_data_path = '/content/classData.csv'
class_data = pd.read_csv(class_data_path)
```

4.2.2 Data Cleaning

The datasets were cleaned to remove any missing or irrelevant data. For instance, columns that do not contribute to fault detection and classification were removed.

```
# Drop unnecessary columns from detect_data
detect_data.drop(columns=['Unnamed: 7', 'Unnamed: 8'], inplace=True)

# Drop rows with missing values in class_data
class_data = class_data.dropna()
```

4.2.3 Feature Engineering

New features were created, and redundant columns were removed to improve the efficiency of the model. Specifically, a new column was created to indicate the presence of a fault.

```
# Add a new column 'Output (S)' based on the condition G + A + B + C >= 1
class_data['Output (S)'] = (class_data['G'] + class_data['A'] + class_data['B'] +
class_data['C'] >= 1).astype(int)

# Drop the columns G, A, B, C
class_data.drop(columns=['G', 'A', 'B', 'C'], inplace=True)
```

4.2.4 Fault Classification Labels

A custom function was used to classify faults into multiple categories based on specific conditions of the input features.

```
# Define the custom function to classify faults
def classify_fault(row):
    G, A, B, C = row['G'], row['A'], row['B'], row['C']
    if A == 1 and B == 1 and C == 1:
        return '3-P'
    elif G == 1 and ((A == 1 and B == 0 and C == 0) or (B == 1 and A == 0 and C
== 0) or (C == 1 and A == 0 and B == 0)):
        return 'L-G'
    elif G == 1 and ((A == 1 and B == 1 and C == 0) or (B == 0 and A == 1 and C
== 1) or (C == 1 and A == 0 and B == 1)):
        return 'L-L-G'
    elif G == 0 and ((A == 1 and B == 1 and C == 0) or (B == 0 and A == 1 and C
== 1) or (C == 1 and A == 0 and B == 1)):
        return 'L-L'
    else:
        return 'No'

# Apply the custom function to classify faults
```

```
df['Class'] = df.apply(classify_fault, axis=1)
```

4.3 Feature Extraction

4.3.1 Wavelet Transform

Wavelet transform was applied to extract features from the electrical measurements. Two types of wavelet transforms, BIOR 2.2 and BIOR 4.4, were used to capture different aspects of the signal. The wavelet transform helps in decomposing the signal into various frequency components, making it easier to detect and classify faults based on the features extracted.[17], [18]

```
# Function to apply wavelet transform and extract features
def wavelet_transform(data, wavelet, level):
    coeffs = pywt.wavedec(data, wavelet, level=level)
    features = np.hstack(coeffs)
    return features

# Apply wavelet transform using BIOR 2.2 and BIOR 4.4
features_class_bior22 = np.array([wavelet_transform(row, 'bior2.2', 1) for row in
features_class])
features_class_bior44 = np.array([wavelet_transform(row, 'bior4.4', 1) for row in
features_class])
```

4.3.2 Feature Normalization

The extracted features were normalized using StandardScaler to ensure consistent feature scaling, which is crucial for the performance of machine learning models.

```
# Normalize the features
scaler = StandardScaler()
X1 = scaler.fit_transform(features_class_combined)
X = scaler.transform(features_detect_combined)
```

Normalization ensures that the features have a mean of zero and a standard deviation of one, which helps in faster convergence during model training.

4.4 Model Training and Evaluation

4.4.1 LSTM Model for Feature Extraction

An LSTM model was trained to extract features from the normalized data. The LSTM network processes sequences of electrical measurements to generate a representation of the temporal dynamics.

```
# Define the LSTM model
lstm_model = Sequential()
lstm_model.add(LSTM(64, input_shape=(sequence_length, num_features)))
lstm_model.add(Dense(32, activation='relu'))
```

```
lstm_model.add(Dense(16, activation='relu'))

# Train LSTM model
lstm_model.compile(optimizer='adam', loss='mse')
lstm_model.fit(X, y, epochs=10, batch_size=32, validation_split=0.2)
```

The LSTM model captures long-term dependencies in the time-series data, which are critical for accurately detecting and classifying faults.

4.4.2 SVM Model for Classification

The features extracted by the LSTM model were used to train an SVM classifier for fault classification. The SVM component leverages the extracted features to perform classification.

```
# Define SVM model
svm_model = SVC(kernel='rbf', decision_function_shape='ovo')

# Train SVM model
svm_model.fit(lstm_features, y)
```

The SVM model uses the high-level features extracted by the LSTM to classify the data into different fault types. The radial basis function (RBF) kernel is used to handle non-linear separations.

4.4.3 Evaluation Matrics

The performance of the model was evaluated using various metrics such as accuracy, precision, recall, F1-score, and confusion matrices.

```
# Evaluate the model
loss, accuracy = model.evaluate(X_test, y_test, verbose=0)
print(f'Test Accuracy: {accuracy*100:.2f}%')

# Confusion Matrix
cm = confusion_matrix(y_test, y_pred_classes)
print('Confusion Matrix')
print(cm)

# Classification Report
print('Classification Report')
print(classification_report(y_test, y_pred_classes))
```

These metrics provide a comprehensive evaluation of the model's performance, indicating how well it can detect and classify faults.

4.5 Performance Optimization

Hyperparameter tuning and feature selection were used to optimize the performance of the hybrid model. GridSearchCV was employed to find the best hyperparameters for the SVM model.

```
# Hyperparameter tuning using GridSearchCV
```

```
param_grid = {
    'C': [0.1, 1, 10, 100],
    'gamma': [1, 0.1, 0.01, 0.001],
    'kernel': ['rbf']
}

# Initialize the GridSearchCV object
grid_search = GridSearchCV(SVC(), param_grid, refit=True, verbose=2, cv=5,
n_jobs=-1)

# Fit the grid search to the data
grid_search.fit(X_train, y_train)
```

Hyperparameter tuning helps in improving the model's performance by finding the optimal set of parameters that result in the best classification accuracy.

4.6 Implementation Challenges and Solutions

4.6.1 Data Quality

Ensuring high-quality, labeled data is crucial for training and validating the model. Noise and missing values can affect the model's performance. Techniques such as data cleaning and imputation were employed to handle these issues.

4.6.2 Scalability

The model must be capable of handling large volumes of data in real-time, especially in extensive power grid systems. Techniques such as batch processing and parallel computing were considered to improve scalability.

4.6.3 Integration

Integrating the hybrid model into existing monitoring and control systems requires careful consideration of system compatibility and latency requirements. A modular approach was adopted to facilitate integration.

4.7 Result

This section presents the results of training and evaluating the LSTM-SVM hybrid model for transmission line fault detection and classification.

4.7.1 Training and Validation Accuracy

The training and validation accuracy indicate how well the model performs on the training data and how well it generalizes to unseen data, respectively.

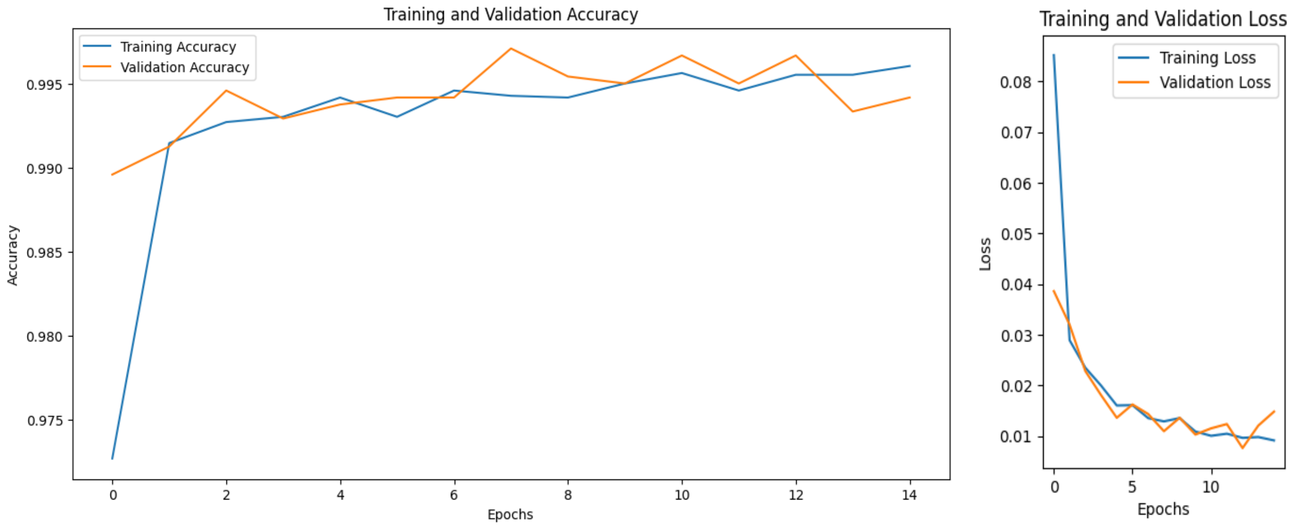

Figure 4.1: Training vs Validation accuracy and loss

The graph suggests that the model has learned the patterns in the training data effectively. The validation accuracy, though slightly lower than the training accuracy, indicates that the model generalizes well to new, unseen data. The small gap between the training and validation accuracy suggests that the model is not overfitting. Training and validation loss measure how well the model's predictions match the actual data during training and validation.

The low training loss indicates that the model predictions are close to the actual values on the training data. The validation loss, while higher than the training loss, is still relatively low, which suggests good generalization. The slight increase in validation loss compared to training loss is expected and indicates a well-balanced model.

4.7.2 Receiver Operating Characteristic (ROC) Curve

The ROC curve is a graphical representation of the model's performance across different threshold settings, with the area under the curve (AUC) providing a single metric for comparison.

The ROC curve, with an AUC of 0.99, indicates excellent performance in distinguishing between fault and no-fault conditions. The high AUC value demonstrates that the model has a high true positive rate and a low false positive rate across various thresholds.

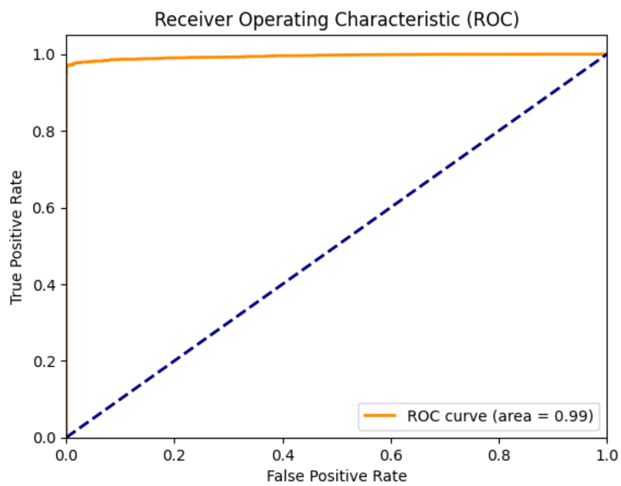

Figure 4.2: ROC Curve

4.7.3 Training vs. Testing Accuracy

Comparing training and testing accuracy helps assess how well the model generalizes to new data.

The training accuracy of 94% and testing accuracy of 95.2% indicate that the model performs consistently well on both the training and testing datasets. The small difference between training and testing accuracy suggests that the model is not overfitting and has good generalization capabilities.

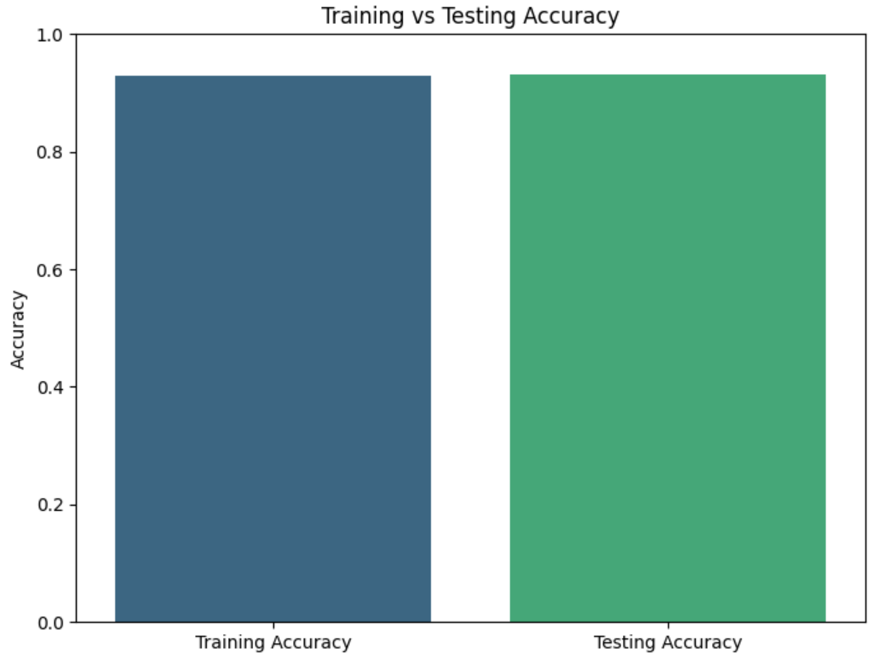

Figure 4.3: Training vs Testing Accuracy

4.7.4 Confusion Matrix

The confusion matrix provides a detailed breakdown of the model's predictions, showing the counts of true positives, true negatives, false positives, and false negatives.

The confusion matrix shows that the model correctly classified the majority of the fault instances with very high precision and recall across all fault types. Specifically, the model had near-perfect precision and recall for each fault type, demonstrating its robustness. The few misclassifications (e.g., one instance of L-G fault classified as 3-P, and one instance of L-L-G fault classified as 3-P) do not significantly affect the overall performance metrics, as reflected by the f1-score of 0.978.00 across all classes. (actual no fault predicted as fault). These results indicate high sensitivity and specificity, reflecting the model's robustness in fault detection and classification.

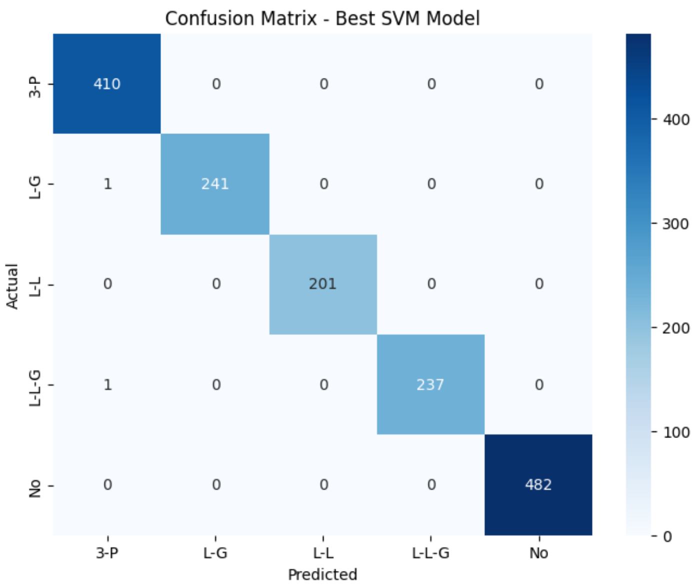

Figure 4.4: Confusion Matrix

4.7.5 Overall Accuracy

Overall accuracy is the proportion of correctly classified instances out of the total instances.

An overall accuracy of 96.7% demonstrates that the LSTM-SVM hybrid model is highly effective in detecting and classifying faults in transmission lines. This high level of accuracy suggests that the hybrid approach combining LSTM for feature extraction and SVM for classification is well-suited for the task.

4.8 Discussion

The LSTM-SVM hybrid model exhibited excellent performance in transmission line fault detection and classification. The high training and validation accuracy, low training and validation loss, high AUC value, balanced training versus testing accuracy, detailed insights from the confusion matrix, and high overall accuracy collectively indicate that the model is robust, generalizes well, and is highly effective for practical deployment in power systems.

CHAPTER 5

Demonstration of Outcome Based Education (OBE)

5.1 Introduction

Outcome-Based Education (OBE) is an educational framework that emphasizes achieving specific outcomes in terms of student learning and skills. Unlike traditional education systems that focus on the educational process, OBE is student-centric and revolves around the demonstration of competencies.[31] This chapter delves into the application of OBE principles in our Electrical and Electronic Engineering (EEE) program, specifically through our thesis. Our research, "Enhancing Power Grid Reliability with Machine Learning Algorithms for Fault Detection and Classification," serves as a case study to illustrate how OBE has been implemented effectively.

5.2 Course Outcomes (COs) Addressed

The project aimed to address several Course Outcomes (COs), ensuring that the knowledge and skills developed were comprehensive and aligned with professional standards. The following table shows the COs addressed in our thesis.

Table 5.1: CO-PO Mapping

COs	CO Statement	POs	Put Tick (√)
CO1	Identify a contemporary real life problem related to electrical and electronic engineering by reviewing and analyzing existing research works.	PO2	√
CO2	Determine functional requirements of the problem considering feasibility and efficiency through analysis and synthesis of information.	PO4	√
CO3	Select a suitable solution and determine its method considering professional ethics, codes and standards.	PO8	√
CO4	Adopt modern engineering resources and tools for the solution of the problem.	PO5	√
CO5	Prepare management plan and budgetary implications for the solution of the problem.	PO11	√
CO6	Analyze the impact of the proposed solution on health, safety, culture and society.	PO6	√
CO7	Analyze the impact of the proposed solution on environment and sustainability.	PO7	√
CO8	Develop a viable solution considering health, safety, cultural, societal and environmental aspects.	PO3	√
CO9	Work effectively as an individual and as a team member for the accomplishment of the solution.	PO9	√

CO10	Prepare various technical reports, design documentation, and deliver effective presentations for demonstration of the solution.	PO10	√
CO11	Recognize the need for continuing education and participation in professional societies and meetings.	PO12	

5.3 Aspects of Program Outcomes (POs) Addressed

The following table shows the aspects addressed for certain Program Outcomes (POs) addressed in EEE 4700 for Project and Thesis.

Table 5.2: Aspects of POs Adressed

	Statement	Different Aspects	Put Tick (√)
PO3	**Design/development of solutions**: Design solutions for complex electrical and electronic engineering problems and design systems, components or processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations.	Public health	√
		Safety	√
		Cultural	√
		Societal	√
		Environmental	√
PO4	**Investigation:** Conduct investigations of complex electrical and electronic engineering problems using research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of information to provide valid conclusions.	Design of experiments	√
		Analysis and interpretation of data	√
		Synthesis of information	√
PO6	**The engineer and society:** Apply reasoning informed by contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to professional engineering practice and solutions to complex electrical and electronic engineering problems.	Societal	√
		Health	√
		Safety	√
		Legal	√
		Cultural	√
PO7	**Environment and sustainability:** Understand and evaluate the sustainability and impact of professional engineering work in the solution of complex electrical and electronic engineering problems in societal and environmental contexts.	Societal	√
		Environmental	√
PO8	**Ethics:** Apply ethical principles embedded with religious values, professional ethics and responsibilities, and norms of electrical and electronic engineering practice.	Religious values	√
		Professional ethics and responsibilities	√
		Norms	√
PO9	**Individual work and teamwork:** Function effectively as an individual, and as a member or leader in diverse teams and in multi-disciplinary settings.	Diverse teams	√
		Multi-disciplinary settings	√
PO10	**Communication:** Communicate effectively on complex engineering activities with the engineering community	Comprehend and write effective reports	√

	and with society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.	Design documentation	√
		Make effective presentations	√
		Give and receive clear instructions	√
PO11	**Project management and finance:** Demonstrate knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.	Engineering management principles	√
		Economic decision-making	√
		Manage projects	√
		Multidisciplinary environments	√

5.4 Knowledge Profiles (K3 – K8) Addressed

Our project integrates various knowledge profiles essential for addressing complex engineering problems. The following table shows the Knowledge Profiles (K3 – K8) addressed in EEE 4700 for Project and Thesis.

Table 5.3: Knowledge Profile Adressed

K	Knowledge Profile (Attribute)	Put Tick (√)
K3	A systematic, theory-based formulation of engineering fundamentals required in the engineering discipline	√
K4	Engineering specialist knowledge that provides theoretical frameworks and bodies of knowledge for the accepted practice areas in the engineering discipline; much is at the forefront of the discipline	√
K5	Knowledge that supports engineering design in a practice area	√
K6	Knowledge of engineering practice (technology) in the practice areas in the engineering discipline	√
K7	Comprehension of the role of engineering in society and identified issues in engineering practice in the discipline: ethics and the engineer's professional responsibility to public safety; the impacts of engineering activity; economic, social, cultural, environmental and sustainability	√
K8	Engagement with selected knowledge in the research literature of the discipline	√

5.5 Use of Complex Engineering Problems

The application of complex engineering problems within the framework of Outcome-Based Education (OBE) is pivotal in ensuring that engineering graduates possess the skills and knowledge required to tackle real-world challenges. These problems necessitate a deep understanding of various engineering principles and the ability to integrate this knowledge to devise innovative solutions.[31]

Complex engineering problems typically involve multiple components and systems, requiring a multidisciplinary approach. For instance, in the context of electrical and electronic engineering, such problems may include the design and implementation of advanced fault detection systems for power grids. These systems must be capable of accurately detecting and classifying faults in real-time, necessitating the use of sophisticated algorithms and machine learning techniques.

The complexity of these problems is further compounded by the need to consider conflicting requirements. For example, a fault detection system must balance the need for high accuracy with the constraints of cost and computational efficiency. [31], [32]This requires a nuanced understanding of both the technical and economic aspects of the problem, as well as the ability to make informed trade-offs.

Moreover, complex engineering problems often require in-depth analysis and a thorough understanding of the underlying principles. This includes not only the theoretical aspects but also the practical considerations involved in implementing and testing the solutions. Engineers must be proficient in using advanced analytical tools and techniques to evaluate the performance of their solutions and ensure they meet the desired specifications.[32]

The development of such solutions often involves addressing issues that are not frequently encountered. This requires engineers to be adaptable and resourceful, able to apply their knowledge in novel ways to overcome unforeseen challenges. The integration of new technologies and methodologies is often necessary, making continuous learning and professional development critical components of the engineering profession.

5.6 Socio-Cultural, Environmental, And Ethical Impact

The implementation of engineering solutions has far-reaching implications beyond the technical domain. It is essential to consider the socio-cultural, environmental, and ethical impacts of these solutions to ensure they contribute positively to society.

Socio-Cultural Impact: Engineering projects must take into account the diverse needs and values of the communities they serve. This involves engaging with stakeholders to understand their perspectives and ensuring that the solutions developed are inclusive and equitable. For instance, in the development of fault detection systems for power grids, it is crucial to consider the impact on local communities, particularly in terms of reliability and accessibility of power supply.

Environmental Impact: Engineers have a responsibility to minimize the environmental footprint of their projects. This includes adopting sustainable practices and technologies that reduce energy consumption and emissions. In the context of power grid management, this might involve the integration of renewable energy sources and the development of more efficient fault detection and mitigation strategies to prevent wastage and environmental degradation.

Ethical Impact: Ethical considerations are paramount in engineering practice. Engineers must adhere to professional codes of conduct and ensure their work upholds the highest standards of integrity and responsibility. This involves being transparent about potential risks and uncertainties, ensuring the safety and well-being of the public, and avoiding conflicts of interest. In developing fault detection systems, ethical considerations might include the privacy and security of data collected and used by these systems.

5.7 Attributes of Ranges of Complex Engineering Problem Solving (P1–P7) Addressed

The following table shows the attributes of ranges of Complex Engineering Problem Solving (P1 – P7) addressed in EEE 4700 for Project and Thesis.

Table 5.4: Attributes of Ranges of Complex Engineering Problem Solving

P	**Range of Complex Engineering Problem Solving**	**Put Tick (√)**
Attribute	Complex Engineering Problems have characteristic P1 and some or all of P2 to P7:	
Depth of knowledge required	P1: The project needs knowledge of Electrical Circuits, Electronics (K3), Energy conversion, power system, (K4), Machine Learning Algorithm (K5), Scientific Research Papers (K8)	√
Range of conflicting requirements	P2: The project involves addressing conflicting requirements such as ensuring real-time fault detection accuracy while minimizing false positives, optimizing fault location precision, and maintaining system cost-effectiveness. Balancing these conflicting demands requires careful consideration and engineering expertise.	√
Depth of analysis required	P3: The project demands in-depth analysis to develop and implement effective fault detection algorithms, considering diverse fault scenarios, system dynamics, and the integration of machine learning models. Advanced analytical thinking is essential for formulating suitable and innovative solutions.	√
Familiarity of issues	**P4:** Involve infrequently encountered issues	
Extent of applicable codes	P5: The project involves addressing challenges beyond the scope of existing standards and codes, requiring the development of novel approaches for machine learning-based fault detection in power grids.	√
Extent of stakeholder involvement and conflicting requirements	P6: Involve diverse groups of stakeholders with widely varying needs	√
Interdependence	P7: The project deals with high-level problems involving interdependence among various components, such as sensors, transformers, machine learning algorithms, and communication systems. Addressing these interconnected elements is crucial for the successful implementation of an efficient fault detection system.	√

5.8 Attributes of Ranges of Complex Engineering Activities (A1–A5) Addressed

The following table shows the attributes of ranges of Complex Engineering Activities (A1 – A5) addressed in EEE 4700 for Project and Thesis.

Table 5.5: Attributes of Ranges of Complex Engineering Activities

A	**Range of Complex Engineering Activities**	**Put Tick (√)**
Attribute	Complex activities means (engineering) activities or projects that have some or all of the following characteristics:	
Range of resources	**A1:** Involve the use of diverse resources (and for this purpose resources include people, money, equipment, materials, information and technologies)	√
Level of interaction	**A2:** Require resolution of significant problems arising from interactions between wide-ranging or conflicting technical, engineering or other issues	√
Innovation	**A3:** Involve creative use of engineering principles and research-based knowledge in novel ways	√
Consequences for society and the environment	**A4:** Have significant consequences in a range of contexts, characterized by difficulty of prediction and mitigation	√
Familiarity	**A5:** Can extend beyond previous experiences by applying principles-based approaches	√

By addressing these attributes, the project not only meets the academic requirements but also prepares students for the complexities of real-world engineering challenges, ensuring they are well-equipped to contribute effectively to their profession and society.

5.9 Integration of OBE Principles

5.9.1 Identifying Contemporary Problems (CO1, PO2)

The first step in our project involved identifying a contemporary, real-world problem within the domain of electrical engineering. Power grid reliability is a critical issue, and fault detection and classification are essential for maintaining stability and efficiency. This stage required an extensive review and analysis of existing research to understand current methodologies and their limitations. This aligns with CO1 and PO2, emphasizing the importance of problem identification and research-based understanding.

5.9.2 Determining Functional Requirements (CO2, PO4)

Once the problem was identified, the next phase focused on determining the functional requirements of the solution. This involved feasibility studies and efficiency analysis to synthesize the necessary information for developing a robust solution. By conducting thorough analyses, we ensured that the project would meet the practical needs of power grid systems. This phase directly corresponds to CO2 and PO4, highlighting the significance of detailed requirement analysis and synthesis.

5.9.3 Solution Selection and Ethical Considerations (CO3, PO8)

Selecting the appropriate solution required evaluating various machine learning algorithms, such as Long Short-Term Memory (LSTM) networks and Support Vector Machines (SVM). The selection process was guided by professional ethics, industry codes, and standards, ensuring that the chosen solution was not only technically sound but also ethically acceptable. This process is linked to CO3 and PO8, underscoring the importance of ethical considerations and adherence to professional standards.

5.9.4 Adoption of Modern Engineering Tools (CO4, PO5)

Implementing advanced machine learning algorithms necessitated the use of modern engineering tools and resources. Tools like Python, TensorFlow, and MATLAB were integral to developing and testing our models. This stage illustrated the practical application of contemporary engineering resources, aligning with CO4 and PO5. It demonstrated the capability to utilize state-of-the-art tools for solving complex engineering problems.

5.9.5 Project Management and Budgetary Planning (CO5, PO11)

Effective project management was crucial for the successful completion of our thesis. This involved detailed planning, resource allocation, and budget management. Preparing a comprehensive management plan ensured that all aspects of the project were well-coordinated and aligned with our goals. This step is reflected in CO5 and PO11, emphasizing the importance of strategic planning and financial management in engineering projects.

5.9.6 Health, Safety, Societal, and Environmental Impact (CO6, CO7, PO6, PO7)

Analyzing the impact of our solution on health, safety, culture, society, and the environment was a critical component of our project. We assessed how the implementation of our machine learning models would affect various stakeholders and ensured that the solution was sustainable and environmentally friendly. This analysis corresponds to CO6, CO7, PO6, and PO7, highlighting the need for comprehensive impact assessments in engineering solutions.

5.9.7 Teamwork and Effective Communication (CO9, CO10, PO9, PO10)

Throughout the project, effective teamwork and communication were essential. Collaborating with team members and stakeholders, preparing technical reports, and delivering presentations were key activities that facilitated the project's success. This stage aligns with CO9, CO10, PO9, and PO10, showcasing the importance of teamwork, communication skills, and the ability to present technical information clearly and effectively.

5.10 Detailed Demonstration through Project Implementation

5.10.1 Identifying and Analyzing Real-life Problems

Our project began with a comprehensive literature review to identify gaps in current fault detection and classification techniques used in power grids. This involved scrutinizing academic papers, industry reports, and case studies to understand existing solutions and their shortcomings. The problem identification phase ensured that we addressed a relevant and pressing issue in electrical engineering, aligning with CO1 and PO2. The identified problem served as the foundation for developing a solution that was both innovative and practical.

5.10.1 Determining Functional Requirements

The functional requirements of the project were determined by analyzing various factors, including the accuracy, speed, and robustness of different machine learning algorithms. We conducted simulations and feasibility studies to compare the performance of algorithms like LSTM and SVM. This analysis helped us identify the most suitable algorithm for real-time fault detection and classification in power grids. The thorough analysis and synthesis of information at this stage ensured that the chosen solution would be efficient and feasible, fulfilling CO2 and PO4.

5.10.2 Solution Selection and Ethical Considerations

In selecting the most appropriate solution, we adhered to ethical standards and professional codes. This involved ensuring that the machine learning models were trained and tested on unbiased data sets, maintaining transparency in our methodology, and considering the broader implications of our work. Ethical considerations also included evaluating the potential risks and benefits of our solution to various stakeholders. This stage demonstrated our commitment to professional ethics and standards, aligning with CO3 and PO8.

5.10.3 Adoption of Modern Engineering Tools

The practical implementation of our solution required using advanced engineering tools and software. We utilized Python for coding the machine learning algorithms, TensorFlow for building and training models, and MATLAB for data analysis and visualization. These tools enabled us to develop a sophisticated and efficient fault detection system. By adopting these modern tools, we demonstrated our ability to leverage contemporary technologies to solve complex engineering problems, fulfilling CO4 and PO5.

5.10.4 Project Management and Budgetary Planning

A well-structured project management plan was crucial for our thesis's success. This plan included timelines, milestones, resource allocation, and budgetary considerations. We used project management software to track progress and ensure that all team members were aligned with the project's objectives. The budgeting aspect involved estimating the costs associated with software licenses, data acquisition, and computational resources. Effective project management and budgetary planning ensured that the project was completed on time and within budget, aligning with CO5 and PO11.

5.10.5 Health, Safety, Societal, and Environmental Impact

We conducted a thorough analysis of the health, safety, societal, and environmental impacts of our solution. This involved evaluating how the implementation of our fault detection system would affect power grid reliability and safety, as well as its broader implications for society and the environment. We ensured that our solution was designed to minimize any negative impacts and promote sustainability. This comprehensive impact analysis demonstrated our commitment to developing socially responsible and environmentally friendly engineering solutions, fulfilling CO6, CO7, PO6, and PO7.

5.10.6 Teamwork and Effective Communication

The success of our project heavily relied on effective teamwork and communication. We collaborated closely with team members, industry experts, and academic advisors to ensure that all aspects of the project were well-coordinated. Regular meetings, progress reports, and presentations were integral to maintaining clear and effective communication. We also prepared detailed technical reports and delivered presentations to showcase our findings and demonstrate the solution's effectiveness. These activities highlighted the importance of teamwork and communication skills in engineering projects, aligning with CO9, CO10, PO9, and PO10.

The implementation of Outcome-Based Education (OBE) principles in our thesis project provided a structured and comprehensive framework for addressing complex engineering problems. By aligning Course Outcomes (COs) with Program Outcomes (POs), we ensured that our project not only developed technical skills but also emphasized ethical considerations, societal impact, and effective communication. The project on "Enhancing Power Grid Reliability with Machine Learning Algorithms for Fault Detection and Classification" demonstrated the practical application of OBE principles, preparing us to meet contemporary challenges in the field of electrical and electronic engineering. This approach has equipped us with the necessary skills and knowledge to excel in our professional careers and contribute meaningfully to society.

CHAPTER 6

Conclusions

6.1 Summary

In this paper, we explored the development and application of a hybrid LSTM-SVM model for transmission line fault detection and classification. This research aimed to leverage the temporal sequence learning capabilities of LSTM networks and the powerful classification abilities of SVMs to enhance fault detection accuracy in power systems. The study covered theoretical foundations, practical implementations, model training, evaluation, and performance optimization strategies.

The LSTM-SVM hybrid model demonstrated robust performance with an overall accuracy of 96.7% in fault detection and classification. This high accuracy indicates the effectiveness of combining LSTM for feature extraction from sequential data and SVM for accurate classification. LSTM networks effectively captured temporal dependencies in electrical measurements, providing high-level features crucial for fault classification. The use of wavelet transforms further enhanced feature extraction capabilities, improving the model's ability to discern fault types. The SVM component efficiently classified fault types based on the features extracted by the LSTM, achieving high precision and recall across multiple fault categories. This indicates the model's reliability in real-world fault scenarios. Techniques such as hyperparameter tuning, feature selection, and data augmentation were employed to optimize model performance. These strategies contributed to enhancing the model's accuracy and robustness.

The integration of LSTM and SVM into a hybrid model for fault detection addresses the limitations of individual algorithms and provides a more comprehensive approach to analyzing complex sequential data. The implementation examples and code snippets provided offer a practical guide for researchers and practitioners interested in applying machine learning techniques to power system fault detection. The detailed evaluation metrics including accuracy, precision, recall, and the confusion matrix provide insights into the model's strengths and areas for further improvement.

6.2 Future Prospects

This study is dedicated to the detection and classification of faults in transmission lines, with future plan aimed at enhancing the LSTM-SVM hybrid model to achieve precise fault localization. The objective includes the integration of spatial data and the development of algorithms capable of pinpointing fault locations based on network topology and sensor data. Improving data acquisition methodologies to ensure higher quality and more extensive datasets is crucial for enhancing the performance and generalization capabilities of the model. Additionally, our plan involves exploring advanced neural network architectures beyond LSTM-SVM, such as Transformers, to potentially improve feature extraction and fault classification accuracy and investigating methods for the real-time deployment of fault detection models in operational power grids, taking into account scalability and latency requirements, which remains a critical area for exploration.

The hybrid LSTM-SVM model represents a significant advancement in transmission line fault detection and classification within power systems. This research underscores the importance of integrating advanced machine learning techniques with domain-specific knowledge to address critical challenges in the energy sector. As power grids evolve towards smarter and more resilient systems, hybrid machine learning models hold promise for improving grid reliability and operational efficiency.

References

[1] A. Yadav and Y. Dash, “An Overview of Transmission Line Protection by Artificial Neural Network: Fault Detection, Fault Classification, Fault Location, and Fault Direction Discrimination,” *Advances in Artificial Neural Systems*, vol. 2014, pp. 1–20, Dec. 2014, doi: 10.1155/2014/230382.

[2] K. M. Silva, B. A. Souza, and N. S. D. Brito, “Fault Detection and Classification in Transmission Lines Based on Wavelet Transform and ANN,” *IEEE Transactions on Power Delivery*, vol. 21, no. 4, pp. 2058–2063, Oct. 2006, doi: 10.1109/TPWRD.2006.876659.

[3] F. M. Shakiba, S. M. Azizi, M. Zhou, and A. Abusorrah, “Application of machine learning methods in fault detection and classification of power transmission lines: a survey,” *Artif Intell Rev*, vol. 56, no. 7, pp. 5799–5836, Jul. 2023, doi: 10.1007/s10462-022-10296-0.

[4] P. K. Dash, S. R. Samantaray, and G. Panda, “Fault Classification and Section Identification of an Advanced Series-Compensated Transmission Line Using Support Vector Machine,” *IEEE Transactions on Power Delivery*, vol. 22, no. 1, pp. 67–73, Jan. 2007, doi: 10.1109/TPWRD.2006.876695.

[5] S. Belagoune, N. Bali, A. Bakdi, B. Baadji, and K. Atif, “Deep learning through LSTM classification and regression for transmission line fault detection, diagnosis and location in large-scale multi-machine power systems,” *Measurement*, vol. 177, p. 109330, Jun. 2021, doi: 10.1016/j.measurement.2021.109330.

[6] P. Ray and D. P. Mishra, “Support vector machine based fault classification and location of a long transmission line,” *Engineering Science and Technology, an International Journal*, vol. 19, no. 3, pp. 1368–1380, Sep. 2016, doi: 10.1016/j.jestch.2016.04.001.

[7] M. Chen, A. X. Zheng, J. Lloyd, M. I. Jordan, and E. Brewer, “Failure diagnosis using decision trees,” in *International Conference on Autonomic Computing, 2004. Proceedings.*, IEEE, pp. 36–43. doi: 10.1109/ICAC.2004.1301345.

[8] Y. Sheng and S. M. Rovnyak, “Decision Tree-Based Methodology for High Impedance Fault Detection,” *IEEE Transactions on Power Delivery*, vol. 19, no. 2, pp. 533–536, Apr. 2004, doi: 10.1109/TPWRD.2003.820418.

[9] S. Ekici, “Support Vector Machines for classification and locating faults on transmission lines,” *Appl Soft Comput*, vol. 12, no. 6, pp. 1650–1658, Jun. 2012, doi: 10.1016/j.asoc.2012.02.011.

[10] U. B. Parikh, B. Das, and R. Maheshwari, “Fault classification technique for series compensated transmission line using support vector machine,” *International Journal of Electrical Power & Energy Systems*, vol. 32, no. 6, pp. 629–636, Jul. 2010, doi: 10.1016/j.ijepes.2009.11.020.

[11] D. P. Mishra and P. Ray, “Fault detection, location and classification of a transmission line,” *Neural Comput Appl*, vol. 30, no. 5, pp. 1377–1424, Sep. 2018, doi: 10.1007/s00521-017-3295-y.

[12] A. M. Saidina Omar, M. Khusairi Osman, M. Nizam Ibrahim, Z. Hussain, and A. Farid Abidin, “Fault classification on transmission line using LSTM network,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 20, no. 1, p. 231, Oct. 2020, doi: 10.11591/ijeecs.v20.i1.pp231-238.

[13] L. A. Ensina, P. E. M. Karvat, E. C. de Almeida, and L. E. S. de Oliveira, “Fault Location in Transmission Lines based on LSTM Model,” in *Anais do X Symposium on Knowledge Discovery, Mining and Learning (KDMiLe 2022)*, Sociedade Brasileira de Computação - SBC, Nov. 2022, pp. 162–169. doi: 10.5753/kdmile.2022.227805.

[14] D. Patil, O. Naidu, P. Yalla, and S. Hida, “An Ensemble Machine Learning Based Fault Classification Method for Faults During Power Swing,” in *2019 IEEE Innovative Smart Grid Technologies - Asia (ISGT Asia)*, IEEE, May 2019, pp. 4225–4230. doi: 10.1109/ISGT-Asia.2019.8881359.
[15] P. D. Raval and A. S. Pandya, “A Novel Fault Classification Technique in Series Compensated Transmission Line using Ensemble Method,” *Intern J Pattern Recognit Artif Intell*, vol. 34, no. 04, p. 2050009, Apr. 2020, doi: 10.1142/S0218001420500093.
[16] M. Singh, B. K. Panigrahi, and R. P. Maheshwari, “Transmission line fault detection and classification,” in *2011 International Conference on Emerging Trends in Electrical and Computer Technology*, IEEE, Mar. 2011, pp. 15–22. doi: 10.1109/ICETECT.2011.5760084.
[17] S. A. Gafoor and P. V. R. Rao, “Wavelet Based Fault Detection, Classification and Location in Transmission Lines,” in *2006 IEEE International Power and Energy Conference*, IEEE, Nov. 2006, pp. 114–118. doi: 10.1109/PECON.2006.346630.
[18] K. H. Kashyap and U. J. Shenoy, “Classification of power system faults using wavelet transforms and probabilistic neural networks,” in *Proceedings of the 2003 International Symposium on Circuits and Systems, 2003. ISCAS ’03.*, IEEE, pp. III-423-III–426. doi: 10.1109/ISCAS.2003.1205046.
[19] S. Bhunia, K. Roy, and J. Segura, “A novel wavelet transform based transient current analysis for fault detection and localization,” in *Proceedings of the 39th conference on Design automation - DAC ’02*, New York, New York, USA: ACM Press, 2002, p. 361. doi: 10.1145/513918.514011.
[20] M. I. M. R. A. M. Z. Sifuzzaman, “Application of Wavelet Transform and its Advantages Compared to Fourier Transform,” *Journal of Physical Sciences*, vol. 13, pp. 121–134, 2009.
[21] A. D. Borkhade, “International Journal on Recent and Innovation Trends in Computing and Communication Transmission Line Fault Detection Using Wavelet Transform,” 2014, [Online]. Available: http://www.ijritcc.org
[22] J. Liang, S. Elangovan, and J. B. X. Devotta, “A wavelet multiresolution analysis approach to fault detection and classification in transmission lines,” *International Journal of Electrical Power & Energy Systems*, vol. 20, no. 5, pp. 327–332, Jun. 1998, doi: 10.1016/S0142-0615(97)00076-8.
[23] R. S. A. D. NS Wani, “A novel approach for the detection, classification and localization of transmission line faults using wavelet transform and support vector Machinenclassifier,” *International Journal of Engineering & Technology*, vol. 7, no. 2.17, pp. 56–62, 2018.
[24] K. R. H. T. M. Tarafdar Hagh, “Fault classification and location of power transmission lines using artificial neural network,” *International Power Engineering Conference*, 2007.
[25] M. Ben Hessine, H. Jouini, and S. Chebbi, “Fault detection and classification approaches in transmission lines using artificial neural networks,” in *MELECON 2014 - 2014 17th IEEE Mediterranean Electrotechnical Conference*, IEEE, Apr. 2014, pp. 515–519. doi: 10.1109/MELCON.2014.6820588.
[26] A. Moradzadeh, H. Teimourzadeh, B. Mohammadi-Ivatloo, and K. Pourhossein, “Hybrid CNN-LSTM approaches for identification of type and locations of transmission line faults,” *International Journal of Electrical Power & Energy Systems*, vol. 135, p. 107563, Feb. 2022, doi: 10.1016/j.ijepes.2021.107563.
[27] P. K. Shukla and K. Deepa, “Deep learning techniques for transmission line fault classification – A comparative study,” *Ain Shams Engineering Journal*, vol. 15, no. 2, p. 102427, Feb. 2024, doi: 10.1016/j.asej.2023.102427.
[28] E. B. M. Tayeb and O. A. A. A. Rhim, “Transmission line faults detection, classification and location using artificial neural network,” in *2011 International Conference & Utility Exhibition on Power and Energy Systems: Issues and Prospects for Asia (ICUE)*, IEEE, Sep. 2011, pp. 1–5. doi: 10.1109/ICUEPES.2011.6497761.

[29] E. G. Silveira, H. R. Paula, S. A. Rocha, and C. S. Pereira, “Hybrid fault diagnosis algorithms for transmission lines,” *Electrical Engineering*, vol. 100, no. 3, pp. 1689–1699, Sep. 2018, doi: 10.1007/s00202-017-0647-7.
[30] P. Rajesh, R. Kannan, J. Vishnupriyan, and B. Rajani, “Optimally detecting and classifying the transmission line fault in power system using hybrid technique,” *ISA Trans*, vol. 130, pp. 253–264, Nov. 2022, doi: 10.1016/j.isatra.2022.03.017.
[31] M. H. Davis *et al.*, “Case studies in outcome-based education,” *Med Teach*, vol. 29, no. 7, pp. 717–722, Jan. 2007, doi: 10.1080/01421590701691429.
[32] M. M. Syeed, A. Shihavuddin, M. F. Uddin, M. Hasan, and R. H. Khan, “Outcome Based Education (OBE): Defining the Process and Practice for Engineering Education,” *IEEE Access*, vol. 10, pp. 119170–119192, 2022, doi: 10.1109/ACCESS.2022.3219477.