Ministry of Higher Education
& Scientific Research
University of Babylon
College of Science for Women
Dept. of Computer Science

IoT-Based Monitoring and Detection System for Epileptic Seizures Using Machine Learning

A Thesis

Submitted to the Council of College of Science for Women\University of Babylon in a Partial Fulfillment of the Requirements for the Degree of Master in Science\ Computer Sciences

Submitted By

Alaa Lateef Noor Al-Hajjar

Supervised By

Asst. Prof. Dr. Ali Kadhum M. Al-Quraby

2023 A.D. 1445 A.H.

بِســــــم اللَّهِ الرَّحْمَٰنِ الرَّحِيمِ

﴿ قَالُواْ سُبْحَانَكَ لاَ عِلْمَ لَنَآ إِلاَّ مَا عَلَّمْتَنَآ إِنَّكَ أَنْتَ العليم الحكيم ﴾

صدق الله العلي العظيم

[سورة البقرة : الآية 32]

SUPERVISORS CERTIFICATION

I certify that the thesis entitled "**IoT-Based Monitoring and Detection System For Epileptic Seizures Using Machine Learning** " is prepared by (**Alaa Lateef Noor**) under our supervision at the Department of Computer Science/ College of Science for Women/ University of Babylon as partial fulfillment of the requirements of the degree of Master in Computer Sciences.

Signature:

Name: Asst. Prof. Dr. Ali Kadhum M. Al-Quraby

Date: / /

Address: University of Babylon/College of Science for Women

THE HEAD OF THE DEPARTMENT CERTIFICATION

I certify that the thesis entitled "**IoT-Based Monitoring and Detection System For Epileptic Seizures Using Machine Learning**" is prepared by (**Alaa Lateef Noor**) in the Department of Computer Science/ College of Science for Women/ University of Babylon as partial fulfillment of the requirements of the degree of Master in Computer Sciences.

Signature:

Name: Dr. SAif Al-Alak Head of the Department

Date: / /

Address: University of Babylon/College of Science for Women

DEDICATION

To those who illuminated my life with their love, to those whose prayers did not leave me, **my dear father and mother.**

To my companion on the road and partner of life and success, **my beloved husband**.

To the beat of my heart, the apple of my eye, my three princes**, Hussien, Redha,** and **Zain al-Abidin**.

To the brothers of my soul and the memories of my childhood and youth, **my dear sisters and brothers**

Alaa

ACKNOWLEDGEMENT

First and foremost, I would like to thank my God, Allah Almighty, for giving me endless graces and gave me the ability to continue and helped me to reach this moment.

My deep sense of gratitude to the beacon of science, to the master of creatures, to the greatest Prophet, Mohammed (Peace be upon Him and His Family).

I would like to take this opportunity to express my sincere gratitude and appreciation to my supervisor, Assistant Professor Dr. Ali Kadhum M. Al-Quraby, for his unwavering encouragement, constant support, and the vast amount of information he taught me. His guidance and instructions gave me the knowledge and skills i needed to finish this thesis.

I would also like to acknowledge the University of Babylon - College of Science for Women-Department of Computer Science for providing all kinds of support in carrying out this thesis.

ABSTRACT

The neurological disorder known as epilepsy has an ongoing negative impact on the brain. Identification of seizures is essential to the clinical care of individuals with epilepsy. Expert doctors frequently use visual electroencephalography (EEG) data analysis to detect epileptic seizures which is a method for observing the nonlinear electrical activity of the brain's nerve cells. It is an epilepsy detection diagnostic tool. Since it takes neurologists a long time to review, the manual interpretation of EEG data derived from the recordings of EEG signals from a single patient is a very difficult and time-consuming process.

In this thesis, we suggest an Internet of Things (IoT) framework for precise and effective seizure detection and monitoring for epileptic patients utilizing machine learning techniques. Three layers make up the proposed Internet of Things framework: the Things/Devices, Fog, and Cloud layers. The proposed method is summarized in transmitting the collected data from the thing layer to the FoG layer where a number of critical steps are carried out starting from segmenting the EEG data by converting it into 2-D table format and creating a (Weighted Visibility Graph) WVG from EEG data. Our suggested method extracts nine features from the WVG and an additional ten statistical features from the original EEG dataset. All these features are fed to the machine learning methods to classify the obtained signal as normal or abnormal. Ten of the most common machine learning methods used in this proposed system named as Logistic Regression LR, K-nearest neighbor KNN, Support Vector Machine SVM, Stochastic Gradient Descent SGD, Naïve Bayes NB, Decision Tree DT, Random Forest RF, Extra Tree Classifier EXT, Gradient Boosting GB, and Extreme Gradient Boosting XGB.

After classifying the signal, one of two actions will be taken depending on the classification state: either sending a notification to any predetermined caretaker in case of the occurrence of a seizure or reducing the data by using the threshold-based method in case of the absence of the seizure. As a result, in both cases, the data is uploaded to the cloud layer to be reviewed later by a specialized medical team.

Four scenarios were used to evaluate our proposed method using performance evaluation metrics such as accuracy, precision, F1_score, Specificity, etc. The power of the provided methods is demonstrated by the proposed strategy, which yields a percentage of 100% in the fourth scenario which uses ML models with hyper-parameters tuning, balanced EEG data, and extracted features.

TABLE OF CONTENTS

Contents		**Page No.**

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ACC	Accelerometer
ANNs	Artificial Neural Networks
ASP	Average Shortest Path
AVGD	Graph Average Degree
AVGW	Average weighted degree
BC	Betweeness centrality
CSL	Cloud Storage Layer
CT	Computed Tomography
DCL	Data Collection Layer
DL	Deep Learning
DML	Decision-Making Layer
DT	Decision Tree
DTL	Data Transmission Layer
DWT	Discrete Wavelet Transform
EDA	Electro-dermal activity
EEG	Electroencephalography
ENN	Edited Nearest Neighbor
ES	Epileptic Seizure
EXT	Extra Tree Classifier
FIM	Features Importance Method
GB	Gradient Boosting
GE	Global Efficiency

HVG	Horizontal Visibility Graph
IAL	Information Analysis Layer
IFTTT	If This Then That
IoMT	Internet of Medical Things
IoT	Internet of Things
KNN	k-nearest neighbor
LE	Local Efficiency
LR	Reinforcement Learning
LR	Logistic Regression
MCOR	Matthew Correlation Coefficient
ML	Machine Learning
MRI	Magnetic Resonance Imaging
MRS	Magnetic Resonance Spectroscopy
NB	Naive Bayes
NVG	Natural Visibility Graph
PET	Positron Emission Tomography
RF	Random Forest
ROC	Receiver Operating Characteristic
SCL	Seizure Classification Layer
SGD	Stochastic Gradient Descent
SMOT	Synthetic Minority Oversampling Technique
SVM	Support Vector Machine
VG	Visibility Graph

WVG	Weighted Visibility Graph
XGB	Extreme Gradient Boosting

LIST OF PUBLICATIONS

PUPLISHED PAPERS

- Al-hajjar, Alaa Lateef Noor, and Ali Kadhum M. Al-Qurabat. "An overview of machine learning methods in enabling IoMT-based epileptic seizure detection." *The Journal of Supercomputing* (2023): 1-48. Clarivate (Web of Science) – Q1
- Al-hajjar, Alaa Lateef Noor, and Ali Kadhum M. Al-Qurabat. "Epileptic Seizure Detection Using Feature Importance and ML Classifiers." *Journal of Education for Pure Science-University of Thi-Qar 13.2* (2023).

SUBMITTED PAPERS

- Improving Seizure Identification in Epilepsy Patients Using SMOTE+ENN and Machine Learning in IoMT. *Journal of Ambient Intelligence and Smart Environments*.
- Analyzing the Performance of Weighted Visibility Graph Features and Machine Learning Algorithms for Classifying Epileptic Seizure EEG Signals in IoMT. *Biocybernetics and Biomedical Engineering*.

CHAPTER ONE

GENERAL INTRODUCTION

CHAPTER ONE

GENERAL INTRODUCTION

1.1 Introduction

Over the past several years, there has been a lot of excitement about the Internet of Things (IoT) in the field of healthcare technology, which is called the Internet of Medical Things (IoMT). The healthcare industry is extremely practical, and IoMT offers a wide range of options to improve it. A wide range of modern medical sensors and gadgets may communicate through different networks, giving access to crucial data regarding patients' ailments. Then, by having a better understanding of the symptoms, this information may be utilized to anticipate disease and recovery, monitor patients from a distance, and overall enhance the diagnostic and treatment process by increasing automation and portability. IoMT-based epileptic seizure detection is one particular instance of healthcare industry [1].

One of the most prevalent neurological disorders is epilepsy, which tends to impact the human brain by causing unpredictable and spontaneous seizures. A number of causes have recently been put up. The main factor is the brain's internal electrical activity being disturbed. This could be caused by a number of factors, including abnormalities, low blood sugar, and a shortage of oxygen during childbirth [2][3]. Around 50 million people around the world possess epilepsy, and 100 million have had it at least once throughout their lives [4].

The primary characteristic of this condition is recurrent seizures, which are caused by an electrical imbalance in the brain. Typically, it causes bodily parts to shake and can even result in fainting. The human brain's typical neural activity pattern is disrupted by epilepsy disease, which causes a rise in important clinical symptoms such as unusual sensations, emotions, abnormal behavior, memory loss, etc. They have a lower quality of life and must rely exclusively on their caregivers for the remainder of their lives. To avoid any form of injury or accident caused by an unclear seizure, this seizure must thus be watched over and identified before it occurs. In order to prevent difficulties connected to seizures, researchers are particularly interested in seizure detection systems [5].

Recordings are frequently used to detect epileptic seizures using the electroencephalography (EEG) technique. Analysis of EEG waves is crucial for identifying neurological conditions like epilepsy. Electric signals from EEG monitors are used to capture the neural activity of the human brain. The brain’s electrical activity is analyzed by EEG, which creates patterns to categorize it as normal or abnormal. EEG typically captures the patterns of brain waves and a piece of equipment that is implanted, such as electrodes positioned on the head, collects the signals [6].

The IoMT is having a significant impact on the healthcare industry by offering valuable solutions for a variety of applications in medicine and healthcare, including remote medical treatment, exercise programs, children’s and elderly care, and the detection and prognosis of various historical illnesses like epilepsy, Alzheimer’s, and schizophrenia. Utilizing wearable devices, IoMT technologies offer continuous and immediate personal medical monitoring [7][8]. In addition to such methods, machine learning ML techniques offer encouraging options for accurate seizure stage detection from incoming EEG signals. Today, the IoMT, along with ML methods and the resources of cloud computing have become powerful technologies that can help with a variety of issues in the healthcare sector [6][9].

1.2. Problem Statement

Patients with epilepsy need special care and continuous monitoring to preserve their lives. The manual epilepsy diagnosis has several shortcomings such as: (It makes clinical diagnosis more arbitrary and susceptible to error because it necessitates the doctors to have extensive clinical diagnosis expertise and professional skills also the EEG signals surrounded by noise may exhibit waveform variations that make diagnosis challenging).

Automated diagnosis also has several limitations such as: (the unbalanced nature of the EEG dataset, large amounts of these data make time and energy consumption and finding optimal hyper-parameters of ML models is a very hard task to solve. From here, the need arose to design an automated system that can get over these limitations and detect epileptic seizures from electroencephalogram (EEG) signals based on IoMT to benefit from the capabilities of IoT and wearable devices in the healthcare sector side by side with machine learning techniques to save a patient life.

1.3. Thesis Objectives

The primary objectives of this thesis are as follows:

1. Propose an IoMT framework for automated epilepsy seizure detection and monitoring.
2. Extract the best features of EEG signals using a weighted visibility graph.
3. Investigate several machine learning algorithms to classify the monitored seizures and make a comparison between them in terms of results.
4. Reduce the amount of data transferred from the fog tier to the cloud tier to reduce the rate of energy consumption.

5. Evaluate the performance of the proposed IoMT framework using different performance metrics such as accuracy, recall, precision, F1 score, etc.

1.4. Thesis Contributions

The main contributions of this thesis are focused on the design and implementation of an automatic epileptic seizure detection system based on IoMT. The contributions of this thesis can be summarized as follows:

1. The EEG data was time-segmented using a non-overlapping moving window to look for seizures.

2. An automated seizure detection system is built on top of the Internet of Medical Things (IoMT) using EEG data. In order to determine the patient's condition in the fog tier, this system employs feature extraction and classification. Additionally, a selective transfer strategy that only sends to the Cloud the information that is really required at the moment.

3. In this thesis, we provide a new method for detecting epilepsy by mapping EEG signals to a weighted network called a weighted visibility graph (WVG). In this way, it is possible to extract both global and local features from time series, making the method independent of time series alignment and outperforming major distance-based time series classification techniques.

4. Combining SMOTE and ENN greatly enhances the risk identification of individuals experiencing epileptic seizures. This strategy is used to address the issue of data class imbalance in the healthcare industry. Moreover, tuning ML models with suitable hyperparameters is notoriously challenging. In this thesis, the "GridSearchCV" algorithm is used to find a solution, and it does so by using a dictionary describing the parameters that may be explored when training a model.

5. We present a threshold-based data reduction strategy for the fog tier, which ranks and selects features depending on their importance. This leads to fewer features, which reduces energy consumption and bandwidth requirements.

1.5. Related Works

This section provides a summary of the earlier research on monitoring and detecting epileptic seizures that has been done. Many studies dealt with patients with epilepsy and how to monitor and control it. We review part of it.

M. A. Sayeed et al. , 2019 [10] suggested an IoMT-based system that uses an individual's EEG signal to foresee the commencement of a seizure. An EEG sensor's readings are continually analyzed and processed by the system to isolate the brain's hyper-synchronous pulses. The onset of a seizure may then be detected using this collection of features. It was implemented by means of a voltage level detector and a signal rejection algorithm (SRA) for filtering out noise and other undesired signal distortions. Using a predetermined threshold for hyper-synchronous pulses over a certain interval, an oncoming seizure may be identified. The device sends out an alert to the appropriate caregivers or physicians as soon as a seizure is detected.

According to [11], G. Regalia (2019) developed wearable automated seizure detection devices that offer tremendous potential to enhance seizure. Machine learning is utilized in a bracelet with the accelerometer (ACC) and electro-dermal activity (EDA) sensors to automatically utilize recognized an event based on symptoms of ongoing generalized tonic-clonic seizure GTCS and send an alarm to a mobile app, which prompts designated caregivers with a call and text via a cloud-based system. The patient's GPS location can be sent to the caregivers. The wearer can quickly silence the alert in the event of a false alarm. Before the signal is delivered to caregivers, the user can promptly silence it.

M. A. Sayeed et al. 2019 [12], the authors explore how discrete wavelet transform (DWT) may be used for detecting epileptic seizures. The IoMT architecture is used in the suggested system. IoT nodes capture the EEG signal and use DWT to analyze it. After the signal is broken down into its component bands, features like standard deviations, signal complexity, and activity may be extracted from them. A deep neural network (DNN) classifier is then trained on these extracted characteristics to make the diagnosis of normal, interictal, or ictal EEG activity. In the experiments, the accuracy for two classes (ictal and normal) was reported at 100%, while the accuracy for all three classes (ictal, normal, and interictal) was 98.6%.

H. Daoud et al. (2020) [13] proposed a deep learning-enabled Internet of Things-based platform for continuously monitoring and predicting epileptic episodes. The raw data from the EEG headset is sent wirelessly to a Field Programmable Gate Array FPGA, where it undergoes preprocessing to extract critical spatiotemporal properties before being input into the integrated deep convolutional neural network (DCNN). When a seizure is identified, the EEG data and the outcome of the CNN model's prediction are sent to a Raspberry Pi, which then incorporates real-time alerts. A cloud is used to store the EEG signal for later review by a medical professional. The suggested approach achieves an experimental prediction accuracy of 96.1.

P. A. E. Akashah (2020) [14] proposed a different method in contrast to the majority of previously published research, which relies on EEG-based monitoring to assess and forecast the start of seizure. They suggested an Internet of Things-based system for monitoring heart rate in order to identify seizures, with heart-rate variability being the primary metric for analysis. Children with neurological problems, aged 15 and younger, are the focus of this research. The authors accomplished their goal by developing a prototype wearable device to track heart rate and uploading the results to a cloud-based database for further study. The outcomes of this study, however, have not yet been recorded, since the research is still ongoing.

S. Hassan et al (2022) [15], proposed a study to detect early warning signs of epileptic behavior specifically, "Grand mal epilepsy Tonic-Clonic (GTC) seizure," which is a kind of generalized epilepsy in people in the hopes of preventing the onset of the disorder employed Electromyography EMGs, Electrocardiogram ECGs, and Accelerometers on all three axes for fall detection, and Dallas temperature sensors. Seizures classified according to a wide variety of criteria, including muscular spasms, body temperature, heart rate, and falls, using fuzzy logic algorithm. These are sent into a system to determine the kind of seizure, with the result displayed graphically on the loT platform’s (ThinkSpeak) dashboard. In terms of monitoring core bodily functions including temperature, heart rate, muscular spasms, and falls, they reported 98.90%, 95.49%, 83.0%, and 87.21% as average accuracy, respectively.

Table 1.1 contrasts the results of several EEG detection methods for the previously explained related works and other recent works concerning the feature extraction method, classifier used, and detection accuracy.

Table 1.1. A synopsis of recent papers that evaluated the usage of common feature extraction methods and ML classifiers

Ref.	Year	Dataset	ML Classifier	Features Method	Performance Metrics	Results
[12]	2019	Bonn	DNN	DWT	accuracy	98.6%
[16]	2019	Bonn, Freiburg	RF	DWT	Sensitivity	99.95%

[17]	2019	Publically available dataset	SVM	DWT	Accuracy	97.87%
[18]	2019		ANN	DWT	Accuracy	95%
[19]	2019	Bonn	SVM, KNN, DNN	Feature Scaling Loss Function	Accuracy	SVM=94 %, KNN=74 %
[20]	2019	CHB-MIT	RF	Wavelet Packet Decompositio n	Accuracy	84%
[21]	2019	Ramaiah College, CHB-MIT	SVM	Successive Decompositio n Index	Sensitivity, F1-score	97.53%, 97.22%
[22]	2019	Bonn	Least Square SVM	DWT, Entropy	Sensitivity, Specificity, Accuracy	100%, 99.4%, 99.5%
[23]	2020	Bonn	ANN, KNN, NB, SVM	DWT	Accuracy	97.82%
[24]	2020	CAP Sleep	ANN	DWT	Accuracy	91.1%
[25]	2020	CHB-MIT	RF, DT	DWT	Accuracy	99.81%
[26]	2020	Kahib-Celebi school of medicine	SVM, KNN, NB	EMD, DWT	Accuracy	SVM=94.5 6%, KNN=95.6 3%, NB=96.8 %
[27]	2020	CHB-MIT	DL, SVM, CNN	DFT, EMD, DWT	Sensitivity, Specificity	92.7%, 90.8%
[28]	2021	CHB-MIT	Fuzzy Classifier	Adaptive Wavelet Transform	Accuracy	96%
[29]	2021	CHB-MIT	SVM	DWT	Sensitivity, Specificity	96.81%, 97.26%
[30]	2021	Bonn	SVM, KNN	DWT, Local Binary Pattern Transition Histogram	Accuracy	99.6%
[31]	2022	Bonn CHB-MIT	LR, NB, KNN, DL, RF, LSTM	Wavelet Transform	Accuracy	RF=97%, LSTM=98 %
[32]	2022	CHB-MIT	SVM/ RF	Tunable Q-Wavelet Transform	Accuracy	SVM=90.4 %, RF=93.5%

[33]	2022	CHB-MIT	SVM	DWT, Eigenvalue	Accuracy, Sensitivity	97%, 96.67%
[34]	2022	CHB-MIT	DT, RF, ANN, Ensemble Learning	DWT, DFT	Sensitivity, Specificity, Accuracy	91.9%, 89.7%, 91.9%
[35]	2023	Bonn	RF	Entropy	Accuracy	96%

1.6. THESIS OUTLINE

The remaining of this thesis is structured as follows:

Chapter 2: This chapter presents the underlying theory of epileptic seizures in based of IoT by applying a general overview of the fields of neuroscience, EEG signals and explain the terms and give descriptions of some research datasets that are freely accessible to the public. Detection of epileptic seizures using several ML classifiers is also covered. We discuss the different types of graph theory in more detail and finally explain briefly some of the methods used to resolve the imbalanced data and give a brief definition to hyper parameters.

Chapters 3: This chapter presents the proposed methodology used to monitor and detect the epileptic seizures. Design of the system and implementation of the network layers are explained.

Chapter 4: The experimental findings are examined, clarified, and discussed in this chapter. It also includes evaluating how well the suggested system operates in light of various performance measures.

Chapter 5: This chapter finishes off the thesis and makes some recommendations for further studies.

CHAPTER TWO

THEORETICAL BACKGROUND

CHAPTER TWO

THEORETICAL BACKGROUND

2.1. Introduction

This chapter introduces short overviews of the fields of neuroscience, and EEG signals. Also, the terms and descriptions of some research datasets that are freely accessible to the public are explained. Then we'll quickly go through how to analyze epileptic seizures from an EEG signal to better understand epilepsy. Detection of epileptic seizures using several ML classifiers along with a thorough comparison of these methods is presented. At the end, quick overview of the IoMT and briefly explain how it can be used in the healthcare field, especially in ML-based epilepsy detection.

2.2 Neuroscience

The comprehensive study of the brain is known as neuroscience. It combines a variety of disciplines, such as neurophysiology, which studies the brain's electrical characteristics; neuropsychology (the branch of psychology that aims to study how the brain's neural networks contribute to our ability to think and reason); both neurochemistry (in which chemists study the brain's chemical characteristics of communication) and neuroanatomy (where neuroanatomists engage with the structures of the human brain) [36]. There are many subfields within neuroscience, including but not limited to, cultural neuroscience, computational neuroscience, developmental neuroscience, clinical neuroscience, cognitive neuroscience, and molecular neuroscience, to mention a few [37].

Neuroimaging employs a variety of techniques to directly or indirectly picture the anatomy and operation of the central nervous system. Imaging techniques can be divided into two major categories: structural imaging, which deals with anatomy, injury or pathology; and functional imaging, concerning metabolic processes, cognition, or pharmacology. Among the most influential and widely used neuroimaging methods are as follows: computed tomography (CT), which generates a sequence of cross-sectional images of the brain by computing the amount of X-rays that were absorbed; brain magnetic resonance imaging (MRI) scans that look at the structure of the brain to see whether there's anything wrong with it; in order to see larger structures in the brain, diffusion MRI tracks the movement of water molecules; cognitive neuroimaging, or functional MRI, which looks at how the brain works, positron emission tomography (PET) that produces a picture of the binding of active molecules, and magnetic resonance spectroscopy (MRS) imaging that studies metabolic changes in brain strokes, and seizures. It is possible to monitor the electrical activity of the brain under diverse physiological circumstances using functional imaging techniques like EEG [38].

2.3. EEG Signals

The electroencephalography (EEG) technique is frequently used for detecting epileptic seizures from EEG recordings. When compared to other neuroimaging modalities, it was popular because of its inexpensive cost and high portability. EEG is trustworthy for real-time applications due to its capacity to obtain readings every milisecond. Hans Berger created the EEG in 1923 as an imaging technique that is non-invasive for studying the brain. Typically, an implanted device, such as electrodes inserted on the scalp, collects the information for EEG, which records the patterns of brain waves [6]. Wired and wireless methods can be used to collect EEG signals, and these signals can be measured using electrode counts ranging from 1 to 256. EEG has a lower spatial resolution than functional MRI but offers a greater temporal understanding of brain activity. For analyzing EEG signals, five frequency bands—Delta, Theta, Alpha, Beta, and Gamma—are typically examined. Table (2.1) summarizes various frequency ranges and how they relate to human behavior. The frequency range of EEG signals is from 1Hz − 100Hz, while the EEG range is from 10μV − 100μV.

Table 2.1 EEG frequency bands and associated research on how the brain works

Frequency Bands	Frequency Range (Hz)	Relation to human behavior
Delta	1-4	Especially at the deepest stages of sleep, both in youngsters and in healthy adults
Theta	4-8	Adults with high-theta rhyme activity display inconsistent cognitive function.
Alpha	8-12	Usually located in the back of the brain in calm, healthy people
Beta	12-26	Found in nervous, alert people's frontal lobes
Gamma	26-100	Characteristic of persons who are worried, happy, or self-aware

2.4 Datasets Publicly Accessible

A dataset is essential for academics and data scientists to evaluate the performance of their suggested models. Similar to this, we must record the brain's electrical activity to identify the seizures of epilepsy. EEG recording is the most widely used method for examining the electrical signals of the brain. This EEG recording data is necessary for machine learning techniques to investigate novel seizure detection methods, such as seizure localization, patient seizure monitoring, and quick seizure detection. The importance of open source datasets is that these datasets provide a benchmark for assessing and contrasting the results [39]. The most well-known datasets that are frequently employed in research on epilepsy are shown in figure (2.1) with their corresponding proportion.

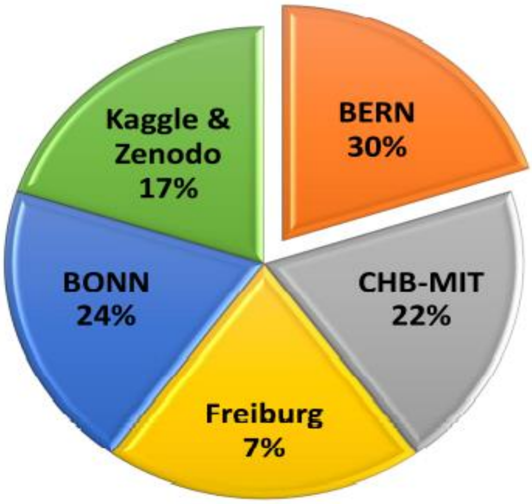


Figure 2.1 The utilization rate of various datasets.

The further details for each dataset are included in Table (2.2).

Table 2.2 Provides a comprehensive description of the publicly available EEG datasets used to identify seizures of epilepsy

Dataset	Recording	Sampling Frequency (Hz)	No. of Patients	No. of Seizure	Time (hours)
CHB-MIT [40][41]	Scalp EEG 256	256	22	163	844
Freiburg [42]	IEEG	256	21	87	708
Bonn [43]	Surface and IEEG	173.61	10	NA	39 m
Bern Barcelona [44]	IEEG	512	5	3750	83 m
Kaggle [42]	IEEG	400/5 KHz	5 dogs, 2 patients	48	627
Zenodo [39]	Scalp EEG	256	79 neonatal	460	74 m

2.5 EEG Signal Analysis for Epilepsy

EEG data analysis is the fundamental method for pinpointing epileptic seizures ES activity in the brain. Recordings from EEG are a crucial clinical tool for distinguishing ES from non-ES. Comparison of pre-and post-episode symptoms of epileptic seizure phases, as well as pre-and post-seizure times, may be distinguished using EEG data. A quick description of these phases is given in the following subsections [45].

2.5.1 Pre-Ictal State

Pre-ictal states only appear during the time period prior to a seizure and do not occur at other times. It might not always be visible to the naked eye. [46].

2.5.2. Pro-Ictal State

Epileptic seizures are more likely to occur but are not guaranteed in this state.

2.5.3. Ictal and Interictal State

The changes in EEG recording signals that take place during a seizure are known as the ictal state, while the interictal state is the period between two subsequent seizure onsets. The amount of epileptogenic neurons, cortical areas, and seizure duration can be altered for the same person.

2.5.4. Post-Ictal State

This condition has developed following a seizure.

The wave pattern might contain insightful data regarding the activity of the brain. Neurologists with experience can identify diseases by looking at the EEG waves. [47]. Figure (2.2) shows the EEG signal plots for the normal stage in the case of a healthy person and the preictal and ictal stages in the case of an epileptic individual.

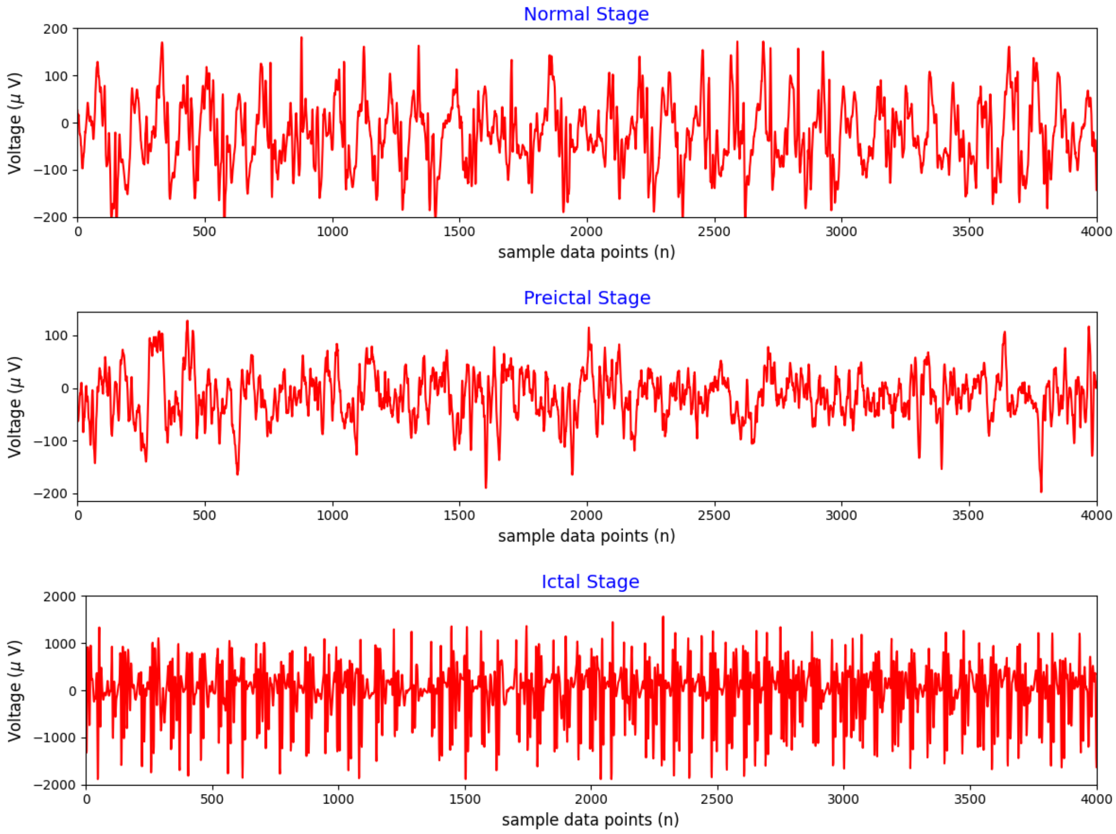


Figure 2.2 EEG signal graphs for normal, preictal, and ictal stages.

2.6. Data Reduction and Feature Extraction

Data reduction involves reducing the feature values by taking out the essential features from a particular region of the input signal. A key component of classifying EEG signal data is feature extraction. The primary goal of feature extraction is to gather trustworthy data for epileptic recognition. In terms of technical terms, a feature denotes a distinguishing quality and a recognizable measurement derived from a section of a pattern. With the least amount of information loss possible, the feature extraction technique compresses the huge volume of EEG data into a useful and significant feature vector set. As a result, it facilitates analysis (classification) by facilitating easier and faster computational speed [48].

Fundamentally, there are two methods for extracting features from an EEG signal, namely manual and automatic extraction. Table (2.3) shows the EEG features that are most commonly used to detect epileptic activity.

Table 2.3 EEG features that are most commonly used to detect epileptic activity

Feature Extraction Method	Relevant Features
Time Domain Features	Standard deviation, approximate, sample entropy, Shannon entropy, power, energy, line length, zero crossings, min, max, variance, Hurst exponent, fuzzy entropy, entropy, mode, median, kurtosis, skewness, and mean.
Frequency Domain Features	Spectral entropy, spectral power, median frequency, peak frequency, and spectral energy.
Time-Frequency Domain Features	Root mean square, approximate entropy, Shannon entropy, median, energy, standard deviation, max, min and line length.
Discrete Wavelet Transformation	Variation, bounded variation, relative power, relative scale energy, coefficients, energy, entropy, relatively bounded, variance, and standard deviation.
Continuous Wavelet Transformation	Energy’s standard deviation, energy, coefficient z-score, entropy.
Fourier Transformation	Median frequency, power, peak frequency, spectral entropy power, spectral edge frequency, total spectral power.
Non-Linear Features	Higher order spectra, Kolmogorov entropy, sample entropy, Hurst exponent, approximate entropy, Largest Lyapunov Exponent.

2.7. Performance Evaluation

Methods are compared and contrasted based on how accurately they provide outcomes. Ten-fold cross-validation is widely used as a training method because in each fold, or a horizontal section of the dataset, the training dataset is utilized for the other nine sections, whereas the testing dataset is used for one portion. The classification performance is evaluated using a variety of performance measures, including accuracy, precision, recall, F1 score, and false-positive rate. These are based on four possible classification outcomes—True-Positive (TP), True-Negative (TN), False-Positive (FP), and False-Negative (FN) as presented in Table (2.4). Table (2.5) discusses several measures that may be used to evaluate the classifier's efficiency.

Table 2.4 Classification outcomes

Acronym	Detection type	Real-world scenario
TP	True-positive	If a person suffers to 'seizure' and also correctly detected as a 'seizure'
TN	True-negative	The person is actually normal and the classifier also detected as a 'non-seizure
FP	False-positive	Incorrect detection, when the classifier detects the normal patient as a 'seizure' case
FN	False-negative	Incorrect detection, when the classifier detects the person with 'seizure(s)' as a normal person. This is a severe problem in health informatics research

Table 2.5 Evaluation metrics

Metrics		Formula
Accuracy	It is the ratio of segments that were successfully identified to all segments.	Accuracy $=\frac{TP+TN}{TP+TN+FP+FN}$
Recall	It is the proportion of segments with seizures that were successfully identified out of all the segments with seizures.	Recall$=\frac{TP}{TP+FN}$

Specificity	It is the ratio of the number of segments accurately identified as not having seizures to all segments that did not have seizures.	S Specificity= $\frac{TN}{TN+FP}$
Precision	It is the proportion of true positives to all cases.	Precision= $\frac{TP}{TP+FP}$
F1score	It is the precision and recall weighted harmonic mean.	F1score= $2 \times \frac{(\text{Precision} * \text{Recall})}{\text{Precision} + \text{Recall}}$
Matthew Correlation Coefficient (MCOR)	With a value between -1 and 1, it is a balanced indicator of the classifier's performance.	MCOR $= \frac{TP*TN-FP*FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$

2.8. Machine Learning Methods for Detecting Epileptic Seizures

In particular, this section offers a thorough analysis of the studies conducted on ML-based techniques for ES detection. At first, we focus on the promise of ML methods in the medical and biomedical fields.

2.8.1. An Overview of ML in Healthcare

Over the past few decades, by using statistical methods to uncover patterns in enormous data sets, ML has extended across a range of academic disciplines. It is now possible to access large-scale biomedical data, which is a big advancement for medical researchers. The growth of ML techniques and data analysis methodologies is essential for the creation of efficient medical instruments. For the purpose of illness diagnosis, ML has found widespread use in the healthcare industry, such as the classification of skin cancer [10], the detection of breast cancer [49], retinal image analysis for diabetic retinopathy detection [50], Alzheimer’s disease diagnosis [51], and prediction of epilepsy [52]. This is because the complicated structure of medical data makes it a difficult challenge to manually discover representations.

2.8.2. Machine Learning for Neuroscience

Neuroscientists have been primarily concentrating on what we've learned recently about the functional architecture and anatomical organization of the human brain. Technological developments have made it possible for neuroscientists to collect, process, and analyze neuroimaging data at a new level of detail; ML and DL are prime examples of the kinds of enabling technologies that might be utilized as a starting point for investigating fundamental questions regarding how the brain works [37]. In this part, we give a general overview of several machine learning (ML) approaches that have been applied to the study of neuroscience, including supervised learning, unsupervised learning, and reinforcement learning.

A. Supervised Learning

In supervised learning, the learning algorithm receives training data together with labels applied by human specialists. This allows the algorithm to extract the relationship between the data and labels in order to accurately categorize the unknown data. For instance, a training dataset might contain photographs labeled with "home," "dog," or "cat," and we'd need a classifier that can foretell the label of a new image that the system hasn't seen before. The support vector machine (SVM), a supervised learning technique frequently employed for ES detection, is one such method with numerous potential applications in computing and theoretical neuroscience. A supervised ML technique is used to analyze brain systems under stress [53].

B. Unsupervised Learning

On a typical day, our brain is given the majority of the information on its own. As information is repeated, the brain creates a functional model and uses it to build perceptions. The patterns in the new knowledge are then found using this perception. The brain learns new things through perception, which is the inspiration for unsupervised learning algorithms. Unsupervised learning uses data that has not been tagged or classed to train algorithms. The diagnosis and categorization of disorders from neurophysiological data makes substantial use of these algorithms.

C. Reinforcement Learning

The development of Reinforcement Learning (RL) was influenced by animal psychology, or how animals relate to one another and their environment. The partnership between AI and neuroscience has resulted in significant technological advancements, one of which is RL. Reinforcement learning is the method of developing a plan of action to maximize the advantages of an agent's interaction with its environment. The strategy, rewards signal, environmental model, and value function are the fundamental elements of a reinforcement learning system [37][54].

2.8.3. ML Classification Techniques

In automated epilepsy detection systems, ML methods are the most common classifiers employed. Features are extracted, analyzed statistically, ranked, selected, and then fed into ML algorithm classifiers using traditional handcrafted feature extraction techniques. Various classification strategies as shown in Figure (2.3), including support vector machines (SVMs), artificial neural networks (ANNs), random forest (RF), logistic regression (LR), naive bayes (NB), decision tree (DT), and k-nearest neighbor (KNN). A variety of classifiers are used in seizure detection, and we will go through the most well-known ones in the following subsections.

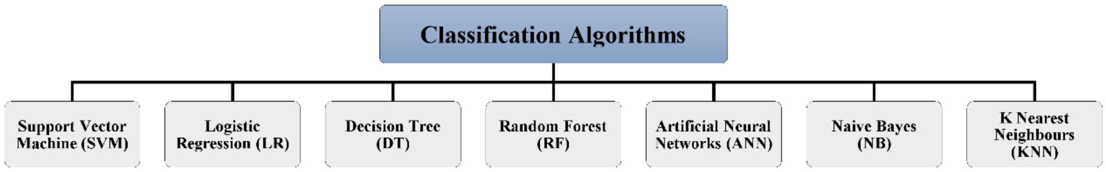


Figure 2.3 Some of the most popular machine learning classifiers.

1. Support Vector Machine (SVM)

The Support Vector Machine (SVM) is a two-stage method. It has several features, such as being able to handle a large number of predictors and variables even with a small sample size. In most cases, SVMs try to locate the optimal hyper-plan for identifying and isolating instances of a given class from all other instances. Whichever hyper-plane produces the largest gap between the two groups is optimal [55]. It is possible to formulate the problem of locating the best-separating hyperplanes as follows:

$$\min \frac{1}{2} ||w||^2 \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (2.1)$$

Subject to:

$$r^t(w^T x^t + w_0) \geq 1, \ \forall t \quad \ldots\ldots\ldots\ldots\ldots \quad (2.2)$$

The term "margin" refers to the largest distance between two adjacent data points on the hyperplane perpendicular to the slab. It is the linear decision surface’s most pronounced gap among borderline patients that inspired its use in SVM as a means of classifying patients into distinct groups. It was excellent at handling non-linear and high-dimensional data. It aids in predicting key features of unidentified testing data based on the structure of training data sets [56].

The linear kernel function, which is often used in SVM, is described by the following equation:

$$K(X, Y) = X^T Y \quad \text{……………………………… (2.3)}$$

Alternatively, a polynomial of degree d may serve as a kernel function, as shown below:

$$K(X_{i,} X_j) = (\mathrm{Xi}, \mathrm{Xj})^d \quad \text{.. (2.4)}$$

In which the total number of polynomials is denoted by d(d ≥ 1).

A function is said to be quadratic in its kernel if the degree of its polynomials is two or three.

2. Naive Bayes (NB) Classifier

In the NB algorithm, characteristics and categories are the focus. One quick method under consideration analyzes all the data in the training sets while requiring less information for classification. NB is a probabilistic classifier that relies only on learning features independent of the class (as per Bayesian theory), such that any characteristic of a given class may be considered separately from the others. This idea of independence is typically a bad one. It is a Bayesian theorem-based algorithm. In order to determine a conditional probability, this classifier considers the association between each characteristic and the class for each occurrence [23].

Given a training set D with n classes and any attribute vector Y with class labels, it can be shown that the class to which attribute Y most likely belongs, based on the posterior probability, is:

$$P(C_i|Y) > P(C_j|Y) \text{ for } 1 \leq j \leq n, j \neq i, \quad \text{……… (2.5)}$$

Where

$$P(C_i|Y) = \frac{P(Y|C_i)P(C_i}{P(Y)} \quad \text{………………..… (2.6)}$$

To clarify, P(C_i) stands for the probabilities of each class; the prior probability of Y is P(Y); the posterior probability of Y is P(C_i|Y); and the posterior probability of Y conditioned on C_i is P(Y|C_i).

3. K- Nearest Neighbours (KNN)

KNN is a simple, nonlinear, and nonparametric sample classification method [37]. In practice, it performs well on big training datasets. Using this approach, a data item is assigned to the category with which its k-closest neighbours have the most familiarity based on the results of a majority vote. Similarity metrics between the training and test data sets are the primary foundation of this approach. To determine which class a new sample belongs to, we look at the other samples in the same or neighbouring k datasets used for training and use the results as our guide. In general, K should have a value between 3 and 10. In comparison to 1, those numbers are astronomically larger [57] [58].

A distance metric, such as the Manhattan distance, may be used to assess how close or similar two entities are to one another. For a p-dimensional space, the Manhattan distance is defned as:

$$d(i, j) = |x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + \ldots + |x_{ip} - x_{jp}| \quad \text{………..(2.7)}$$

4. Artificial Neural Network (ANN)

ANNs are functions that are made up of neurons and weights. The input values are processed by the neurons, and the output is the outcome; the weights transport the information between the neurons. There are three primary layers of neurons: input, hidden, and output. Each piece of data that will be supplied into the network is represented by an input unit, and this layer comprises several such units. The hidden layers include the hidden units, which are based on two things: (a) the activities of the input units; and (b) the weights that are based on the relationships between the input and the hidden units [23]. When given input, the ANN computes the weighted total of the inputs and incorporates a bias. A transfer function is used to visualize this calculation.

$$\sum_{i=1}^{n} W_i \times X_i + \text{bias} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \text{(2.8)}$$

$$\text{Output} = \begin{cases} 1 & \text{if } \sum_{i=1}^{n} W_i \times X_i + \text{bias} \geq 0 \\ 0 & \text{otherwise} \end{cases} \quad \ldots\ldots\ldots\ldots\ldots\ldots \text{(2.9)}$$

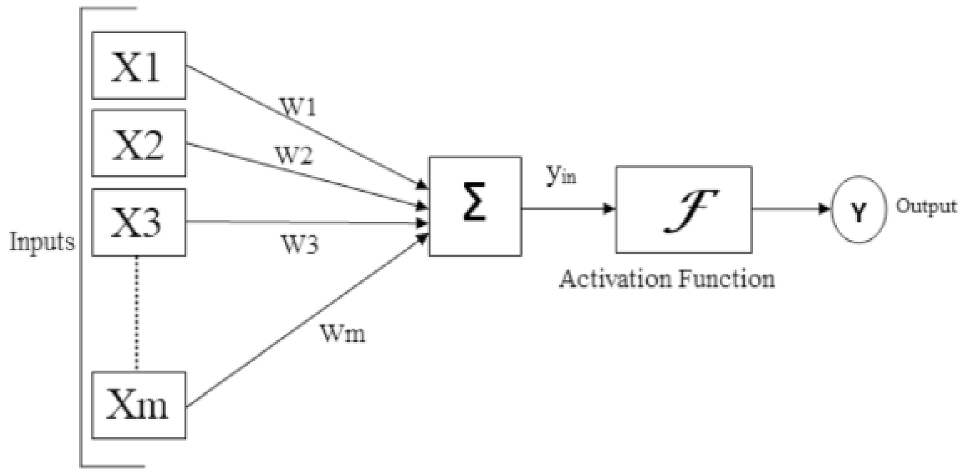


Figure 2.4 Artificial neural network layers.

5. Logistic Regression (LR)

LR is a supervised technique for performing binary classification of events based on whether they occur or not. Given that it's a probability, the dependent variable can only take on values between zero and one. For use in LR, the probabilities are changed to logit form by dividing the chance of success by the chance of failure. This logistic function is also called the natural logarithm of chances or the log odds [59].It is shown by the following expressions:

$$P(x)=\frac{1}{1+e^{-(x-\mu)/s}} \quad \text{………………………………… (2.10)}$$

where p(μ) = 1⁄2 denotes the middle of the parabola μ, and s is the scale factor. Equation (2.10) can also be expressed as:

$$P(x)=\frac{1}{1+e^{-(\beta_0+\beta_i x)}} \quad \text{……………………..…………(2.11)}$$

Most of the time, maximum likelihood estimation (MLE) is used to figure out this model's beta parameter (or coefcient). The corresponding diagram of LR is shown in Figure (2.5).

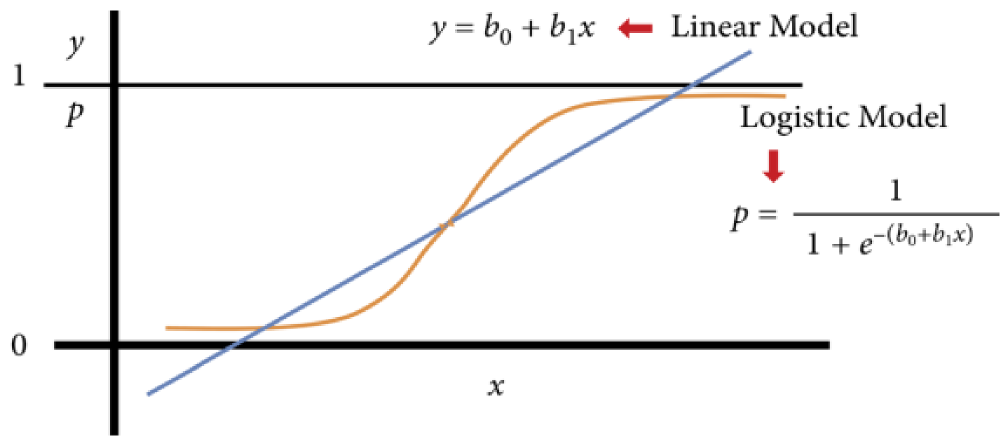


Figure 2.5 Logistic regression

6. Decision Tree (DT) Classifier

The data are organized into a tree with each node indicating a test on one of the input variables, and the last node, the "leaf," reflecting the final decision. The testing and branching process is repeated by the classification algorithm. Learning from DTs uses a divide-and-conquer tactic by employing a greedy search to locate the best possible node splits within a tree. This partitioning is then iterated recursively from the top down until all or nearly all of the data have been assigned to their respective classes [60]. DT models often use either Gini impurity or information gain as a way to separate options. They are useful for judging how well a given set of test conditions can reliably classify test data. The "information gain" is the change in entropy from before the separation to after the separation of a specific characteristic. The best split will come from the attribute with the most information gain because it does a better job of classifying the training data according to its target classification. Typically, the following formula is used to express information gain:

$$I_{Gain}(S,a) = Entropy(S) \sum_{v \in values(a)} \frac{|S_v|}{|S|} \text{Entropy}(S_v) \ldots\ldots\ldots (2.12)$$

Where a is a specific class label or attribute.

Whereas the Gini impurity would indicate the likelihood that the classification would be wrong, if an arbitrary data point in the dataset were to be classified based on the distribution of classes. If a set, S, is pure (i.e., it only contains elements from one class), then its impurity is also zero. The following equation represents this:

$$\text{Gini} = 1 - \sum_{i=1}^{c} (p_i)^2 \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (2.13)$$

7. Random Forest (RF)

RF is a classifier ensemble made up of several decision trees. Diverse components of the data collection are used to train the decision trees. Each DT receives a fresh sample as input, and the forest chooses the classification with the greatest votes [61].

8 Extra Tree Classifier (EXT)

An example of ensemble learning, Extra Trees Classifier takes a "forest" of independent decision trees and uses their combined classification accuracy to draw conclusions. The main conceptual difference between this and a Random Forest Classifier is in how the decision trees in the forest are built [62].

9. Extreme Gradient Boosting (XGB)

One kind of ensemble learning is XGBoost or in short XGB. Results from a single machine learning model may not always be reliable. With ensemble learning, it's possible to systematically pool the foresight of numerous students. As a consequence, we have a unified model that synthesizes the results of several models [63].

2.9. An Overview of the Internet of Medical Things

Initial research and development of remote patient monitoring systems sparked the application of IoT in healthcare. Since then, research on IoMT's uses has increased steadily, and current studies focus on finding ways to integrate IoT into many areas of healthcare, such as preventing the spread of illness and facilitating accurate automated diagnosis and treatment. Furthermore, there is a big demand right now for smart healthcare that is high-quality, affordable, and patient-centered. A huge need for real-time, intelligent, and remote healthcare services has been generated by the growth of IoMT and cloud technologies under the umbrella of smart cities [64],[65]. We provide a working definition of an IoMT system and a high-level overview of its architectural components in this section. Classifications of IoMT applications from the literature are shown in Figure (2.6).

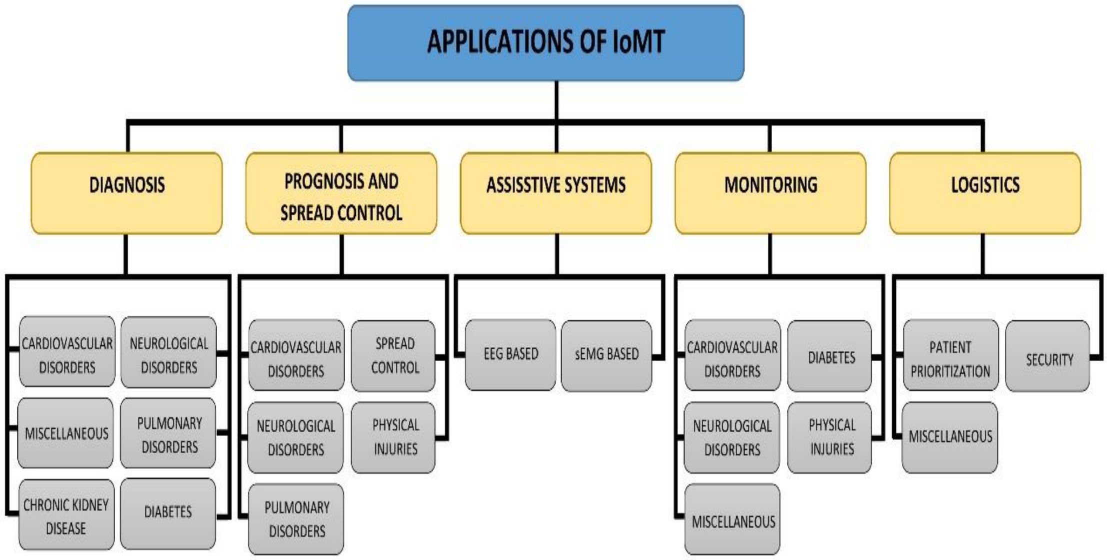


Figure 2.6. IoMT Application

2.9.1. The IoMT Definition

The Internet of Medical Things (IoMT) is a network of all possible healthcare resources that are interconnected to facilitate speedy data transmission via the Internet [1]. What this implies is that the whole healthcare system—doctors, hospitals, rehabilitation centers, medical equipment, sensors, and patients themselves—will be linked together for seamless, real-time data sharing. Many kinds of IoT devices are designed for the healthcare industry, such as smart wearable devices like portable insulin syringes, blood pressure monitors, stress monitors, weight trackers, hearing aids, fitness trackers, and EEG and ECG monitors. Previous research demonstrated that a smart city may offer real-time smart healthcare services when cloud technologies, IoT, and smart sensors are merged [66]. Therefore, in order to deliver high-quality healthcare services, an intelligent healthcare monitoring system ought to be capable of quickly processing multimedia signals and sensor data. However, the fundamental issue with epileptic patients is that they require prompt, high-quality care. For people suffering from seizures, any delay in receiving care or accessing medical facilities or hospitals could be catastrophic. As a result, people with epilepsy need an intelligent healthcare monitoring system, which is essential and might typically solve this problem. There are three basic requirements [67] that must be met for a framework to develop and work well.

– **Interoperability**: The wide range of devices being used in the framework should be able to cooperate among each other to enable the desired functionality.

– **Bounded latency and reliability**: The network's entities need to be able to send and receive information quickly and accurately so that crises can be handled properly and the massive volume of data can be analyzed in unison.

– **Privacy and security**: The IoMT architecture requires authentication and security mechanisms to ensure that only the appropriate organizations get the private data being exchanged. There are a number of methods, including encryption and physical unclonable functionalities, for verifying the identity of Internet of Things devices and ensuring their authenticity.

2.9.2. IoMT and ML-based Epilepsy Seizure Detection

Everything from farms and smart homes to clinics and at-home health monitors is using loT and ML-powered monitoring devices to keep an eye on things. Over time, several practical uses for IoMT (Internet of Medical Things) have emerged [48], [64].

Improvements in loMT technology have opened several doors in the healthcare industry, particularly in areas like lowering service costs, facilitating monitoring outside of hospitals, and spotting anomalies as soon as they occur in real time. Acute disorder treatment and monitoring are also a part of these cutting-edge methods. An epilepsy disorder is a neurological condition characterized by repeated seizures that are caused by an abrupt electrical disruption in the brain. Consequently, prompt detection and prediction are of the utmost importance so that the patient may get the best possible care and therapy [64].

2.10. Graph Theory

Graph-based network analysis can be used to comprehend the topological design that defines human brain structures, including small-worldness, modularity, and highly connected or centralized hubs. Many nodes in some networks can be accessed from each other with only a few steps, even if most of them are not neighbors.

This phenomenon is known as small-worldness. This property confirms effective information separation and combination in the brains of human networks with minimal energy and wiring costs, making it well suited for the research of complex brain dynamics [68].

With brain regions represented as a set of nodes and connections represented as edges, graph theory is being increasingly utilized to study networks of brain connectivity. Since graphs are conceptually very simple and flexible, they are used to represent mathematical abstractions of networks, which can be described as a structure consisting of a set of vertices (also known as nodes or points) and a set of edges (also known as arcs or links) that connect the vertices [69].

Graphs are a common tool in computer science, biology, social science, and other fields for interacting with various interactions and processes. The definition of a graph in mathematics is:

$$G = (V, E) \qquad (2.14)$$

V: is a collection of vertices, or node, as it is sometimes referred to.

E: edge linking two vertices

Each edge is identified by a pair (p, q), linking two vertices p and q [70]. Each node is established by an integer index p = 1,..., N. Additionally, a time series-derived graph typically possesses the following features:

1. Connectivity: this quality denotes that each node can identify at least its left and right nearest neighbors.
2. Un-directedness: A graph produced in this fashion has no directions in its connections.
3. The visibility criteria is unaffected by changes to the axes' horizontal and vertical scales as well as their translations [71].

Finally, unique numbers are frequently used to identify the nodes. Additionally, the edges may bear labels in the form of weights, which may describe attributes like the separation between nodes, node degree, node distribution, path and path length, among others [72].

2.10.1 Visibility Graph (VG)

Traditional VG has some drawbacks when used in biological or medical time series analysis, such as being unable to capture all points below the baseline. Sinusoidal and cosinusoidal time series are mapped to various graphs by the VG [73]. There are thus various graph theory developments.

It is noteworthy that visibility graph preserves the structure of the time series in the related graph, acting as a natural bridging between complex network theory and time series analysis [74]. By assigning each value to a node and connecting two nodes with an edge if the respective data points are visible from one another, one can create a visibility graph from an ordered series of values [75]. The definition of a visibility graph in mathematics is [76]:

$$\frac{x_i-x_n}{i-n} > \frac{x_i-x_j}{i-j} \quad \text{……..…………………… (2.15)}$$

The i node and the j node are visible if i<n<j; otherwise, they are invisible [77]. Visibility graph is clarified in Figure (2.7.a).

There are three types of visibility graph that's:

1. Natural Visibility Graph Algorithms (NVG): In order to explore the complexity of the original time series, NVG was developed as a way to map irregular time series into networks. The resulting network is always connected, undirected, and invariant under affine transformations of the series [78].

The vision line between a and b is allowed to have any slope under the Natural Vision Criteria (NVC). if each intermediate point in the time series (t_c, y_c) is true, then two points (t_a, y_a) and (t_b, y_b) are considered to be naturally visible [75].

The algorithm's mathematical equation, $t_a < t_c < t_b$, states what the intermediate point must be [69]:

$$Y_c \leq Y_b + (Y_a + Y_b)\frac{t_c-t_a}{t_b-t_a} \quad \text{…………………….. (2.16)}$$

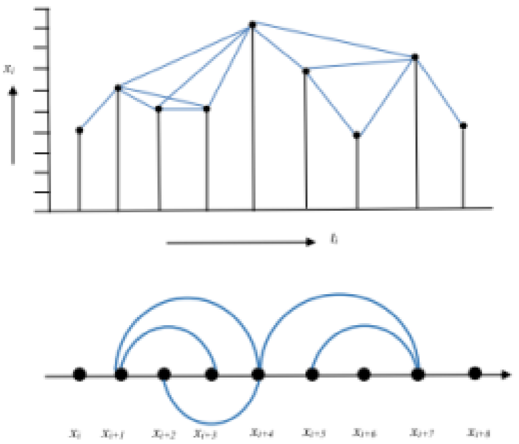
Figure (2.7.b) clarifies the graph of natural visibility**.**

2. Horizontal Visibility Graphs Algorithm (HVG) : Visibility graphs include HVG as a sub graph. The definition of the construction of a horizontal visibility graph is based on an alternative visibility graph criterion, which states that two nodes i and j in the graph are connected if a horizontal line connecting xi and xj can be drawn in the time series and that approach does not intersect any intermediate data height. The figure (2.7.c) clarifies horizontal visibility graph [79].

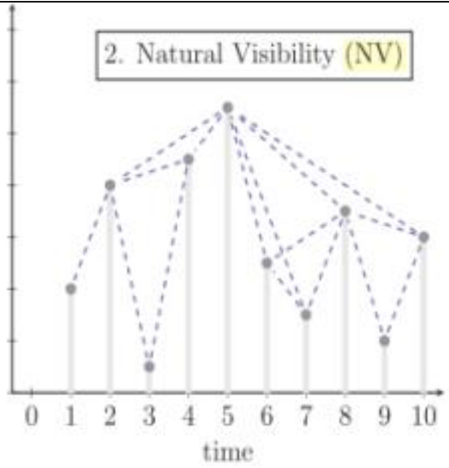
3. Weighted Visibility Graph (WVG): Since the strength of edges between different time nodes cannot be expressed by a visibility graph, Supriya [56] proposed a (WVG) and demonstrated that as the values of the EEG signals fluctuate, the link weights will also change, making the construction of the EEG Signal WVG easier. Thus edge Eij's weight Wij between the two nodes (ti, xi) and (tj, xj) is as follows [80]:

$$C=\begin{cases} \arctan\left(\frac{x(t_i)-x(t_j)}{t_j-t_i}\right) & i < j \; e_{ij} \text{ exist} \\ 0 & otherwise \end{cases} \quad \ldots\ldots\ldots\ldots\ldots (2.17)$$

Studies have shown that weighted visibility graph (WVG) networks when compared to connectivity networks retain more structural information on the time series. The weighted visibility graph (WVG) has been proposed to calculate weight through the radian function as the criterion, which obtained promising results in the detection of epilepsy [81]. Clarify the weighted visibility graph in Figure (2.7.d).



a. Visibility graph



b. Natural visibility Graph [78].

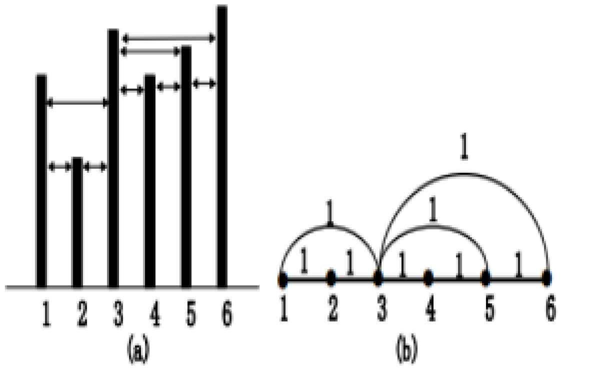
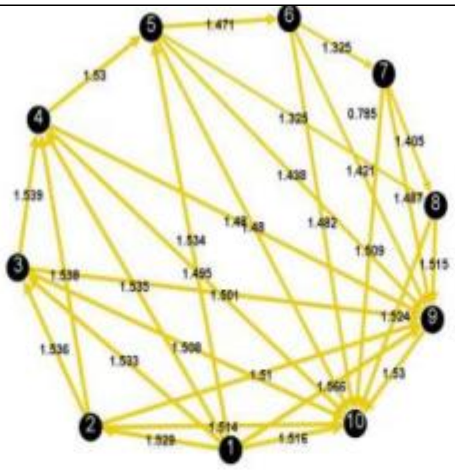
 c- Horizontal visibility graph of a time series [82].	 d. Weighted Visibility Graph [56].

Figure 2.7 The descriptions of Visibility Graph

2.10.2. Global Graph Features

Time domain EEG signals are converted using a feature extraction technique into significant graph features, in the following section some of these features:

1. **The Graph Density (GD):** measures the ratio of the edges number compared to the total number of possible edges in a graph, which helps to define the size of the graph [83].

2. **The clustering coefficient** is a metric for connectedness and cohesiveness. It was created with the intention of measuring the degree to which a network exhibits the characteristic that the mean clustering coefficient reflects the propensity for nodes in a graph to cluster together [84]. The local clustering coefficient of a graph is given as follows for a weighted network:

$$C= \frac{1}{N} \sum j \in G_i \; C(i) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (2.18)$$

3. Global Efficiency (GE): is the average inverted shortest path length throughout the network. In addition, represents the effectiveness of the interaction throughout the entire graph and is a measure connected to information exchange [85]. The mathematical definition of GE is:

$$E_{global} = \frac{1}{N(N-1)} \sum i, j \in N, i \neq j \frac{1}{d_{i,j}} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2.19)$$

4. Local Efficiency (LE): is a metric for the system's fault tolerance that assesses how effectively nodes communicate with one another when a node is removed [86]. The definition of mathematics is.

$$E_{local} = \text{N} \sum i \in G \; E_{global}(G_I) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (2.20)$$

5. Average Weighted Degree (AVGW): of each node represents the sum of the weights of all the edges that are related to that node. Different types of EEG produce varying edge weights in their nodes, which causes the edge weight to differ between them [56].

The following is the AVGW mathematical definition:

$$\text{AVGW} = \sum_{j \in B(i)} Wij \;\; \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (2.21)$$

Where B(i) denotes the area around node i, while wij stands for the weight of the edges connecting nodes i and j. Additionally, the network's AVG is the network's average mean of all the weights of incident links across all of its vertices.

6. Graph Average Degree (AVGD): in graphing, the term "average degree" refers to a graph invariant that is equal to the arithmetic mean of all vertex degrees. It has several uses, including calculating the irregularity degrees of networking and in the social sciences [87]. In other words, the average degree of a graph is just the average number of edges per node. Calculating this is a rather simple process.

$$AVG=\frac{Total\ Edges}{Total\ N0des}=\frac{M}{N} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (2.22)$$

6. Modularity: Newman first proposed the idea of modularity idea. Instead of being a way to find modules (communities), modularity is a quality function. It is a metric for how well the network has been divided into clusters (or communities).

Newman established the idea of modularity in a complicated network. Modularity is not a way to find modules (communities), but a quality function [88]. Newman states that the modularity Q is defined as follows if G is a weighted network and A is that network's weighted adjacency matrix:

$$Q=\frac{1}{2m}\sum_{i,j}\left(A_{i,j}-\frac{k_i k_j}{2m}\right)\delta\left(C_i,C_j\right) \quad \ldots\ldots\ldots\ldots\ldots \quad (2.23)$$

Where $A_{i,j}$ represent the weight of the edges (links) connecting node i.; $m=1/2\sum_{ij}A_{i,j}$ indicate the total number of network links, $k_i=\sum_j A_{i,j}$ is the weighted vertex degree of i, C_i. is the name of the cluster to which vertex i belongs., $\delta\left(C_i,C_j\right)$ is 1 if both nodes i and node j belongs to the same cluster otherwise $\delta\left(C_i,C_j\right)$ is equal to 0.

7. **Betweeness Centrality (BC):** BC measures how frequently a node operates as a link between any two nodes' most powerful connections. As a result, nodes with high BC values take part in a lot of shortest paths.

9. Average Shortest Path (ASP): in order to minimize the cost (distance, time, risk, etc.), the decision-maker must generally determine the shortest path between the source and destination nodes in the sequence of edges and vertices (or arcs) that takes a specified source and destination [89]. The number of steps taken along the shortest paths for all potential pairs of network nodes is the average shortest path length, which is a notion in network topology. It gauges how well people can move about on a network using information or other means.

2.11 Unbalance Dataset

The features of the data tend to change with time, and as there are not an equal number of learning instances in each class under consideration, this distribution makes it challenging to categorize the data sets. Class imbalances, which arise from the situation that some data streams that do not match the conditions are disregarded when dealing with actual issues because of the lack of instances and low priority, are the primary characteristic of the unbalanced data set. Because occurrences of the minority class are typically portrayed as positive instances and cases of the majority class are typically portrayed as negative instances [90]. Due to the underrepresentation of minority classes, this problem arises frequently.

As a result, Minority class outcomes could be subpar. Whenever the minority class is especially important, special attention must be given to it. The majority of evaluations of classification techniques for unbalanced data sets start with data-level techniques, which are usually classified into under-sampling and over-sampling techniques. There are numerous cutting-edge technologies available today for the categorization of unbalanced data sets, from the most fundamental under-sampling and over-sampling methods to real-valued negative selective over-sampling.

2.11.1. SMOT-Based Sampling Method

The sampling algorithms used by (Synthetic Minority Oversampling Technique) SMOTE are based on an oversampling technique put out by Chawla et al. [91], which builds on the program's strong points and makes a number of enhancements. By integrating between clustered minority samples, the SMOTE algorithm increases the sample size. Instead of oversampling by replacement, it creates "fake" examples of the minority class. A comprehensive instance is constructed in a less application-specific manner by operating in the "function space" as opposed to the "data space" as in the past.

The minority class is oversampled by using samples from each minority class and providing thorough examples along any minority class's nearest neighbor line segment. Samples from neighbors are randomly selected based on the quantity of oversampling needed.

2.11.2. Edited Nearest Nighbor method ENN

ENN is the common under-sampling algorithm, which deletes samples by searching whether the classes of majority samples are the same as those of the *k*-nearest neighbors. ENN eliminates "noisy sample" to balance two groups of samples. The samples that can be eliminated, however, are few in number because majority samples are frequently the majority samples' neighbors [92].

Edited Nearest Neighbors Rule employs k = 3 nearest neighbors to find the dataset's incorrectly categorized samples, after which they are deleted in order to apply the K = 1 classification rule. Dennis Wilson put up this resampling and categorization strategy in 1972.

2.12 Hyper Parameters

For various issues or datasets, multiple ML methods are appropriate. Choosing the best algorithm and fine-tuning the model's hyper-parameters (HPs) are two of the many difficult and time-consuming steps involved in creating an efficient machine learning model. Machine learning models have two different types of parameters: model parameters, which are initialized and updated during the data learning process (for example, the weights of neurons in neural networks), and hyper-parameters, which cannot be directly estimated from data learning and must be set prior to training an ML model because they define the model architecture [93].

Models may have a large number of hyper parameters, and determining the ideal set of parameters can be approached as a search issue. The "GridSearchCV" method is the most effective one for tweaking hyper parameters. The machine learning model is assessed for a variety of hyper parameter values in the GridSearchCV technique. This method is known as GridSearchCV because it analyzes a grid of hyper parameter values to find the optimum set of hyper parameters. For instance, suppose we wanted to set two hyper parameters of the Logistic Regression Classifier model, C and Alpha, with various sets of values as shown in figure 2.8. The best model will be created using the grid search technique by building multiple iterations of the model using all feasible combinations of hyper parameters. For C = [0.1, 0.2, 0.3, 0.4, 0.5] and Alpha = [0.1, 0.2, 0.3, 0.4], as in Figure (2.7). The performance score is 0.726 (Highest) for the pairing of C=0.3 and Alpha=0.2, hence it is chosen.

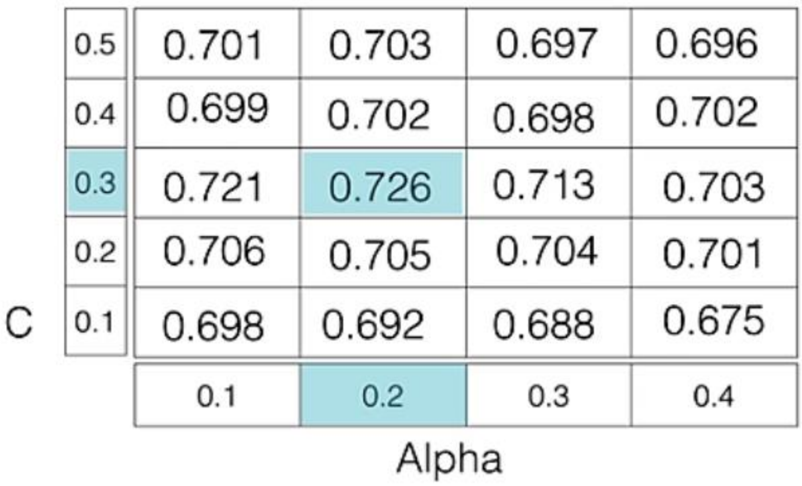


Figure 2.8 An example to represent hyper parameters of Logistic Regression Classifier

CHAPTER THREE

THE PROPOSED SYSTEM

CHAPTER THREE

THE PROPOSED SYSTEM

3.1. Introduction

This chapter introduces an overview of the proposed system at all its levels and its basic steps in more detail. Also, it provides a comprehensive illustration of the feature extraction techniques, machine learning models, an IFTTT protocol, and compression methods. The overall design and each method that was utilized to construct the healthcare system should aid in improving knowledge of the suggested protocol.

3.2. The Proposed Method

In this thesis, we present a real-time, accurate, and efficient IoMT framework for epileptic seizure identification. The suggested model is split into three tiers: the end device tier, which includes the Bluetooth-enabled EEG headset; the edge/fog tier, which includes the computational services; and the cloud tier, which includes cloud storage and other services.

To begin processing at the device tier, an EEG collection module, in the form of a head-mounted EEG headset, is used to collect raw EEG data. An EEG headset is used to record electrical brain activity by placing electrodes on the head. These electrodes can measure small changes in voltage caused by current waves travelling through neurons in various regions of the brain. When an EEG headset is used, the raw EEG signals are sent over Bluetooth to the subsequent layer periodically.

An edge/fog-based tier collects these signals digitally, and then sends them to a cloud-based server at the cloud tier through a wireless or cellular network. Within the context of this thesis, the edge/fog computing tier provides an additional processing unit between the loT devices and the cloud tier as an intermediate tier between them. This thesis proposes implementing an edge/fog computing tier as an extra processing unit to decrease latency while also improving system reliability and energy efficiency. Additionally, it offers benefits beyond those of cloud computing by providing access to processing power, storage space, networking resources, and real-time data analysis.

The proposed approach for tiered automated epileptic seizure detection consists of six layers, particularly, the Data Collection Layer (DCL), Data Transmission Layer (DTL), Information Analysis Layer (IAL), Seizure Classification Layer (SCL), Decision-Making Layer (DML), and Cloud Storage Layer (CSL). The flowchart of the proposed system is shown in Figure 3.1. A detailed explanation of these layers will be given in the following subsections:

3.2.1 Data Collection Layer (DCL)

The epilepsy patient's scalp EEG data samples are collected by the data collecting layer (DCL) through a wearable EEG headset. This EEG device can be used in real-time because it collects patient data every second, enabling it to successfully serve as a lifesaver for epileptic patients. We need to find out a different approach to acquire data until a fully-functional, live IoMT system is available that can capture from patients health factors important to epileptics. Fortunately, there exist online repositories where users may have free access to high-quality datasets.

Throughout this thesis, we have studied the freely accessible EEG repository of epilepsy data that has been established and provided by the department of epilepsy at Bonn University, Germany. The five sets referred to as A, B, C, D, and E make up the entire EEG database. 100 single channels, 23.6-s EEG signals from five different classes are included in each set.

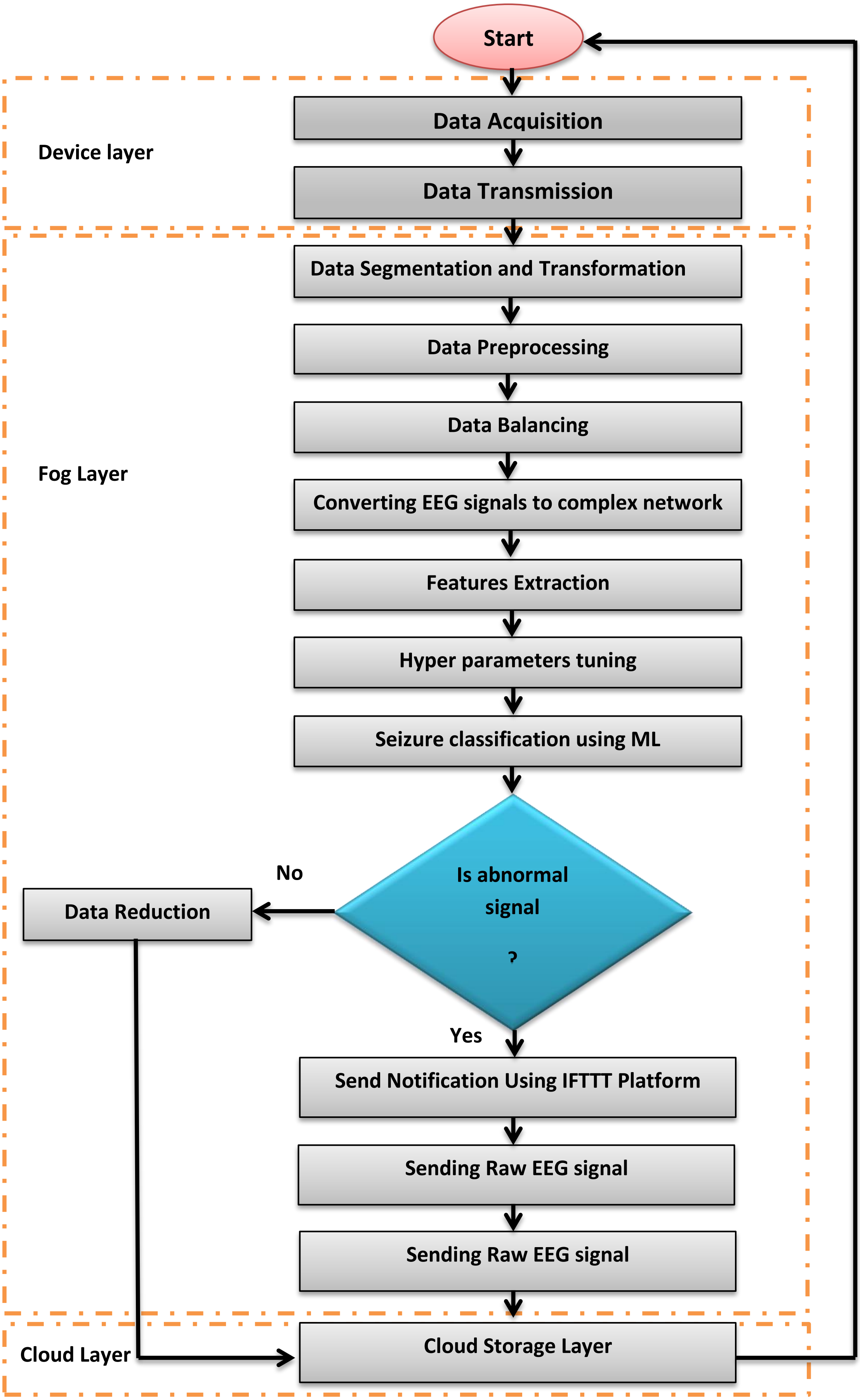


Figure 3.1 The flowchart of the proposed system.

3.2.2 Data Transmission Layer (DTL)

The data transmission layer (DTL), which is a part of the device tier in this paradigm, is responsible for sending the data of EEG that has been detected by the EEG headset to the fog server. Low-energy Bluetooth technology is used to send the digitally stored EEG data samples that the EEG headset senses to the edge/fog layer. Then, these data samples are sent from edge/fog layer to the cloud server by using the 4G network or Wi-Fi to connect to the Internet after preprocessed.

3.2.3 Information Analysis Layer (IAL)

In this architecture, this layer serves as a sub layer of the edge/fog tier. The main task of the IAL is to carry out the data segmentation and transformation, data preprocessing, balancing the dataset, converting EEG signals to complex network, and feature extraction for the dimensionality reduction of raw samples of EEG. Dimensionality reduction is used to eliminate duplicate features and choose just the appropriate information.

A- Data Segmentation and Transformation

The next critical stage, following data collecting, aims to transform a signal from its original 1-D format into a 2-D table. The goal here is to streamline analytical processes and provide vital information. This information is in its unprocessed "raw" form at this time. Therefore, it is inappropriate to provide such details at this time. In addition, this phase introduces the dataset as supervised by including possible value suggestions for the class feature.

The original dataset from the reference consists of 5 different folders, each with 100 files, with each file representing a single subject/person. Each file is a recording of brain activity for 23.6 seconds. The corresponding time-series is sampled into 4097 data points. Each data point is the value of the EEG recording at a different point in time. So, we have total 500 individuals with each has 4097 data points for 23.6 seconds.

To illustrate the frequency-dependent differences between epileptic and non-epileptic events, we have randomly drawn a few examples of each kind of waveform as shown in figure (3.2). As a result, the data on epileptic seizures seems smoother and tends to follow a certain pattern.

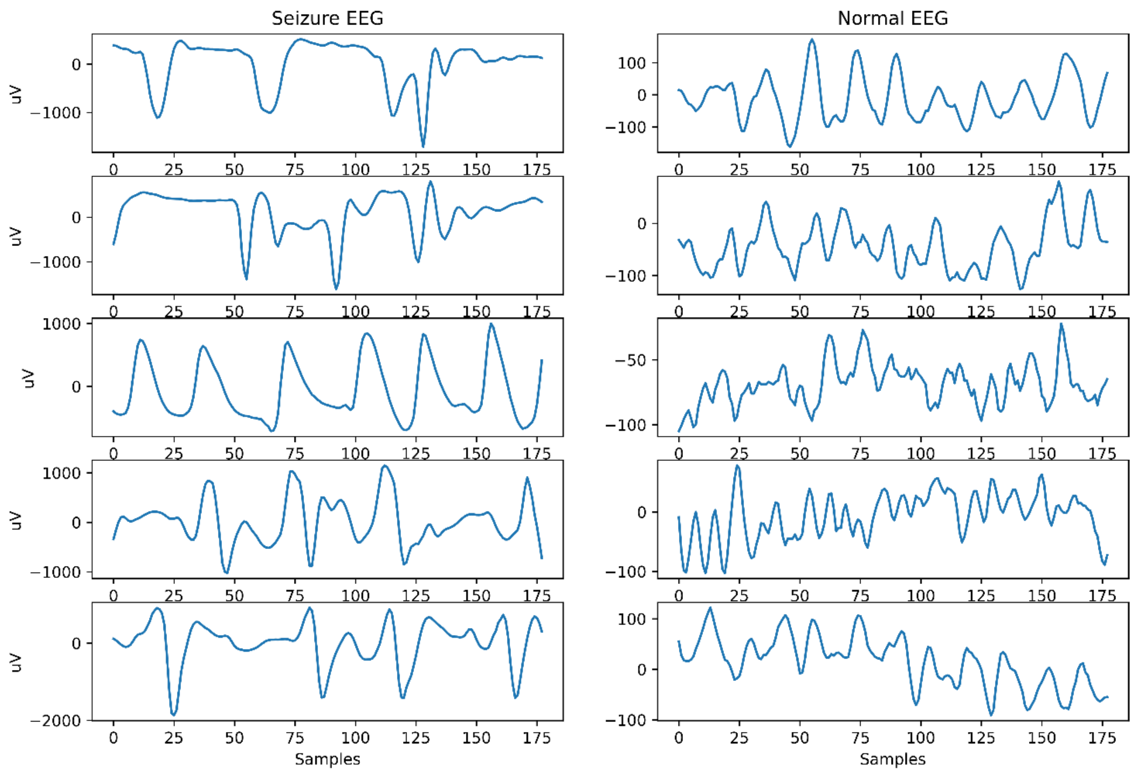


Figure 3.2. Frequency-dependent differences between epileptic and non-epileptic.

In order to make the system work periodically and every second and reduce the computation time, Over the EEG data, a moving window of a particular size is moved, with the moving distance being equal to the window's width. This has been done in order to statistically compile all of the data found in the window area for a specific period of time. Understanding the pattern of the EEG signals over time and how the signal varies within a specific time frame is the primary goal of employing a moving window.

Hence, we had segmented every 4097 data points into 23 chunks using a moving window of 1 second length, each chunk contains 178 data points for 1 second, and each data point is the value of the EEG recording at a different point in time. So, now we have 23 x 500 = 11500 pieces of information (rows), each information contains 178 data points (columns) for 1 second, the last column represents the label y, see Figure (3.3).

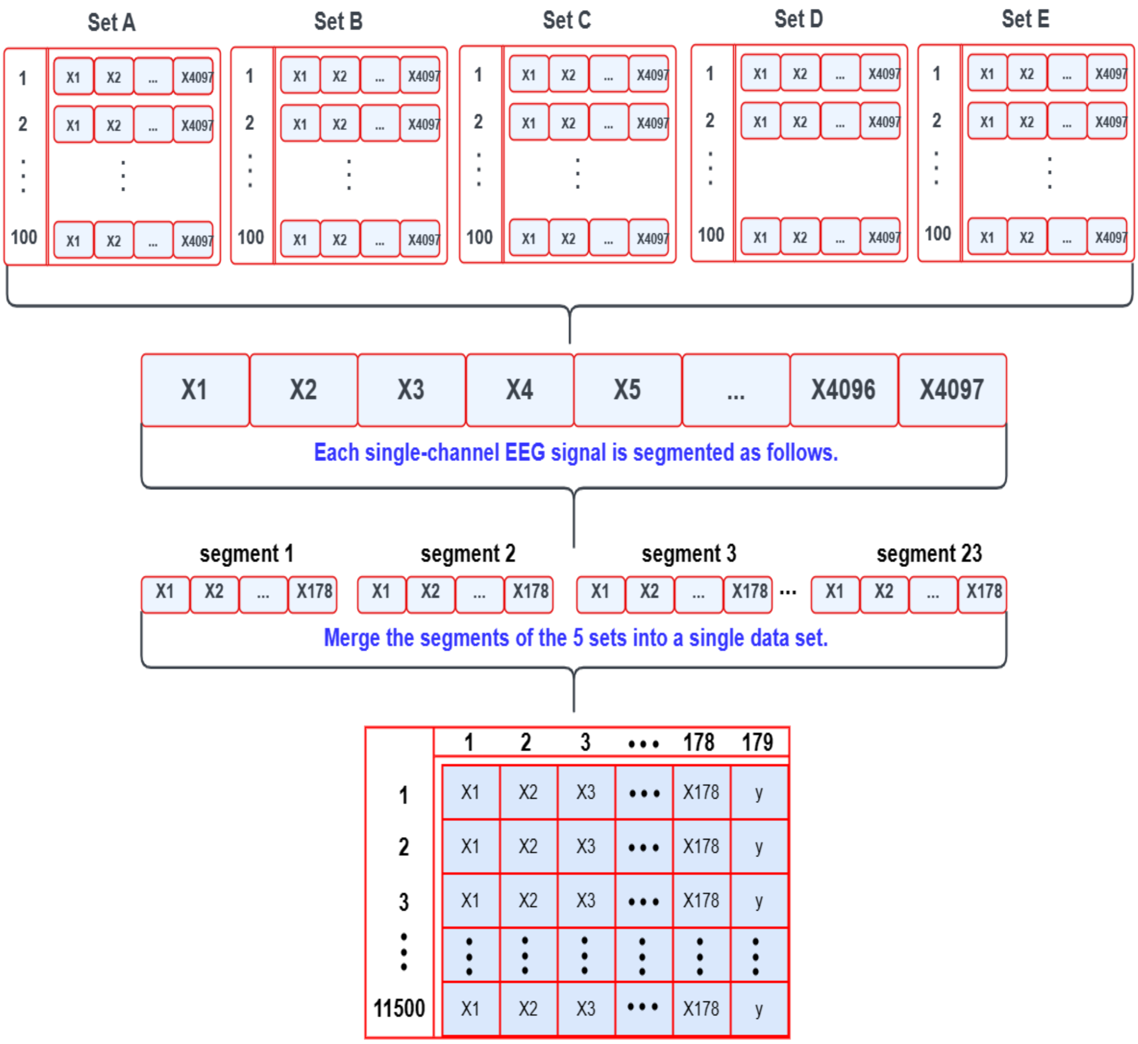


Figure 3.3 Datasets segmentation and transformation.

The response variable is y in column 179 was the classification of seizure activity (label) on a scale of 1-5. The description for the values in the response variable is provided in Table (3.1).

Table 3.1 Explanation of the values of the EEG response variables.

Response Value	Description
1	The patient exhibits seizure activity
2	Section of the brain has a tumor, but no seizure activity
3	Section of the brain previously had a tumor, but no seizure activity
4	The patient has eyes closed, but no seizure activity
5	The patient had eyes open, but no seizure activity

From the first view we can assume we need to solve multi-classification task, but, after accurate exploring definitions of classes of y, we can realize we can reform our multi-classification task to a binary classification task. All of the subjects in classes 2, 3, 4, and 5 did not experience epileptic seizures. Only class 1 has epileptic seizures. For this, we may simply merge classes 2, 3, 4, and 5 into class 0 (not epileptic seizure), while maintaining class 1 as 1 (epileptic seizure). Algorithm 3.1 depicts the steps of segmentation and transformation.

B- Data Preprocessing

Prior to completing feature extraction or classification activities, data processing is necessary. The acquired input EEG signals typically contain duplicate data along with unwelcome noise and artifacts. Denoising the signal is crucial in order to remove noise and artifacts that taint the original signal during recording and processing. Some examples of these artifacts are muscle artifacts, environmental artifacts, power line interference, etc. It is crucial to eliminate these uncertainties in order to get the data ready for subsequent processing. Only pertinent signal-related characteristics will be included into EEG signals.

Algorithm 3.1: Segmentation and Transformation.

Input: $EEG_{datasets}$: comprises of five sets A, B, C, D and E;

$\mathcal{W}$: size of moving window

Output: EEG_{SEG}: $segmented\ EEG_{datasets}$

$EEG_{SEG} \leftarrow [\ \emptyset\ ]$

$\boldsymbol{For}\ i \leftarrow 1\ to\ 5\ \boldsymbol{do}$

  $\mathcal{R} \leftarrow EEG_{datasets}[i]$

  $\mathcal{S} \leftarrow 0$

  $\boldsymbol{For}\ j \leftarrow 1\ to\ len(\mathcal{R}/\mathcal{W})\ \boldsymbol{do}$

    $k \leftarrow 0$

    $\boldsymbol{For}\ p \leftarrow \mathcal{S}\ to\ len(\mathcal{R})\ \boldsymbol{do}$

      $X[k] \leftarrow \mathcal{R}[p]$

      $\boldsymbol{if}\ (k < \mathcal{W})\ \boldsymbol{then}$

        $k \leftarrow k + 1$

      $\boldsymbol{else}$

        $y \leftarrow \boldsymbol{class}(EEG_{datasets}[i])$

      $\boldsymbol{end\ if}$

    $\boldsymbol{End\ for}$

    $EEG_{SEG} \leftarrow \boldsymbol{append}([X, y])$

    $\mathcal{S} \leftarrow \mathcal{S} + \mathcal{W}$

  $\boldsymbol{End\ for}$

$\boldsymbol{End\ for}$

$EEG_{SEG}[:, \boldsymbol{y}] \leftarrow \boldsymbol{map}(\{5,4,3,2\}, 0)$

Return EEG_{SEG}

The preprocessing operations executed in this thesis are as follows:

1- Remove unnamed column (it has information which we don’t need).

2- Reorganize the data due to earlier merging and mapping of the 2, 3, 4, and 5 classes as 0 classes (not epileptic seizures).

3- Checking missing data.

4- Normalizing EEG data using standard scalar function, in the raw EEG data, there is a very big difference in mean and standard deviation between epileptic and non-epileptic data, so this will demand that we normalize or scale our data.

C- Balancing the Data Set

The collected data sets include two class values: seizure and non-seizure, which demonstrate that seizure records are in the lowest percentage and non-seizure records are in the greatest percentage. Due to the lengthy recording times (over an hour), the EEG data sets are quite unbalanced, and patients only have seizures briefly (a few seconds at most).

A description of the imbalanced data sets has been presented in the proportion of seizure and non-seizure recordings in each patient data set are shown in Figure (3.4). In cases when the distribution of class values for a particular training data set is unbalanced, the conventional classifiers may not function properly. Since the class value in the minority has the least influence on accuracy, they become prejudiced and disposed toward the class value that is in the majority.

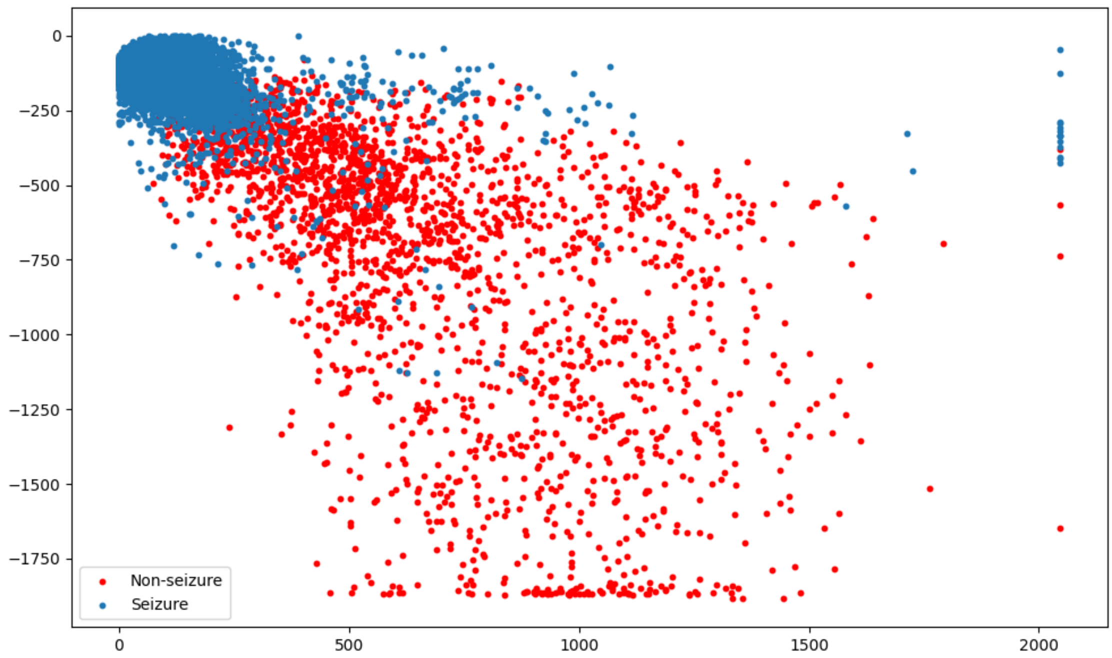


Figure 3.4 Scatter plot of values for seizure and non-seizure occurrences.

One of the solutions to overcome that weakness is to generate new examples that are synthesized from the existing minority class. In this thesis, we suggest combining oversampling and under sampling methods to balance the dataset better by combining ***Synthetic Minority Oversampling Technique*** and the ***Edited Nearest Neighbor*** method — or in short, SMOTE-ENN. The suggested technique combines the strengths of SMOTE and ENN as shown in figure (3.5). The former being able to produce synthetic instances for the minority class and the latter being able to eliminate observations from both classes when they are found to have a different class from their K-nearest neighbour in the majority class. Figure (3.6) displays the flowchart of SMOTE-ENN.

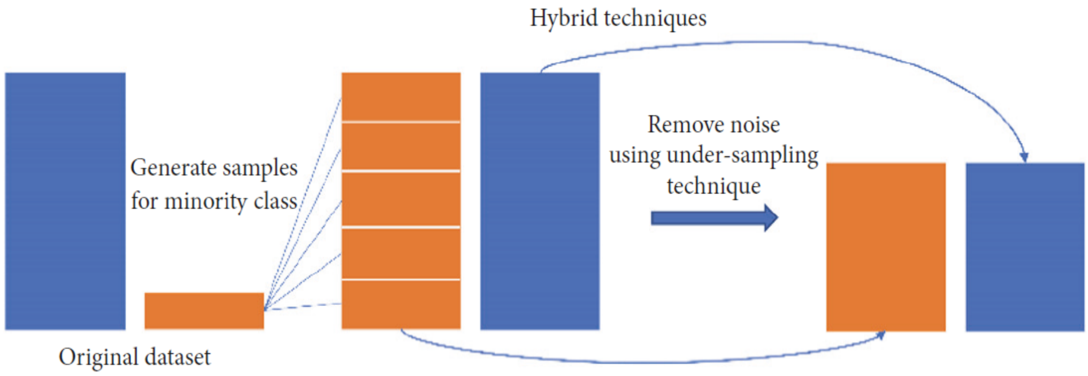


Figure 3.5 The illustration of oversampling and under sampling hybrid technique.

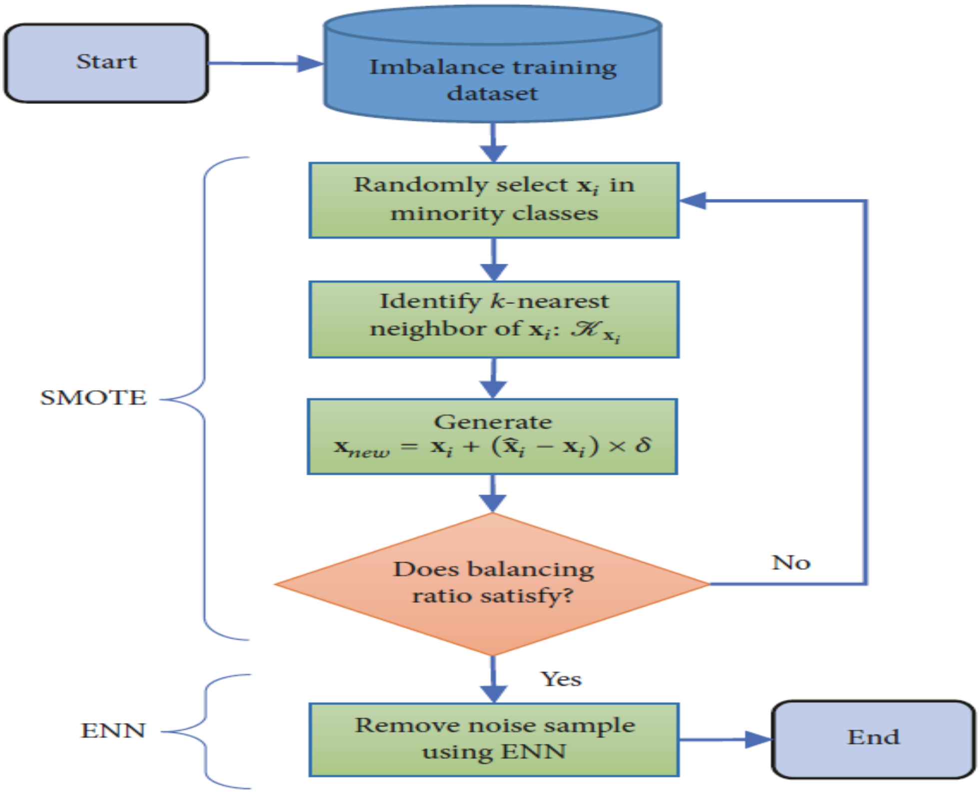


Figure 3.6 The flowchart of SMOTE-ENN algorithm.

The steps involved in SMOTE-ENN are as follows:

1. (Beginning of SMOTE) Select arbitrary information from the minority class.

2. Determine the separation between the randomly generated data and its k nearest neighbors.

3. To create a synthetic sample, multiply the difference by a random value between 0 and 1, and then add the result to the minority class.

4. Keep going back to steps 1-3 until the necessary percentage of the minority class is reached. (SMOTE ends here)

5. (Beginning of ENN) K is the number of neighbors that are closest to the observation. Unless otherwise specified, K=3.

6. Find the K-nearest neighbor of the observation among the other observations in the dataset, then return the majority class from the K-nearest neighbor.

7. If the class of the observation and the majority class from the observation's K-nearest neighbor is different, then the observation and its K-nearest neighbor are deleted from the dataset.

8. Repeat step 2 and 3 until the desired proportion of each class is fulfilled. (**End of ENN**)

D. Creating a Complex Network from EEG Signals

Graph theory, statistical physics, and data analysis are all topics covered under the complexity science subfield known as complex network theory. Visibility Graph (VG) is one of the methods among them. The electrical activity of brain cells as measured on the scalp or cerebral cortex surface is known as the EEG signal. It clearly exhibits dynamic, non-stationary, and non-linear properties. The VG approach offers a tool to investigate the EEG data's underlying dynamics. Since the VG may inherit the dynamic nature of creating time series data, this technique has the characteristics of defining time series from the perspective of graph theory. In this thesis, Weighted Visibility Graphs (WVG) are created using EEG time series data. The weighted visibility graph is created utilizing the natural visibility graph and the subsequent stages.

- **STEP I**

Regarded each time series data sample point as a node in the network depicted in Figure (3.7). In order to create a WVG from EEG time series data, consider $G = (V, E)$ be a graph where $V = \{n_i\}, i = 1, 2, \dots, N$, are the nodes and $E = e_i, i = 1, 2, 3, \dots, N$, are the edges of the graph. A time series $x(t_i), i = 1, 2, \dots, N$ of N sampling points and node n_i correspond to data sample point x_i.

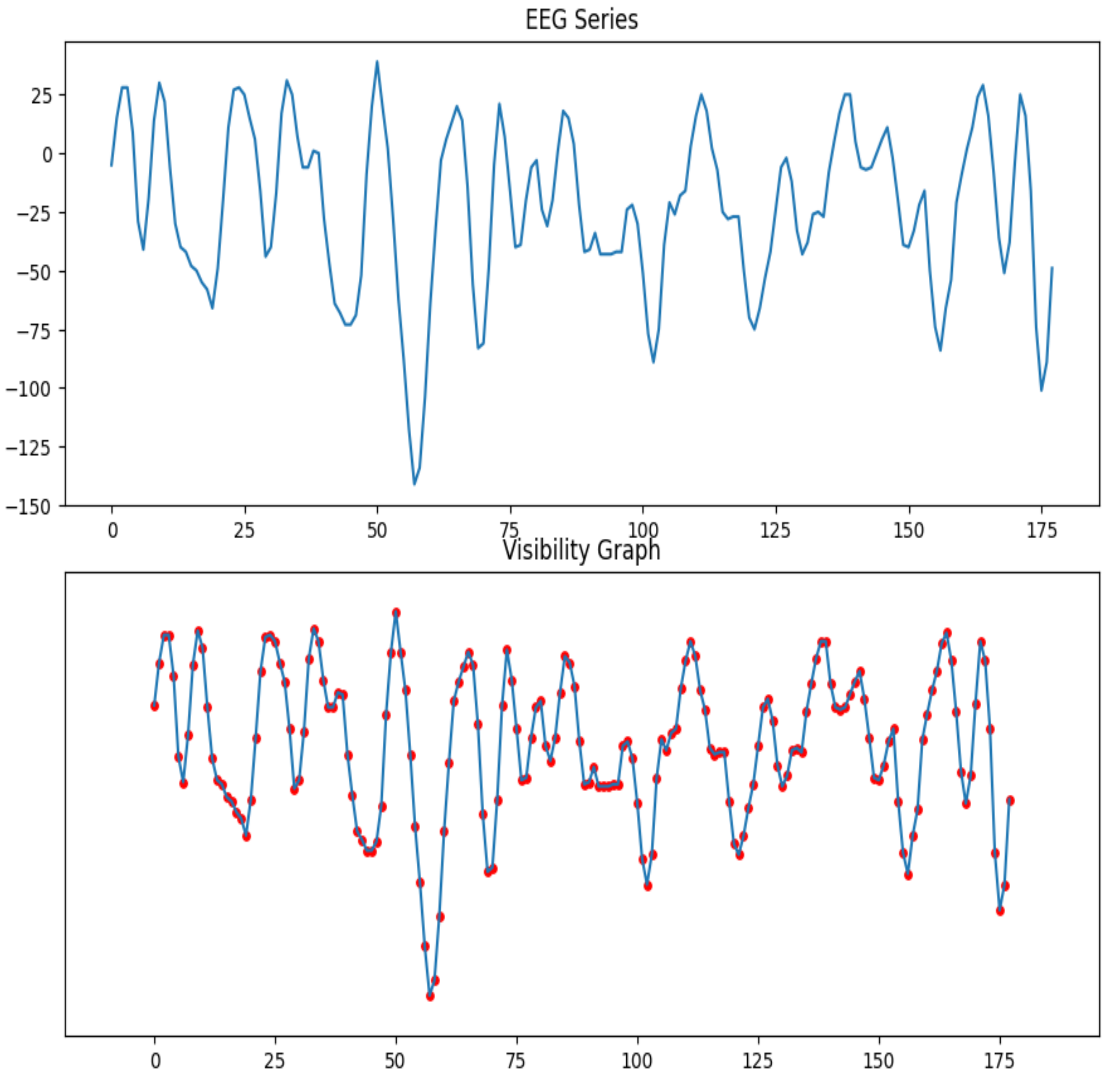


Figure 3.7 Consider each data sample points of EEG as the nodes of the graph.

- **STEP II**

The natural visibility graph equation serves as the foundation for the edges (links) connecting the WVG's nodes to determine the relationships between the different nodes of the weighted visibility graph, we used the natural visibility graph technique. The visibility graph is built on the concept of the Euclidean plane, where each vertex represents the location of a point and links between linked nodes are only possible if there is any visibility between them. The edges connecting any two pairs of nodes only leave the graph using the VG approach if they comply with the following rule:

$$x(t_c) < x(t_a) + \left(x(t_b) - x(t_a)\right)\frac{t_c - t_a}{t_b - t_a}, \; a < c < b \qquad (3.1)$$

Where, $x_a = x(t_a)$ and $x_b = x(t_b)$ are the data sample points and t_a and t_b are any two arbitrary time events and t_c is any event exits between them i.e. $t_a < t_c < t_b$. The VG of time series data that represents edge creation based on visibility among them is shown in Figure (3.8) with a corresponding edge diagram.

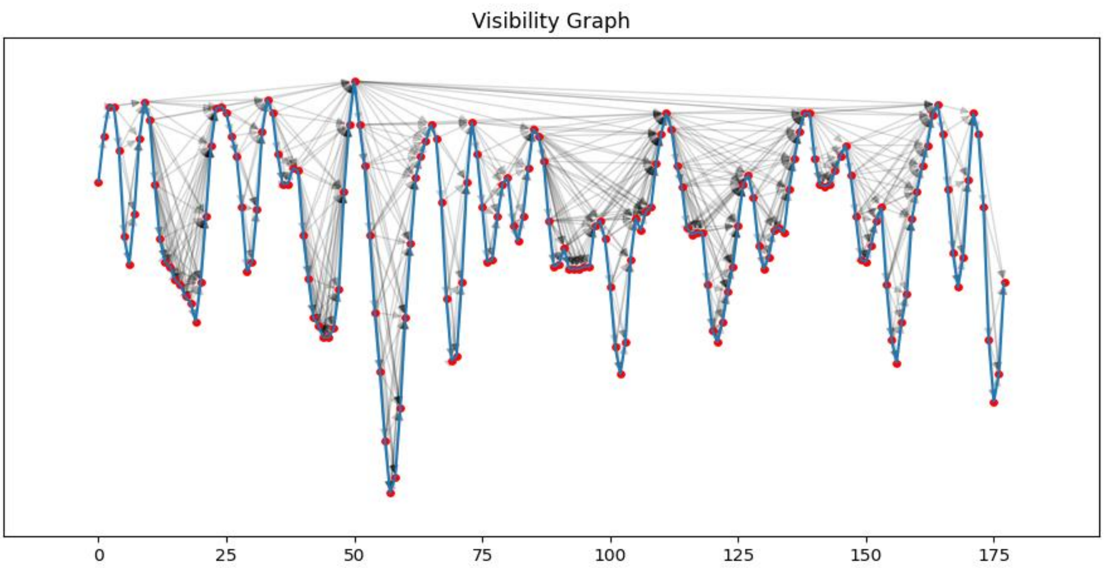


Figure 3.8 Represents the edge construction based on visibility among them.

- **STEP III**

Find the difference in edge weight between two nodes. By retaining weight information in a complex network as opposed to a binary network (which simply provides information on the links' status as either exits between two nodes or not), more reliable results may be produced, according to literature research in the field of network theory Because various edges have varying strengths, weighted complex networks play a vital role in identifying weak and perhaps less relevant edges (links). In this thesis, a WVG is created to characterize EEG time series data. . A graph's edge has a weight value that describes it. A triple $G\ (V, E, w)$ can be used to represent a weighted graph where w: $E \rightarrow R$ weighted function.

In this thesis, all the edges of the graph are directional in nature as a link between node $n_a = x_a$ and node $n_b = x_b$ is seen as having direction from n_a to node n_b where $a < b$ as shown in Figure (3.9).

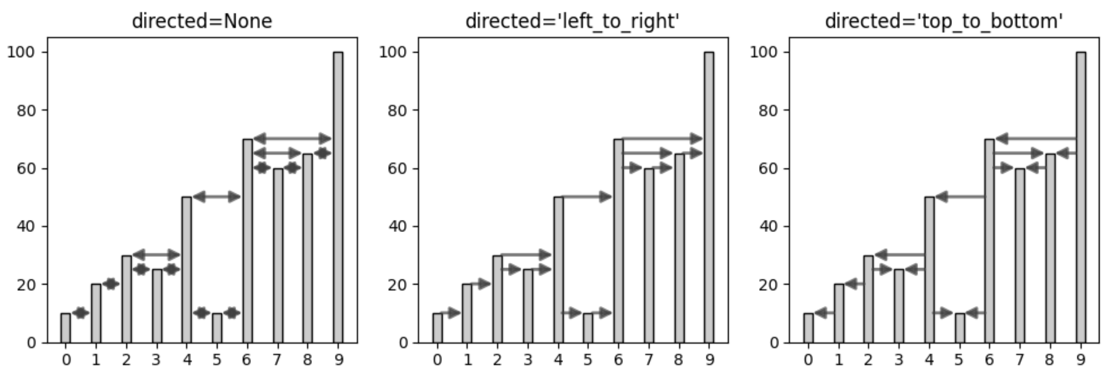


Figure 3.9 Represents the different directions of visibility graph.

The absolute value of edge weight has been taken into consideration in this thesis. Equation 3.2 is used to determine the edge weight.

$$w_{ab} = \arctan\frac{x(t_b) - x(t_a)}{t_b - t_a},\ \ a < b \qquad \ldots\ldots\ldots\ldots\ldots\ (3.2)$$

Where, in this thesis, we have taken into consideration all of the edge weight values in radian function, w_{ab}denotes the weight of the edge between nodes n_a and n_b. The inverse trigonometric function is known as the $arctan$ assists in identifying rapid changes in EEG data. Figure (3.10) represents the different weights that can be used in a weighted visibility graph. In example 3.1, we succinctly outline the advantages of using the aforementioned edge's weight while processing EEG data.

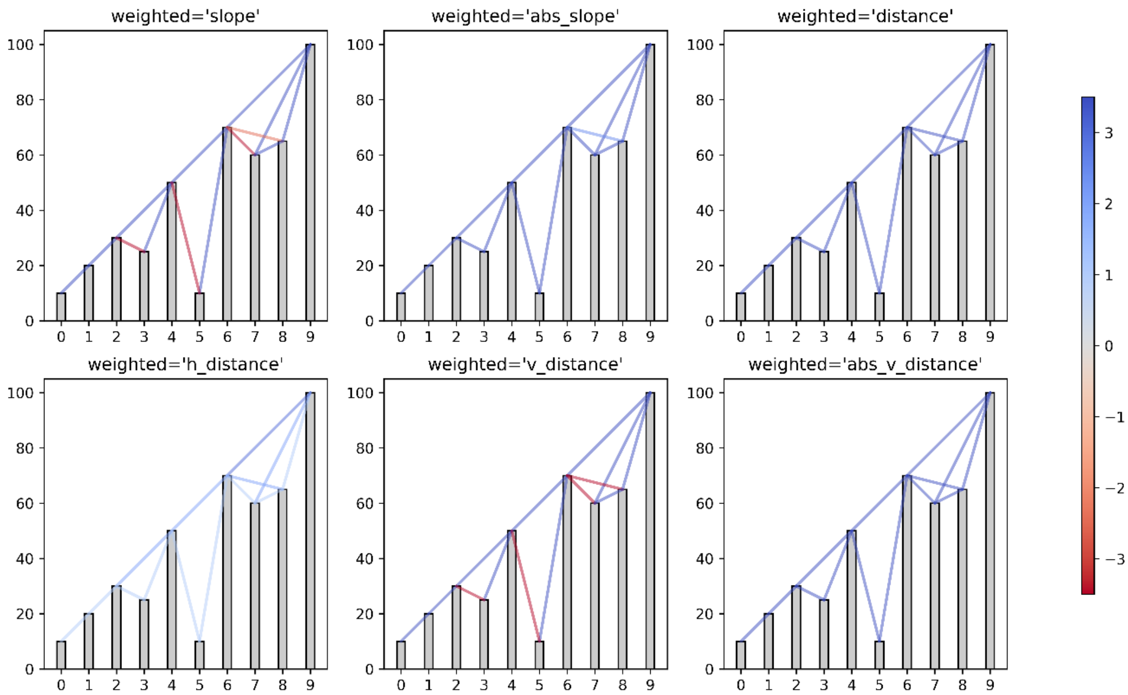


Figure 3.10 The different weights that can be used in a WVG.

Example 3.1: Let's consider an EEG time series $X = \{10, 20, 30, 25, 50, 10, 70, 60, 65, 100\}$ with associated time $t = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$. As it is clear from Figure 3.11 that $x(t_1) = 10$ and $x(t_6) = 10$, i.e. both are having the same value and also there is a sudden fluctuation at $x(t_7) = 70$. The angle between $x(t_1)$ and $x(t_7)$ is α_1 and α_2 is the angle between $x(t_6)$ and $x(t_7)$.

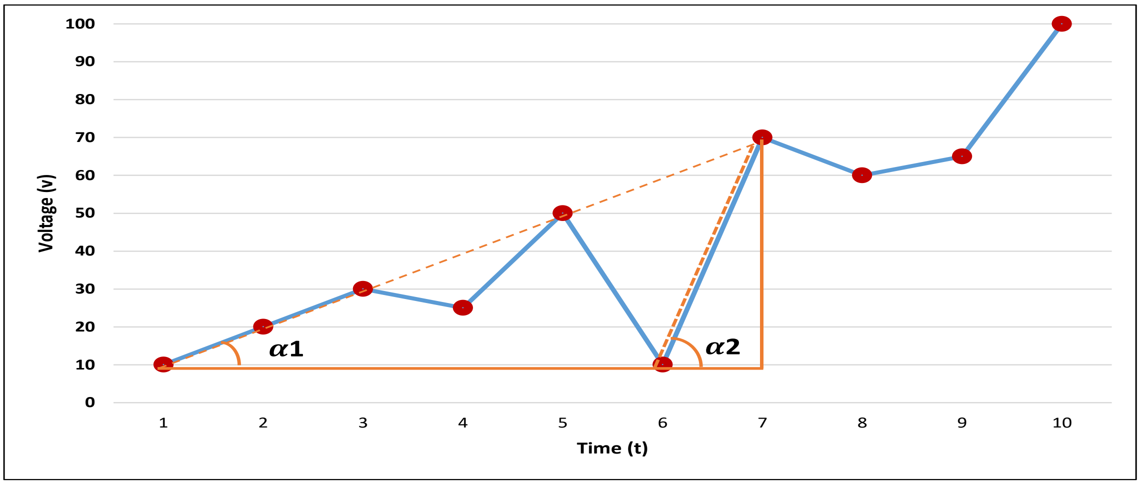


Figure 3.11. An illustration of edge weights between several data points.

According to the equation 3.2 the edge weight between $x(t_1)$ and $x(t_7)$ is:

$$w_{17} = \arctan\frac{70-10}{7-1} = 1.471 = \alpha_1$$

Similarly the edge weight is calculated between $x(t_6)$ and $x(t_7)$ which is:

$$w_{67} = \arctan\frac{70-10}{7-6} = 1.554 = \alpha_2$$

The aforementioned example so demonstrates clearly that even if the two nodes (data sample points) have the identical values, their ability to link with the third node will vary due to the weight of their edges. Edge weights will also fluctuate along with the EEG signal values, aiding in the differentiation of the various EEG signal types.

- **STEP IV**

WVG construction: As illustrated in Figure (3.12), the WVG is finally built using the edge weight value determined in the previous stage.

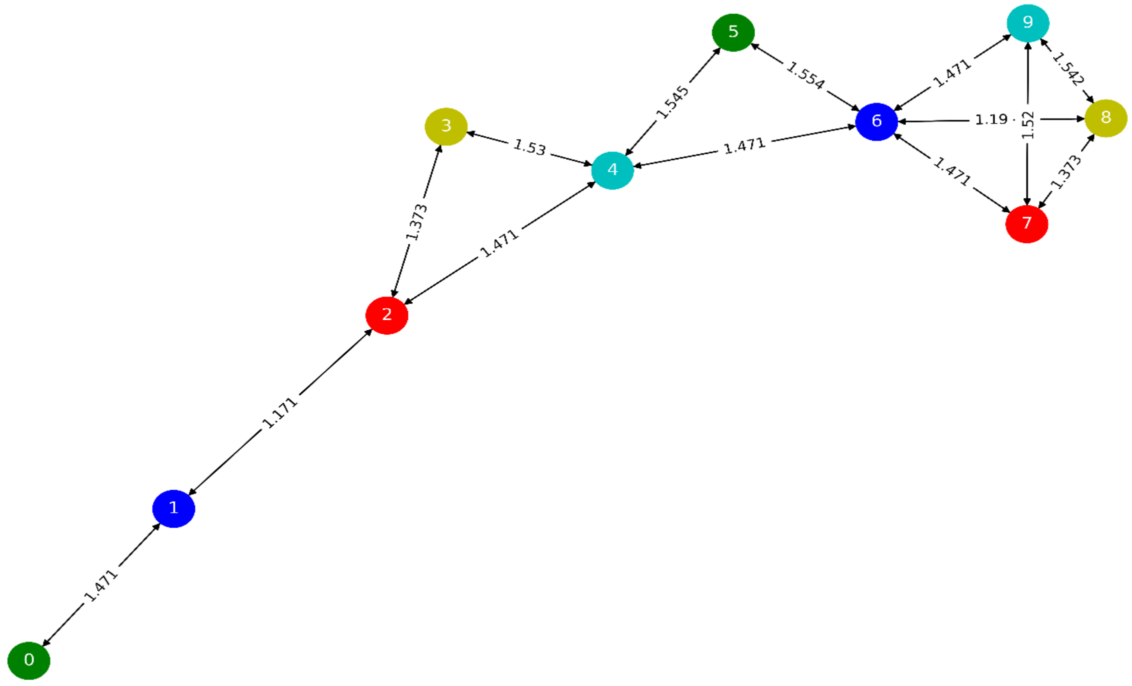


Figure 3.12 Weighted Visibility Graph.

E- Feature Extraction

The next stage in developing a monitoring system that is both dependable and energy-efficient for the identification and reporting of epileptic seizures is to apply feature extraction to the raw data gathered at the DCL layer in order to discover a collection of characteristics associated to epilepsy.

To increase the accuracy of Computer-Aided Design (CAD) systems, particularly IoMT devices, it is imperative to extract relevant information with fewer parameters. Therefore, an important step in classifying EEG signal data is feature extraction. Technically speaking, a feature denotes the distinguishing quality and a measurable quantity derived from a section of a pattern. With the least amount of information loss possible, the feature extraction procedure reduces the huge volume of EEG data into a useful and significant feature vector set. As a result, by facilitating easier computation and speed, it aids in the analysis (classification) process. In this thesis, we have extracted two types of features: statistical (time domain features) from the original signal and graph features from the constructed graph. Ten statistical features shown in figure (3.13) named as min, max, median, mean, entropy, sum, standard deviation, variance, range and absolute difference signal as features are extracted from the original EEG dataset. And nine statistical features of network are extracted named as the clustering coefficient, average weighted degree, global efficiency, local efficiency, graph density, graph average degree, average shortest path, modularity and betweeness of network as features from the weighted visibility graph as shown in Figure (3.14). These features are able to concentrate on the important information that may be learned about the time series by looking at the structural pattern of complicated networks. The weighted visibility graph has a modularity feature that we've introduced to help distinguish between distinct EEG signal types by identifying communities within their complex network. The second introduced feature is average weighted degree. It is evident from section II that because of fluctuations in epileptic EEG signals, the edge weight will diverge, and different types of EEG signals have different edge weights among their nodes, resulting in different average weighted degree values in their resulting networks.

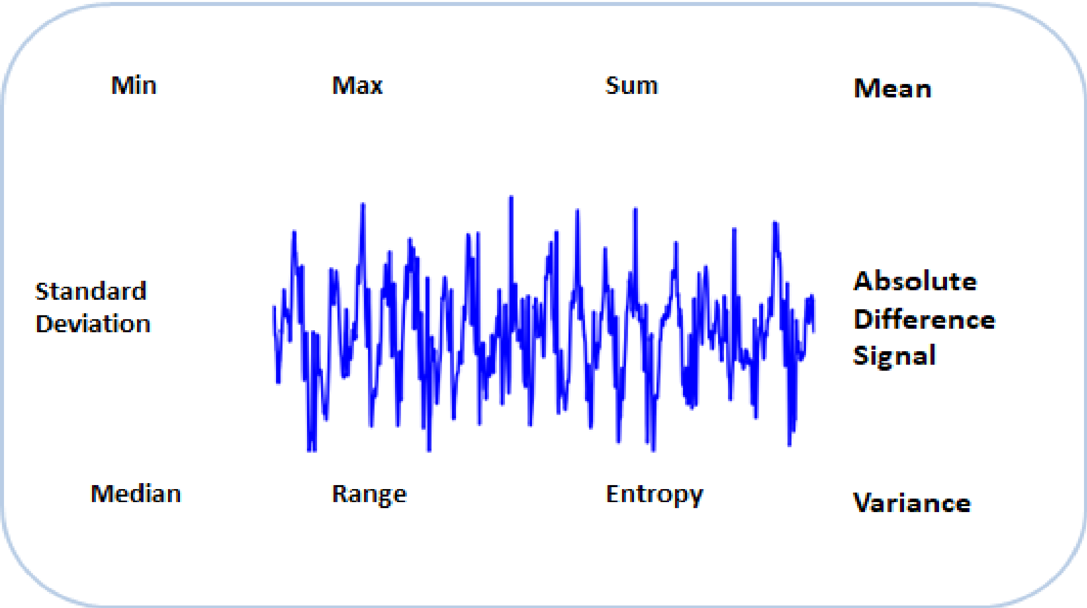


Figure 3.13 Time series-based feature extraction

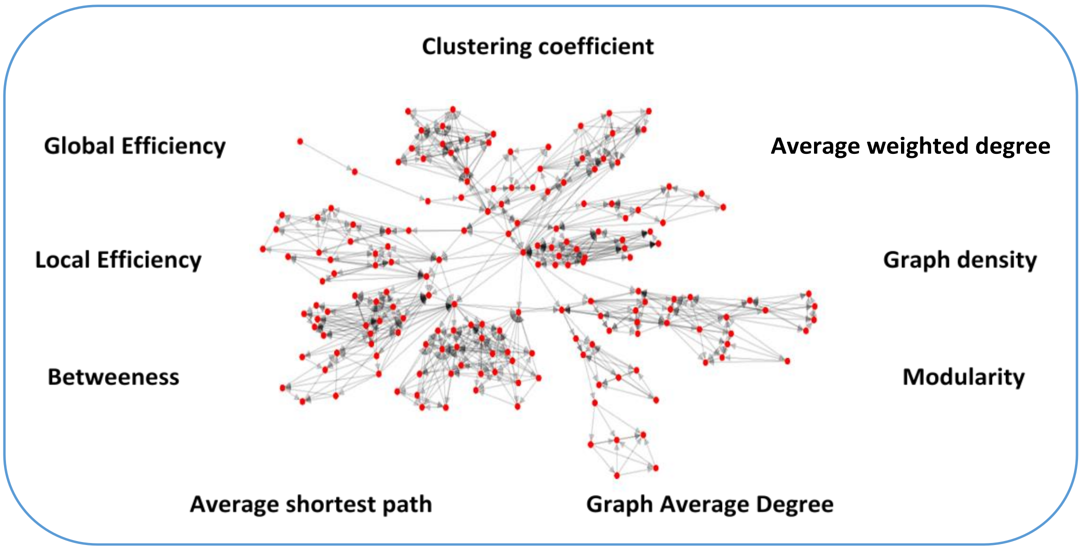


Figure 3.14 Graph-based feature extractions.

1) Modularity

The modularity of the network as described in chapter 2 (section 2.10.2) and also shown in Figure (3.15) is the first extracted feature of the WVG. Because it is a quick and effective approach that is frequently used in the field of community discovery from complex networks, we employed the Louvain method in this thesis to calculate the modularity of the complex network of EEG data.

The Louvain technique has two stages. Each node is initially added to its nearby communities to see which one will increase modularity gain ΔQ. The tiny community discovered in the first stage serves as the node of the newly built network in the second phase, and the weights of new linkages are determined by the sum of the weights of the links between nodes in the corresponding two previous communities. The two procedures will be done again up until the maximum modularity is reached and no further node movement is possible.

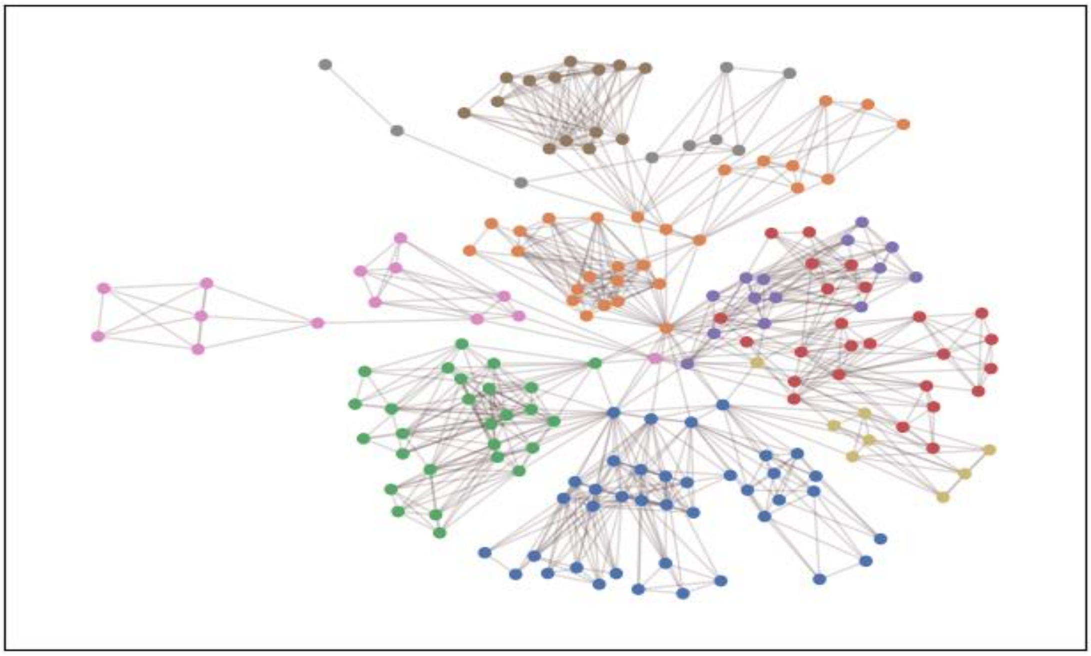


Figure 3.15 The Modularity of WVG.

2) Average Weighted Degree

The second extracted characteristic of the WVG is the network's average weighted degree as specified in section 2.10.2. If a $A_{N\times N} = \{a_{ij}\}$ is utilized to represent the WVG as an adjacency matrix with N nodes, then $a_{ij} = 1$ if node i has an edge leading to node j, else it is 0.

The sum weights of all the edges connected to node i make up the node's weighted degree.

In light of this, for a specific patient, the aforementioned 19 features will serve as an accurate representation of the patient's condition throughout an N-sample time window.

3.2.4 Seizure Classification Layer

Now we present another layer in the edge/fog layer in the direction of a trustworthy, epileptic seizure energy-efficient detection and notification: patient categorization. Scalp samples from a patient with epilepsy by using the SCL layer, EEG are divided into two seizure phases: normal and abnormal. Machine learning models are used to categorize EEG signals based on several selected features derived from IML. Particularly, we make use of the feature extraction methods that were previously reported, and then ten of the different classifiers were tested in this thesis to obtain a higher performance classifier than the others. The classifiers used are support vector machine (SVM), K-nearest neighbors (KNN), logistic regression (LR), decision tree (DT), Naïve Bayes (NB), Random Forest (RF), Extra Tree Classifier (EXT), Gradient Boosting, Extream Gradient Boosting (XGB), and Stochastic Gradient (SG).

This layer is crucial for accurately forecasting the abnormal stage with the greatest accuracy and the shortest time of classification in order to provide real-time services as quickly as possible.

i. Hyper-parameter Tuning of ML

The definition of a machine learning (ML) model is a mathematical model having a set of parameters that must be learned from the data. However, some parameters referred to as "Hyper-parameters" cannot be discovered using simple learning processes. Prior to starting training, individuals frequently choose them based on some intuition or trial and error. Finding optimal hyper-parameter is a very hard task to solve. To solve this problem, in this thesis, the "GridSearchCV" is employed, which employs a dictionary of possible model training parameters. The parameter grid is conceptualized as a dictionary, with the parameters serving as the keys and the settings to be evaluated as the values. This method of searching is used to identify the best hyper-parameters and thus enhance prediction accuracy.

ii. Training of the Module

In machine learning, training is the most crucial phase. ML model receives the extracted features and uses it to identify patterns and generate predictions during training. Each classifier is trained separately. In order for the model to do the task at hand, it must learn from the data. The model improves its prediction skills with training over time.

iii. Testing of the trained Module

An entirely separate set of test samples has been employed as the input for assessing the effectiveness of the approaches. The test samples' features are retrieved using the same feature extraction methods. Then the already-trained classifier is provided with the retrieved features as input. To evaluate the performance of all 10 classifiers, each classifier is tested independently, and the results are then compared. The findings of the analysis of the seizure detection modules' testing are detailed in chapter 4.

3.2.5 Decision-Making Layer

The alert notification and seizure detection should both function very instantly. This implies that the alarms should alert the family members and the medical services as soon as a seizure begins, in the shortest amount of time possible.

The DML is in charge of deciding whether or not a patient is in a safe state depending on the output of the SCL layer. The following formula describes the process of seizure detection:

$$DML = \begin{cases} Seizure & if\ SCL = 1 \\ non\ Seizure & if\ SCL = 0 \end{cases}$$

With the goal of creating a monitoring system that consumes minimal energy, we build a technique that allows a DML to automatically choose the most efficient configuration for sending patient data, based on the patient's status. According to the output of this equation, two actions will be taken in this layer: Either sending a notification or reducing the EEG data.

A. Send Notification

In the event that SCL identifies the epileptic seizure as the result, the DML will decide to send an alert message to the caregiver, his or her family, and any local hospitals or ambulance services in case the patient has a seizure. Also, all of the raw EEG readings are uploaded to a remote cloud service, where they will be archived and analyzed afterward. This way, the right steps can be taken to save the patient’s life. As part of our design, we included the IFTTT (IF This Then That) protocol, a task automation solution for IoT and web services based on a trigger–action paradigm, to enable the sending of email or SMS to doctors and emergency services.

When using IFTTT, there are two scenarios that can happen: IF THIS, and THEN THAT. The suggested system receives the patient's condition—a seizure—and the moment at which it began in the IF THIS state. In this case, if the IF THIS rule is true, the system will transmit an alarm, therefore the suggested system inputs the actions to be taken in the THEN THAT state. THEN THAT takes a patient ID, a period, and an address as inputs.

The patient name and ID are included in the SMS. The doctor can see historical and recent information on the patient's condition. A line graph of the time forecasts were recorded compared to the predictions, as illustrated in Figure 3.16, is one example of how the doctor might use the platform's functionalities to visually display data received.

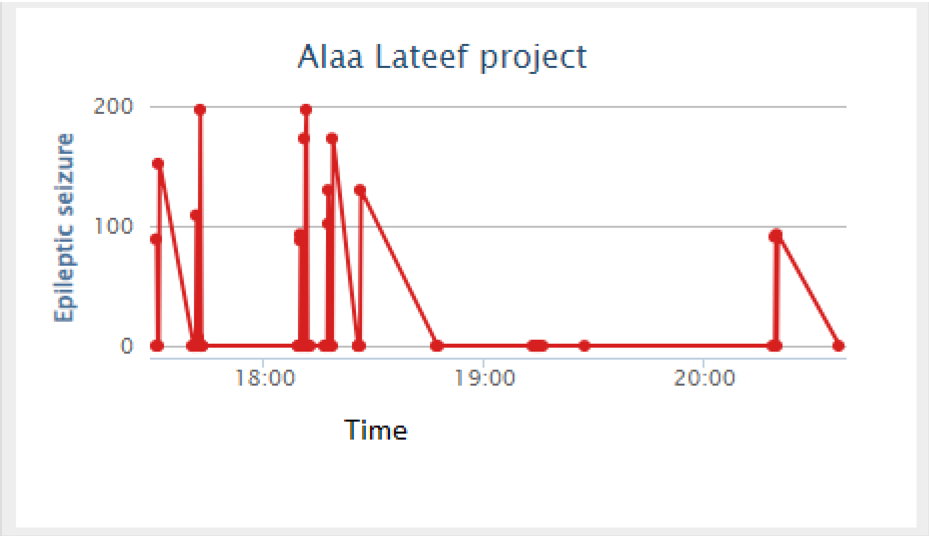


Figure 3.16 Example on ThingSpeak visualization

B. Send Compressed Data

We now discuss the second action of DML in the epileptic seizure detection system, which involves sending data regarding patients who do not have epileptic seizures to a remote server in the cloud. The suggested system employs a method of data compression known as threshold-based data reduction to minimize the size of the EEG data set, with the threshold being changeable.

In the event that SCL identifies the non-epileptic seizure as the result, then the DML will be sending much less information. By transmitting compressed data to the cloud layer, the DML helps preserve resources (i.e., memory space, time, and energy) for the non-epileptic patient. It's vital to highlight that releasing raw data won't give clinicians any new information for patients without seizures while their condition is stabilized (and no additional analysis is needed).While, in the case of epilepsy, doctors need to do more than just look at the EEG's features to predict when seizures could occur.

Here, we explain how the DML compresses EEG data using a threshold-based method.

i. Feature Importance

The first step in the proposed method is to calculate the importance of features. Typically, the input data for EEG is made up of a number of variables or features. The feature (variable) significance identifies the contribution of each feature to the model prediction. It establishes the usefulness of a certain variable for the forecast and current model. Feature importance is a score between 0 and 1 assigned to each column (or feature), telling how powerful is that feature in predicting the target variable, with 0 indicating that the feature has no importance and 1 indicating that the feature is absolutely essential. Note that we also require that the sum of all features should be 1.

Here, the proposed method used the XGB and EXT algorithms to compute feature importance scores based on Gini impurity measurements to estimate how each feature contributes to the prediction.

In general, we quantify the relevance of a feature using a score, which is a sum of all scores that equals one. The characteristic is more significant the higher its score value. A feature importance score has a lot of advantages. For example, the link between independent variables (features) and dependent variables (targets) may be established. We might identify and eliminate unnecessary characteristics by looking at variable significance ratings. The model may run faster or perhaps perform better if the number of unnecessary features is decreased. Figure 3.17 visualizes the importance of the top twenty EEG features.

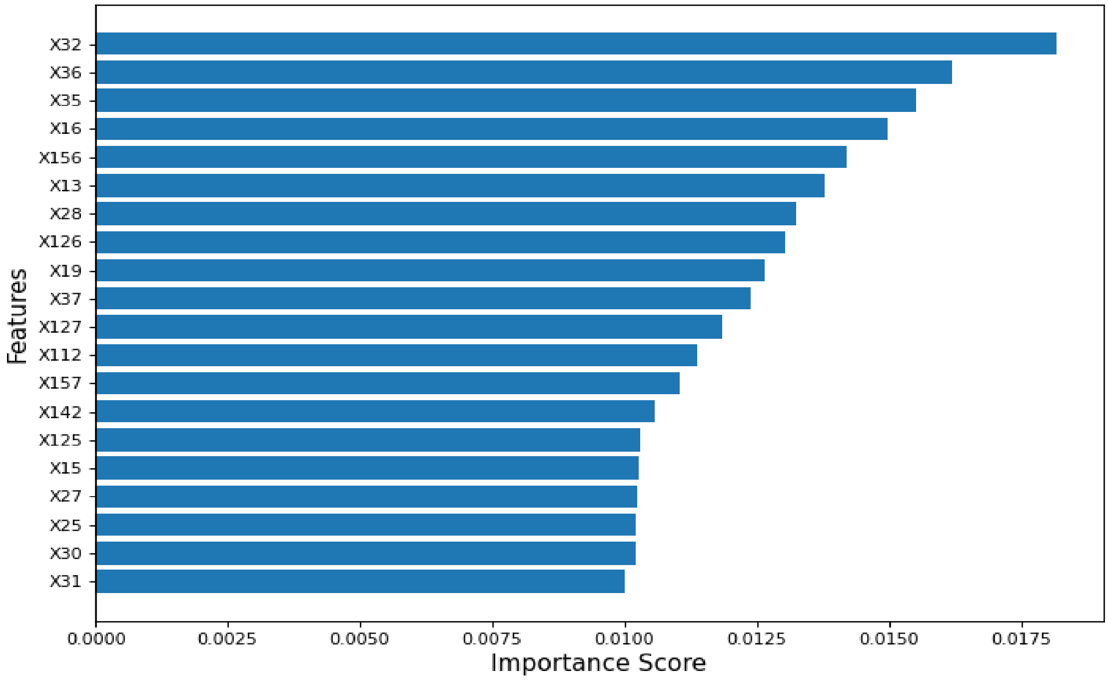


Figure 3.17. Visualize the importance of the top twenty EEG features.

ii. Feature Ranking

The next step in the proposed method after assigning an importance score for each feature is feature ranking. The feature ranking mapping the feature's score into a percentage.

$$F_{Rank} = F_{Score} * 100\% \quad \ldots\ldots\ldots\ldots\ldots \quad (3.6)$$

The purpose of this step is to facilitate how to deal with feature selection in the following step.

iii. Feature Selection

Another crucial task for dimensionality reduction to handle the big data problem is feature selection, which comes after feature importance and scoring. Proper feature selection avoids redundant features and allows for accurate classification to be carried out for the detection of various phases of patients with epileptic seizures.

For this work, the threshold-based feature selection approach has been used. Consequently, the following gives the compression ratio (stated as a percentage):

$$COM_R = \left(1 - \frac{M}{N}\right) \times 100 \qquad \text{.................. (3.7)}$$

Where N is the length of the original signal, and M is the number of chosen features obtained following the threshold-based data reduction.

In this method, the threshold amount is adopted in the process of selecting features, for example when a threshold of more than 25% is used, the features with a higher percentage of importance than the threshold are selected, and other features are discarded. The dataset's feature count is decreased or compressed once this approach is applied to it. The proposed threshold-based data reduction is presented in Algorithm 3.2.

	Algorithm 3.2: Threshold-based Data Reduction
	Input: $Epileptic\ Seizure\ Recognition\ dataset, threshold$ **Output:** $Select_{Feature}$:*reduced dataset*

$Select_{Feature} \leftarrow \emptyset$

$F_{Score} \leftarrow \emptyset$

$F_{Rank} \leftarrow \emptyset$

`# load data`

$dataset \leftarrow load\_data('Epileptic\ Seizure\ Recognition')$

`# Split data into X and y`

$X = dataset[:,0:178]$

$Y = dataset[:,178]$

`# Split data into train and test sets`

$X\_train, X\_test, y\_train, y\_test \leftarrow split(X, Y)$

`# fit model on all training data using XGBClassifier()`

$model \leftarrow Classifier(XGBClassifier, Gini)$

$model.fit(X\_train, y\_train)$

`# make predictions for test data and evaluate`

$y\_pred \leftarrow model.predict(X\_test)$

$accuracy \leftarrow accuracy\_score(y\_test, y\_pred)$

`# Fit model using each importance`

$F_{Score} \leftarrow model.feature\_importances$

`# Sort features in descending order`

$F_{Score} \leftarrow sort(F_{Score})$

`# Scoring features as a percentage (%)`

$\boldsymbol{for}\ i \leftarrow 1\ \boldsymbol{to}\ len(F_{Score})\ \boldsymbol{do}$

  $F_{Rank}[i] \leftarrow F_{Score}[i] \times 100\%$

$\boldsymbol{end\ for}$

$\boldsymbol{for}\ i \leftarrow 1\ \boldsymbol{to}\ len(F_{Rank})\ \boldsymbol{do}$

  $\boldsymbol{if}\ (F_{Rank}[i] > threshold)\ \boldsymbol{then}$

    $Select_{Feature} \leftarrow Select_{Feature} \cup F_{Rank}[i]$

$\quad \boldsymbol{end\ if}$

$\boldsymbol{end\ for}$

$return\ \boldsymbol{select_{Feature}}$

3.2.6 Cloud Storage Layer

The daily collecting and processing of detected EEG data samples from patients with epilepsy depends on this layer. The raw EEG signals collected from an epileptic patient are retained on this layer together with the patient's personal information in order to uniquely identify the raw EEG signals received for a particular patient. Along with their name, age, gender, home address, family members' names, phone numbers, etc., the patient's information also includes their social security number, commonly known as their Unique ID (UID). This layer gives DML specific details it needs to handle emergency circumstances, such as the location and phone numbers of a nearby hospital and registered family members. Many hospitals, healthcare institutions, and research and development companies also receive summarized EEG data samples for use in creating new drugs or vaccines and conducting additional studies on the subject of early epileptic seizure detection.

The IoT Platform also serves as a repository for the patient's seizure detection and recording history. All of our computations and analyses are performed in the fog layer, thus we choose to use an open-source IoT platform in our design. ThingSpeak™, an IoT platform, was integrated into this design. ThingSpeak™, an open-source IoT analytics platform service, can gather, visualize, and analyze real-time data streams. Data supplied by our system is instantly visualized on ThingSpeak™ using the HTTP protocol via the Internet.

In the suggested system, we have a specific channel labeled "Epileptic Seizure" which is the heart of the IoT platform. This is the place where EEG readings are recorded continuously. Through a RESTful application programming interface (API), medical practitioners can access the IoT cloud and the database.

Figure (3.18) illustrates the relationships between the aforementioned parts.

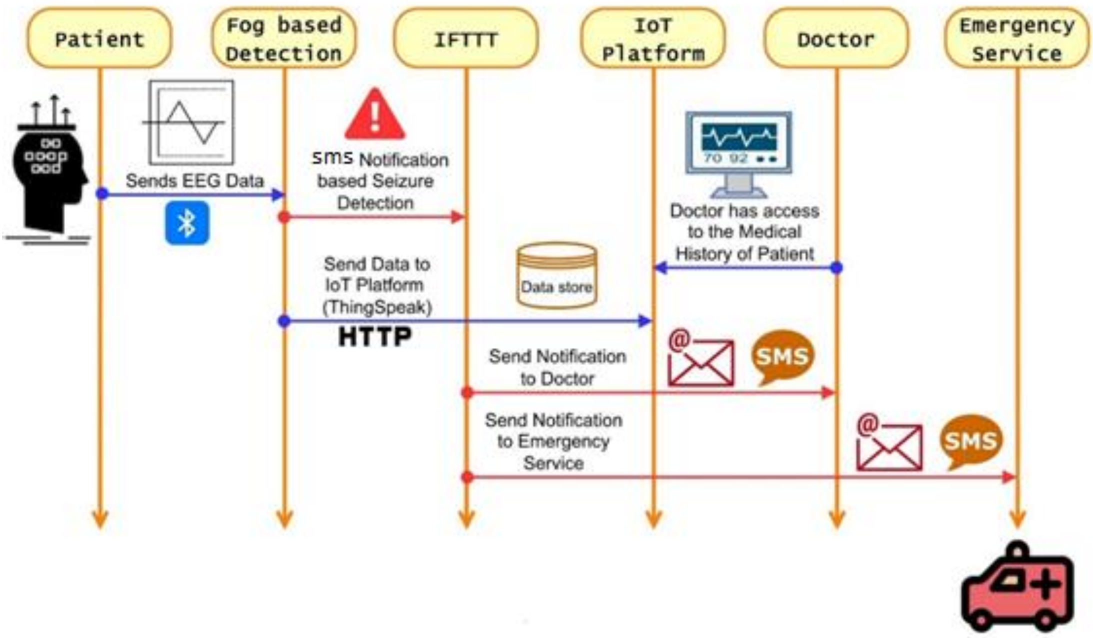


Figure 3.18 The relationships between the proposed system's parts.

CHAPTER FOUR

SIMULATION RESULTS AND DISCUSSIONS

CHAPTER FOUR

SIMULATION RESULTS AND DISCUSSION

4.1. Introduction

The performance evaluation and simulation findings for the suggested methods presented in Chapter 3 are presented as graphs and tables and discussed in this chapter. Using EEG data to compare the effectiveness of strategies using various performance indicators is the main objective.

4.2. Simulation Environment

Extensive simulation experiments are performed using a personal computer with an Intel(R) Core(TM) i3-7100U CPU running at 2.40GHz and 4.00GB of RAM to evaluate our proposed system. The proposed technique was implemented in Python 3.9 using the PyCharm editor and was based on a dataset from Bonn University in Germany.

To demonstrate the efficiency of the suggested approach, various simulated scenarios are used. For each of these scenarios, we used performance evaluation measures to compare the results obtained when applying each of these cases and to find out which scenario gave us the best results, taking into consideration the reduction of storage space and the reduction of energy consumption. Table (4.1) shows these scenarios with the corresponding description.

Table 4.1 Model Evaluation Scenarios.

Scenario No.	Representation	Brief Description
1.	Scenario #1	Using ML models with raw EEG data.
2.	Scenario #2	Using ML models with extracted features.
3.	Scenario #3	Using ML models with balanced EEG data and extracted features.
4.	Scenario #4	Using ML models with hyper-parameters, balanced EEG data and extracted features.

4.3. Evaluation Metrics

The designed model's performance is examined using evaluation parameters like Accuracy, F1 score, Recall, Precision, and specificity. For each of the machine learning models, we will apply these performance measures in all four scenarios, so that the results of the comparison with performance are clear, and we can determine which model provided us with the best results.

4.4. Imbalanced Dataset and Hyper Parameters Tuning

The nature of the Bonn dataset and as shown in Figure (4.1) is unbalanced data because of the big difference between the number of records (EEG signals) that are classified as epileptic seizures and those that are free of seizures.

After applying the SMOTE-ENN technique to solve the problem of the unbalanced nature of the used dataset, we get a balanced dataset as shown in Figure (4.2).

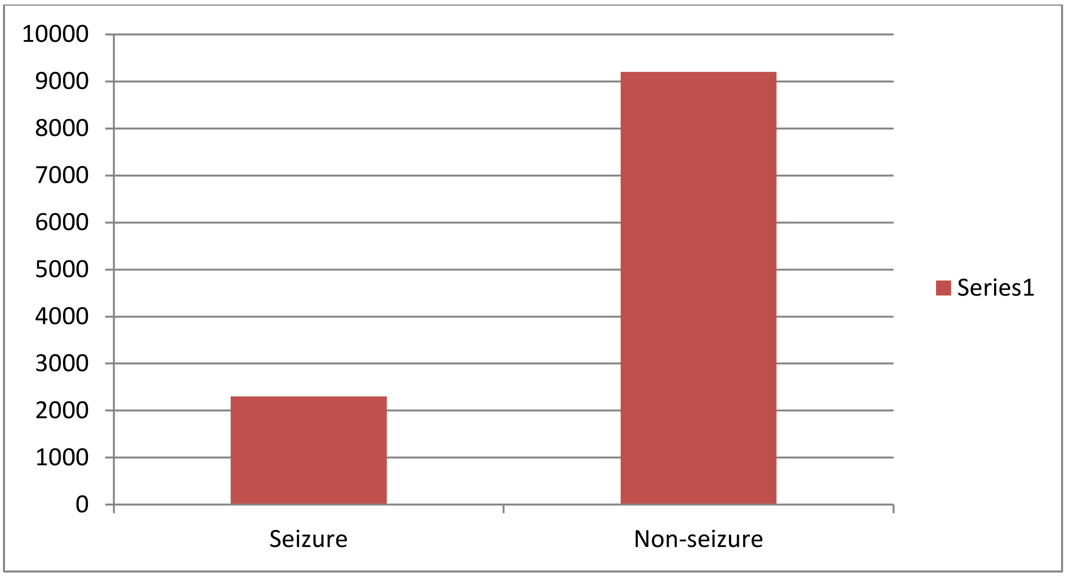


Figure 4.1. The unbalanced nature of the Bonn dataset.

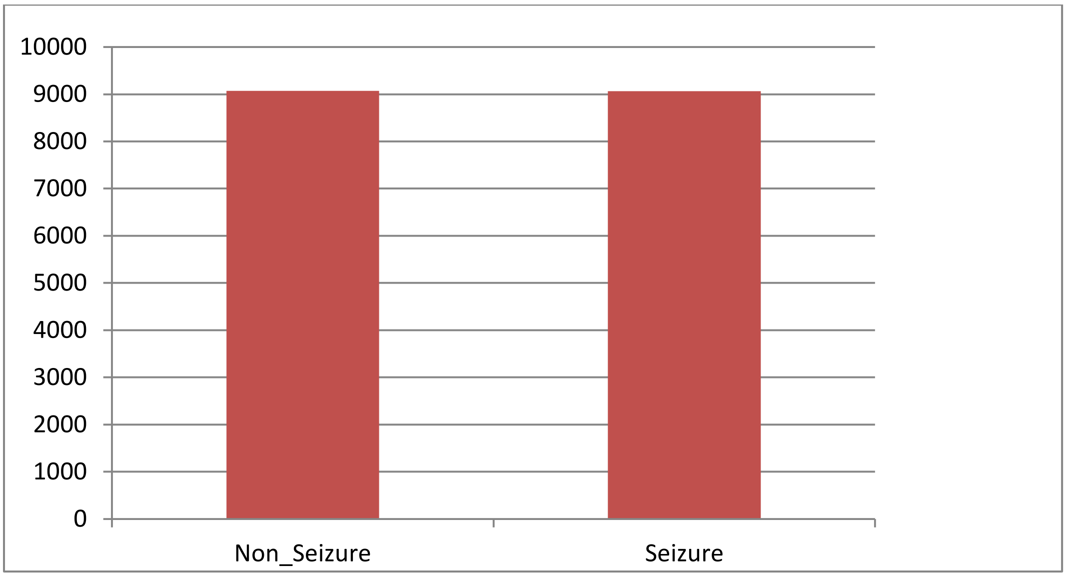


Figure 4.2 The balanced dataset after using the SMOTE-ENN technique.

There are various factors unique to each classifier that can affect how well it performs and must be improved. As a result, "GridSearchCV" using a cross-validation method is used to optimize classifier parameters. To guarantee robustness and the dependability of the performance of the proposed method, classifiers with parameters that are optimized are trained using the Repeated Stratified K-Fold cross-validation method. The classifier's parameters utilized for optimization are listed in Table (4.2).

Table.4.2 Hyper parameters of the used machine learning models.

Method	Parameter	Parameters Grid	Selected Value
LR	solver,	newton-cg, lbfgs, liblinear	newton-cg
	penalty,	l2	l2
	Max_iter	100, 10, 1.0, 0.1, 0.01	100
SVM	Regularization parameters,	50, 10, 1.0, 0.1, 0.01	50
	Gamma,	1, 0.1, 0.01, 0.001, 0.0001	0.01
	Kernel	poly, rbf, sigmoid	Rbf
DT	Split criterion	Gini, entropy	Gini
	Max depth	None, 2, 4, 6, 8, 10	8
	Max features	None, sqrt, log2, 0.2, 0.4, 0.6, 0.8	None
	Splitter	best, random	Best
RF	No. of trees	10, 100, 1000	100
	Max features	sqrt, log2	log2
	Max depth	None, 2, 4, 6, 8, 10	None
EXT	Random state	0,1,2,3,4,5	0
	No. of trees	320,340,360,380,400	360
	Max depth	25, 30, 32, 34, 38, 45	25
	Split criterion	Gini, entropy	Entropy
	Max features	sqrt, log2, 0.2, 0.4, 0.6, 0.8, 1.0	1.0
GB	Learning Rate	0.15, 0.1, 0.05, 0.01, 0.005, 0.001	0.15
	Max depth	3, 4, 7, 9	4
	Max features	None, sqrt, log2, 0.2, 0.4, 0.6, 0.8	Sqrt
	No. of trees	100, 250, 500, 750, 1000, 1250, 1500, 1750	750
	Subsample	0.5, 0.7, 1.0	1.0
XGB	Learning Rate	0.1, 0.01, 0.001	0.1
	Max depth	3, 5, 7	7
	Subsample	0.5, 0.7, 1	0.5
KNN	Metric	euclidean, manhattan, minkowski	Manhattan
	No. of neighbors	1-21	3
	weights	uniform, distance	Uniform

Where the resulting feature vector is randomly divided into ten subsets, one of which is retained for testing while the other nine are utilized to train the model. Three times are needed to complete the process. (i.e., in this thesis, the number of repeats is 3 of 10-fold cross-validation is used to estimate the model performance, this means that (3 * 10) or 30 different models would be fit and evaluated), and performance measurements are calculated for each iteration before the average classification metrics are taken into account as the classifier's performance..

4.5. Experimental Results

This section discusses and analyzes the results of the suggested method for detecting epileptic seizures based on evaluation metrics and displays the simulation results. Classification of EEG data to identify seizure activity and evaluate classifier performance is the initial stage. Many different types of classification models were tested in the experiments.

4.5.1. Classification Accuracy

The accuracy assessment is given in Table (4.3). Table (4.3) illustrates the comparative analysis of the classification accuracy of all four test scenarios with different classifiers that are applied in the above experiments.

In relation to the results of the classification shown in the table, it is clear that the first scenario is the least fortunate in terms of the accuracy of the classification, and the reason is due to the fact that the data used in this scenario is of its raw nature without any processing. While we find that the second and third scenarios improved the accuracy rate as a result of conducting different treatments on the data, such as extracting good features and making the data balanced, which led to improved results. Also, we find that most of the methods used achieved results that were very close to each other. As for the fourth scenario, we find that the results achieved are highly accurate, as it can be noted that five of the methods used achieved 100% classification accuracy, namely LR, DT, ExT, GB and XGB. The reason for this is that this scenario used the best parameters of ML in addition to the extracted features and data balancing, which yielded very good results.

Table 4.3 The accuracy metric of ML models using the four scenarios.

Metric	Method	Scenario #1	Scenario #2	Scenario #3	Scenario #4
Accuracy (%)	LR	81.83 %	95.13 %	94.06 %	100 %
	KNN	82.82 %	95.39 %	95.47 %	99.77 %
	SVM	97.82 %	96.73 %	96.27 %	99.78 %
	SGD	97.83 %	96.74 %	96.27 %	99.78 %
	NB	97.83 %	96.73 %	96.27 %	99.77 %
	DT	94.17 %	94.70 %	96.47 %	100 %
	RF	96.30 %	96.57 %	96.66 %	99.61 %
	EXT	97.48 %	96.74 %	97.13 %	100 %
	GB	97.04 %	95.35 %	97.46 %	100 %
	XGB	97.43 %	96.48 %	97.87 %	100 %

4.5.2. Classification Precision

Another measure that was used in these experiments is classification precision, which is considered one of the important measures that show the percentage of true positives. Table (4.4) illustrates the comparative analysis of the classification precision of all four test scenarios with different classifiers that are applied in the above experiments.

Table 4.4 The Precision metric of ML models in the four scenarios.

Metric	Method	Scenario #1	Scenario #2	Scenario #3	Scenario #4
Precision (%)	LR	84.84 %	89.13 %	94.04 %	100 %
	KNN	85.86 %	94.54 %	95.72 %	99.94 %
	SVM	97.81 %	94.11 %	96.15 %	99.55 %
	SGD	84.45 %	90.19 %	93.99 %	100 %
	NB	96.39 %	85.07 %	95.40 %	99.88 %
	DT	87.19 %	87.85 %	96.38 %	100 %
	RF	96.34 %	94.88 %	96.08 %	99.21 %
	EXT	97.64 %	94.52 %	96.84 %	100 %
	GB	95.63 %	87.92 %	97.23 %	100 %
	XGB	96.55 %	93.23 %	97.47 %	100 %

From the table, we find that the first and second scenarios do not give good results, and the reason for this is due to the fact that the data used is not balanced. Where in an imbalanced classification problem with two classes precision calculates the accuracy for the minority class. While the results of the third scenario show a good improvement in the results, after using the SMOTE+ENN method, as it ensures that there is no bias towards the majority class, and thus the precision of the classification increases.

As for the fourth scenario, we note that the results have improved significantly, after making an optimization in the parameters of the ML models and balancing the data. In this scenario, the model predicted many examples as belonging to the minority class, where the ratio of correct positive examples is much better. Also, noticed that 60% of the ML models achieved full or perfect precision 100%, namely LR, SGD, DT, EXT, GB, and XGB.

4.5.3. Classification Sensitivity

In this experiment, we deal with another measure of performance, which is sensitivity also known as the true positive rate or recall. Sensitivity is important in scenarios where missing positive epilepsy instances (false negatives) is more critical than incorrectly classifying negative epilepsy instances as positive (false positives). In this thesis, it's crucial to correctly identify individuals with epilepsy to ensure they receive proper treatment. Table (4.5) illustrates the comparative analysis of the classification sensitivity of all four test scenarios with different classifiers that are applied in these experiments.

From the results shown in Table (4.5), it is clear that the first and second scenarios are characterized by poor results. The reason is that the dataset used is unbalanced. In datasets where one class significantly outweighs the other (class imbalance), sensitivity becomes particularly important. In these two scenarios, the models simply predict the majority class which achieves accepted accuracy (as shown in Table 4.3) but has poor sensitivity for the minority class. Sensitivity helps assess how well the model performs in the minority class.

Whereas, in the third and fourth scenarios, we notice a good improvement in sensitivity. Where SMOTE+ENN methods were used in order to balance the dataset as shown in the results of the third scenario, in addition to the optimization of the ML models parameters, as shown in the results of the fourth scenario. In the fourth scenario, we note that 90% of the methods achieved a full or perfect sensitivity of 100%, except for the KNN method.

Table 4.5 The Sensitivity metric of ML models in the four scenarios.

Metric	Method	Scenario #1	Scenario #2	Scenario #3	Scenario #4
Sensitivity Recall (%)	LR	10.53 %	86.45 %	93.78 %	100 %
	KNN	15.05 %	81.93 %	94.96 %	99.60 %
	SVM	92.25 %	89.46 %	96.21 %	100 %
	SGD	4.51 %	39.56 %	91.18 %	100 %
	NB	91.18 %	61.29 %	86.88 %	100 %
	DT	83.44 %	85.59 %	96.38 %	100 %
	RF	83.87 %	87.74 %	97.11 %	100 %
	EXT	89.67 %	89.03 %	97.28 %	100 %
	GB	89.46 %	89.24 %	97.56 %	100 %
	XGB	90.53 %	89.03 %	98.19 %	100 %

4.5.4. Classification Specificity

In machine learning classification problems, specificity—also known as the True Negative Rate—is a crucial performance parameter. Out of all the real negative examples in the dataset, it assesses a model's capacity to properly detect negative instances. Specificity is the complement to sensitivity. While sensitivity focuses on correctly identifying positive instances, specificity focuses on correctly identifying negative instances. Table (4.6) illustrates the comparative analysis of the classification specificity of all four test scenarios with different classifiers that are applied in these experiments.

Table 4.6 The Specificity metric of ML models in the four scenarios.

Metric	Method	Scenario #1	Scenario #2	Scenario #3	Scenario #4
Specificity (%)	LR	99.94 %	97.33 %	94.33 %	100 %
	KNN	100 %	98.80 %	95.95 %	99.94 %
	SVM	99.23 %	98.58 %	96.33 %	99.56 %
	SGD	100 %	98.91 %	94.44 %	100 %
	NB	97.71 %	97.27 %	96 %	99.89 %
	DT	96.89 %	97 %	96.54 %	100 %
	RF	99.45 %	98.80 %	96.22 %	99.23 %
	EXT	99.45 %	98.69 %	96.97 %	100 %
	GB	98.96 %	96.89 %	97.35 %	100 %
	XGB	99.18 %	98.36 %	97.57 %	100 %

From the results shown in Table (4.6), it is clear that there's often a trade-off between sensitivity and specificity. Increasing sensitivity might lead to a decrease in specificity and vice versa. Specificity is important when dealing with imbalanced datasets as in the first and second scenarios. If the majority of instances belong to the negative class, a model that always predicts the negative class could achieve high specificity but would likely have poor sensitivity for the minority class. Also, noticed that 60% of the ML models achieved full or perfect precision 100%, namely LR, SGD, DT, EXT, GB, and XGB.

4.5.5. Classification F1-Score

The F1-score is a widely used performance metric in machine learning classification tasks that provides a balance between precision and recall. It provides a single value that combines these two metrics, making it useful for scenarios where you want to balance the trade-off between minimizing false positives and false negatives. In cases where one class greatly outweighs the other as in the first and second scenarios, accuracy alone might not provide an accurate representation of model performance. The F1-score can be a better indicator, as it considers the precision and recall for both classes, providing insight into how well the model performs across the entire dataset.

Table 4.7 The F1-score metric of ML models in the four scenarios

Metric	Method	Scenario #1	Scenario #2	Scenario #3	Scenario #4
F1 (%)	LR	19.02 %	87.77 %	93.91 %	100 %
	KNN	26.16 %	87.79 %	95.34 %	99.77 %
	SVM	94.49 %	91.73 %	96.18 %	99.77 %
	SGD	8.64 %	55 %	92.56 %	100 %
	NB	91.08 %	71.25 %	90.94 %	99.94 %
	DT	85.27 %	86.71 %	96.38 %	100 %
	RF	90.17 %	91.17 %	96.59 %	99.60 %
	EXT	93.49 %	91.69 %	97.06 %	100 %
	GB	92.44 %	88.58 %	97.40 %	100 %
	XGB	93.45 %	91.08 %	97.83 %	100 %

From the results shown in Table (4.7), a low F1-score as in the first and second scenarios could indicate a skewed balance between precision and recall, which might require model adjustments. While the high F1-score as in the third and fourth scenarios indicates that the models have a good balance between precision and recall, meaning it is performing well in correctly classifying positive instances (precision) while also capturing a high proportion of actual positives (recall).

4.5.6. ML Training Time

The training time of ML models refers to the amount of time it takes for the model to learn from a given dataset and adjust its internal parameters to make accurate predictions or classifications.

Table (4.8) illustrates the comparative analysis of the classification training time of all four test scenarios with different classifiers that are applied in these experiments.

From the results shown in the table, it's clear that the training time varies significantly depending on several factors, including the complexity of the model, the size of the dataset, the hardware resources available, and the optimization techniques used. Sometimes, training is stopped early if the model's performance on a validation set starts to degrade. This can save time by avoiding unnecessary training epochs.

4.5.7. Receiver Operating Characteristic

A graphic representation of the Receiver Operating Characteristic (ROC) curve shows how well a binary classification model performs at various classification levels. It charts the relationship between the True Positive Rate (Sensitivity) and the False Positive Rate (1 - Specificity) when the threshold for categorizing positive examples changes.

Table 4.8 The training time taken by the classifiers in the four scenarios.

Metric	Method	Scenario #1	Scenario #2	Scenario #3	Scenario #4
Training time (milliseconds)	LR	937.5	15.625	78.125	1406.25
	KNN	15.625	16.875	15.5	15.625
	SVM	6921.875	2234	4640	2443.5
	SGD	78.125	72.15	1525	31.25
	NB	31.25	35.625	46.37	46.875
	DT	3109.375	4600	0.187	125
	RF	4796.875	1187	3125	4671
	EXT	10906.25	406	921	828.125
	GB	55968.75	4796	14780	2453.125
	XGB	11140.625	1953	4500	6406.25

The ROC curve helps assess and compare the trade-offs between sensitivity and specificity for different threshold settings. The ROC curve is created by plotting the TPR (Sensitivity) on the y-axis against the FPR (1 - Specificity) on the x-axis for different threshold values.

The ROC curve is a powerful tool for comparing the performance of different models or algorithms. Models with ROC curves which are closer to the top left corner (higher TPR and lower FPR) generally perform better across a range of thresholds. Figures (4.3 – 4.6) provide a comprehensive visualization of a model's performance across different threshold settings. It helps to understand how the model's sensitivity and specificity change as the threshold varies. From the figures, it is clear that models with ROC curves that are more inclined towards the upper left corner perform better.

(higher TPR and lower FPR) generally perform better across a range of thresholds.

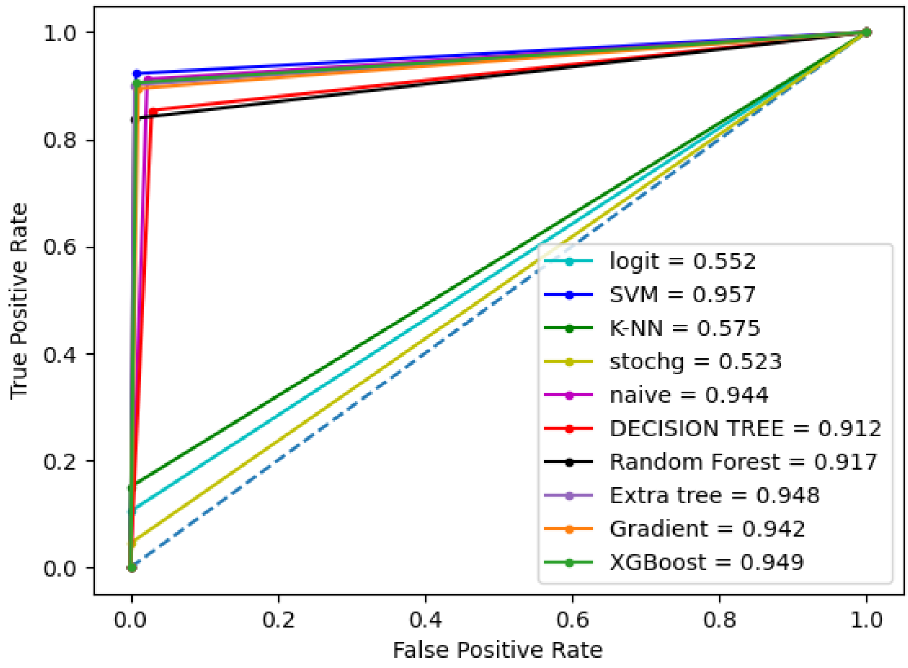


Figure 4.3. The ROC curve of the first scenario.

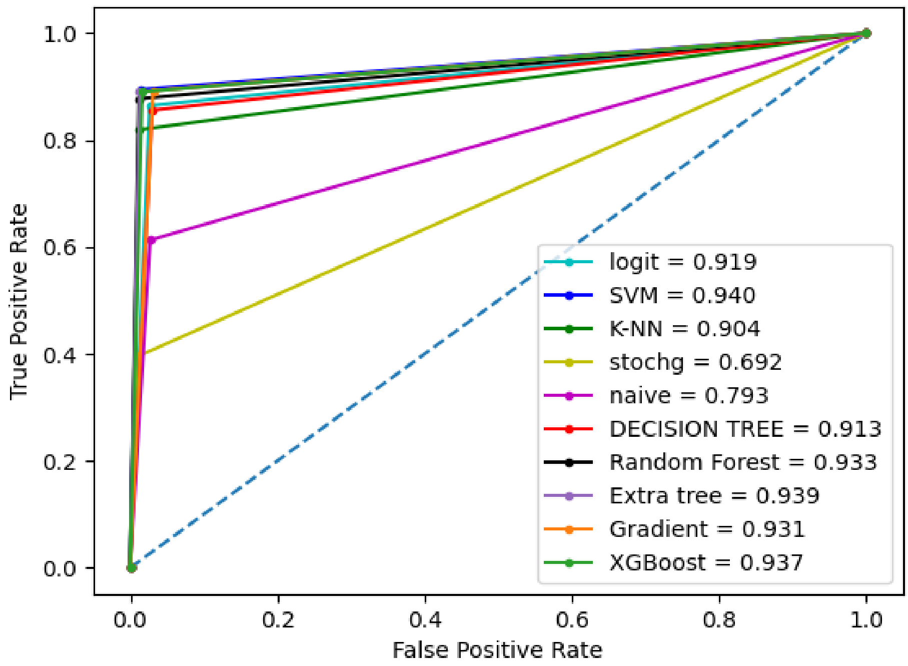


Figure 4.4. The Roc curve of the second scenario.

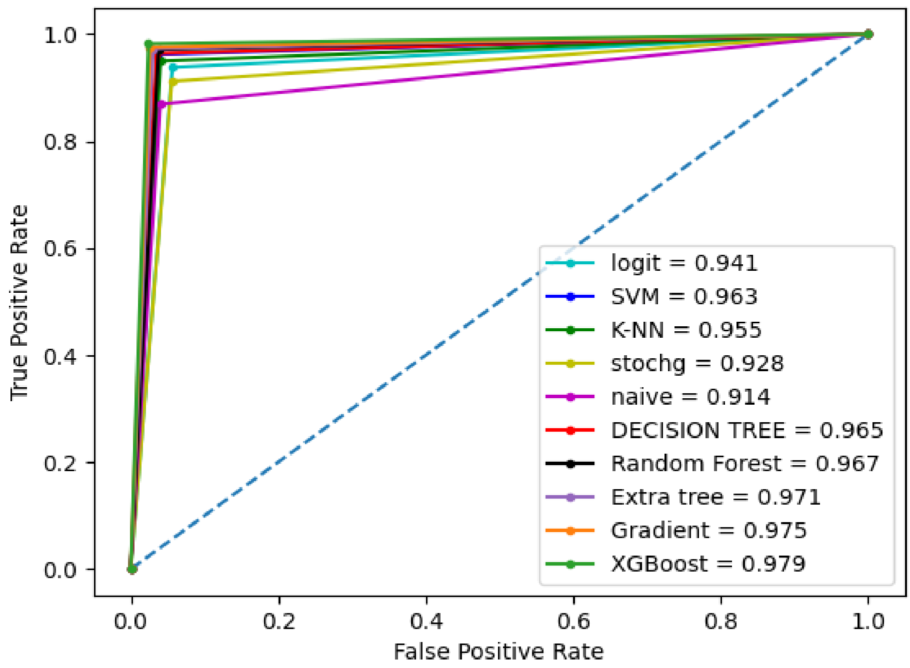


Figure 4.5. The Roc curve of the third scenario.

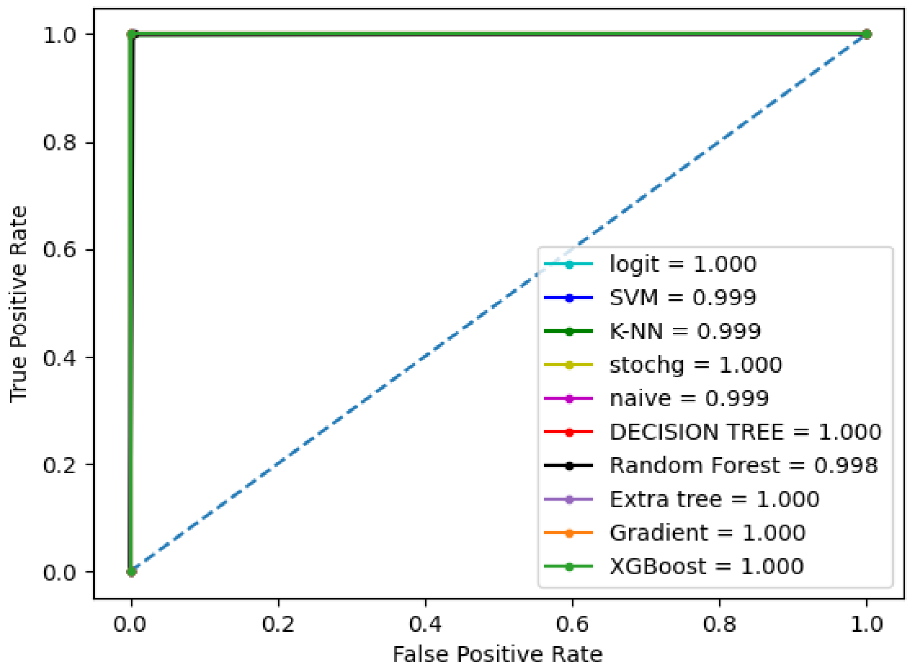


Figure 4.6. The Roc curve of the fourth scenario.

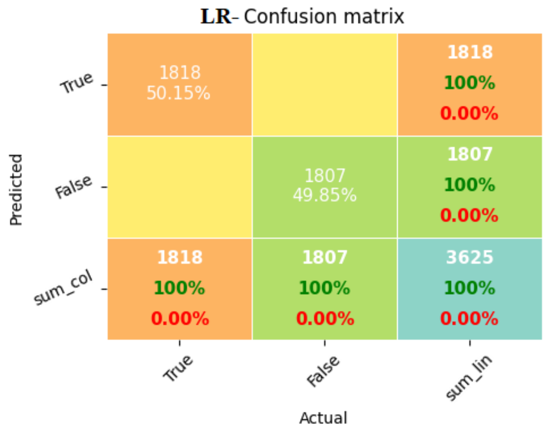
In conclusion, the ROC curve is a valuable tool for evaluating, comparing, and fine-tuning the performance of binary classification models. It provides a visual representation of a model's sensitivity-specificity trade-off and helps make informed decisions based on the problem's context and requirements.

4.5.8. Confusion Matrices

A confusion matrix is a fundamental concept in ML that provides a clear and structured way to evaluate the performance of a classification model. It is especially useful when dealing with binary (two-class) classification problems. The matrix visually summarizes the results of the classification process and helps to understand how well the model is performing.

Due to the large number of experiments as a result of using ten different models for classification and for four different scenarios, we will display the confusion matrices for the fourth scenario only in Figures (4.7 – 4.16).

From the figures we can discover that the diagonal of the confusion matrix (from top left to bottom right) contains the instances that the model has correctly classified. These are the true positives and true negatives. They represent instances where the model's predictions align with the actual labels. Also, let's look at the off-diagonal elements (top right and bottom left) of the confusion matrix. These components stand in for the mistakes the model has made. False Positives (FP) happens when the model predicts the wrong class, one which is positive and the true class is negative. False Negatives (FN) happen when the model predicts the wrong class, usually a negative one when the true class is a positive one. Depending on the problem, certain errors might be more critical than others. As in epilepsy diagnosis, false negatives (missed positive cases) could be more concerning than false positives.



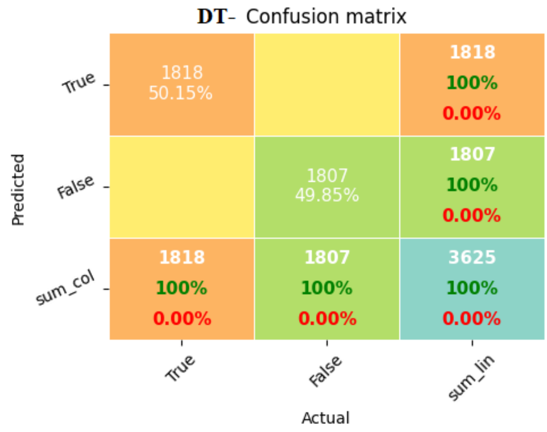
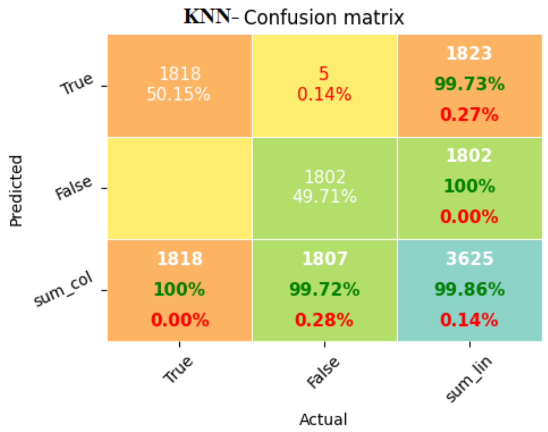


Figure 4.7. LR confusion matrix.

Figure 4.8. DT confusion matrix.



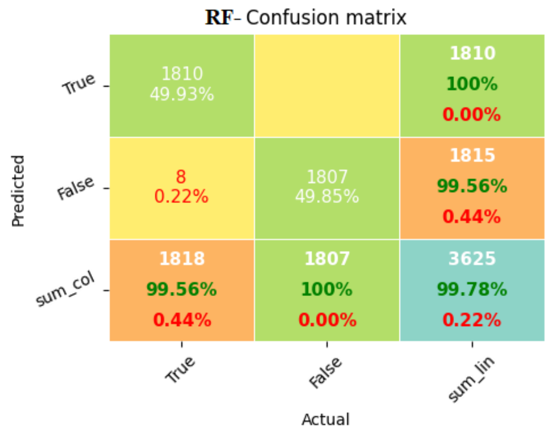
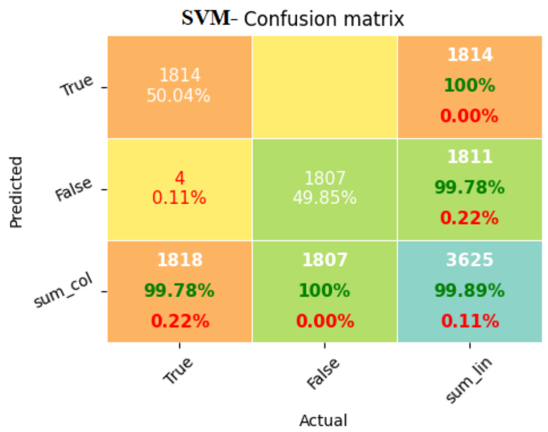


Figure 4.9. KNN confusion matrix.

Figure 4.10. RF confusion matrix.



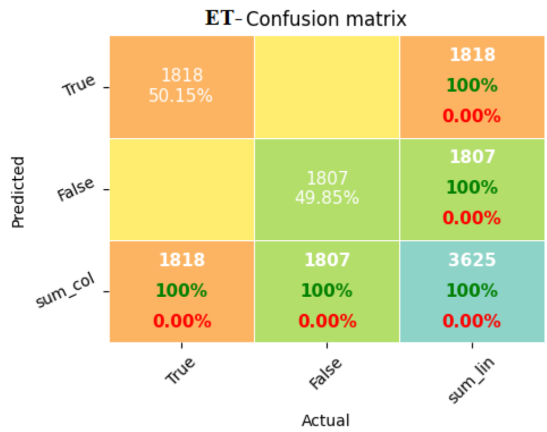


Figure 4.11. SVM confusion matrix.

Figure 4.12. EXT confusion matrix.

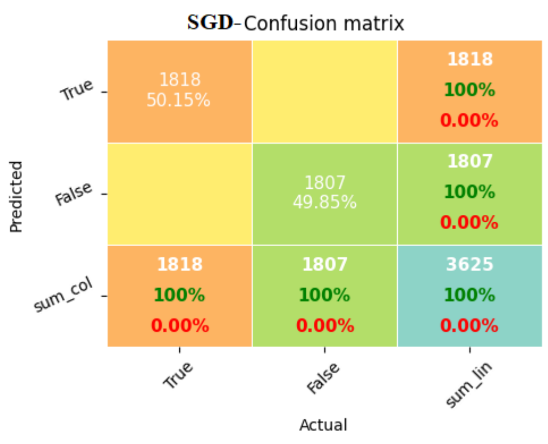


Figure 4.13. SGD confusion matrix.

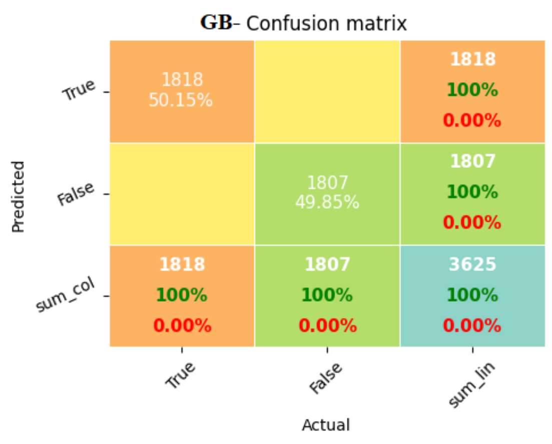


Figure 4.14. GB confusion matrix.

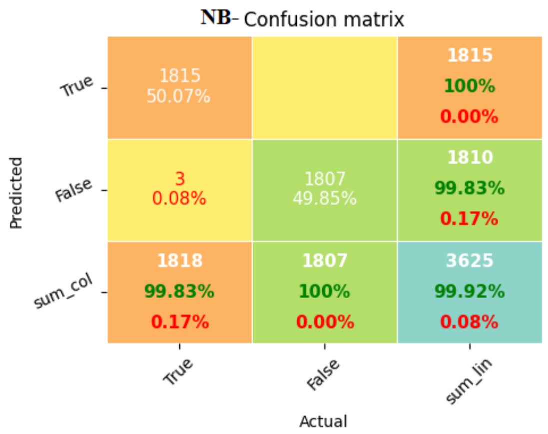


Figure 4.15. NB confusion matrix.

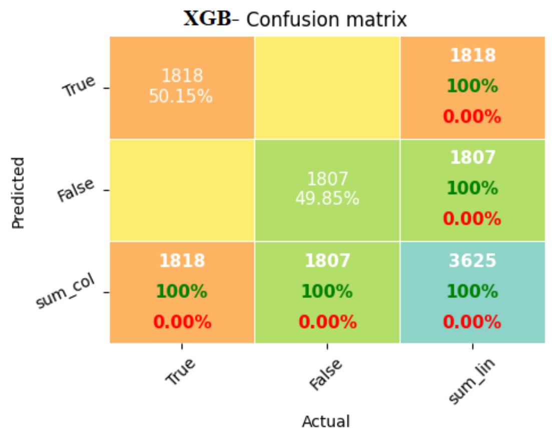


Figure 4.16. XGB confusion matrix.

4.6. Results of Threshold-based Data Reduction

Data reduction in the context of the IoMT refers to the process of minimizing the amount of data transmitted and stored by medical devices and sensors while still preserving the necessary information for meaningful analysis and decision-making. However, the sheer volume of data generated by these devices can be overwhelming and pose challenges related to storage, transmission bandwidth, and data processing. Thus, in this thesis data reduction technique is employed to address these challenges and make IoMT systems more efficient and practical. Where in the proposed system, as we explained in the third chapter, the complete data is sent in the case of a patient diagnosed with epilepsy, while the data is reduced or compressed in the case of a patient diagnosed without epilepsy.

A threshold-based data reduction technique is proposed in this thesis. Instead of transmitting or storing all raw data to a centralized cloud server for processing, feature selection and extraction involve identifying the most relevant and informative attributes (features) of the data based on their importance at the fog level or within a local network. This reduces the dimensionality of the data and retains only the most important or critical aspects for analysis then transmitted to the central cloud server, which reduces the overall data traffic. This is particularly useful for time-series data from wearable EEG devices, where extracting relevant features can help in diagnosing epileptic conditions or monitoring health parameters.

4.6.1. The percentage of deleted features

One of the experiments that we conducted in this section is the quantity of deleted features, depending on their importance. Figure (4.17) shows a comparison of the quantity of deleted features relative to their importance, based on two methods for calculating the importance of these characters, namely XGB and EXT.

From the figure, we notice that the greater the importance of the features, the greater the number of deleted features, and thus a small group of features remain, which are characterized by their high importance in classifying the patient’s condition with appropriate accuracy. We note from the results, in the case of keeping the features whose importance is 50% or more, the number of deleted features is 146 and 168 out of 178 using EXT and XGB, respectively.

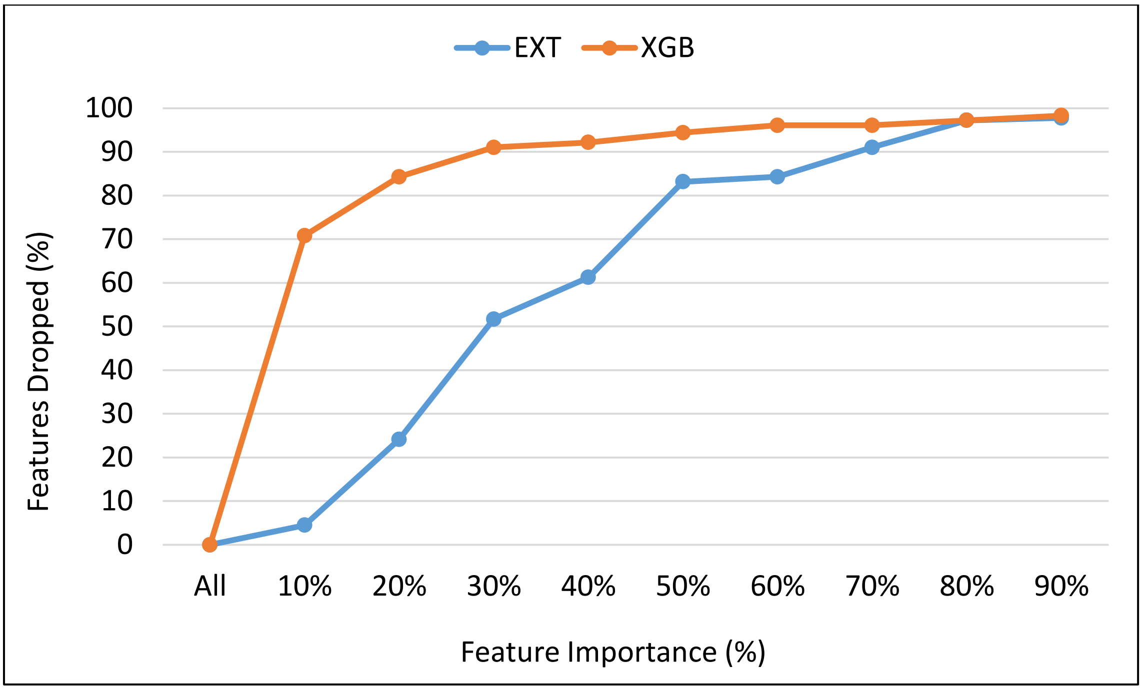


Figure 4.17. The percentage of deleted features using EXT, XGB classifiers

4.6.2. The Amount of Sent Data

Another experiment conducted through this thesis is to calculate the amount of data sent based on the proposed method of data reduction, which is threshold-based data reduction. In this approach, only data or features that surpass a certain threshold or meet specific criteria (i.e., their importance exceeded the threshold) are transmitted. This reduces the amount of unnecessary data sent, as only relevant features are reported. The goal of this proposed method is to strike a balance between preserving data quality and minimizing energy consumption, communication overhead, and processing load on the fog and cloud tiers.

From the results shown in Figure 4.18, we can discover that the amount of data transmitted from the fog tier to the cloud tier is within the percentage from 93.8% to 1.69% and from 29.2% to 1.69% for the features of importance from 10% to 90% and for the EXT and XGB methods, respectively. Note that the XGB method sends fewer features than the EXT method.

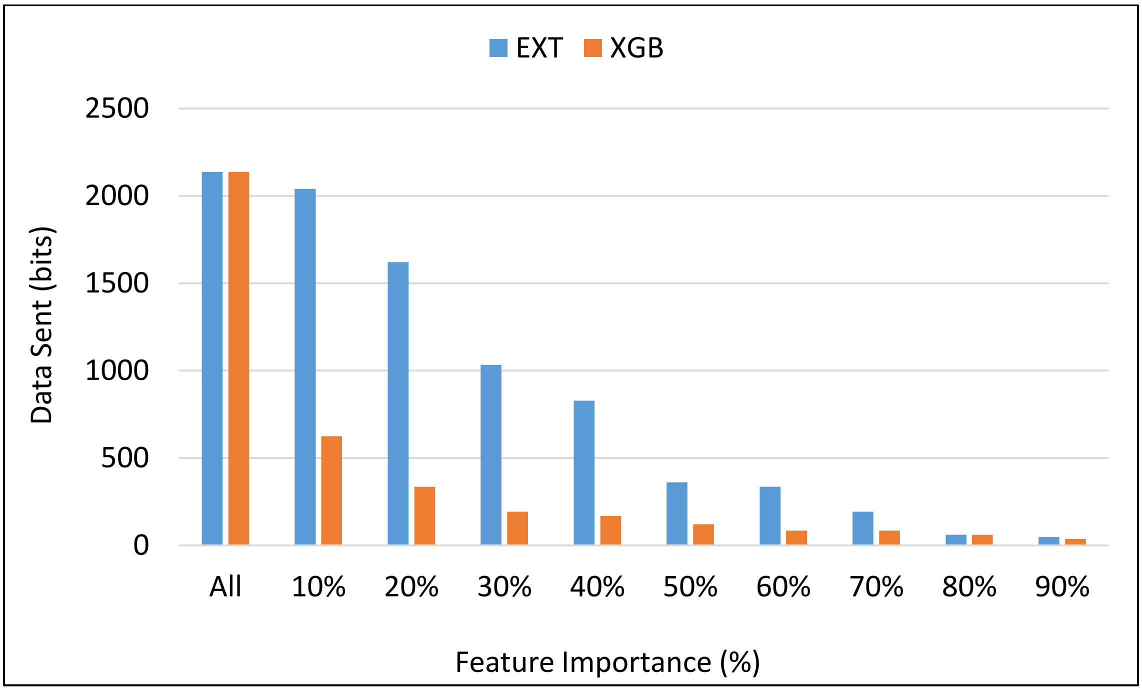


Figure 4.18. Sent data according to different feature importance.

4.6.3. Compression Ratio

Another measure used in this thesis is the data compression ratio. We used this metric to quantify how much the size of the data has been reduced after applying the proposed threshold-based data reduction.

We notice in Figure (4.19) that there is an exponential relationship between the importance of features and the data compression ratio. The more important the features, the higher the compression ratio, and vice versa.

From the results shown in the figure, we can discover that the ratio of data compression achieved by the proposed method ranges from 1.04 to 44.5 and from 3.4 to 59.3 for the features of importance from 10% to 90% and for the EXT and XGB methods, respectively. Where a compression ratio greater than 1 indicates that the data has been compressed, resulting in a smaller size. It is noted that the XGB method compresses data by a greater percentage compared to EXT.

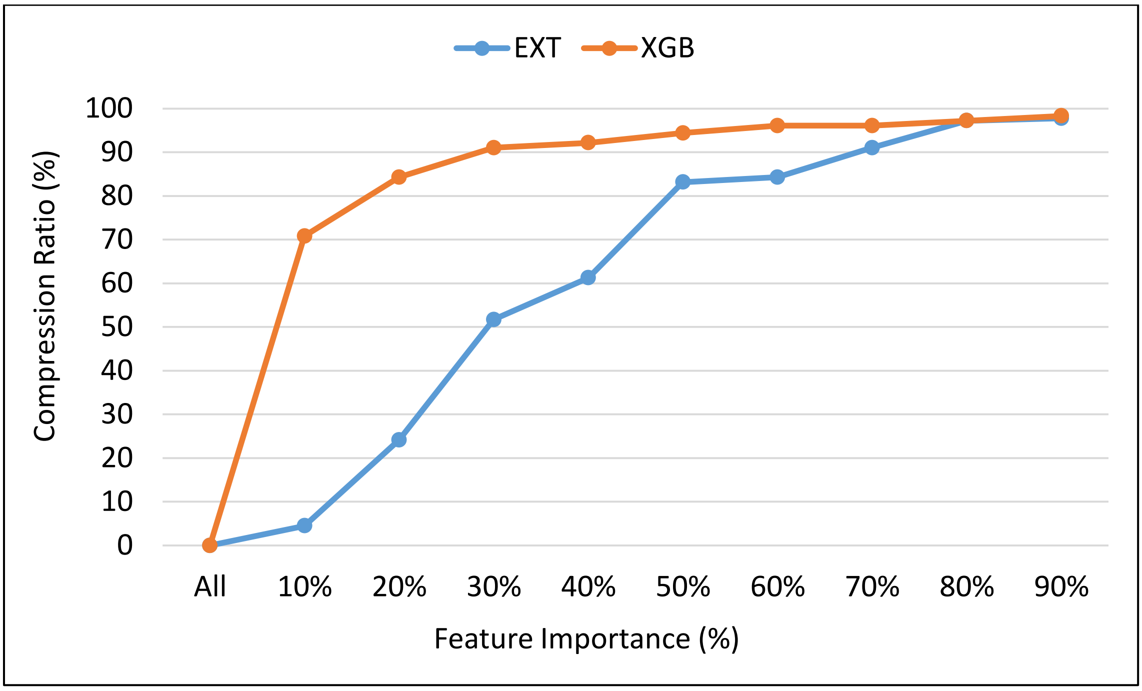


Figure 4.19 The compression ratio of the sent data.

4.6.4. Energy Consumption

Given that the proposed system depends mainly on the IoMT, therefore, one of the important things that this system must achieve is to reduce energy consumption. In this thesis, we propose to reduce energy consumption at the fog tier by reducing the amount of data transmitted to the cloud tier. Our proposed method, which depends on sending data on its importance, gives good results, as shown in Figure (4.20).

From the figure, we notice that there is an inverse relationship between the importance of features and the energy expenditure ratio. The more important the features, the lower the energy consumption as a result of the decrease in the number of transmitted data, as shown in Figure (4.18), and vice versa.

From the results shown in the figure, we can discover that the improvement in the amount of energy consumed in the fog layer is within the percentage from 4.46% to 97.7% and from 70.7% to 98.3% for the features of importance from 10% to 90% and for the EXT and XGB methods, respectively. Note that the XGB method sends fewer features than the EXT method.

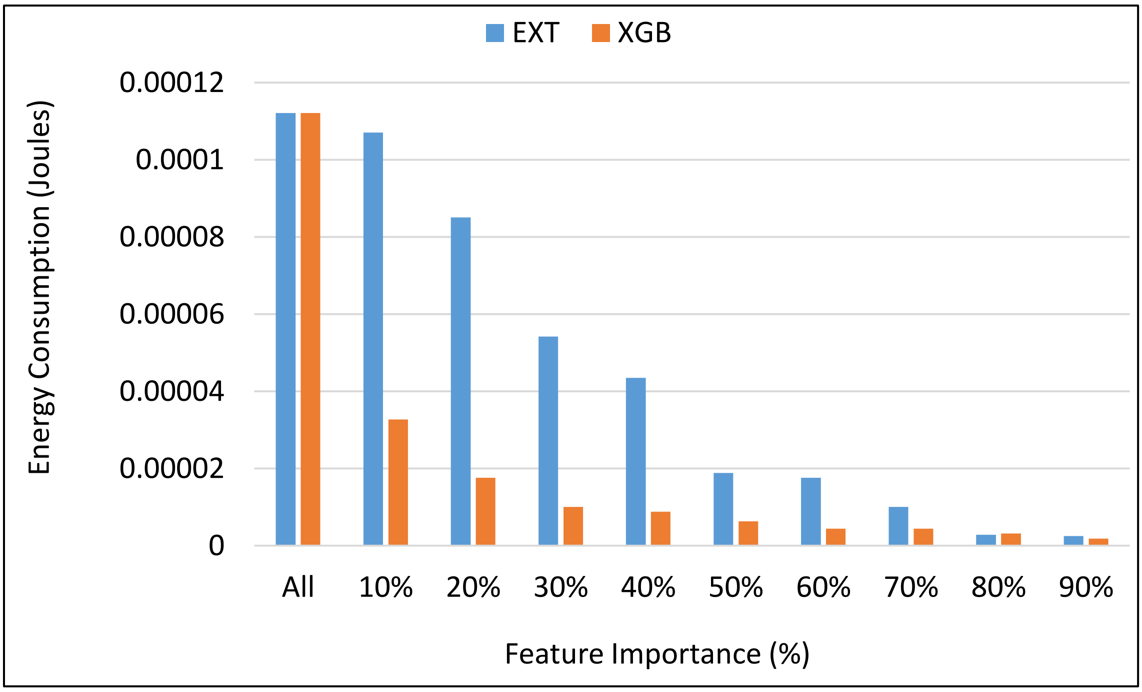


Figure 4.20. Energy consumption by IoMT by using FIM

4.6.5. Transmission Time

The proposed system is based on the use of the Internet of medical things to treat epilepsy cases, and therefore the process of sending data for the purpose of diagnosis should be as soon as possible and does not need a long transmission time. In this experiment, we show the amount of time needed to transmit data using Wi-Fi 4 and Wi-Fi 5 as displayed in Figure (4.21).

As we noticed in previous experiments, there is a direct relationship between the importance of attributes and the time required to send data.

The greater the importance of attributes, the less time is required for transmission as a result of deleting a large number of attributes.

Wi-Fi 4 and 5 use different data transmission rates, but in this experiment, we assumed the use of a data rate of 72 Mbps for Wi-Fi 4 and a data rate of 433 Mbps for Wi-Fi 5. From the results shown in Figure (4.21), we can discover that the transmission time depends on the size of the data and the chosen data rate. The smaller size of the data and higher data rates generally result in shorter transmission times. Also, it's clear from the results that Wi-Fi 5's higher potential data rates can lead to shorter transmission times compared to Wi-Fi 4. We also note that the XGB method takes less time in the transmission process as a result of deleting more features compared to the EXT method.

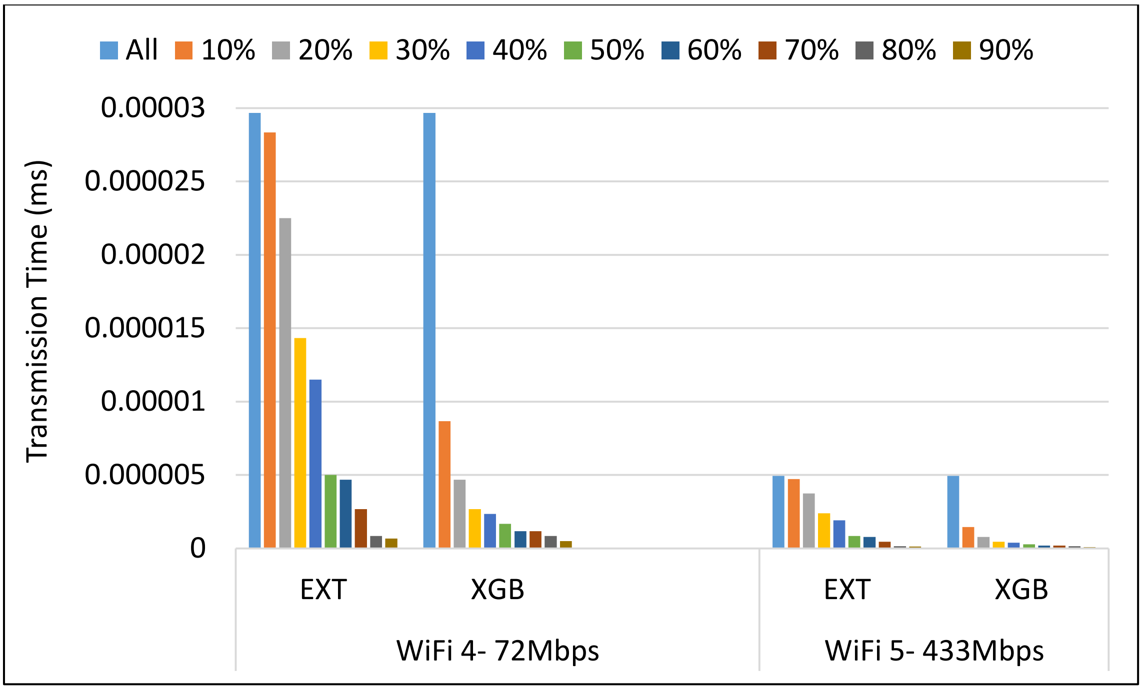


Figure 4.21. The transmission time of sent data via IoMT

It is clear to us through the results shown in the figures that XGB deletes a greater number of features compared to the EXT method. EXT computes feature importance by measuring impurity reduction across randomized decision trees, while XGB calculates feature importance by tracking how much each feature contributes to reducing the loss function during boosting iterations. EXT tends to have a higher level of model randomness due to their construction process, potentially leading to simpler trees. XGB aims to build complex models through boosting iterations, which can capture more intricate relationships in the data.

XGB's boosting process iteratively builds and combines weak learners to create a strong ensemble model that improves predictive accuracy by focusing on correcting errors made by previous models. This iterative process, combined with regularization and other techniques, makes XGB a powerful and widely used algorithm in machine learning. XGB is often considered one of the most powerful algorithms for structured/tabular data due to its sophisticated boosting process and regularization techniques.

In summary, while both EXT and XGB are ensemble methods that can provide feature importance scores, their approaches to constructing decision trees and calculating feature importance differ, leading to variations in their model performance and behavior.

4.6.6. Classification Performance

After the features have been selected in accordance with their importance threshold, we can apply any supervised classification models to predict the outcome. We will use the XGB and EXT methods to perform the classification process to detect whether or not there is epilepsy in the cloud layer based on the data sent from the fog layer. The aim of these experiments is to find out the accuracy of the results that can be obtained after deleting a certain percentage of the features according to their importance.

Figures (4.22 – 4.26) show the results of accuracy, precision, specificity, recall, and AUC that we obtained after reducing the number of transmitted features to conserve the energy of the fog tier for the features of importance from 10% to 90% and for the EXT and XGB classifiers, respectively. The figures clearly show that there is a trade-off between the different measures of classification and the amount of data transmitted. The higher amount of data transmitted (i.e. the less important the features), the higher the accuracy of the classification, and vice versa.

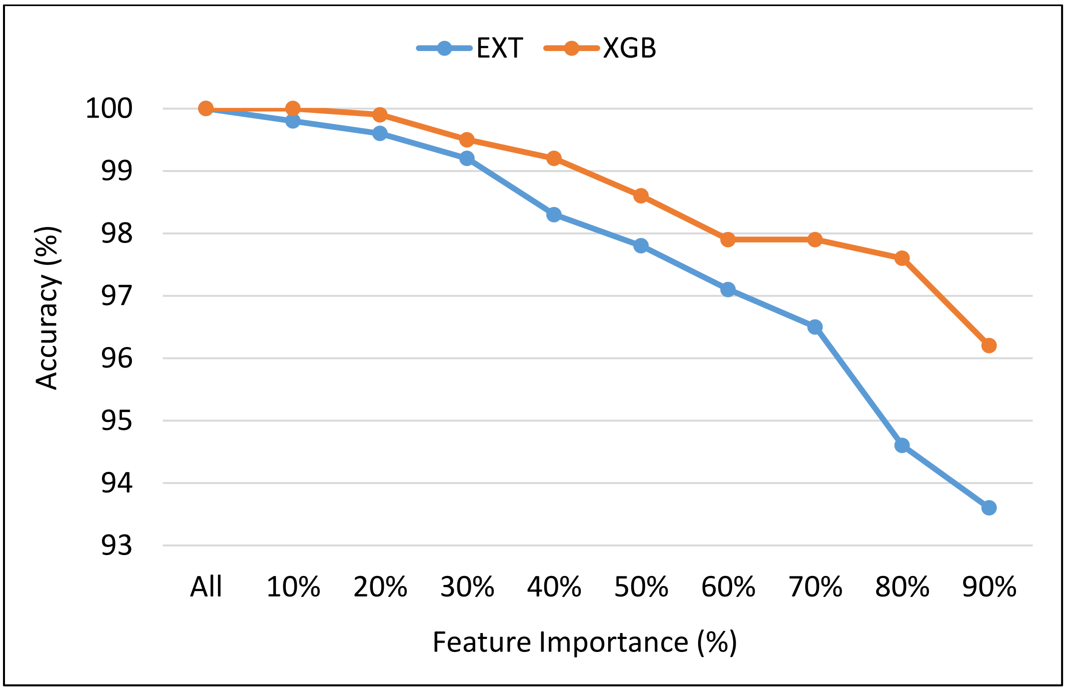


Figure 4.22. The accuracy metric using FIM and EXT, XGB

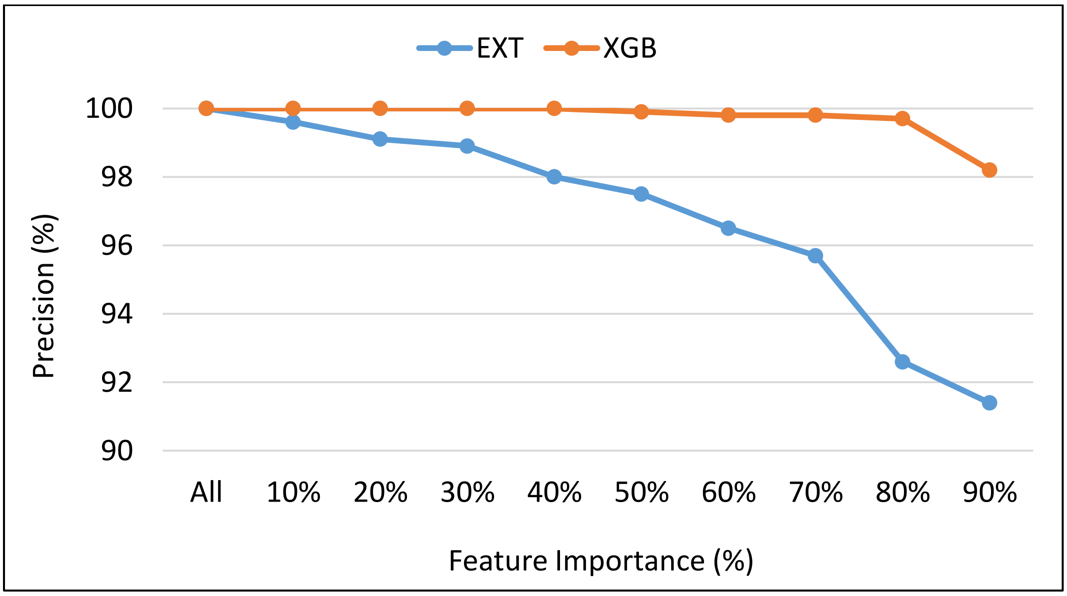


Figure 4.23 The precision metric using FIM and EXT, XGB

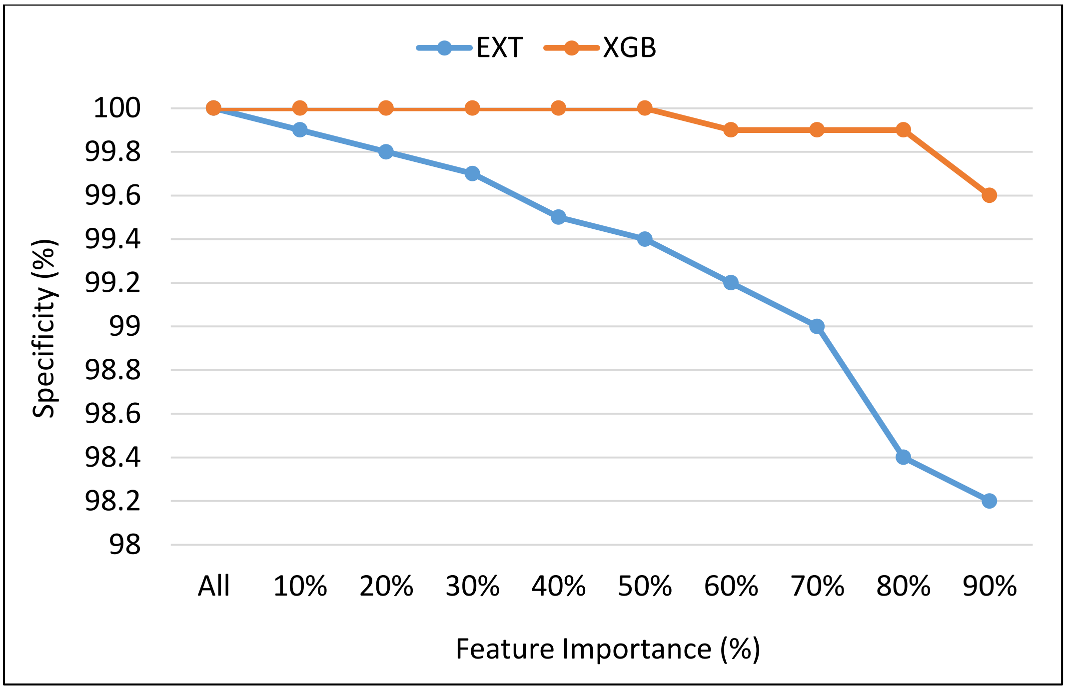


Figure 4.24. The specificity metric using EXT, XGB

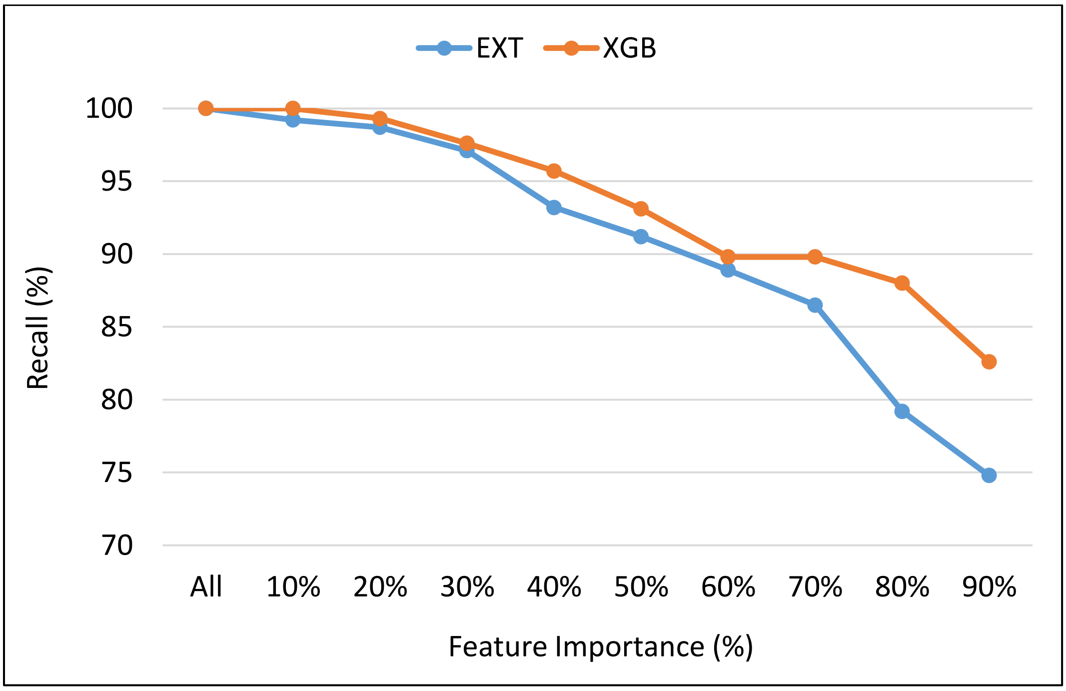


Figure 4.25. The recall metric uses EXT, and XGB.

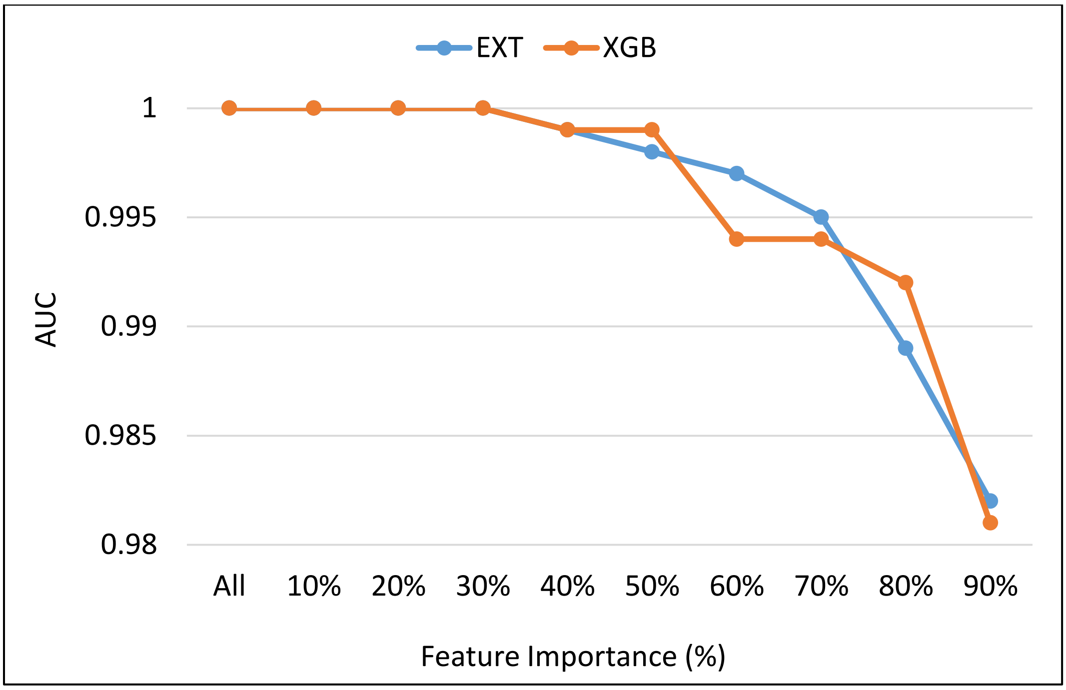


Figure 4.26. The AUC metric using EXT, XGB.

4.7. Comparison Results

Finally, the proposed methodology (in the fourth scenario) can be demonstrated to be more accurate for the detection of epileptic seizures in comparison to other recent works as shown in Table (4.9) which presents a comparative analysis of the performance evaluation of the proposed method with them. This analysis was conducted using the same EEG data set.

Table 4.9. Comparative analysis of the performance of the proposed work with existing works using the same EEG dataset.

Ref.	ML Classifier	Features	Performance metric	Findings	Our Proposed Method finding
[16] 2019	RF	DWT	Sensitivity	99.95%	100%
[19] 2019	SVM, KNN, DNN	Feature Scaling Loss Function	Accuracy	SVM=94%, KNN=74%	SVM=99.77% KNN=99.78%
[23] 2020	ANN, KNN, NB, SVM	DWT	Accuracy	ANN=97.82% KNN=97.58% NB=97.32% SVM=97.15%	KNN=99.78% NB=97.77% SVM=99.77%
[61] **2021**	RF	Fast Fourier Transform	Accuracy	96%	99.61%
[30] 2021	SVM, KNN	DWT, Local Binary Pattern Transition Histogram	Accuracy	SVM=99.6% KNN=99.6%	SVM=99.77% KNN=99.78%
[31] 2022	LR, NB, KNN, RF, LSTM	Wavelet Transform	Accuracy	RF=97%, KNN=91% NB=82% LR=81% LSTM=98%	RF=99.61% KNN=99.78% NB=99.77% LR=100%
[58] **2022**	KNN, LR, SGD, DT, RF, SVM, NB,	-	ROC	KNN=88.15% LR=55.163% SGD=66.44% DT=87.22% RF=96.05% SVM=94.32% NB=93.04%	KNN=99% LR=100% SGD=100% DT=100% RF=99% SVM=99% NB=99%
[35] 2023	RF	Entropy	Accuracy	96%	99.61%

The above analysis results reveal that this work which combines the ML models with the hyper-parameters, balanced EEG data, and the features extracted from WVG and the original dataset is very useful in detecting the epileptic seizures accurately.

The strength of the proposed system lies in the fact that it was able to overcome most of the limitations such as (the unbalanced nature of the data, which could make the classification stage tend towards the largest class, and choosing the best parameters for the machine learning models which is a difficult and time-consuming process).

These limitations could make the system provide low performance in detecting epileptic seizures, and this is unacceptable because it could put a person's life at risk.

Therefore, addressing these weak points reflected positively on the system's performance and thus on the results we obtained which demonstrate the efficiency of the proposed system.

CHAPTER FIVE

CONCLUSIONS AND FUTURE WORK SUGGESTIONS

CHAPTER FIVE

CONCLUSIONS AND FUTURE WORK SUGGESTIONS

5.1. Introduction

From the findings of the experiments based on the proposed method, conclusions are presented in this chapter. This chapter also suggested a number of follow-up projects to aid researchers working on EEG signals, particularly in the medical domains.

5.2. Conclusions

1. This thesis introduces an improved methodology for the automatic analysis and categorization of various EEG data based on IoMT with a specific focus on epileptic seizure detection.

2. In order to diagnose epilepsy, this study investigates that the EEG signals can be best represented by a weighted network, where the nodes interact with one another in variable degrees of strength.

3. As a result, several extraction features focus on gathering all crucial information from both the constructed graph and the original EEG data that represent time series.

4. The suggested system will give specialists the ability to accurately and quickly diagnose brain degenerative illnesses. Epilepsy patients will benefit from the results in terms of their quality of life.

5. This work helps to reduce the volume of data transmitted over the network for the purpose of preserving resources (i.e., memory space, time, and energy) for the non-epileptic patient by using the features importance technique.

5.3. Recommendations for Future Works

Plans and ideas for more research that could be done after the work detailed in this thesis are presented in this part:

1. In the future, researchers may use EEG to study dementia, autism, Alzheimer's disease, and other brain disorders. They may also use it to study motor imagery and mental imagery tasks.

2. It is urged that future researchers concentrate on the absence of epilepsy in kids and create a separate dataset specifically for this type of epilepsy.

3. Encourage the researchers in future to use the capabilities of deep learning techniques to detect this disorder.

REFERENCES

REFERENCES

[1] M. K. Jabar and A. K. M. Al-Qurabat, “Human Activity Diagnosis System Based on the Internet of Things,” in *Journal of Physics: Conference Series*, 2021, vol. 1879, no. 2, p. 22079.

[2] M. Iftikhar, S. A. Khan, and A. Hassan, “A survey of deep learning and traditional approaches for EEG signal processing and classification,” in *2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, 2018, pp. 395–400.

[3] U. R. Acharya, Y. Hagiwara, and H. Adeli, “Automated seizure prediction,” *Epilepsy Behav.*, vol. 88, pp. 251–261, 2018.

[4] Y. Roy, H. Banville, I. Albuquerque, A. Gramfort, T. H. Falk, and J. Faubert, “Deep learning-based electroencephalography analysis: a systematic review,” *J. Neural Eng.*, vol. 16, no. 5, p. 51001, 2019.

[5] B. P. Prathaban, R. Balasubramanian, and R. Kalpana, “ForeSeiz: An IoMT based headband for Real-time epileptic seizure forecasting,” *Expert Syst. Appl.*, vol. 188, p. 116083, 2022.

[6] P. Dhar, V. K. Garg, and M. A. Rahman, “Enhanced Feature Extraction-based CNN Approach for Epileptic Seizure Detection from EEG Signals,” *J. Healthc. Eng.*, vol. 2022, 2022.

[7] C. V. Anikwe *et al.*, “Mobile and Wearable Sensors for Data-driven Health Monitoring System: State-of-the-Art and Future Prospect,” *Expert Syst. Appl.*, p. 117362, 2022.

[8] A. K. M. Al-Qurabat, Z. A. Mohammed, and Z. J. Hussein, “Data traffic management based on compression and MDL techniques for smart agriculture in IoT,” *Wirel. Pers. Commun.*, vol. 120, no. 3, pp. 2227–2258, 2021.

[9] K. Singh and J. Malhotra, “IoT and cloud computing based automatic epileptic seizure detection using HOS features based random forest classification,” *J. Ambient Intell. Humaniz. Comput.*, 2019, doi: 10.1007/s12652-019-01613-7.

[10] M. A. Sayeed, S. P. Mohanty, E. Kougianos, and H. P. Zaveri, “eSeiz: An edge-device for accurate seizure detection for smart healthcare,” *IEEE Trans. Consum. Electron.*, vol. 65, no. 3, pp. 379–387, 2019.

[11] G. Regalia, F. Onorati, M. Lai, C. Caborni, and R. W. Picard, “Multimodal wrist-worn devices for seizure detection and advancing

research: focus on the Empatica wristbands," *Epilepsy Res.*, vol. 153, pp. 79–82, 2019.

[12] M. A. Sayeed, S. P. Mohanty, E. Kougianos, and H. P. Zaveri, "Neuro-Detect: A Machine Learning-Based Fast and Accurate Seizure Detection System in the IoMT," *IEEE Trans. Consum. Electron.*, vol. 65, no. 3, pp. 359–368, Aug. 2019, doi: 10.1109/TCE.2019.2917895.

[13] H. Daoud, P. Williams, and M. Bayoumi, "IoT based efficient epileptic seizure prediction system using deep learning," in *2020 IEEE 6th World Forum on Internet of Things (WF-IoT)*, 2020, pp. 1–6.

[14] P. A. E. Akashah and A. N. Shita, "An IoT platform for seizure alert wearable device," in *IOP Conference Series: Materials Science and Engineering*, 2020, vol. 767, no. 1, p. 12012.

[15] S. Hassan, E. Mwangi, and P. K. Kihato, "IoT based monitoring system for epileptic patients," *Heliyon*, vol. 8, no. 6, Jun. 2022, doi: 10.1016/j.heliyon.2022.e09618.

[16] K. D. Tzimourta *et al.*, "A robust methodology for classification of epileptic seizures in EEG signals," *Health Technol. (Berl).*, vol. 9, pp. 135–142, 2019.

[17] A. Subasi, J. Kevric, and M. Abdullah Canbaz, "Epileptic seizure detection using hybrid machine learning methods," *Neural Comput. Appl.*, vol. 31, pp. 317–325, 2019.

[18] S. M. Wani, S. Sabut, and S. L. Nalbalwar, "Detection of epileptic seizure using wavelet transform and neural network classifier," in *Computing, Communication and Signal Processing: Proceedings of ICCASP 2018*, 2019, pp. 739–747.

[19] D. K. Thara, B. G. PremaSudha, and F. Xiong, "Auto-detection of epileptic seizure events using deep neural network with different feature scaling techniques," *Pattern Recognit. Lett.*, vol. 128, pp. 544–550, 2019.

[20] Y. Wang, J. Cao, X. Lai, and D. Hu, "Epileptic state classification for seizure prediction with wavelet packet features and random forest," in *2019 Chinese Control And Decision Conference (CCDC)*, 2019, pp. 3983–3987.

[21] S. Raghu, N. Sriraam, S. Vasudeva Rao, A. S. Hegde, and P. L. Kubben, "Automated detection of epileptic seizures using successive decomposition index and support vector machine classifier in long-

term EEG," *Neural Comput. Appl.*, vol. 32, pp. 8965–8984, 2020.

[22] S. Chen, X. Zhang, L. Chen, and Z. Yang, "Automatic diagnosis of epileptic seizure in electroencephalography signals using nonlinear dynamics features," *IEEE Access*, vol. 7, pp. 61046–61056, 2019.

[23] W. Mardini, M. M. B. Yassein, R. Al-Rawashdeh, S. Aljawarneh, Y. Khamayseh, and O. Meqdadi, "Enhanced detection of epileptic seizure using EEG signals in combination with machine learning classifiers," *IEEE Access*, vol. 8, pp. 24046–24055, 2020.

[24] R. M. Aileni, S. Pasca, and A. Florescu, "EEG-brain activity monitoring and predictive analysis of signals using artificial neural networks," *Sensors*, vol. 20, no. 12, p. 3346, 2020.

[25] Z. Hu, C. Han, F. Guo, Q. Qin, S. Li, and Y. Qin, "Epileptic Seizure Prediction from the Scalp EEG Signals by using Random Forest Algorithm," in *2020 13th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, 2020, pp. 669–674.

[26] O. Karabiber Cura, S. Kocaaslan Atli, H. S. Türe, and A. Akan, "Epileptic seizure classifications using empirical mode decomposition and its derivative," *Biomed. Eng. Online*, vol. 19, pp. 1–22, 2020.

[27] S. M. Usman, S. Khalid, and M. H. Aslam, "Epileptic seizures prediction using deep learning techniques," *Ieee Access*, vol. 8, pp. 39998–40007, 2020.

[28] V. Harpale and V. Bairagi, "An adaptive method for feature selection and extraction for classification of epileptic EEG signal in significant states," *J. King Saud Univ. Inf. Sci.*, vol. 33, no. 6, pp. 668–676, 2021.

[29] A. Zarei and B. M. Asl, "Automatic seizure detection using orthogonal matching pursuit, discrete wavelet transform, and entropy based features of EEG signals," *Comput. Biol. Med.*, vol. 131, p. 104250, 2021.

[30] M. Yazid, F. Fahmi, E. Sutanto, W. Shalannanda, R. Shoalihin, and G.-J. Horng, "Simple detection of epilepsy from EEG signal using local binary pattern transition histogram," *IEEE Access*, vol. 9, pp. 150252–150267, 2021.

[31] H. O. Lekshmy, D. Panickar, and S. Harikumar, "Comparative analysis of multiple machine learning algorithms for epileptic seizure prediction," in *Journal of Physics: Conference Series*, 2022, vol.

2161, no. 1, p. 12055.

[32] S. Pattnaik, N. Rout, and S. Sabut, “Machine learning approach for epileptic seizure detection using the tunable-Q wavelet transform based time–frequency features,” *Int. J. Inf. Technol.*, vol. 14, no. 7, pp. 3495–3505, 2022.

[33] M. Shen, P. Wen, B. Song, and Y. Li, “An EEG based real-time epilepsy seizure detection approach using discrete wavelet transform and machine learning methods,” *Biomed. Signal Process. Control*, vol. 77, p. 103820, 2022.

[34] S. Chakrabarti, A. Swetapadma, and P. K. Pattnaik, “An improved method for recognizing pediatric epileptic seizures based on advanced learning and moving window technique,” *J. Ambient Intell. Smart Environ.*, vol. 14, no. 1, pp. 39–59, 2022.

[35] S. Goel, R. Agrawal, and R. K. Bharti, “Automated Epilepsy Detection using Machine Learning Classifiers based on Entropy Features,” in *2023 International Conference on Computational Intelligence, Communication Technology and Networking (CICTN)*, 2023, pp. 757–761.

[36] J. Weyhenmeyer and E. A. Gallman, “Rapid Review Neuroscience E-Book,” 2006.

[37] K. Rasheed *et al.*, “Machine Learning for Predicting Epileptic Seizures Using EEG Signals: A Review,” Feb. 2020, [Online]. Available: http://arxiv.org/abs/2002.01925

[38] N. Zhang and Z. Zhang, “The application of cognitive neuroscience to judicial models: recent progress and trends,” *Front. Neurosci.*, vol. 17, p. 1257004, 2023.

[39] N. J. Stevenson, K. Tapani, L. Lauronen, and S. Vanhatalo, “A dataset of neonatal EEG recordings with seizure annotations,” *Sci. data*, vol. 6, no. 1, pp. 1–8, 2019.

[40] P. PhysioBank, “Physionet: components of a new research resource for complex physiologic signals,” *Circulation*, vol. 101, no. 23, pp. e215–e220, 2000.

[41] A. H. Shoeb, “Application of machine learning to epileptic seizure onset detection and treatment.” Massachusetts Institute of Technology, 2009.

[42] M. Ihle *et al.*, “EPILEPSIAE–A European epilepsy database,” *Comput. Methods Programs Biomed.*, vol. 106, no. 3, pp. 127–138, 2012.

[43] R. G. Andrzejak, K. Lehnertz, F. Mormann, C. Rieke, P. David, and C. E. Elger, "Indications of nonlinear deterministic and finite-dimensional structures in time series of brain electrical activity: Dependence on recording region and brain state," *Phys. Rev. E*, vol. 64, no. 6, p. 61907, 2001.

[44] R. G. Andrzejak, K. Schindler, and C. Rummel, "Nonrandomness, nonlinear dependence, and nonstationarity of electroencephalographic recordings from epilepsy patients," *Phys. Rev. E*, vol. 86, no. 4, p. 46206, 2012.

[45] U. R. Acharya, S. V. Sree, G. Swapna, R. J. Martis, and J. S. Suri, "Automated EEG analysis of epilepsy: a review," *Knowledge-Based Syst.*, vol. 45, pp. 147–165, 2013.

[46] L. Kuhlmann, K. Lehnertz, M. P. Richardson, B. Schelter, and H. P. Zaveri, "Seizure prediction—ready for a new era," *Nat. Rev. Neurol.*, vol. 14, no. 10, pp. 618–630, 2018.

[47] N. D. Truong, L. Kuhlmann, M. R. Bonyadi, D. Querlioz, L. Zhou, and O. Kavehei, "Epileptic seizure forecasting with generative adversarial networks," *IEEE Access*, vol. 7, pp. 143999–144009, 2019.

[48] I. D. I. Saeedi and A. K. M. Al-Qurabat, "A systematic review of data aggregation techniques in wireless sensor networks," in *Journal of Physics: Conference Series*, 2021, vol. 1818, no. 1, p. 12194.

[49] K. Kourou, T. P. Exarchos, K. P. Exarchos, M. V Karamouzis, and D. I. Fotiadis, "Machine learning applications in cancer prognosis and prediction," *Comput. Struct. Biotechnol. J.*, vol. 13, pp. 8–17, 2015.

[50] V. Gulshan *et al.*, "Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs," *Jama*, vol. 316, no. 22, pp. 2402–2410, 2016.

[51] E. Moradi, A. Pepe, C. Gaser, H. Huttunen, J. Tohka, and A. D. N. Initiative, "Machine learning framework for early MRI-based Alzheimer's conversion prediction in MCI subjects," *Neuroimage*, vol. 104, pp. 398–412, 2015.

[52] Y. Yang *et al.*, "Epileptic seizure prediction based on permutation entropy," *Front. Comput. Neurosci.*, vol. 12, p. 55, 2018.

[53] M.-A. T. Vu *et al.*, "A shared vision for machine learning in neuroscience," *J. Neurosci.*, vol. 38, no. 7, pp. 1601–1607, 2018.

[54] M. Mahmud, M. S. Kaiser, A. Hussain, and S. Vassanelli, "Applications of deep learning and reinforcement learning to

biological data," *IEEE Trans. neural networks Learn. Syst.*, vol. 29, no. 6, pp. 2063–2079, 2018.

[55] H. Wang, G. Zhuo, and Y. Zhang, "Analyzing EEG signal data for detection of epileptic seizure: introducing weight on visibility graph with complex network feature," in *Databases Theory and Applications: 27th Australasian Database Conference, ADC 2016, Sydney, NSW, September 28-29, 2016, Proceedings 27*, 2016, pp. 56–66.

[56] S. Supriya, S. Siuly, H. Wang, J. Cao, and Y. Zhang, "Weighted Visibility Graph with Complex Network Features in the Detection of Epilepsy," *IEEE Access*, vol. 4, pp. 6554–6566, 2016, doi: 10.1109/ACCESS.2016.2612242.

[57] U. I. Awan, U. H. Rajput, G. Syed, R. Iqbal, I. Sabat, and M. Mansoor, "Effective classification of EEG signals using K-nearest neighbor algorithm," in *2016 International Conference on Frontiers of Information Technology (FIT)*, 2016, pp. 120–124.

[58] A. H. Attia and A. M. Said, "Brain seizures detection using machine learning classifiers based on electroencephalography signals: a comparative study," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 27, no. 2, pp. 803–810, 2022.

[59] A. A. Ein Shoka, M. H. Alkinani, A. S. El-Sherbeny, A. El-Sayed, and M. M. Dessouky, "Automated seizure diagnosis system based on feature extraction and channel selection using EEG signals," *Brain Informatics*, vol. 8, no. 1, Dec. 2021, doi: 10.1186/s40708-021-00123-7.

[60] H. Rajaguru and S. K. Prabhakar, "Sparse PCA and soft decision tree classifiers for epilepsy classification from EEG signals," in *2017 International Conference of Electronics, Communication and Aerospace Technology (ICECA)*, 2017, vol. 1, pp. 581–584.

[61] A. Basri and M. Arif, "Classification of seizure types using random forest classifier," *Adv. Sci. Technol. Res. J.*, vol. 15, no. 3, 2021.

[62] J. M. Montgomery and S. Olivella, "Tree-Based Models for Political Science Data," *Am. J. Pol. Sci.*, vol. 62, no. 3, pp. 729–744, 2018.

[63] P. Carmona, F. Climent, and A. Momparler, "Predicting failure in the US banking sector: An extreme gradient boosting approach," *Int. Rev. Econ. Financ.*, vol. 61, pp. 304–323, 2019.

[64] F. Sadoughi, A. Behmanesh, and N. Sayfouri, "Internet of things in medicine: a systematic mapping study," *J. Biomed. Inform.*, vol. 103,

p. 103383, 2020.

[65] A. K. M. Al-Qurabat and S. A. Abdulzahra, “An overview of periodic wireless sensor networks to the internet of things,” in *IOP Conference Series: Materials Science and Engineering*, 2020, vol. 928, no. 3, p. 32055.

[66] A. K. M. Al-Qurabat, “A lightweight Huffman-based differential encoding lossless compression technique in IoT for smart agriculture,” *Int. J. Com. Dig. Sys*, vol. 11, no. 1, 2022.

[67] H. K. Bharadwaj *et al.*, “A review on the role of machine learning in enabling IoT based healthcare applications,” *IEEE Access*, vol. 9, pp. 38859–38890, 2021.

[68] T. Roine *et al.*, “Reproducibility and intercorrelation of graph theoretical measures in structural brain connectivity networks,” *Med. Image Anal.*, vol. 52, pp. 56–67, 2019.

[69] G. Soni, “Signed visibility graphs of time series and their application to brain networks.” University of British Columbia, 2019.

[70] Y. Zou, R. V Donner, N. Marwan, J. F. Donges, and J. Kurths, “Complex network approaches to nonlinear time series analysis,” *Phys. Rep.*, vol. 787, pp. 1–97, 2019.

[71] H. Huanrui, “New mixed kernel functions of SVM used in pattern recognition,” *Cybern. Inf. Technol.*, vol. 16, no. 5, pp. 5–14, 2016.

[72] F. Karim, S. Majumdar, H. Darabi, and S. Harford, “Multivariate LSTM-FCNs for time series classification,” *Neural networks*, vol. 116, pp. 237–245, 2019.

[73] M. Zheng, S. Domanskyi, C. Piermarocchi, and G. I. Mias, “Visibility graph based community detection for biological time series,” *bioRxiv*, pp. 2003–2020, 2020.

[74] X. Lan, H. Mo, S. Chen, Q. Liu, and Y. Deng, “Fast transformation from time series to visibility graphs,” *Chaos An Interdiscip. J. Nonlinear Sci.*, vol. 25, no. 8, p. 83105, 2015.

[75] D. F. Yela, F. Thalmann, V. Nicosia, D. Stowell, and M. Sandler, “Online visibility graphs: Encoding visibility in a binary search tree,” *Phys. Rev. Res.*, vol. 2, no. 2, p. 23069, 2020.

[76] S. Sannino, S. Stramaglia, L. Lacasa, and D. Marinazzo, “Visibility graphs for fMRI data: Multiplex temporal graphs and their modulations across resting-state networks,” *Netw. Neurosci.*, vol. 1, no. 3, pp. 208–221, 2017.

[77] J. Liu and Q. Li, “Planar visibility graph network algorithm for two dimensional timeseries,” in *2017 29th Chinese Control And Decision Conference (CCDC)*, 2017, pp. 1352–1357.

[78] L. Guzmán-Vargas, B. Obregón-Quintana, D. Aguilar-Velázquez, R. Hernández-Pérez, and L. S. Liebovitch, “Word-length correlations and memory in large texts: A visibility network analysis,” *Entropy*, vol. 17, no. 11, pp. 7798–7810, 2015.

[79] G. Zhu, Y. Li, and P. P. Wen, “Epileptic seizure detection in EEGs signals using a fast weighted horizontal visibility algorithm,” *Comput. Methods Programs Biomed.*, vol. 115, no. 2, pp. 64–75, 2014.

[80] M. Zeng, W. Xu, C. Zhao, Q. Li, and J. Han, “Weighted Complex Network Based on Visibility Angle Measurement,” in *2020 39th Chinese Control Conference (CCC)*, 2020, pp. 1138–1143.

[81] H. Yu *et al.*, “Identification of Alzheimer’s EEG With a WVG Network-Based Fuzzy Learning Approach,” *Front. Neurosci.*, vol. 14, Jul. 2020, doi: 10.3389/fnins.2020.00641.

[82] T. Kong, J. Shao, J. Hu, X. Yang, S. Yang, and R. Malekian, “Eeg-based emotion recognition using an improved weighted horizontal visibility graph,” *Sensors*, vol. 21, no. 5, pp. 1–22, Mar. 2021, doi: 10.3390/s21051870.

[83] Z. Zhang, Y. Qin, L. Jia, and X. Chen, “Visibility graph feature model of vibration signals: a novel bearing fault diagnosis approach,” *Materials (Basel).*, vol. 11, no. 11, p. 2262, 2018.

[84] G. P. Clemente and R. Grassi, “Directed clustering in weighted networks: A new perspective,” *Chaos, Solitons & Fractals*, vol. 107, pp. 26–38, 2018.

[85] F. De Pasquale, S. Della Penna, O. Sporns, G. L. Romani, and M. Corbetta, “A dynamic core network and global efficiency in the resting human brain,” *Cereb. Cortex*, vol. 26, no. 10, pp. 4015–4033, 2016.

[86] M.-C. Ho *et al.*, “Detect AD patients by using EEG coherence analysis,” *J. Med. Eng.*, vol. 2014, 2014.

[87] C. Groenland, T. Kaiser, O. Treffers, and M. Wales, “Graphs of low average degree without independent transversals,” *J. Graph Theory*, vol. 102, no. 2, pp. 374–387, 2023.

[88] M. E. J. Newman, “Analysis of weighted networks,” *Phys. Rev. E*, vol. 70, no. 5, p. 56131, 2004.

[89] K.-W. Jie, S.-Y. Liu, X.-J. Sun, and Y.-C. Xu, “A dynamic ripple-spreading algorithm for solving mean–variance of shortest path model in uncertain random networks,” *Chaos, Solitons & Fractals*, vol. 167, p. 113081, 2023.

[90] F. Li, X. Zhang, X. Zhang, C. Du, Y. Xu, and Y.-C. Tian, “Cost-sensitive and hybrid-attribute measure multi-decision tree over imbalanced data sets,” *Inf. Sci. (Ny).*, vol. 422, pp. 242–256, 2018.

[91] N. V Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, “SMOTE: synthetic minority over-sampling technique,” *J. Artif. Intell. Res.*, vol. 16, pp. 321–357, 2002.

[92] F. Yang, K. Wang, L. Sun, M. Zhai, J. Song, and H. Wang, “A hybrid sampling algorithm combining synthetic minority over-sampling technique and edited nearest neighbor for missed abortion diagnosis,” *BMC Med. Inform. Decis. Mak.*, vol. 22, no. 1, p. 344, 2022.

[93] Y. Xia, C. Liu, Y. Li, and N. Liu, “A boosted decision tree approach using Bayesian hyper-parameter optimization for credit scoring,” *Expert Syst. Appl.*, vol. 78, pp. 225–241, 2017.

الخلاصة

إن الاضطراب العصبي المعروف بالصرع له تأثير سلبي مستمر على الدماغ. يعد تحديد النوبات أمرًا ضروريًا للرعاية السريرية للأفراد المصابين بالصرع. كثيرًا ما يستخدم الأطباء الخبراء تحليل بيانات تخطيط كهربية الدماغ البصري (EEG) للكشف عن نوبات الصرع، وهي طريقة لمراقبة النشاط الكهربائي غير الخطي للخلايا العصبية في الدماغ. إنها أداة تشخيصية للكشف عن الصرع. نظرًا لأن الأمر يستغرق وقتًا طويلاً من قبل أطباء الأعصاب للمراجعة، فإن التفسير اليدوي لبيانات تخطيط كهربية الدماغ المستمدة من تسجيلات إشارات تخطيط كهربية الدماغ من مريض واحد يعد عملية صعبة للغاية وتستغرق وقتًا طويلاً.

في هذه الأطروحة، نقترح إطار عمل إنترنت الأشياء (IoT) للكشف الدقيق والفعال عن النوبات ومراقبتها لمرضى الصرع باستخدام تقنيات التعلم الآلي. تشكل ثلاث طبقات إطار عمل إنترنت الأشياء المقترح: طبقات الأشياء/الأجهزة، والضباب، والسحابة. تتلخص الطريقة المقترحة في نقل البيانات المجمعة من طبقة الشيء إلى طبقة FoG حيث يتم تنفيذ عدة خطوات حاسمة بدءاً من تجزئة بيانات مخطط كهربية الدماغ (EEG) وتحويلها إلى تنسيق جدول ثنائي الأبعاد وإنشاء Weighted Visibility Graph (WVG) من بيانات تخطيط كهربية الدماغ (EEG). تستخرج طريقتنا المقترحة تسع ميزات من WVG وعشر ميزات إحصائية إضافية من مجموعة بيانات EEG الأصلية. يتم تغذية كل هذه الميزات إلى أساليب التعلم الآلي لتصنيف الإشارة التي تم الحصول عليها على أنها طبيعية أو غير طبيعية. عشرة من أساليب التعلم الآلي الأكثر شيوعًا المستخدمة في هذا النظام المقترح تسمى Logistic Regression LR ، K-Nearest Neighbor KNN، Support Vector Machine SVM، Stochastic Gradient Descent SGD، Naïve Bayes NB، Decision Tree DT، Random Forest RF، Extra Tree Classifier EXT، Gradient Boosting GB، و Extreme Gradient Boosting XGB.

بعد تصنيف الإشارة، سيتم اتخاذ أحد الإجراءين اعتمادًا على حالة التصنيف: إما إرسال إشعار إلى أي مقدم رعاية محدد مسبقًا في حالة حدوث نوبة أو تقليل البيانات باستخدام الطريقة المعتمدة على العتبة في حالة غياب النوبة. ونتيجة لذلك، وفي كلتا الحالتين، يتم تحميل البيانات إلى الطبقة السحابية لمراجعتها لاحقًا من قبل فريق طبي متخصص.

تم استخدام أربعة سيناريوهات لتقييم طريقتنا المقترحة باستخدام مقاييس تقييم الأداء مثل accuracy، precision، F1 score، Specificity ، وما إلى ذلك. وتتجلى قوة الأساليب

المقدمة من خلال الاستراتيجية المقترحة، والتي تنتج نسبة 100٪ في السيناريو الرابع الذي يستخدم نماذج ML مع ضبط المعلمات الفائقة وبيانات EEG المتوازنة والميزات المستخرجة.

وزارة التعليم العالي و البحث العلمي

جامعـــــــــة بابـــــــــل

كليـــــــــة العلـــوم للبنـــــــات

قســـــم علـــــوم الحـاســـــوب

نظام المراقبة والكشف المعتمد على إنترنت الأشياء لنوبات الصرع باستخدام التعلم الآلي

رسالة مقدمة الى مجلس كلية العلوم للبنات في جامعة بابل وهي جزء من متطلبات الحصول على درجة الماجستير في العلوم / قسم علوم الحاسوب

مقدمة من قبل

الاء لطيف نور

باشراف

الاستاذ المساعد الدكتور

علي كاظم محمد الغرابي

1445 هـ **2023 م**