NATIONAL ENGINEERING SCHOOL OF TUNIS

Information and Communication Technologies Department

END OF THE YEAR PROJECT II

Study of Deep Learning Models for the Classification of Epileptic Patients

Elaborated by:

Soussi Eya

Abidi Ayoub

2nd Year Telecommunications

Supervised by :

Ms Bouzouita Ines : Assistant Professor
Ms Brari Zayneb : Postdoctoral Researcher

Conducted at
LR-RISC-ENIT

University Year : 2024/2025

Acknowledgments

We would like to express our deepest gratitude to all those who have supported and guided us throughout the completion of this final-year project. First and foremost, we extend our sincere thanks to our academic supervisors, **Ms. Ines Bouzouita**, Assistant Professor, and **Ms. Zayneb Brari**, Postdoctoral Researcher, for their continuous support, valuable advice, and insightful feedback. Their expertise and mentorship were vital in helping us navigate the different stages of this research.

This project was conducted within the **LR-RISC-ENIT** laboratory (Laboratoire de Recherche en Robotique, Informatique & Systèmes Complexes) at the National Engineering School of Tunis (ENIT). We are particularly thankful to the laboratory for providing us with a productive and well-equipped research environment that facilitated the successful execution of our work.

Our research journey began in October 2024, and was structured around a series of weekly meetings, during which we discussed our progress, addressed challenges, and refined our objectives. The first phase of our project was dedicated to exploring the fundamentals of Artificial Intelligence and deep learning models. This was followed by an in-depth study of preprocessing techniques, which formed the foundation for the experimental and analytical components of our work. Over time, this project has not only enhanced our technical and academic competencies but also reinforced our skills in scientific reasoning, teamwork, and research methodology.

We are sincerely appreciative of the opportunity to carry out this research within such a dynamic and inspiring environment, and we thank everyone at LR-RISC-ENIT who contributed, directly or indirectly, to the realization of this project.

Abstract

Epilepsy affects about 6.38 per 1,000 people worldwide, making accurate seizure classification a key challenge for effective treatment. Electroencephalography (EEG), a non-invasive method of recording brain activity, is commonly used for seizure detection. However, analyzing the complex and nonlinear EEG signals is time-consuming and prone to human error. Recent advances in deep learning (DL) offer promising solutions for automating EEG analysis.

This study evaluates three deep learning models: Artificial Neural Network (ANN), Convolutional Neural Network (CNN), and a hybrid CNN-LSTM model for seizure type classification using the Bonn EEG database. The models were tested in both binary and multi-class classification tasks. In the multi-class experiment, the highest accuracy achieved was 89.17% for ANN and CNN, and 84.44% for CNN-LSTM. For binary classification , the best accuracies were 90.00% for ANN, 95.56% for CNN, and 94.44% for CNN-LSTM.

These results show that deep learning models can effectively classify epileptic seizures, offering a potential tool for faster and more reliable diagnosis.

Keywords : EEG signal classification, Deep learning , Binary and multi-class classification , Bonn database

Contents

List of Figures

List of Tables

List of Acronyms

- **Adam:** Adam Optimizer
- **ANN:** Artificial Neural Network
- **CNN:** Convolutional Neural Network
- **CNN-LSTM:** Convolutional Neural Network - Long Short-Term Memory
- **DER:** Derivative Signal Processing
- **DL:** Deep Learning
- **EEG:** Electroencephalogram
- **Keras:** High-Level Neural Networks API
- **LSTM:** Long Short-Term Memory
- **ML:** Machine Learning
- **NumPy:** Numerical Python (Library for Data Manipulation)
- **OPT:** Optimizer Tuning
- **RAW:** Raw EEG data (no preprocessing)
- **SVM:** Support Vector Machine
- **STN:** Feature Standardization
- **SQLite:** Structured Query Language Database
- **TensorFlow:** Open Source Machine Learning Framework
- **PyWavelets:** Python Wavelet Transform Library
- **WAV:** Wavelet Decomposition

General Introduction

Epilepsy is a neurological disorder marked by recurrent seizures due to abnormal electrical activity in the brain, affecting millions globally. Early detection is crucial, as timely treatment can significantly reduce seizure frequency and improve patient outcomes. However, diagnosing epilepsy through traditional methods, like Electroencephalography (EEG), presents challenges. While EEG is an effective tool for monitoring brain activity, manually analyzing the data is labor-intensive and susceptible to human error, given the complex and noisy nature of the signals.
To address this, there has been increasing interest in automating seizure detection using advanced computational techniques. Despite advancements, challenges persist, such as the non-stationary nature of EEG signals, patient-specific differences, and noise interference, highlighting the need for more reliable automated methods.
This project aims to enhance seizure detection by applying deep learning methods to EEG signals. Specifically, it will utilize Convolutional Neural Networks (CNNs) and hybrid CNN-LSTM models to automatically classify seizure and non-seizure events, improving accuracy and reducing manual effort. The project will also develop a user-friendly interface for real-time seizure detection, making it practical for clinical use.
The methodology involves preprocessing EEG data to reduce noise and extract meaningful features, followed by training deep learning models to classify the signals. The final output will be a classification interface that can assist healthcare professionals in diagnosing epilepsy more efficiently.
This report is organized into four chapters. Chapter 1 provides the background on epilepsy, EEG analysis, preprocessing techniques, and the deep learning architectures employed in this project. Chapter 2 reviews previous studies on epileptic seizure classification and presents the methodology proposed for this study. In Chapter 3, the results of the experiments are presented, evaluating the performance of the models and comparing different deep learning approaches. Finally, Chapter 4 discusses the development of the EEG classification interface, detailing the system architecture, core modules, and the validation process.

Chapter 1

Context and Theoretical Foundations

This chapter introduces the main concepts used throughout this study. It first explains epilepsy and how EEG signals are important for detecting seizures. Then, it discusses different EEG preprocessing techniques to prepare the data for analysis. Finally, it presents deep learning models, such as Artificial Neural Networks, Convolutional Neural Networks, and the CNN-LSTM hybrid model, which will be used later for seizure detection.

1.1 Understanding Epilepsy and EEG Analysis

In this section, we provide an overview that sets the foundation for understanding epilepsy and its clinical implications. We begin by defining epilepsy and describing its key features, including the nature of seizures and their impact on brain function. Following this, we discuss the significance of EEG signal analysis as a primary tool for detecting and studying epileptic activity, emphasizing its critical role in the diagnosis and management of epilepsy.

1.1.1 Definition and Characteristics of Epilepsy

Epilepsy is a chronic neurological disorder characterized by sudden and abnormal discharges of brain neurons, leading to temporary brain dysfunction. It affects approximately 50 million people worldwide, impacting individuals of all ages, from infants to older adults. People with epilepsy experience a variety of symptoms, including generalized convulsions, loss of consciousness, and, in some cases, involuntary bodily functions such as urinary incontinence. The unpredictable nature of seizures can pose severe risks, including irreversible brain damage and life-threatening situations. It is the second most common neurological disorder after migraines and is caused by disturbances in the electrical activity of brain cells. These uncontrolled electrical discharges in neurons can remain localized in a specific brain region or spread to other areas, leading to more severe seizures[3][4].

According to a 2017 World Health Organization (WHO) report, approximately 10% of people experience an epileptic episode at some point in their lives. Epilepsy disrupts normal brain activity, leading to episodes of altered consciousness, involuntary muscle movements, and sensory disturbances. These seizures originate from sudden electrical discharges in the cerebral cortex, temporarily affecting motor functions, cognitive abilities, and sensory perception. On average, epilepsy affects around 6.38 per 1,000 individuals globally, highlighting its prevalence and the need for effective management strategies. In addition to seizures, many individuals with epilepsy experience comorbid conditions such as depression, anxiety, dementia, migraines, and cardiovascular diseases. Proper classification of seizure types is crucial for accurate diagnosis and treatment. Misdiagnosing epilepsy or failing to identify seizure variations can lead to worsened health conditions, decreased quality of life, and ineffective medical interventions. Understanding epilepsy's diverse manifestations is essential for developing better treatment approaches, improving patient care, and reducing the risks associated with uncontrolled seizures[5].

1.1.2 EEG Signal Analysis and Its Importance in Seizure Detection

Electroencephalography (EEG) is a widely used non-invasive technique for recording electrical activity in the brain. By placing an array of electrodes on the scalp, EEG captures neural oscillations that are critical for diagnosing and managing various neurological disorders, including epilepsy. Since its introduction in the 1940s, EEG has played a fundamental role in epilepsy assessment, enabling clinicians to monitor seizure activity and abnormal brain functions. However, the continuous, nonlinear, and non-stationary nature of EEG signals presents significant challenges in analysis and interpretation, often requiring extensive expertise and prolonged observation. One of the main difficulties in EEG-based epilepsy detection is the randomness of the signal, as brain activity is highly dynamic and influenced by numerous internal and external factors. EEG signals are susceptible to noise from muscle movements, eye blinks, and external electromagnetic interference, making it difficult to distinguish true epileptic patterns from artifacts. Additionally, the variability in seizure manifestations across patients complicates the classification process, requiring advanced computational techniques for accurate detection[6].

A typical EEG recording setup is illustrated in Figure 1.1, highlighting electrode placement on the scalp and the capture of brainwave activity.

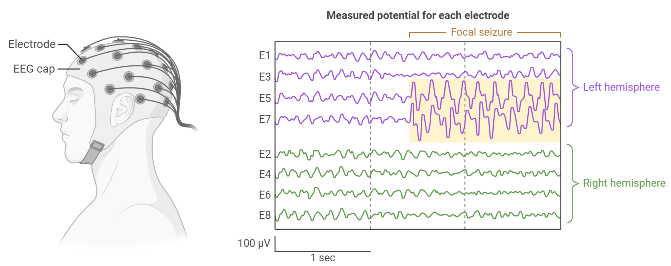


Figure 1.1: EEG recording setup [1].

Traditionally, EEG seizure detection relied on binary classification (seizure vs. non-seizure), however, a more detailed approach categorizing seizures into types such as focal, generalized, tonic-clonic, absence, and myoclonic is essential for precise diagnosis and treatment. Misclassification of seizure types can lead to ineffective treatment strategies and delays in medical intervention. Another major challenge is the manual review of EEG recordings, which can take hours or even days, increasing the burden on neurologists. Continuous monitoring is required to capture interictal (between seizures) and ictal (during seizures) events, but the interpretation remains complex due to the variability in EEG waveforms. In some cases, other conditions, such as syncope, can mimic epileptic seizures, further complicating diagnosis. The integration of deep learning (DL) techniques has significantly advanced EEG analysis, enabling more accurate and automated classification of seizure patterns. Artificial Neural Networks (ANNs) provide a foundation for recognizing complex patterns in EEG signals, while specialized architectures like Convolutional Neural Networks (CNNs) excel in extracting spatial features by detecting characteristic signal variations [3]. However, EEG signals also have strong temporal dependencies, making hybrid models such as CNN-LSTM (Long Short-Term Memory) particularly effective. CNNs capture spatial features, while LSTMs process sequential dependencies, allowing for improved recognition of seizure onset and progression [7]. However, despite these technological improvements, the inherent unpredictability and complexity of EEG signals remain key challenges, necessitating further research to enhance the reliability and efficiency of seizure detection systems.

1.2 EEG Data Preprocessing

In this section, we will explore the various data preprocessing techniques that are essential for enhancing the quality of EEG signals. Specifically, we will focus on methods such as wavelet decomposition and derivative-based preprocessing, which play a critical role in reducing noise, filtering artifacts, and highlighting important features in the raw EEG data.[8]

1.2.1 Data Preprocessing Techniques

Preprocessing plays a crucial role in enhancing the performance of EEG signal analysis. Raw EEG signals are often contaminated by various sources of noise and artifacts that can obscure important features, making it difficult for models to effectively identify seizure patterns. Without proper preprocessing, the accuracy and reliability of seizure detection systems would be significantly compromised.

The primary goal of preprocessing is to remove unwanted noise and artifacts, ensuring that the features extracted from the data reflect the underlying brain activity rather than extraneous factors. Furthermore, preprocessing helps in optimizing the signal for feature extraction by selecting the appropriate frequency bands and channels. EEG signals are recorded from multiple electrodes, each capturing different aspects of the brain's electrical activity. By focusing on the most relevant channels and frequency bands, preprocessing reduces the dimensionality of the data, making the analysis more efficient and effective [8].

Techniques such as wavelet decomposition and derivative-based methods are often employed during this stage to enhance the signal and prepare it for subsequent analysis. These preprocessing steps are critical in ensuring the quality and relevance of the data, ultimately improving the performance of machine learning models used for seizure detection and classification.

1.2.2 Wavelet Decomposition for EEG Preprocessing

Wavelet decomposition is a powerful mathematical tool for analyzing and processing EEG signals, particularly due to its ability to capture both time-domain and frequency-domain characteristics. Unlike traditional Fourier-based methods, which assume stationarity in signals, wavelet decomposition provides a multi-resolution approach that allows the examination of EEG signals at different scales. This is particularly important for EEG data, as brain activity exhibits complex, transient patterns that vary over time[9].

The **Discrete Wavelet Transform (DWT)** is widely used for EEG preprocessing because it effectively separates neural activity from noise and artifacts. The process involves successive decomposition of the signal into approximation coefficients (which retain the low-frequency components of the signal) and detail coefficients (which capture the high-frequency components). This hierarchical approach allows the extraction of relevant EEG features while filtering out unwanted interference [10]. Mathematically, the wavelet decomposition of a signal $x(t)$ is expressed as follows:

$$x(t) = \sum_{j,k} c_{j,k}\psi_{j,k}(t) + \sum_{j,k} d_{j,k}\phi_{j,k}(t) \tag{1.2.1}$$

where:

- $c_{j,k}$ are the approximation coefficients, representing the coarse, low-frequency structure of the signal.
- $d_{j,k}$ are the detail coefficients, which preserve fine, high-frequency variations.
- $\psi_{j,k}(t)$ and $\phi_{j,k}(t)$ are the wavelet and scaling functions, respectively, which form the basis for signal reconstruction.

To efficiently process EEG signals, the **Daubechies-4 (db4) wavelet** is commonly used due to its compact support and optimal time-frequency localization. In our work, we employ a **five-level wavelet decomposition**, which provides a balanced trade-off between signal smoothing and preservation of critical neural features. This approach enhances the quality of EEG data, ensuring that meaningful brain activity is retained while minimizing the impact of noise and artifacts[9].

1.2.3 Derivation-based Preprocessing

EEG signals are known for their strong non-stationary nature, where their statistical properties, such as mean and variance, evolve over time. This non-stationarity complicates direct analysis and necessitates specific preprocessing strategies to stabilize the signals before feature extraction. Based on the theoretical foundations laid by Box and Jenkins [11], differentiation or derivation is recognized as an essential tool for addressing non-stationarity in time series data. By applying derivation, slow-varying trends and baseline drifts are suppressed, while the rapid fluctuations and transient components, which often contain crucial information about neurological events such as seizures, are accentuated. In this context, third-order differentiation is employed, guided by the research findings reported by Dickey and Pantula [12], which highlight its effectiveness in enhancing the signal's informative components without excessively amplifying noise. The application of third-order derivation allows for the amplification of subtle variations linked to seizure activity, making the underlying patterns more prominent and improving the sensitivity of subsequent machine learning models. This preprocessing step therefore plays a critical role in transforming raw EEG data into a form that is more amenable to accurate detection and classification of epileptic seizures.

1.3 Deep Learning Architectures for EEG Analysis

To improve EEG data analysis, Deep Learning (DL) has emerged as a powerful tool in this field. As a branch of machine learning, deep learning centers on training artificial neural networks biologically inspired models made up of multiple interconnected layers that apply both linear and nonlinear transformations to the input data, often consisting of segments of multichannel EEG recordings. These models have demonstrated the ability to capture intricate patterns by learning multi-level representations of the data.[8]

In this section, we present some of the most widely used deep learning architectures for EEG analysis. We first introduce Artificial Neural Networks (ANNs), the fundamental building blocks of deep learning, before exploring Convolutional Neural Networks (CNNs), which have proven effective in extracting spatial features from EEG signals. Finally, we examine hybrid models combining CNNs with Long Short-Term Memory (LSTM) networks to capture both spatial and temporal dependencies in EEG data.

1.3.1 Artificial Neural Networks

1.3.1.1 Definition of Artificial Neural Networks

An *Artificial Neural Network (ANN)* is a computational model inspired by the structure and functioning of the human brain. It has the capacity to automatically learn from data and is particularly effective in recognizing complex patterns. This characteristic renders ANNs highly valuable for tasks such as detecting seizures from EEG signals. The learning process involves the progressive adjustment of internal parameters, known as *weights*, as the network processes increasing amounts of data [13].

1.3.1.2 Structure of Artificial Neural Networks

Artificial Neural Networks are organized into multiple layers, each fulfilling a specific role in data processing and pattern learning. The main components are the input layer, hidden layers, and the output layer.

The input layer is responsible for receiving the initial data, which may consist of either raw EEG signals or pre-processed features. The input is commonly represented as a vector $X = [x_1, x_2, ..., x_n]$, where each element x_i corresponds to an EEG feature. Neurons in the input layer act as transmitters, forwarding the received inputs directly to the next layer without performing any transformations.

The hidden layers perform the essential task of learning underlying patterns from the input data. Within these layers, each neuron computes a weighted sum of its inputs, adds a bias term, and applies a non-linear transformation through an activation function. Mathematically, the output of a neuron j in a hidden layer can be expressed as:

$$Z_j = \sum_{i=1}^{n} W_{ij} x_i + b_j$$

where W_{ij} denotes the weight associated with the connection between the input i and neuron j, x_i represents the input feature, and b_j is the corresponding bias. The resulting value Z_j is subsequently passed through an activation function, such as ReLU or sigmoid, to introduce the necessary non-linearity that allows the network to model complex relationships.

The output layer generates the final classification or prediction based on the features extracted by the hidden layers. In binary classification problems, a sigmoid activation function is typically employed to provide a probability value between 0 and 1. For multi-class classification tasks, a softmax function is applied to calculate the probabilities of each possible class, as shown by:

$$P(H), \quad P(E), \quad P(S)$$

where H, E, and S represent different classification categories, such as various seizure states.

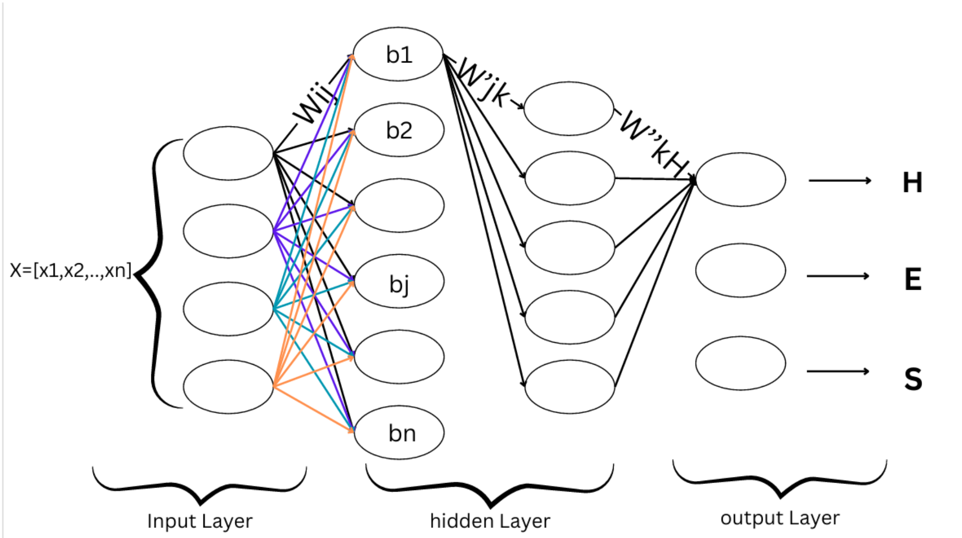


Figure 1.2: Proposed ANN architecture.

1.3.1.3 Activation Functions

Activation functions are fundamental components of artificial neural networks, serving to introduce non-linearity into the model. This non-linearity is crucial, as it enables the network to approximate complex functions and learn intricate patterns within data. Without activation functions, a neural network would behave as a simple linear regression model, significantly limiting its representational capacity and effectiveness in solving real-world problems [14].
Real-world data, including biomedical signals such as EEG recordings, frequently exhibit highly non-linear behaviors. The integration of activation functions allows neural networks to capture and model these complex relationships, thereby enhancing their predictive performance and generalization capabilities.

Several types of activation functions have been developed, each suited for specific tasks and exhibiting distinct advantages and limitations.

Linear Activation Function: The linear activation function is very simple:

$$f(x) = x$$

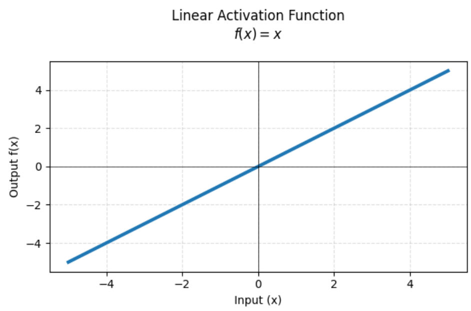


Figure 1.3: Linear activation function plot

It is mostly used when we need to predict continuous numbers, like in regression problems. However, because it is a straight line, it cannot help the network learn complicated patterns.

Sigmoid Activation Function: The sigmoid activation function changes the input into a value between 0 and 1:

$$f(x) = \frac{1}{1 + e^{-x}}$$

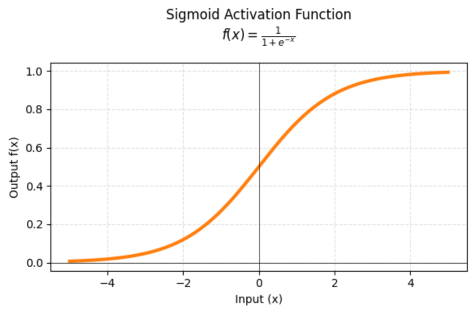


Figure 1.4: Sigmoid activation function plot

It is mainly used for binary classification. One disadvantage is that when the input is too big or too small, the function's gradient becomes very small, making the training slower. This is called the vanishing gradient problem.

Tanh (Hyperbolic Tangent) Activation Function: The tanh activation function is written as:

$$f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

It is similar to the sigmoid but gives outputs between -1 and 1. This helps make the data more balanced around zero, which can help training. However, like sigmoid, tanh can also suffer from the vanishing gradient problem.

ReLU (Rectified Linear Unit) Activation Function: The ReLU activation function is defined as:

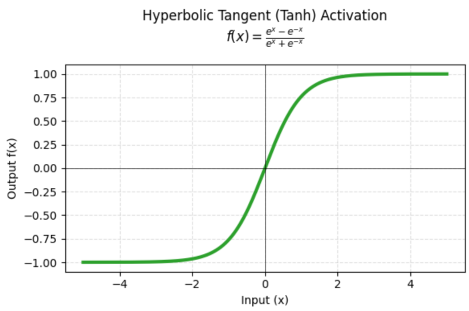


Figure 1.5: Hyperbolic tangent activation function plot

$$f(x) = \begin{cases} 0, & \text{if } x < 0 \\ x, & \text{if } x \geqslant 0 \end{cases}$$

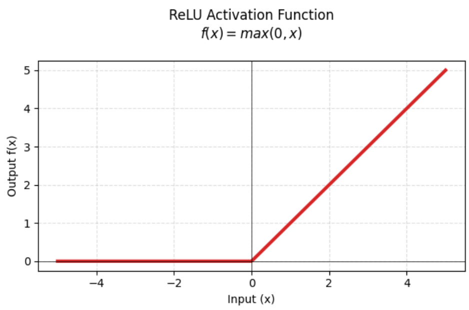


Figure 1.6: ReLU activation function plot

ReLU is very popular because it is simple and helps the network learn faster. It avoids the vanishing gradient problem for positive values. However, sometimes neurons can "die" and only output zero if the input is always negative. This is called the "dying ReLU" problem.

Softmax Activation Function: The softmax activation function is used when there are many possible classes. It turns numbers into probabilities that add up to 1:

$$f(x_i) = \frac{e^{x_i}}{\sum_j e^{x_j}}$$

where x_i is the input for class i, and the sum is over all classes. It is mainly used for multi-class classification tasks.

Choosing the right activation function is very important. For binary classification, the **sigmoid** function is often used. For multi-class classification, **softmax** is the best choice. In regression tasks, the **linear activation** is used. In deep networks, **ReLU** is usually preferred because it makes training faster and easier.

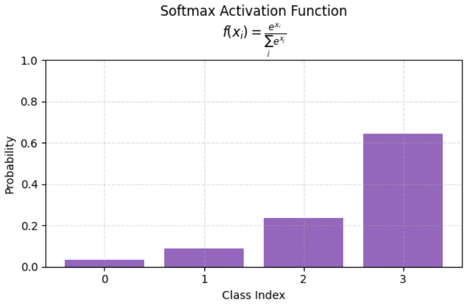


Figure 1.7: Softmax activation function plot (example with 4 classes)

1.3.2 Convolutional Neural Networks

A Convolutional Neural Network (CNNs) is composed of multiple layers that sequentially process the input EEG data to extract relevant features and classify brain activity patterns. The model consists of three main types of layers: convolutional layers, pooling layers, and fully connected layers, each serving a specific role in the feature extraction and classification process[4].
The convolutional layer is the fundamental component of the CNN, designed to automatically detect spatial and temporal dependencies in the EEG signal. It applies a set of filters (kernels) that slide over the input data, computing feature maps by performing a convolution operation at each step[3]. This operation can be expressed mathematically as:

$$y[i] = \sum_j X[i+j]W[j]$$

where X represents the input signal, W denotes the filter weights, and $y[i]$ is the resulting feature map.

In our implementation, we used 1D convolution with a 1×3 kernel, which is well-suited for analyzing time-series EEG data as it captures localized patterns within the signal[3].

To reduce the dimensionality of the feature maps while preserving essential information, a pooling layer is applied after the convolutional operation. We used max pooling, which selects the maximum value from each segment of the feature map, thereby highlighting the most dominant characteristics while reducing computational complexity. The max pooling operation is defined as:

$$y_i = \max(x_{i-1}, x_i, x_{i+1})$$

This step helps the network become more robust to small variations in the input signal while ensuring efficient processing.

After passing through multiple convolutional and pooling layers, the extracted features are flattened into a one-dimensional vector and fed into fully connected layers. These layers act as classifiers by transforming the learned feature representations into final predictions. The last layer employs an activation function, such as sigmoid for binary classification or softmax for multi-class classification, to assign probabilities to different classes[4].

Since EEG data is inherently a time series, our CNN architecture was designed to capture temporal dependencies effectively. The combination of convolutional and pooling operations ensures that relevant features are extracted automatically without requiring manual preprocessing [4]. As shown in Figure 2.16, the overall architecture consists of stacked convolutional layers followed by Max pooling layers, culminating in a fully connected output layer responsible for classification.

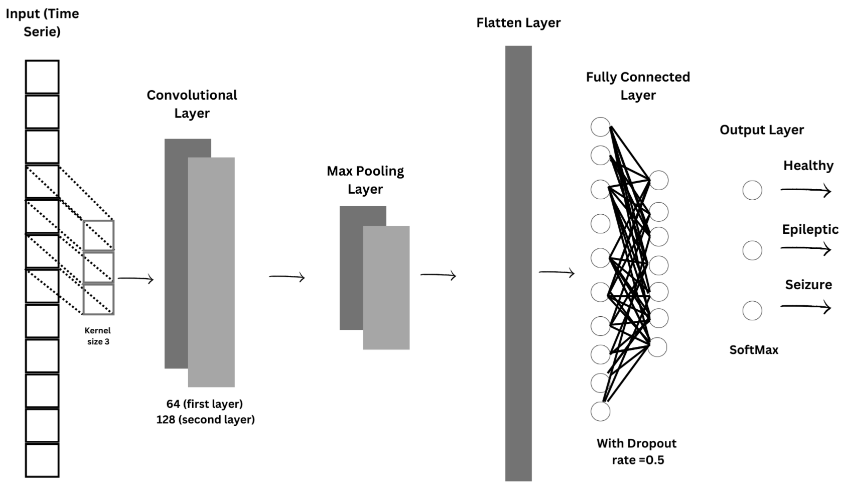


Figure 1.8: Proposed CNN architecture.

1.3.3 CNN-LSTM Hybrid Model

Recurrent Neural Networks (RNN) and Their Limitations

Recurrent Neural Networks (RNNs) are a class of neural networks specifically designed for sequential data processing. Unlike traditional feedforward neural networks, RNNs introduce a recurrent connection that enables the network to maintain a form of memory by passing information from one step to the next in a sequence. However, standard RNNs face significant limitations, primarily due to the vanishing and exploding gradient problem. When back propagating through long sequences, the gradients of earlier time steps tend to shrink exponentially (vanish) or grow excessively (explode), making it difficult to learn long-range dependencies. As a result, RNNs struggle to retain information over extended sequences, leading to performance degradation in tasks that require long-term memory retention[15].

Long Short-Term Memory (LSTM) Networks

Long Short-Term Memory (LSTM) networks are a type of Recurrent Neural Network (RNN) specifically designed to capture long-term dependencies in sequential data[2]. LSTM architectures contain memory cells, which are controlled by three primary gates: the input gate, the forget gate, and the output gate. These gates regulate the flow of information into, out of, and within the memory cells, allowing the network to retain or discard information as needed, as shown in Figure 1.9.

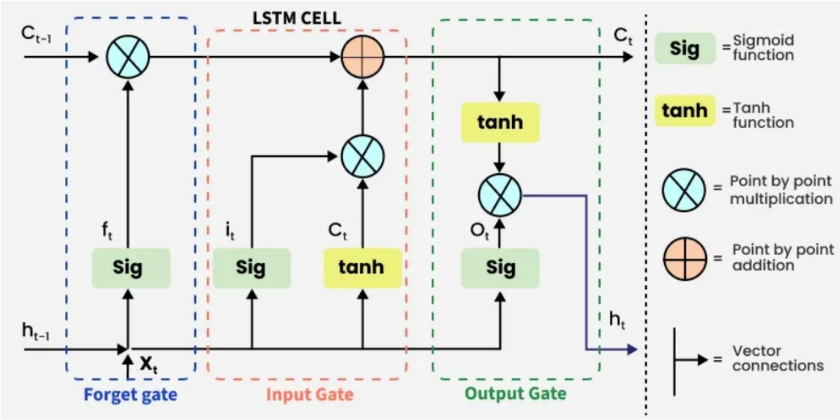


Figure 1.9: Typical LSTM structure [2].

Input Gate The input gate controls the addition of new information to the memory cell. It processes the current input x_t and the previous hidden state h_{t-1} by applying a sigmoid activation function, which produces values between 0 and 1. This result, denoted by i_t, determines how much of the new candidate information should be retained. Additionally, a new candidate cell state $\hat{C}_t$ is created by applying a hyperbolic tangent (tanh) activation function to a combination of x_t and h_{t-1}. The candidate $\hat{C}_t$ proposes potential new content to add to the cell state. The combination of i_t and $\hat{C}_t$ controls what new information will be integrated into the memory cell. Mathematically, the input gate is defined as:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$
$$\hat{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c)$$

where σ denotes the sigmoid activation function, and tanh is the hyperbolic tangent function. W_i and W_c are weight matrices, and b_i and b_c are bias terms.

Forget Gate The forget gate determines what information should be removed from the memory cell. It takes the current input x_t and the previous hidden state h_{t-1}, applying a sigmoid function to generate a forget factor f_t between 0 and 1. A value close to 0 indicates that the information should be discarded, while a value close to 1 means it should be retained. The forget gate allows the network to adapt by removing irrelevant or outdated information from the cell state. The equation for the forget gate is:

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

where W_f is the weight matrix and b_f is the bias for the forget gate.

Output Gate The output gate regulates what information is output from the memory cell to the next hidden state. It processes the current input x_t and the previous hidden state h_{t-1} using a sigmoid function to compute o_t. Meanwhile, the updated cell state C_t is passed through a tanh activation to squish its values between -1 and 1. Finally, the output hidden state h_t is obtained by element-wise multiplication of o_t and $\tanh(C_t)$. The output gate equations are:

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

where W_o is the weight matrix and b_o is the bias for the output gate, and $\odot$ represents element-wise multiplication.

Working of LSTM The LSTM network consists of a chain of repeating modules, each containing a memory cell regulated by the input, forget, and output gates. The memory cell is updated at each time step through a combination of these gates, enabling the network to capture and manage long-term dependencies while mitigating the issues of vanishing or exploding gradients commonly found in traditional RNNs.

CNN-LSTM Hybrid Model for EEG Signal Analysis

A CNN-LSTM hybrid model combines the strengths of Convolutional Neural Networks (CNNs) for spatial feature extraction and Long Short-Term Memory (LSTM) networks for sequential dependency learning. This hybrid approach enables effective extraction of both spatial and temporal features, leading to superior performance in tasks like seizure detection.

Feature Extraction with CNNs
By applying convolutional filters to the EEG signal, CNNs can capture local dependencies and reduce noise, allowing them to extract essential features from the raw signal. These spatial features are crucial as EEG data often contains high-frequency noise, which CNNs can filter out while retaining the meaningful information.

Temporal Pattern Recognition with LSTMs
Once spatial features have been extracted by the CNN layers, the model transitions to LSTM layers, which excel at learning sequential patterns in data. EEG signals are temporal in nature, with sequential dependencies across time. LSTMs are specifically designed to capture long-term temporal dependencies, making them ideal for analyzing time-series data like EEG signals. The integration of LSTM layers allows the model to better understand the progression of signals over time, thus improving classification accuracy[7][4].

Detailed Architecture of the CNN-LSTM Model

The CNN-LSTM hybrid model is composed of the following key components:

- **Convolutional Layers (CNN Block):** The model begins with multiple 1D convolutional layers that apply various filters to the input EEG signal to extract relevant spatial features. Each convolutional layer is followed by an activation function like ReLU (Rectified Linear Unit), which introduces non-linearity and enhances the model's ability to learn complex relationships.

- **Pooling Layers:** Following the convolutional layers, max-pooling or average-pooling layers are used to reduce the dimensionality of the feature maps. Pooling helps in retaining the most significant spatial features while discarding less important information.

- **LSTM Layers:** After spatial feature extraction, the model feeds the data into LSTM layers. These layers help the model understand how the signal evolves over time, ensuring that both short-term and long-term dependencies are captured.

- **Dropout Layers:** Dropout layers are added during training to prevent overfitting. These layers randomly deactivate a fraction of the neurons to reduce the risk of the model becoming too reliant on specific features during training.

- **Fully Connected Layer Softmax Activation:** After the LSTM layers, the model passes the learned feature representations through a fully connected (dense) layer, which helps in transforming the features into a format suitable for classification. For binary classification, a sigmoid activation function is used, while for multi-class classification, the softmax activation function is employed.

Conclusion

In summary, this chapter provided the essential background needed for the study. It described epilepsy and the importance of EEG signals in seizure detection, outlined key preprocessing techniques for EEG data, and introduced deep learning models that will be applied later. This foundation is crucial for understanding the methods and results presented in the following chapters.

Chapter 2

State of the Art and Proposed Work

This chapter provides a comprehensive overview of existing research on epileptic seizure detection and classification using EEG signals. It begins with a review of previously proposed models, particularly those leveraging deep learning techniques. After identifying strengths and limitations in the current literature, we present the motivation behind our proposed methodology. The latter sections introduce the core components of our approach, including data preprocessing, deep learning architectures, classification strategies, and evaluation metrics.

2.1 Previous Studies on Epileptic Seizure Classification

This section presents a detailed review of the literature focused on the automatic detection and classification of epileptic seizures using EEG signals. It emphasizes both traditional and deep learning-based approaches, comparing their methodologies and performances in clinical and experimental settings[16] [17] [18].

2.1.1 Existing Models and Approaches

Various models have been proposed for the classification of epileptic seizures from EEG signals, each with unique methodologies for preprocessing, feature extraction, and classification. Traditional methods often rely on handcrafted features, but recent advances have increasingly shifted towards deep learning-based models. These models, particularly convolutional neural networks (CNNs), have shown significant improvements in classification accuracy by automating the feature extraction process. Additionally, preprocessing techniques such as wavelet transforms and filters are widely used to enhance the quality of EEG signals before they are passed into classifiers.

In particular, three prominent approaches include kernel-based classifiers, hybrid CNN models, and deep learning-based CNNs with discrete wavelet transform preprocessing. These methods have gained attention for their ability to achieve high accuracy in seizure detection tasks while reducing the need for manual feature extraction.

2.1.2 Comparison of Three Studies on Seizure Detection Using the Bonn EEG Dataset

Study 1: R-ProCRC with Wavelet Transform Preprocessing

The first study [16] focuses on a kernel-based robust probabilistic collaborative representation-based classifier (R-ProCRC) combined with wavelet transform preprocessing. This method achieved an impressive 99.3% accuracy on the Bonn dataset. The approach highlights the utility of wavelet transforms for extracting key features from EEG signals, enabling the R-ProCRC classifier to outperform traditional models in terms of accuracy and robustness. Sensitivity and specificity were both above 96%, with certain patients showing near-perfect results (100%).

Study 2: Hybrid CNN with Machine Learning Classifiers

The second study [17] employs a hybrid approach that combines convolutional neural networks (CNNs) for feature extraction with machine learning classifiers for further classification. In this model, EEG signals are preprocessed using a Butterworth filter, followed by automatic feature extraction via CNNs. The extracted features are then selected based on their mutual information gain, and various classifiers (including k-NN and SVM) are used for final classification. The model demonstrated excellent performance on the Bonn dataset, achieving 100% accuracy for two classes, 99% for three classes, 94.6% for four classes, and 94% for five classes.

Study 3: CNN with Discrete Wavelet Transform Preprocessing

The third study [18] applies convolutional neural networks (CNNs) with discrete wavelet transform (DWT) preprocessing. DWT is used to decompose EEG signals into sub-bands, enhancing the feature extraction process. This method achieved 99.5% accuracy for the two-class problem on the Bonn dataset, and 97% on the CHB-MIT dataset. The study emphasized the power of combining CNNs with wavelet transform for automatic feature extraction and classification, enabling the model to handle complex and multichannel EEG datasets.

Study 4: 1D CNN with Data Augmentation for Robust Classification

The fourth study [19] investigates a one-dimensional CNN model trained using data augmentation techniques to enhance generalization. EEG signals from the Bonn dataset were segmented into overlapping windows, and additional noise was introduced to diversify the training set. This approach reduced overfitting and improved performance. The 1D CNN achieved 98.6% accuracy in two-class classification, with both sensitivity and specificity above 95%. The study highlights the importance of data augmentation, especially when working with smaller datasets, to build robust seizure detection models[19].

2.1.3 Key Insights from the Comparison

Preprocessing Techniques

Each study employed different preprocessing techniques. The first study relied on wavelet transforms to extract features from EEG signals. The second study used a

Butterworth filter for signal preprocessing, followed by feature extraction using CNNs. The third study combined discrete wavelet transform (DWT) with CNNs for more detailed signal decomposition and feature extraction. The fourth study introduced signal segmentation and data augmentation, applying sliding windows to create overlapping segments and injecting random noise into training samples. These techniques demonstrate the importance of preprocessing in enhancing signal quality and feature extraction, which is crucial for achieving high classification performance.

Feature Extraction and Selection

The second and third studies relied heavily on CNNs for feature extraction, automating the process and reducing the complexity associated with manual feature selection. The first study, on the other hand, used wavelet transforms for feature extraction. Additionally, the second and third studies incorporated feature selection methods based on mutual information and other techniques to ensure that only the most relevant features were used for classification, improving model efficiency. The fourth study did not focus explicitly on feature selection, but its use of data augmentation and segmentation increased feature diversity, supporting more robust model training.

Classification Performance

All four studies showed excellent classification performance. The first study achieved 99.3% accuracy for two classes on the Bonn dataset. The second study demonstrated 100% accuracy for two classes and achieved solid results for multiclass classification, maintaining 94% accuracy for five classes. The third study achieved 99.5% accuracy for the two-class problem and 97% for the CHB-MIT dataset, indicating that its model generalizes well across different datasets. The fourth study reported 98.6% accuracy, with strong sensitivity and specificity, highlighting its reliability even with limited training data. These results collectively underscore the effectiveness of different methods for seizure detection.

Multiclass Classification

The second and third studies placed a particular emphasis on multiclass classification. The second study showed that, while accuracy dropped slightly as the number of classes increased, it still maintained high performance, with up to five classes. The third study demonstrated the ability of the model to handle multiple classes efficiently, achieving high accuracy even with multisubject and multichannel datasets. The fourth study, while focused on binary classification, could potentially be adapted to multiclass experiments through the use of its data augmentation and segmentation techniques.

Evaluation Metrics

While all four studies primarily reported accuracy, sensitivity, and specificity, the second and third studies expanded the evaluation by considering additional metrics such as precision, recall, and F1-score. The second study also utilized bagging with k-NN classifiers to improve precision and recall, ensuring a more balanced trade-off between false positives and false negatives, especially in multiclass experiments. The fourth study highlighted generalization through data augmentation, which helped maintain high performance even in the presence of noise and variability, thus making it more robust in practical use cases.

2.1.4 Challenges and limitations

Several challenges and limitations were identified across the reviewed studies, impacting the performance and applicability of the proposed seizure detection methods.

Study 1 :

The proposed method faces several challenges and limitations. One issue is the small amount of seizure data in some of the databases, which can make it hard to train the model for detecting all types of seizures effectively. Another challenge is the impact of technical problems, like electrode disconnection, which affected the accuracy of the results, especially for one patient. Additionally, while the wavelet transform helps in processing EEG signals, it can filter out short seizures, causing them to be misclassified as non-seizures. There were also some false detections, mainly caused by high-amplitude activities being mistaken for seizures. Although the method improved detection accuracy, it may face difficulties in real-time systems due to the use of linear operators. The method showed good performance on different databases, but it may still have limitations when applied in real clinical environments. Seizure detection remains a complicated task because EEG signals are irregular and non-linear, and they often contain seizure-like activities throughout the recording. Lastly, the preprocessing steps, like wavelet filtering, add complexity to the process, and if not done correctly, they can affect the performance of seizure detection.[16].

Study 2 :

The model faces several limitations that hinder its performance. One significant limitation is the decrease in classification accuracy as the number of classes increases, particularly for multiclass problems, where the accuracy drops below 90%. Although a 2-layered CNN architecture works well for binary classification, increasing the number of layers to 4 or 6 does not provide significant improvements and only adds complexity, increasing the computational cost. Additionally, the model's reliance on feature selection presents challenges, as redundant features can cause longer training times, higher power consumption, and a greater risk of overfitting, especially in mobile healthcare applications. Another limitation is the model's tendency to overfit despite using CNN for automatic feature extraction and 10-fold cross-validation. This issue, coupled with long training times, highlights the difficulty in tuning hyperparameters, which remains a time-consuming process. Furthermore, the model's performance is heavily dependent on the quality and quantity of training data, limiting its ability to generalize effectively across various datasets. These limitations suggest that while the CNN architecture outperforms traditional machine learning methods, it still requires optimization and further development to handle different types of time-series signals and datasets[17].

Study 3 :

The study on the combined CNN and LSTM method for epileptic EEG recognition highlights several limitations. Firstly, the model relies on pre-processed EEG data, and the quality of the data, such as noise levels and signal integrity, can significantly affect performance. Despite improvements in feature extraction by CNN, low signal-to-noise ratios remain a challenge, impacting the recognition of epileptic events. Additionally, EEG signals captured through multi-channel electrodes may not always fully exploit spatial information, leading to lower classification accuracy. Deep learning models like CNN-LSTM are also prone to overfitting, especially with limited or non-diverse training data, raising concerns about generalization to unseen data. Another limitation is the lack of interpretability in deep learning models, making it difficult to understand the reasoning behind specific predictions, which is crucial in medical applications. The integration of this model into clinical practice could be hindered by the need for large, annotated datasets and the complexity of real-time processing for immediate diagnosis. Furthermore, the model's ability to generalize across different patient populations and seizure types remains uncertain. The scalability of the model with increasing medical data could also pose challenges, requiring significant computational resources and time. Finally, while the study suggests future work on mobile applications for epilepsy detection, integrating such systems with real-time monitoring devices presents additional technical and logistical challenges[18].

Study 4 :

This study faced several challenges that impacted the accuracy of seizure classification models. One key issue was the limited amount of data for certain seizure types, such as TNSZ (Tonic Seizure), SPSZ (Simple Partial Seizure), FNSZ (Focal Non-Motor Seizure), and CPSZ (Complex Partial Seizure). For example, TNSZ and SPSZ each had only three patient records, which made it difficult for the model to accurately classify these seizures. Misclassification was also common between FNSZ and CPSZ, especially in the Network 1D Ra model, where CPSZ was often misidentified as FNSZ. The conversion of EEG data into spectrograms led to the loss of important information, negatively affecting the classification of ABSZ (Absence Seizure). The small sample size for TCSZ (Tonic-Clonic Seizure) resulted in high variability and poor performance. Additionally, the lack of powerful GPUs and the reliance on personal laptops slowed down the training process, while using only 5-second EEG recordings limited the model's effectiveness. Longer recordings might improve classification accuracy. The absence of a non-seizure class in the dataset made it harder to distinguish between seizure types. There is also a need for further exploration of hyperparameters and network configurations. Tuning these hyperparameters and experimenting with different network setups could help improve the model's performance. These adjustments may provide more accurate results and help the model better handle different seizure types.[19].

2.2 Proposed Methodology

While many recent studies have explored various techniques for the classification of epileptic EEG signals, our work does not seek to introduce a fundamentally novel approach. Instead, the focus is on performing a comprehensive comparative analysis of three well-established deep learning architectures, exploring their individual and combined performance in both binary and multi-class classification of EEG data. The deep learning models selected for this study are:

- Artificial Neural Network (ANN)
- Convolutional Neural Network (CNN)
- Hybrid CNN–LSTM Model

These models were trained on EEG signals derived from the well-known Bonn dataset. The analysis was carried out under different preprocessing conditions, across multiple classification experiments, and using varying training configurations. The ultimate goal was to evaluate the performance of each model under realistic and challenging conditions, and to understand how different input representations and preprocessing techniques influence their effectiveness.

Figure 2.1 below illustrates the overall methodology followed in this study, including data preprocessing, model design, and evaluation phases.

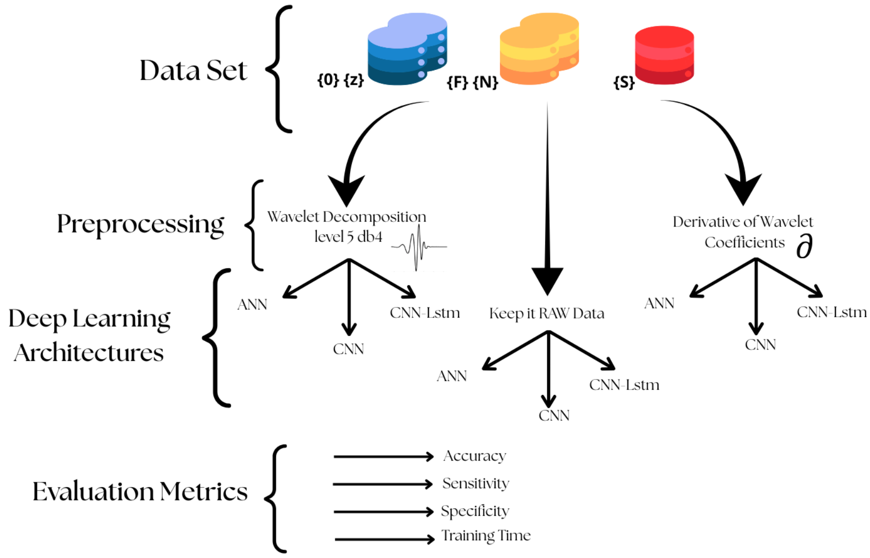


Figure 2.1: Proposed methodology for EEG signal classification using deep learning models.

2.2.1 Dataset Description

For this study, we utilized the publicly available EEG dataset provided by the University of Bonn, Germany[1], which has become a standard benchmark in the field of epilepsy detection. The dataset comprises recordings from 25 individuals, including both healthy subjects and patients with pharmacoresistant partial epilepsy. EEG signals were acquired noninvasively and categorized into five subsets: **F, N, O, Z, and S**, each containing 100 single-channel EEG segments of 23.6 seconds duration.

The five subsets differ in terms of physiological and pathological conditions:

- Subsets **O** and **Z** represent EEG data from healthy individuals, recorded with eyes open and closed, respectively.

[1] `http://epileptologie-bonn.de/cms/upload/workgroup/lehnertz/eegdata.html`

- Subsets **F** and **N** were obtained from epileptic patients during seizure-free intervals, using intracranial electrodes placed in different regions of the brain.
- Subset **S** corresponds to EEG data collected during active seizure episodes.

For the purpose of classification, we grouped the subsets into three distinct classes:

- **Healthy class:** combining subsets **Z** and **O**,
- **Interictal (Epileptic) class:** combining subsets **F** and **N**,
- **Ictal (Seizure) class:** represented solely by subset **S**.

This tripartite categorization allows us to explore multi-class classification experiments reflecting various stages of epileptic brain activity. A detailed summary of the dataset structure and class distribution is presented in Table 2.1.

Set	Condition	Foci	Electrodes	Area	Samples	Parameters
O	Healthy	Scalp	Surface	All areas	100	173.61 Hz filter
Z	Healthy	Scalp	Surface	All areas	100	–
F	Intermittent epilepsy	Intracranial	Depth	Lesion outside	100	0.53–40 Hz (12 dB/oct)
N	Intermittent epilepsy	Intracranial	Depth	Lesion outside	100	–
S	Ictal epilepsy	Intracranial	Depth	Lesion inside	100	Segment 23.6 s

Table 2.1: Summary of the Bonn University EEG dataset (4096 data points per sample).

Example EEG signals from the different subsets of the dataset are shown in Figure 2.7, representing various conditions: healthy (subsets O and Z), epileptic (subsets F and N), and seizure (subset S).

Figure 2.2: EEG Signal from Folder {O} (Healthy Subject)

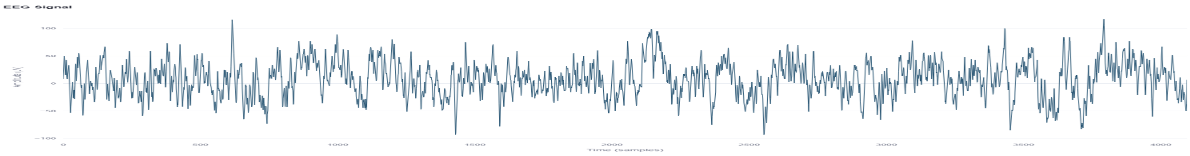


Figure 2.3: EEG Signal from Folder {Z} (Healthy Subject)

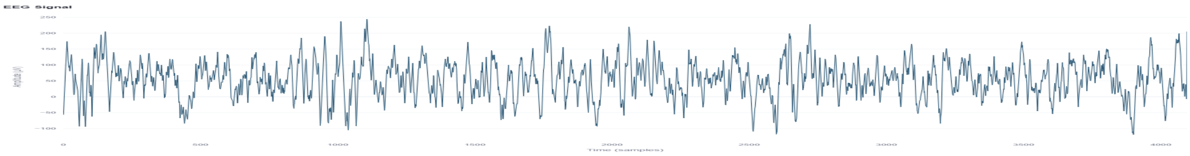


Figure 2.4: EEG Signal from Folder {F} (Epileptic)

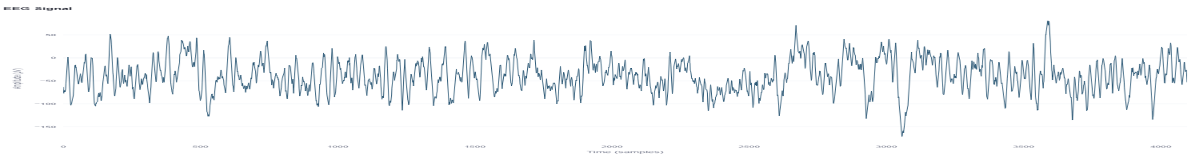


Figure 2.5: EEG Signal from Folder {N} (Epileptic)

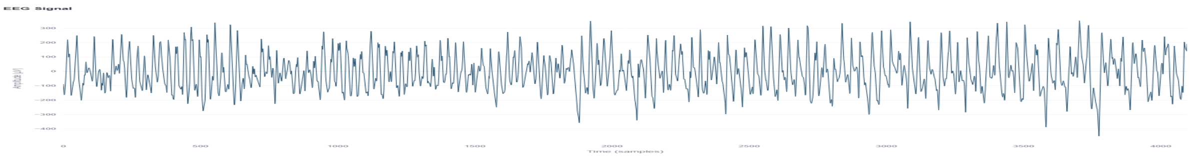


Figure 2.6: EEG Signal from Folder {S} (Seizure)

Figure 2.7: Example EEG signals from different conditions: {O} and {Z} (healthy subjects), {F} and {N} (interictal epileptic activity), and {S} (ictal seizure activity).

2.2.2 Input Representations and Preprocessing

A crucial step in any EEG classification task lies in how the raw signal is represented and preprocessed before being fed into a model. In our study, we considered three distinct types of input representations:

- **Raw EEG Signals:** This approach involved using the EEG signals in their raw time-domain form, without applying any transformation or feature extraction. This representation retains the complete original structure of the signal, allowing the models—especially deep architectures like CNN and LSTM—to learn discriminative features directly from the waveform. While raw data offers the highest signal fidelity, it may also contain noise and redundant information that could affect learning performance.

- **Wavelet Decomposition:** The second input representation was based on the Discrete Wavelet Transform (DWT). We employed the Daubechies 4 (db4) mother wavelet with a level-5 decomposition, a configuration frequently used in EEG analysis due to its ability to capture both high-frequency components (useful for detecting seizure spikes) and low-frequency background activity. Wavelet decomposition breaks the EEG signal into approximation and detail coefficients, representing different frequency bands localized in time. This multi-resolution analysis allows models to capture both transient changes and long-term trends in the EEG signal, which are essential for distinguishing between healthy and epileptic brain activity.

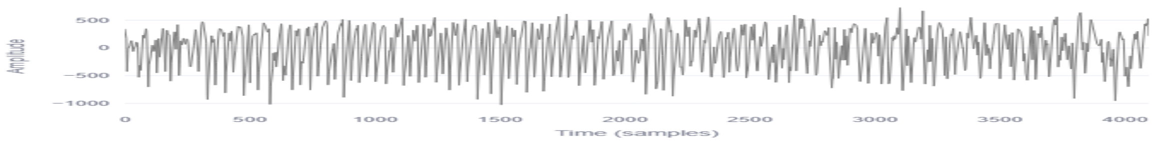


Figure 2.8: Original EEG signal

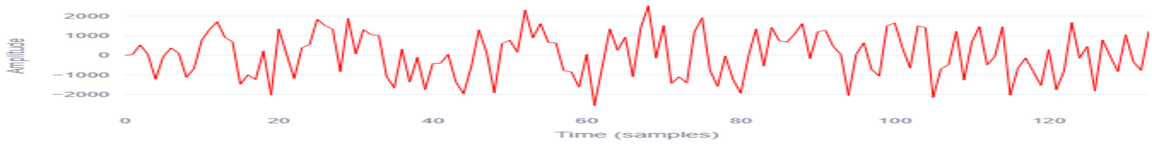


Figure 2.9: Level 1 detail coefficients

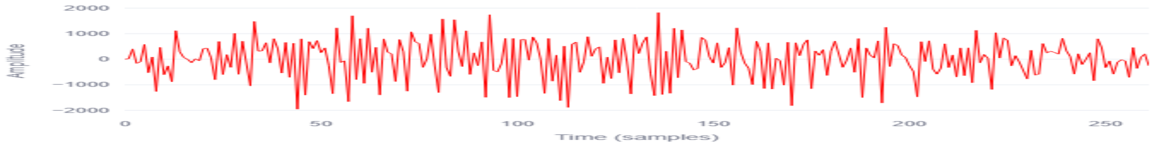


Figure 2.10: Level 2 detail coefficients

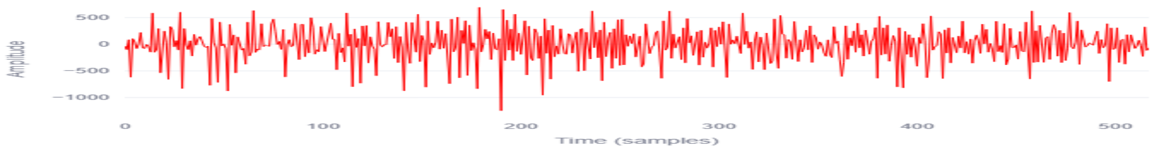


Figure 2.11: Level 3 detail coefficients

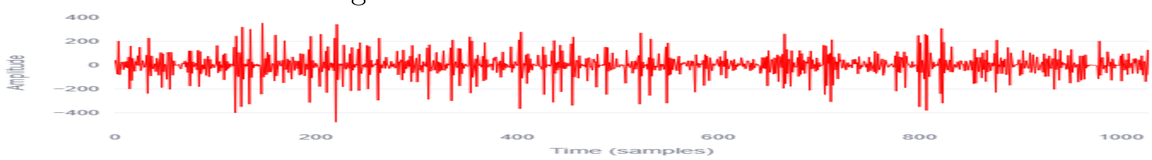


Figure 2.12: Level 4 detail coefficients

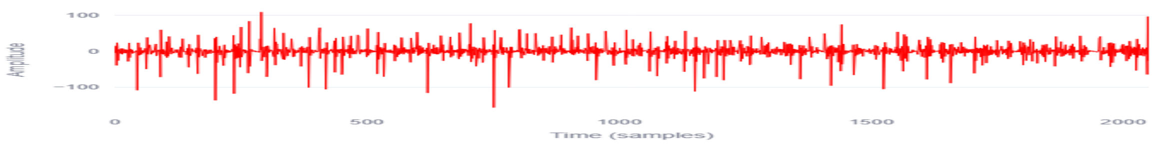


Figure 2.13: Level 5 detail coefficients

Figure 2.14: Wavelet decomposition of an EEG signal using Daubechies 4 (db4) up to level 5. Each level captures different frequency components important for seizure analysis.

Wavelet decomposition breaks the EEG signal into approximation and detail coefficients, representing different frequency bands localized in time. This multi-resolution analysis allows models to capture both transient changes and long-term trends in the EEG signal, which are essential for distinguishing between healthy and epileptic brain activity.

- **Derivative of Wavelet Coefficients:** In addition to the raw and wavelet-transformed signals, we introduced a third variant based on the temporal derivative of the wavelet coefficients. This approach highlights rapid changes and transition points in the signal, which are often linked to seizure events. By computing the first-order derivative of the wavelet coefficients, we aimed to amplify subtle differences between signal classes and help the models learn transition dynamics more effectively.

  To further demonstrate the effectiveness of the derivative transformation, Figure 2.9 presents a composite image that includes the original EEG signal alongside its first, second, and third-order temporal derivatives of the wavelet coefficients.

Figure 2.15: Composite visualization of an EEG signal and its wavelet-based temporal derivatives. From top to bottom: (a) Original EEG signal, (b) First-order derivative, (c) Second-order derivative, and (d) Third-order derivative. Each level reveals different temporal dynamics, useful for modeling transitions.

These three input types were used independently to train each of the three deep learning models. By doing so, we aimed to determine which preprocessing pipeline provided the most useful information for classification under various conditions. The inclusion of both raw and transformed data allows for a fair and insightful comparison between signal-level and feature-level learning approaches.

2.2.3 Deep Learning Architectures

Artificial Neural Network (ANN):

The ANN used in this study consists of several dense (fully connected) layers with

ReLU activation functions. The input layer corresponds to the length of the EEG signal segment, while the output layer varies based on the number of classes in each classification experiment. ANN serves as a baseline model to understand how non-convolutional, feedforward networks handle EEG signal data.

Convolutional Neural Network (CNN):

The CNN model developed in this work consists of two convolutional layers with 64 and 128 filters, respectively, each using a kernel size of 3 and the ReLU activation function. Each convolutional layer is followed by a max pooling layer with a pool size of 2 to reduce the feature map dimensions. After feature extraction, a flatten layer is applied, followed by a dense layer with 128 units and ReLU activation. To prevent overfitting, a dropout layer with a rate of 0.5 is included. For the output, a dense layer is used: in the case of binary classification, a sigmoid activation function is applied, whereas for multiclass classification, a softmax activation function is used.

The figure 2.16 below presents our proposed CNN architecture for EEG classification:

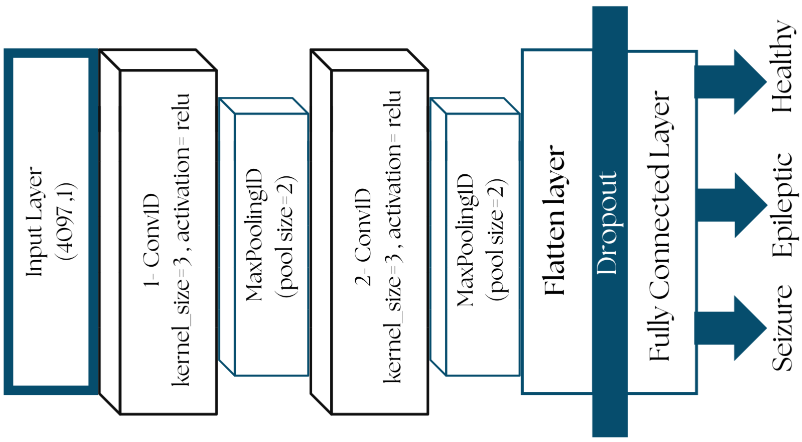


Figure 2.16: Proposed CNN Architecture for EEG Classification.

CNN-LSTM Hybrid Model:

The hybrid CNN-LSTM model combines the strengths of both spatial and temporal feature extraction. The architecture consists of two 1D convolutional layers: the first uses 64 filters and the second 128 filters, both with a kernel size of 3 and ReLU activation, to extract spatial features from EEG signals. Each convolutional layer is followed by a max pooling layer with a pool size of 2 to reduce dimensionality. The output is then passed to an LSTM layer with 128 units and ReLU activation to capture long-term temporal dependencies. A dropout layer with a rate of 0.5 is added to reduce overfitting. Finally, a fully connected dense layer with a softmax activation function is used for classification whether binary or multiclass nby adjusting the number of output neurons accordingly. This hybrid approach enables the model to extract meaningful spatial features while also modeling their temporal evolution, which is essential for detecting epileptic seizure patterns in EEG recordings , as shown in Figure 2.17.

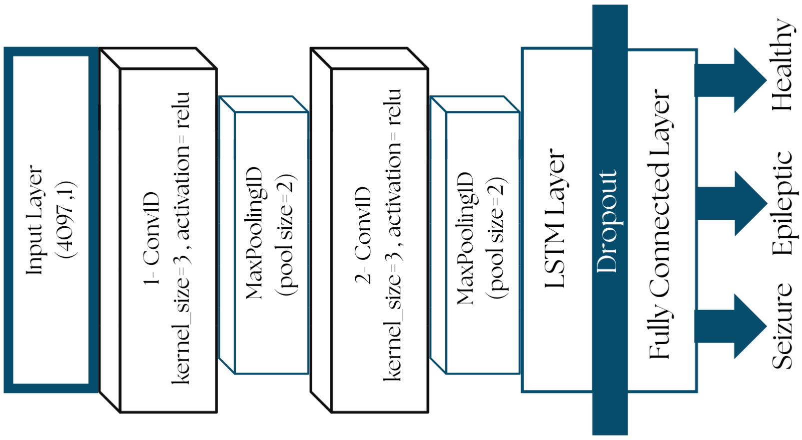


Figure 2.17: Proposed architecture of CNN–LSTM model for EEG signal classification

2.2.4 Classification Experiments

To thoroughly evaluate the models, we designed six classification experiments involving binary and multi-class setups:

1. Healthy vs Seizure
2. Epileptic vs Seizure
3. Healthy vs Epileptic
4. Healthy vs (Epileptic and Seizure)
5. (Healthy and Epileptic) vs Seizure
6. Healthy vs Epileptic vs Seizure

These experiments simulate real-world diagnostic challenges, such as distinguishing between healthy subjects, patients with epileptic brain activity without seizures, and those actively experiencing seizures.

2.2.5 Configuration Modes

To assess the impact of training optimizations, we conducted experiments under four different configurations for each model and each classification experiment:

- **No Scaler / No Optimizer**
- **Scaler Only:** Standardization or normalization of the input data.
- **Optimizer Only:** Using optimization algorithms such as Adam or SGD.
- **Scaler and Optimizer:** Combination of both scaling and optimization.

This systematic evaluation helps isolate the effects of preprocessing and training strategies on model performance.

2.2.6 Evaluation Metrics

To objectively compare the results, four performance metrics were computed for each experiment[7]:

- **Accuracy:** Measures the proportion of correctly classified instances.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

- **Sensitivity (Recall):** Measures the proportion of actual positives correctly identified.

$$\text{Sensitivity} = \frac{TP}{TP + FN}$$

- **Specificity:** Measures the proportion of actual negatives correctly identified.

$$\text{Specificity} = \frac{TN}{TN + FP}$$

- **Training Time:** The total duration required to train the model under each configuration.

Here, TP (True Positives), TN (True Negatives), FP (False Positives), and FN (False Negatives) represent the values from the confusion matrix.

Conclusion

In this chapter, we explored different methods used for epileptic seizure detection with EEG signals. After reviewing several approaches, we concluded that deep learning models, especially CNN and hybrid CNN–LSTM, show strong potential for improving detection accuracy. Based on these findings, we decided to compare ANN, CNN, and CNN–LSTM models using different input types to identify the most effective solution. The next chapter will focus on the results of these models, providing an analysis of their performance, comparing different training configurations, and discussing their strengths and limitations.

Chapter 3

Results and Discussion

This chapter presents and analyzes the results of our models for epileptic seizure detection using EEG signals. We focus on comparing the performance of the ANN, CNN, and CNN-LSTM models. The chapter begins with a detailed explanation of the training methodology and model configurations, followed by an in-depth performance analysis. We will also explore how different feature transformation methods impact the accuracy of the models. Lastly, the results are discussed to highlight the strengths and limitations of each approach.

3.1 Training Methodology and Hyperparameter Settings

In this section, we describe in detail how each model is trained from data preparation through stopping criteria and explain the rationale behind every hyperparameter choice. By applying a consistent protocol across the fully-connected ANN, the convolutional network, and the convolutional-LSTM hybrid, we ensure that differences in performance reflect architectural strengths rather than mismatched training regimes.

3.1.1 Data Preparation and Classification Experiments

Continuous EEG recordings are divided into non-overlapping windows of 4 097 samples, capturing sufficient temporal context for seizure patterns. We conduct experiments on six tasks:

1. Five binary classification experiments (e.g., healthy vs. seizure under different experimental conditions).

2. One multi-class experiment .

For binary tasks, outputs use either a single probability (sigmoid head) or two-class softmax; for the multi-class task, a three-class softmax head is used.

3.1.2 Common Training Protocol

- **Mini-Batch Size:** 32 samples per update, balancing gradient stability and computational efficiency.

- **Epoch Limits:** Up to 50 epochs for binary tasks (fully connected) or 30 epochs for convolutional-based models; up to 100 epochs for the multi-class fully connected model.
- **Early Stopping:** Monitors validation loss, with patience set to 8–10 epochs depending on the model; best weights are restored when training halts.
- **Regularization:** Dropout (50%) applied in dense heads of convolutional and hybrid models; L_2 weight decay ($\lambda = 1 \times 10^{-4}$) on all convolutional kernels.
- **Activations:** ReLU in all hidden layers and recurrent units; sigmoid or softmax in output layers according to task.
- **Loss Functions:** Binary cross-entropy for single-output heads; categorical cross-entropy for softmax heads.

3.1.3 Fully-Connected ANN Configuration

Each EEG window is fed as a 4,097 dimensional vector through three dense layers ($512 \rightarrow 256 \rightarrow 128$ units) with ReLU activations, terminating in a softmax output layer (2 or 3 units). Training minimizes categorical cross-entropy using mini-batches of 32. Early stopping halts training when validation loss fails to improve for 10 epochs; convergence typically occurs between 20 and 60 epochs.

3.1.4 Convolutional Network Configuration

Raw windows reshaped to ($4{,}097 \times 1$) pass through two Conv1D–ReLU blocks (64 then 128 filters, kernel size 3), each followed by max-pooling (pool size 2). A flattened 128-unit dense layer (ReLU + 50% dropout) precedes a single sigmoid neuron for binary tasks or a three-unit softmax for multi-class. Models train for up to 30 epochs with early stopping (patience = 8) and L_2 decay on convolutional layers.

3.1.5 Convolutional–LSTM Hybrid Configuration

After the same two convolutional blocks, the pooled sequence enters an LSTM layer with 64 ReLU-activated units, followed by 50% dropout. The final head mirrors the convolutional network's output. Training uses the same batch size and epoch cap, with early stopping (patience = 8) and a record of wall-clock training time for each run.

3.1.6 Hyperparameter Summary

The following table summarizes the key hyperparameters used in training the ANN, CNN, and CNN-LSTM models. Each model follows consistent training protocols to ensure fair performance comparisons.

Parameter	ANN	CNN	CNN-LSTM
Batch Size	32	32	32
Epochs	50 / 100	30	30
Loss Function	Sigmoid + BCE / Softmax + CCE	Sigmoid + BCE / Softmax + CCE	Sigmoid + BCE / Softmax + CCE
Pooling	–	MaxPooling1D (pool size = 2)	MaxPooling1D (pool size = 2)
Flatten	–	Flatten	–
LSTM Layer	–	–	LSTM (64 units)

Table 3.1: Summary of model hyperparameters for ANN, CNN, and CNN-LSTM architectures.

By maintaining these consistent training protocols and regularization techniques, we ensure that subsequent performance comparisons reflect each architecture's original capabilities under fair conditions.

3.2 Performance of Our Methodology

3.2.1 Impact of Training Configurations

Across the different experiments carried out in this work, a consistent trend was observed concerning the effect of various training configurations on model performance. As described in the methodology section, each architecture was evaluated under different conditions specifically through combinations involving data scaling and the choice of optimizer.

Figure 3.3 summarizes these differences by comparing four setups: training with neither scaling nor optimizer adjustments, with only scaling, with only an optimizer, and with both. The results clearly demonstrate that combining both strategies leads to the highest improvements in accuracy, sensitivity, and specificity. For instance, one ANN experiment showed a jump from 72.5% accuracy, 50.0% sensitivity, and 79.3% specificity with no enhancements, to 90.0%, 57.1%, and 100.0%, respectively, when both components were applied.

This pattern of improvement was consistent across all tested models ANN, CNN, and the hybrid CNN–LSTM regardless of whether the classification task was binary or multiclass. The findings underline the importance of applying the right configuration strategies to maximize model reliability before introducing more complex signal transformation methods like wavelet decomposition, which will be discussed in the following sections.

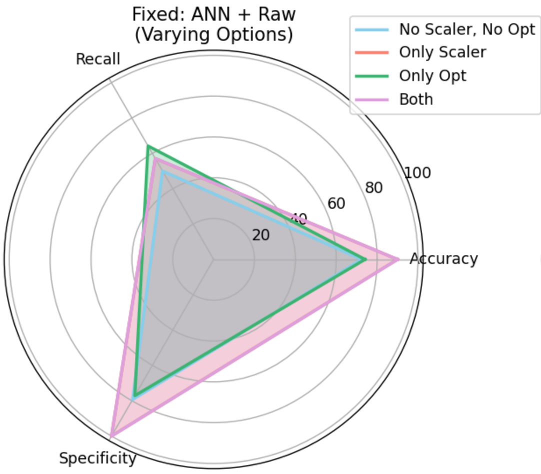


Figure 3.1: Radar chart showing performance metrics (accuracy, sensitivity, and specificity) under different training configurations: raw data (no scaler, no optimizer), only scaler, only optimizer, and both scaler and optimizer.

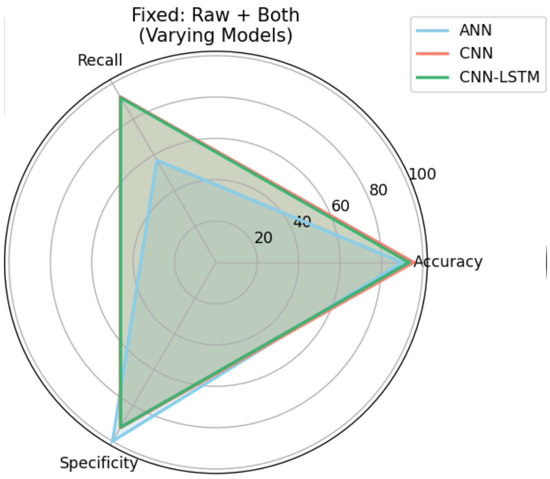


Figure 3.2: Radar chart showing Performance comparison of ANN, CNN, and CNN–LSTM models for epileptic classification.

3.2.2 Comparative Analysis of ANN, CNN, and CNN–LSTM Models

In our comparison on the same EEG dataset, the ANN's performance consistently falls into the mid-range overall accuracy between roughly 75 % and 90 %, sensitivity (true-positive rate) around 55 %–70 %, and specificity near 100 %. When we switch to convolutional architectures, the CNN pushes those numbers dramatically higher: accuracy typically lies between 90 % and 98 %, sensitivity in the same high band, and specificity remains essentially perfect. Adding an LSTM on top of the CNN brings a modest shift: the hybrid CNN–LSTM records accuracies of about 85 %95 % and sensitivities of 80 %92 %, with specificity still very strong but just below the pure CNN's.

Turning to computational cost, the plain ANN completes a full train-and-test cycle on our EEG data in under one minute (0.5–1 min), making it attractive for ultra-lightweight deployments but with limited detection reliability. The CNN, owing to its convolutional feature extraction, requires on the order of 4–6 minutes yet delivers a 20 + point jump in both accuracy and sensitivity over the ANN. The CNN–LSTM incurs the longest runtimes (8–12 minutes) because its recurrent layer processes sequences frame by frame.

Together, these results underscore the classic trade-off in EEG seizure classification: convolutional filters unlock the most significant gains in correctly identifying seizure patterns, while recurrent layers offer incremental temporal modeling at a non-negligible time penalty. For real-time or resource-constrained settings, the CNN emerges as the optimal choice achieving consistently high accuracy (> 90 %), sensitivity (> 90 %), and specificity (100 %) with an acceptable execution time.

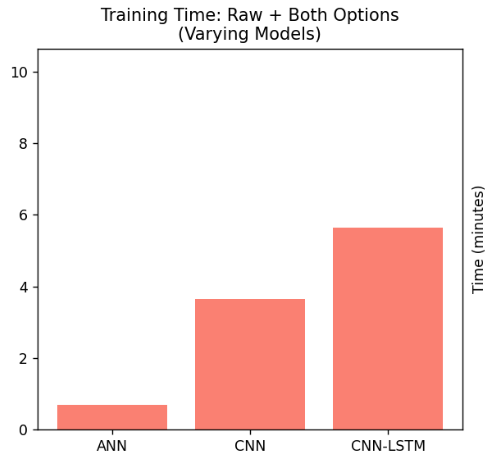


Figure 3.3: Average runtime of ANN, CNN, and CNN–LSTM models.

3.2.3 Impact of Feature Transformation Methods

he effect of two feature transformation methods—wavelet decomposition and derivative-based extraction—versus raw EEG input is shown in Figure 3.4. Across all three architectures, wavelet decomposition provides consistent, moderate boosts in classification metrics: accuracy increases by approximately 2–10 percentage points and sensitivity by 3–20 pp, while specificity remains essentially unchanged. Derivative based features yield a more mixed outcome: in some experiments they match or exceed wavelet driven sensitivity gains (up to +15 pp), but in others they slightly reduce accuracy (down by 3–5 pp). Importantly, these relative shifts are uniform across ANN, CNN, and CNN–LSTM models no single architecture benefits disproportionately from one method over the other.

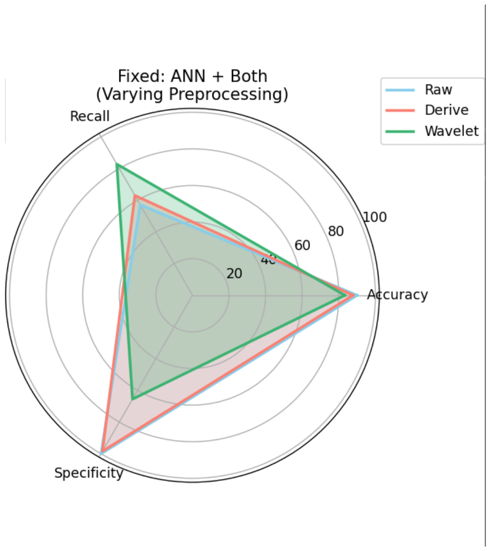


Figure 3.4: Classification performance ranges for raw data, wavelet decomposition, and derivative-based feature extraction across all models (accuracy, sensitivity, specificity).

The divergent computational costs of the two approaches are highlighted in Figure 3.5. Applying wavelet decomposition introduces only a modest overhead, with training and inference cycles lengthening by roughly 10–25% compared to raw data. In contrast, derivative extraction can double or even triple the total runtime, particularly for the CNN–LSTM model, where the sequential feature derivation significantly compounds the cost of the recurrent layer.

Figure 3.5: Execution-time overhead for wavelet decomposition and derivative extraction versus raw data across ANN, CNN, and CNN–LSTM.

Taken together, these results suggest that if maximal classification accuracy and sensitivity are the sole objectives, derivative based preprocessing may be justified. However, for most real-time or resource-constrained EEG applications, wavelet decomposition offers the best balance delivering reliable performance gains with only moderate impact on execution time.

3.3 Discussion

Our systematic evaluation of three deep learning architectures ANN, CNN, and CNN–LSTM under multiple training configurations and feature transformation strategies has yielded several clear insights. First and foremost, convolutional filtering emerges as the cornerstone of effective seizure pattern recognition: the pure CNN consistently outperforms the ANN and the CNN–LSTM in accuracy and sensitivity, regardless of whether raw waveforms or transformed features are used. This superiority stems from the CNN's ability to learn localized, time invariant features directly from the EEG signal, capturing spike and wave complexes and other salient seizure markers far more reliably than fully connected layers alone.

Second, the addition of an LSTM layer to the convolutional backbone offers only marginal gains in classification metrics over the CNN, while imposing a substantial runtime penalty. The CNN–LSTM's strength lies in modeling temporal dependencies across segments, which can prove useful for subtle pre-ictal pattern detection; however, our results show that these gains are often outweighed by doubled or even tripled inference times. In practical settings where real-time responsiveness or hardware constraints matter, the pure CNN strikes a more favorable balance between performance and speed.

Third, our exploration of training configurations combining data scaling with optimizer tuning demonstrates that thoughtful "front-end" choices can yield performance jumps comparable to, or even exceeding, those obtained by complex feature extraction. In nearly every experiment, the joint application of a standard scaler and an optimized learning rate/optimizer delivered the largest single boost in accuracy, sensitivity, and specificity, underscoring the importance of stable, well conditioned input distributions before any architectural refinements.

Finally, when we compare the two advanced feature transformation methods, wavelet decomposition proves to be the most pragmatic enhancement. It reliably adds a few percentage points to accuracy and sensitivity across all three models, while incurring only a modest (10–25%) increase in training and inference time. In contrast, deriving higher-order temporal derivatives can sometimes amplify sensitivity but at the cost of doubling or tripling execution time an overhead rarely justified outside of offline, exploratory analyses.

Taken together, these findings paint a coherent picture: for real world EEG seizure classification, a CNN trained on appropriately scaled data and optionally augmented by lightweight wavelet-based features represents the most effective and efficient approach. More complex architectures (CNN–LSTM) or heavier preprocessing (derivatives) offer niche advantages, but their incremental benefits must be weighed carefully against the additional computational burden. As practitioners, we must therefore prioritize not only raw model capacity but also the often-overlooked "upstream" choices—data conditioning and training configuration—that can profoundly shape final performance.

Conclusion

In this chapter, we presented the practical part of our project. We trained and compared three deep learning models (ANN, CNN, and CNN-LSTM) to classify epileptic patients using EEG signals. Different preprocessing methods and training options were tested to evaluate their impact on performance and training time. The results showed that CNN models offered the best balance between accuracy and efficiency, while pre-processing and optimization significantly improved performance.

Chapter 4

Development of the EEG Classification Interface

This chapter presents the design and implementation of the NeuroScan EEG Analysis System, a modular platform for clinical-grade electroencephalogram (EEG) data classification. The system integrates signal preprocessing, model-based inference, and automated reporting within a cohesive architecture designed for flexibility, scalability, and clinical usability.

4.1 System Overview

The NeuroScan Interface (Figure 4.1) follows a three-tier architecture that distinctly separates concerns among the user interface, data processing, and storage components. This layered approach enhances maintainability, facilitates updates, and allows modular testing of subsystems.

- **Presentation Layer**
  Developed using Streamlit, this layer offers a lightweight web interface with a clinical theme. It supports patient data entry, signal upload, model selection, and visualization of results in an intuitive format suitable for clinical environments.

- **Processing Layer**
  A Python-based backend performs signal preprocessing, feature extraction, and model inference. The modular design supports plug-and-play integration of new preprocessing techniques or classification models without changes to the interface.

- **Data Layer**
  A local SQLite database stores patient records in a structured format. Each EEG entry is timestamped and linked with model configurations and output results, ensuring traceability and auditability.

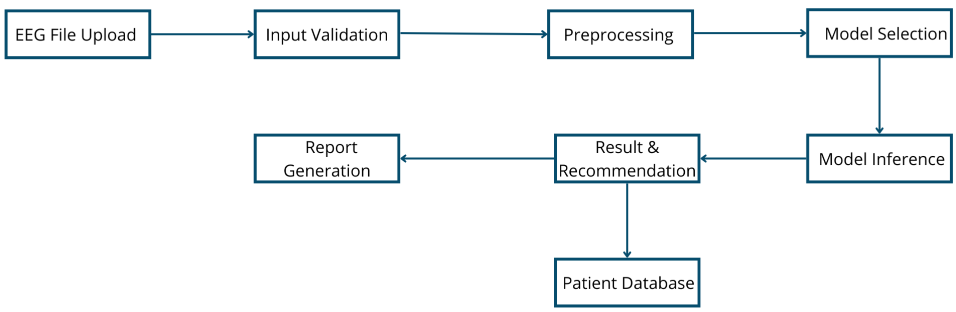


Figure 4.1: System Architecture Diagram

4.2 Model Integration Framework

The NeuroScan interface supports modular and dynamic integration of pre-trained classification models stored within a structured directory hierarchy. This design ensures compatibility with multiple machine learning experiments while maintaining ease of access and system robustness.

4.2.1 Modular Repository Structure

All classification models are stored in a root directory labeled `Models/`, which is hierarchically organized into subfolders based on three main criteria:

- **Preprocessing Technique**
  - `Raw_Data`
  - `Derive` (derivative transformation)
  - `Wavelet_dec` (wavelet decomposition)
- **Model Architecture**
  - `ANN` (Artificial Neural Network)
  - `CNN` (Convolutional Neural Network)
  - `CNN-LSTM` (Hybrid Convolutional-Recurrent Network)
- **Experiment Variant**
  - Labeled as `Exp1`, `Exp2`, etc., corresponding to specific experimental setups or training configurations

Models are saved with filenames that indicate the applied optimization techniques:

- `No_Options.keras` (baseline model)
- `With_Standard.keras` (standardization applied)
- `With_Optimizer.keras` (using Adam optimizer)
- `With_Options.keras` (combination of standardization and Adam)

This logical folder structure enables the system to retrieve the correct model using only the user's selected options, without requiring manual navigation or user-side knowledge of file organization.

An illustration of the model selection interface, including the preprocessing, architecture, experiment, and optimization options, is shown in Figure 4.2.

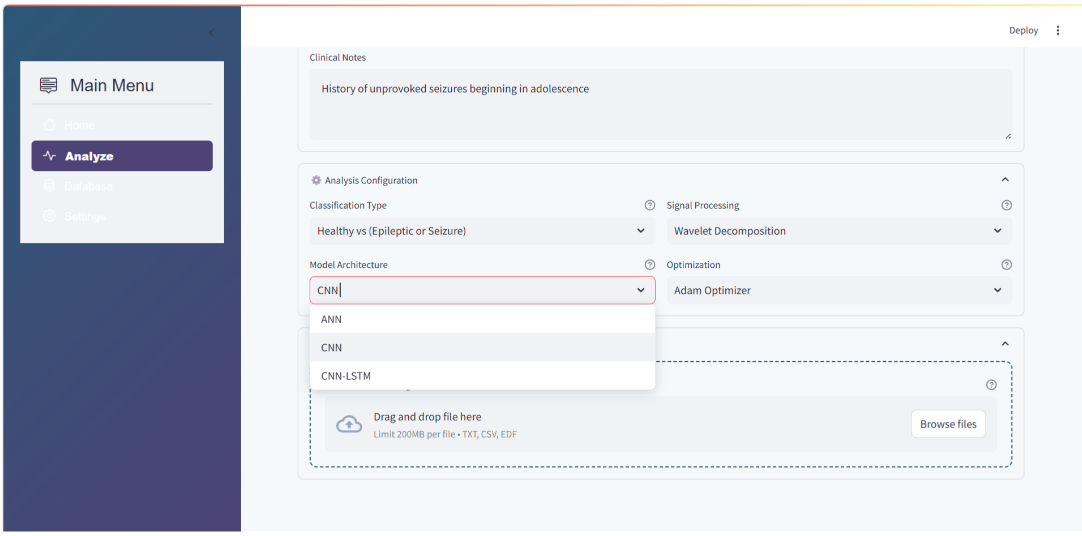


Figure 4.2: Model selection interface highlighting preprocessing, architecture, experiment, and optimization dropdowns

4.2.2 Intelligent Path Construction

When the user finalizes their configuration via the interface, the backend uses these selections to dynamically resolve the path to the correct model. The resolution system is based on mappings of UI labels to folder names and filename conventions. This ensures that even with 216 different valid combinations (3 preprocessing types × 3 architectures × 6 experiments × 4 optimization settings), each model can be automatically and reliably located.

To guarantee robustness, the system includes validation checks to verify model existence. If the resolved path points to a missing model, the application gracefully reverts to a fallback configuration such as the baseline model from the same experiment and notifies the user accordingly.

An example of such a fallback notification is shown in Figure 4.3.

4.3 Core Functional Modules

The interface implements a modular pipeline that processes EEG signals from acquisition to diagnostic output through three specialized components.

4.3.1 Signal Preprocessing

The system transforms raw EEG inputs into model-compatible representations using one of three user-selected methods. A summary of the preprocessing methods, including their transformations and resulting output dimensions, is provided in Table 4.1.

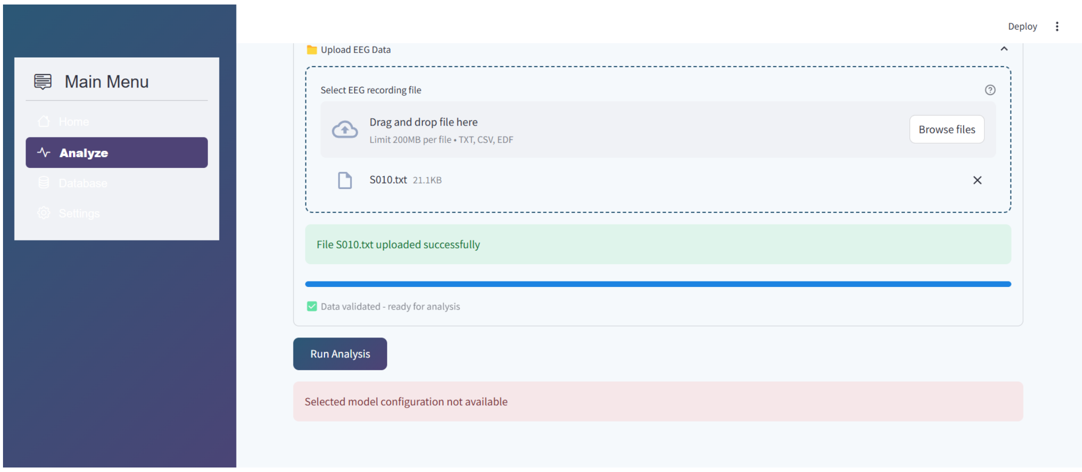


Figure 4.3: System notification for fallback model activation

Table 4.1: Preprocessing Method Specifications

Method	**Transformation**	**Output**
Raw	Sequence length normalization	(1, 4097)
Derivative	Temporal difference calculation	(1, 16382)
Wavelet	Multi-resolution decomposition	(1, 4129)

- **Raw**: Preserves original signal characteristics while standardizing dimensions
- **Derivative**: Enhances transient features through differential analysis
- **Wavelet**: Provides joint time-frequency representation using Daubechies-4 basis

4.3.2 Signal Visualization

The system provides interactive EEG visualizations for clinical review prior to classification, as shown in Figure 4.4 below

Technical implementation:

- Time-domain plot: Matplotlib with custom cycler for channel colors
- PSD plot: SciPy's `signal.welch()` with Hann windowing
- Interactivity: Streamlit's `pyplot` integration

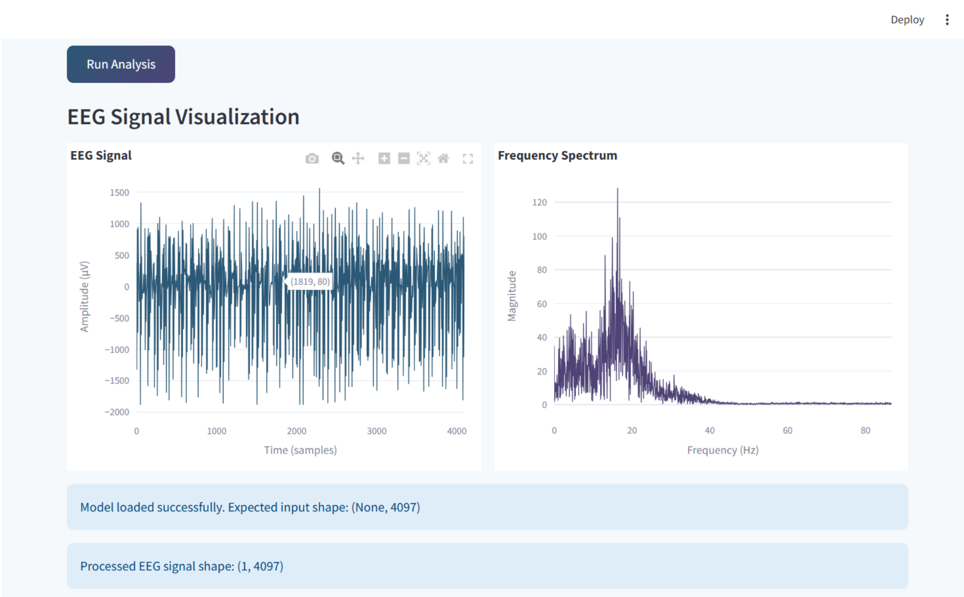


Figure 4.4: EEG signal visualization showing

4.3.3 Classification Engine

Once preprocessing is complete, the transformed signal is passed into a neural model for classification. The interface supports both binary (e.g., seizure vs. non-seizure) and multi-class (e.g., epileptic patterns) classification modes.

Workflow:

- **Model Execution:** The selected .keras file is loaded, and inference is run.
- **Output Interpretation:**
  - For binary models, a sigmoid activation returns a probability which is thresholded at 0.5.
  - For multi-class models, a softmax activation returns class probabilities from which the highest is selected.

An example of a classification result, showing the predicted probability and the corresponding final class label, is presented in Figure 4.5.

4.3.4 Clinical Reporting

The final step is to summarize the analysis in a report tailored for medical professionals. Reports are auto-generated and exported in PDF format, integrating several components:

- **Patient Metadata** (ID, age, date/time of analysis)
- **Signal Visualization**
- **Diagnostic Findings:**
  - Includes the predicted class and associated confidence

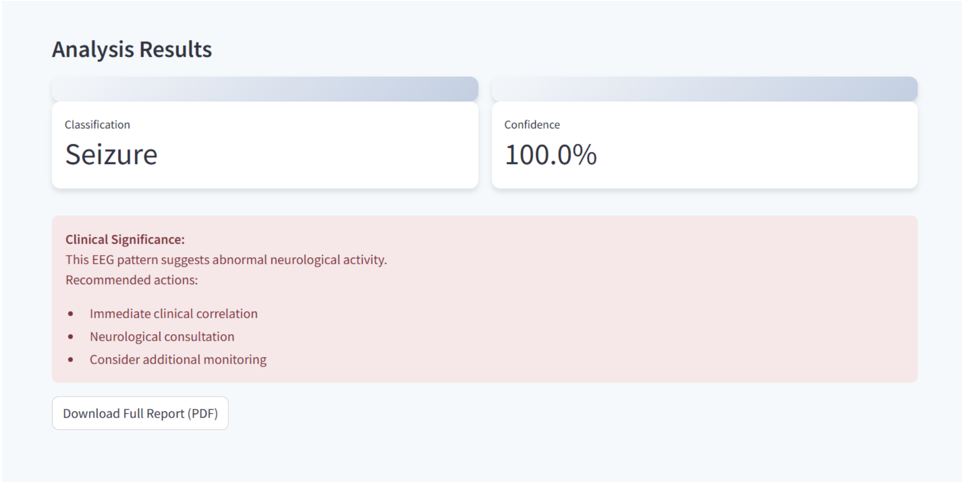


Figure 4.5: Results display panel showing probability and final class label

- Highlights abnormal patterns, if detected

- **Auto-generated Recommendations**

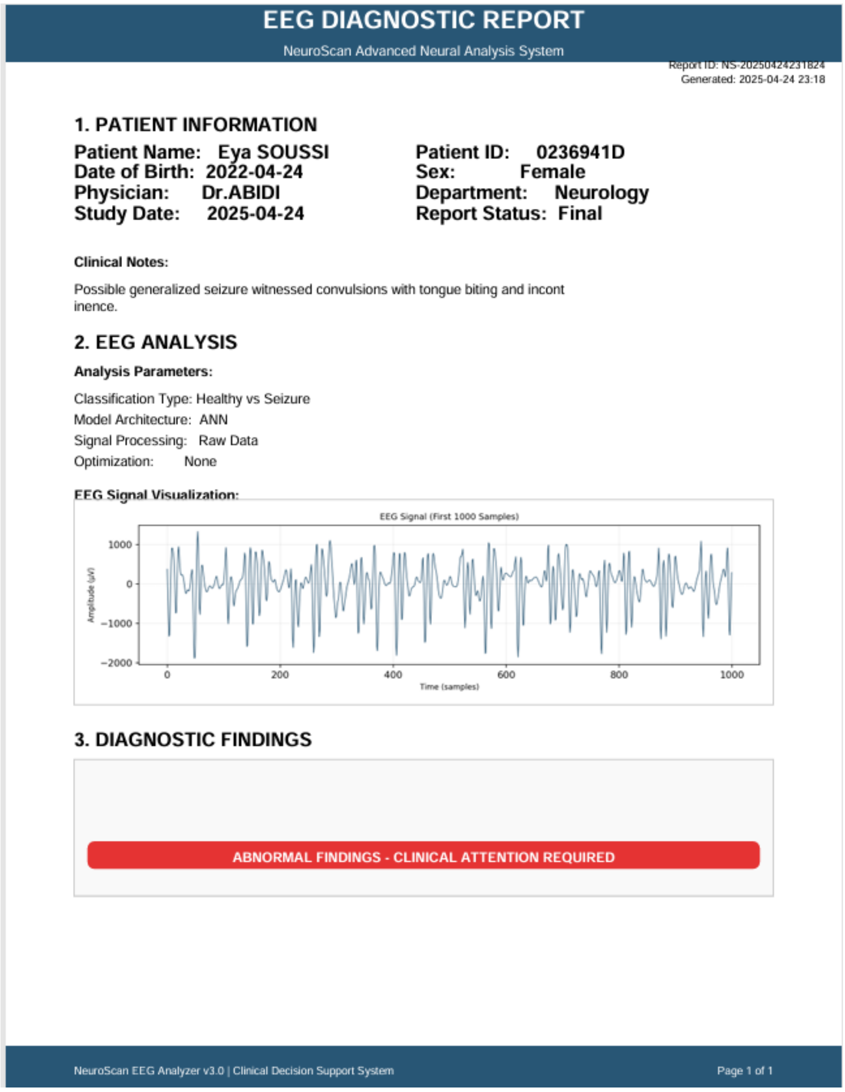
The system offers real-time clinical guidance based on the detected class
The report generation system utilizes the following Python libraries:

- `Report!Lab` - Core PDF generation engine for creating clinical-grade reports
- `PyPDF2` - Merging multiple PDF components into final documents
- `Pillow` - Embedding and resizing graphical elements
- `datetime` - Timestamping reports with analysis time
- `pandas` - Structuring patient metadata tables

The reporting pipeline executes three key operations:

1. Assembles all analysis components (text, plots, tables) into PDF pages
2. Applies hospital-branded templates and watermarks
3. Outputs encrypted PDFs with audit trails

The final output of the reporting system is illustrated through a generated clinical report for a sample patient, as shown in Figure 4.6



EEG DIAGNOSTIC REPORT
NeuroScan Advanced Neural Analysis System
Report ID: NS-20250424231824
Generated: 2025-04-24 23:18

1. PATIENT INFORMATION

Patient Name: Eya SOUSSI	Patient ID: 0236941D
Date of Birth: 2022-04-24	Sex: Female
Physician: Dr.ABIDI	Department: Neurology
Study Date: 2025-04-24	Report Status: Final

Clinical Notes:

Possible generalized seizure witnessed convulsions with tongue biting and incont inence.

2. EEG ANALYSIS

Analysis Parameters:

Classification Type: Healthy vs Seizure
Model Architecture: ANN
Signal Processing: Raw Data
Optimization: None

EEG Signal Visualization:

3. DIAGNOSTIC FINDINGS

ABNORMAL FINDINGS - CLINICAL ATTENTION REQUIRED

NeuroScan EEG Analyzer v3.0 | Clinical Decision Support System
Page 1 of 1

Figure 4.6: Sample PDF preview showing metadata, signal plot, and clinical note

4.3.5 Patient Record Interface

After generating the clinical report, the system offers direct access to the patient database interface, allowing users to:

- Review historical EEG submissions
- View associated model configurations and results
- Download or regenerate reports
- Filter records by patient ID, date, or diagnosis

The interface (shown in Figure 4.7) is designed for ease of navigation and clinical tracking, showing a structured list with the following columns

Patient ID	Upload Date	Diagnosis	Confidence	View Signal

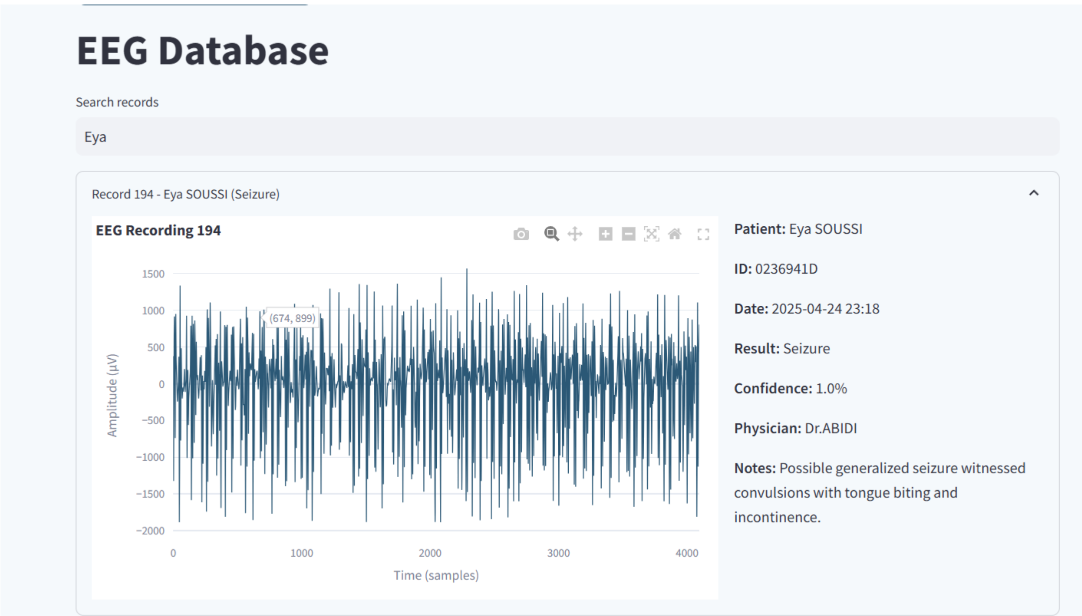


Figure 4.7: Patient database interface showing the interactive table with filtering options

This feature supports longitudinal follow-up, helping clinicians monitor diagnostic changes over time and maintain a complete digital patient history.

Python libraries used:

- `sqlite3` - For patient record storage and retrieval
- `pandas` - For handling the data table operations

4.4 Validation & Error Handling

To ensure reliability and avoid false or misleading results, the NeuroScan EEG Classification Interface implements robust validation and error-handling mechanisms throughout the system. This ensures data integrity, proper model function, and clear user feedback in case of any issues.

4.4.1 EEG File Validation

The system performs checks on the EEG data before processing to ensure that the input meets necessary quality standards. The file is validated in the following ways:

- **Minimum Data Points**: The system checks that the input file contains at least 100 data points to ensure that the signal is of adequate length for analysis. An error message is displayed if the requirement is not met, as shown in Figure 4.8
- **Numerical Content**: The system verifies that the content of the file is purely numerical, as non-numerical data can interfere with further processing.
- **Multi-format Support**: The system accepts various EEG file formats, including .txt, .csv, and .edf. It will gracefully reject unsupported file formats.

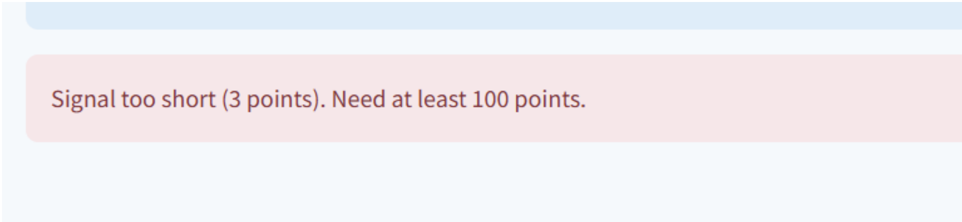


Figure 4.8: Error message displayed for insufficient data points

4.4.2 Model Safety Checks

The system ensures that only properly preprocessed data is passed to the model, and model outputs are validated to meet the necessary standards for clinical use:

- **Input Shape Verification**: Checks that processed data matches the expected input shape of the selected model. Any mismatch triggers an error message.
- **NaN Detection in Outputs**: Flags results when numerical instability produces NaN values and alerts the user.
- **Confidence Thresholding**: Rejects outputs with confidence below 60% and prompts user verification.

4.4.3 Error Handling Cases

The system logs errors in a structured way, categorizing each issue and providing clear feedback. The different types of errors, along with their detection methods and user feedback, are summarized in Table 4.2. An example of the system's log output displaying these error types is shown in Figure 4.9.

Table 4.2: Error Handling Specifications

Error Type	**Detection Method**	**User Feedback**
Invalid File Format	File signature analysis	"Unsupported file type"
Short Signal	Signal length $<$ 100 points	"Insufficient data points"
Model Mismatch	Input shape mismatch	"Configuration incompatible"
NaN Detection	NaN values in output	"Invalid output detected"

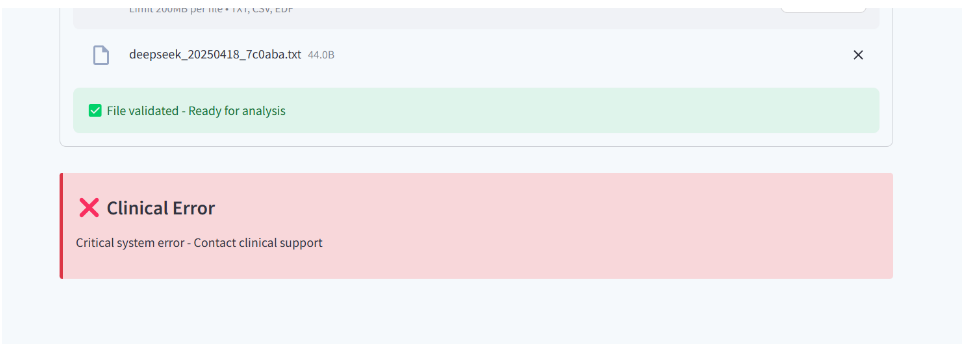


Figure 4.9: System log output showing error types and corresponding user feedback

Conclusion

The NeuroScan system delivers a clinically viable EEG analysis platform that combines automated classification with interactive visualization and robust validation. Its modular architecture ensures reliable performance while enabling future expansion, bridging the gap between research algorithms and clinical diagnostic workflows.

Conclusion

Epileptic seizure detection from EEG signals remains a critical challenge in the field of biomedical engineering due to the complex, non-stationary nature of EEG data and the need for fast, accurate, and reliable detection systems to support clinical diagnosis and management. This research has explored the use of deep learning models for epileptic seizure detection from EEG signals, focusing on both the preprocessing techniques and advanced model architectures. The importance of effective data preprocessing, including methods like wavelet decomposition and derivation-based preprocessing, was highlighted as crucial steps in improving the quality of EEG signals for accurate seizure detection. Among the deep learning models tested, the Convolutional Neural Network (CNN) and the CNN-LSTM hybrid model showed superior performance in comparison to simpler architectures, particularly when feature transformation techniques were applied. The models were successful in meeting the initial objective of creating a reliable and efficient seizure detection system, with the CNN-LSTM hybrid model offering the best trade-off between complexity and performance.

However, the study was not without challenges. The limited size and diversity of the dataset posed constraints on the generalization capabilities of the models, and the computational cost associated with training the deep learning models was significant, potentially hindering real-time applications. Despite these challenges, the results indicate that there is strong potential for further improvement in the field. Future research could focus on expanding the dataset to include a wider range of EEG recordings from diverse populations to improve model robustness and generalization. Additionally, exploring more advanced architectures, such as transformer networks or hybrid models combining attention mechanisms, could lead to even higher performance.

From our perspective, the integration of such advanced models into real-time clinical settings, combined with continuous learning approaches and adaptive mechanisms, offers a promising path toward more accurate, timely and patient-centered epilepsy management solutions.

Bibliography

[1] BioRender. Electroencephalography (eeg) recording. https://www.biorender.com/template/electroencephalography-eeg-recording. Accessed: April 29, 2025.

[2] GeeksforGeeks. Deep learning | introduction to long short term memory (lstm). https://www.geeksforgeeks.org/deep-learning-introduction-to-long-short-term-memory/, n.d. Accessed: 2025-04-28.

[3] Manuel J. Rivera, Javier Sanchis, Oscar Corcho, Miguel A. Teruel, and Juan Trujillo. Evaluating cnn methods for epileptic seizure type classification using eeg data. *IEEE Access*, 12:75483–75495, 2024. https://doi.org/10.1109/ACCESS.2024.3406332 doi:10.1109/ACCESS.2024.3406332.

[4] Mengni Zhou, Cheng Tian, Rui Cao, Bin Wang, Yan Niu, Ting Hu, Hao Guo, and Jie Xiang. Epileptic seizure detection based on eeg signals and cnn. *Frontiers in Neuroinformatics*, 12, 2018. URL: https://www.frontiersin.org/journals/neuroinformatics/articles/10.3389/fninf.2018.00095, https://doi.org/10.3389/fninf.2018.00095 doi:10.3389/fninf.2018.00095.

[5] El-Sayed A Hemdan EE Ein Shoka AA, Dessouky MM. Eeg seizure detection: concepts, techniques, challenges, and future trends. *national library of medicine*.

[6] Dessouky M.M. El-Sayed Ein Shoka, A.A. Eeg seizure detection: concepts, techniques, challenges, and future trends. *mbient Intell Human Comput*.

[7] Shilpa Jain, Vedant V. Gupta, Ankur Singh Bist, Mukesh Joshi, and Ashish Garg. Automated epileptic seizure detection of eeg signals using machine learning. In Paulo J. Sequeira Gonçalves, Pradeep Kumar Singh, Sudeep Tanwar, and Gregory Epiphaniou, editors, *Proceedings of Fifth International Conference on Computing, Communications, and Cyber-Security*, pages 349–357, Singapore, 2025. Springer Nature Singapore.

[8] Federico Del Pup, Andrea Zanola, Louis Fabrice Tshimanga, Alessandra Bertoldo, and Manfredo Atzori. The more, the better? evaluating the role of eeg preprocessing for deep learning applications. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 33:1061–1070, 2025. https://doi.org/10.1109/TNSRE.2025.3547616 doi:10.1109/TNSRE.2025.3547616.

[9] Shruti Jadon. Introduction to different activation functions for deep learning. *Medium, Augmenting Humanity*, 16, 2018.

[10] Federico Del Pup, Andrea Zanola, Louis Fabrice Tshimanga, Alessandra Bertoldo, and Manfredo Atzori. The more, the better? evaluating the role of eeg preprocessing for deep learning applications. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 33:1061–1070, 2025. https://doi.org/10.1109/TNSRE.2025.3547616 `doi:10.1109/TNSRE.2025.3547616`.

[11] George E.P. Box and Gwilym M. Jenkins. *Time Series Analysis: Forecasting and Control*. Holden-Day, 1976.

[12] David A. Dickey and Sastry G. Pantula. Determining the order of differencing in autoregressive processes. *Journal of Business & Economic Statistics*, 5(4):455–461, 1987.

[13] Techopedia. Artificial neural network (ann), 2025. Accessed: 2025-01-22. URL: `https://www.techopedia.com/definition/5967/artificial-neural-network-ann`.

[14] DataCamp. Introduction to activation functions in neural networks, 2025. Accessed: 2025-01-28. URL: `https://www.datacamp.com/tutorial/introduction-to-activation-functions-in-neural-networks`.

[15] Alex Sherstinsky. Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network. *Physica D: Nonlinear Phenomena*, 404:132306, 2020. https://doi.org/10.1016/j.physd.2019.132306 `doi:10.1016/j.physd.2019.132306`.

[16] Zuyi Yu, Weidong Zhou, Fan Zhang, Fangzhou Xu, Shasha Yuan, Yan Leng, Yang Li, and Qi Yuan. Automatic seizure detection based on kernel robust probabilistic collaborative representation. *Medical & biological engineering & computing*, 57:205–219, 2019.

[17] Fatima Hassan, Syed Fawad Hussain, and Saeed Mian Qaisar. Epileptic seizure detection using a hybrid 1d cnn-machine learning approach from eeg data. *Journal of Healthcare Engineering*, 2022(1):9579422, 2022.

[18] Xiashuang Wang, Yinglei Wang, Dunwei Liu, Ying Wang, and Zhengjun Wang. Automated recognition of epilepsy from eeg signals using a combining space–time algorithm of cnn-lstm. *Scientific Reports*, 13(1):14006, 2023. https://doi.org/10.1038/s41598-023-41537-z `doi:10.1038/s41598-023-41537-z`.

[19] Manuel J Rivera, Javier Sanchis, Oscar Corcho, Miguel A Teruel, and Juan Trujillo. Evaluating cnn methods for epileptic seizure type classification using eeg data. *IEEE Access*, 2024.

Annexes

Table 4.3: Exp.1 : Healthy vs Seizure

Raw EEG												
	ANN				**CNN**				**CNN-lstm**			
	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)
$-STN$, $-OPT$	72.5	50.0	79.3	0.60	78.89	74.34	74.34	3.55	96.67	96.56	96.56	5.79
$+STN$	90.0	57.1	100.0	0.56	96.67	94.44	94.44	3.85	82.22	71.73	71.43	5.85
$+OPT$	74.1	64.2	77.1	0.67	83.33	76.46	76.46	3.85	92.22	93.39	93.39	5.40
$+STN$ and $+OPT$	90.0	57.1	100.0	0.70	95.56	92.59	92.59	3.64	93.33	92.06	92.06	5.64
DERV												
	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)
$-STN$, $-OPT$	75.56	81.48	73.02	3.00	82.22	79.89	79.89	19.40	93.33	91.01	91.01	36.27
$+STN$	92.22	74.07	100.00	2.59	92.22	87.04	87.04	19.47	91.11	91.53	91.53	35.05
$+OPT$	67.78	88.89	58.73	3.48	88.89	85.71	85.71	18.18	96.67	94.44	94.44	38.64
$+STN$ and $+OPT$	87.78	62.96	98.41	3.37	94.44	90.74	90.74	18.87	94.44	92.86	92.86	35.48
WT												
	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)
$-STN$, $-OPT$	55.00	56.34	96.55	0.26	92.60	98.05	87.96	0.31	91.00	89.66	93.50	4.55
$+STN$	90.00	89.66	79.31	0.25	98.33	96.55	100.00	1.24	91.00	89.00	93.00	4.35
$+OPT$	51.67	52.78	86.21	0.27	98.33	96.55	100.00	0.16	51.00	75.00	29.00	0.22
$+STN$ and $+OPT$	83.33	82.76	65.52	0.33	95.00	89.66	100.00	0.15	88.00	86.00	90.00	4.31

Table 4.4: Exp.2 : Epileptic vs Seizure

Raw EEG												
	ANN				**CNN**				**CNN-lstm**			
	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)
$-STN$, $-OPT$	83.33	82.76	83.87	0.50	88.33	88.04	88.04	2.43	93.33	93.12	93.65	3.87
$+STN$	90.00	79.31	100.00	0.49	98.33	98.28	100.00	2.55	94.44	94.97	93.65	4.09
$+OPT$	85.00	86.21	83.87	0.62	93.33	93.33	93.55	2.35	92.22	93.39	93.39	5.40
$+STN$ and $+OPT$	83.33	65.52	100.00	0.55	98.33	98.28	100.00	2.27	92.22	91.27	93.65	3.70
DERV												
	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)
$-STN$, $-OPT$	88.33	93.10	83.87	1.32	98.33	98.39	96.77	10.29	93.33	92.06	92.06	35.88
$+STN$	80.00	58.62	100.00	1.16	98.33	98.28	100.00	9.92	70.00	50.00	50.00	37.25
$+OPT$	80.00	68.97	90.32	1.26	96.67	96.77	93.55	10.23	64.67	57.97	57.97	47.89
$+STN$ and $+OPT$	85.00	68.97	100.00	1.33	98.33	98.28	100.00	2.27	95.56	94.71	94.71	35.18
WT												
	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)
$-STN$, $-OPT$	81.67	70.97	70.97	0.28	98.33	96.55	100.00	0.13	51.66	51.60	75.80	9.33
$+STN$	80.00	100.00	100.00	0.26	96.67	93.10	100.00	0.11	51.67	51.67	75.80	9.69
$+OPT$	81.67	100.00	64.52	0.29	95.00	96.55	93.33	0.14	83.30	65.52	87.81	4.44
$+STN$ and $+OPT$	80.00	100.00	100.00	0.31	93.33	89.66	96.30	0.16	51.66	51.11	75.80	9.23

Table 4.5: Exp.3 : Healthy vs Epileptic

Raw EEG												
	ANN				**CNN**				**CNN-lstm**			
	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)
-STN , -OPT	68.33	79.31	58.06	0.72	96.67	97.62	100.00	3.67	82.50	82.56	84.48	8.43
+STN	83.33	81.03	85.48	0.67	98.89	99.21	100.00	3.76	85.00	84.59	72.41	8.86
+OPT	66.67	91.38	43.55	0.85	93.33	95.24	100.00	3.60	92.22	93.39	93.39	5.40
+STN and +OPT	75.00	65.52	83.87	0.84	96.67	97.62	100.00	3.69	82.50	82.68	87.93	8.34
DERV												
	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)
-STN , -OPT	65.00	89.66	41.94	2.24	92.50	92.63	96.55	19.33	84.17	84.12	84.12	38.03
+STN	69.17	75.86	62.90	2.06	95.83	95.80	94.83	19.44	75.00	74.14	74.14	38.38
+OPT	65.00	81.03	50.00	2.29	86.67	86.99	96.55	19.03	87.50	87.40	87.40	39.90
+STN and +OPT	74.17	72.41	75.81	2.30	95.00	95.11	98.28	19.82	90.00	90.21	90.21	40.48
WT												
	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)
-STN , -OPT	81.67	64.00	64.52	0.28	81.67	65.52	96.77	0.17	51.66	51.60	75.80	9.33
+STN	86.67	100.00	100.00	0.28	95.00	93.10	96.43	3.19	51.67	51.67	75.80	9.69
+OPT	80.00	79.00	70.97	0.28	98.33	100.00	96.67	3.05	83.30	65.52	87.81	4.44
+STN and +OPT	86.67	100.00	100.00	0.28	96.67	93.10	100.00	3.02	51.66	51.11	75.80	9.23

Table 4.6: Exp.4 : Healthy vs (Epileptic and Seizure)

Raw EEG												
	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)
-STN , -OPT	68.33	79.31	58.06	0.72	81.67	82.04	82.04	4.68	89.33	89.72	91.94	9.60
+STN	83.33	81.03	85.48	0.67	83.33	83.20	83.20	4.95	75.33	74.45	69.35	9.61
+OPT	66.67	91.38	43.55	0.85	81.67	81.70	81.70	4.61	92.67	91.84	87.10	9.52
+STN and +OPT	75.00	65.52	83.87	0.84	90.83	90.68	90.68	4.74	74.67	71.74	54.84	9.68
DERV												
	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)
-STN , -OPT	65.00	89.66	41.94	2.24	87.33	85.39	85.39	36.61	88.67	88.20	88.20	59.84
+STN	69.17	75.86	62.90	2.06	88.00	88.82	88.82	36.41	91.33	91.42	91.42	39.85
+OPT	65.00	81.03	50.00	2.29	85.33	85.83	85.83	35.92	85.33	84.88	84.88	41.43
+STN and +OPT	74.17	72.41	75.81	2.30	90.00	91.00	91.00	36.45	93.33	94.08	94.08	40.27
WT												
	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)
-STN , -OPT	85.00	84.50	84.50	38.34	68.89	98.18	22.86	1.91	82.22	82.22	60.00	5.37
+STN	86.50	86.20	86.20	38.67	84.44	80.00	91.43	0.38	85.56	90.91	77.14	5.08
+OPT	84.20	83.50	83.50	36.76	84.44	92.73	71.43	0.20	85.56	96.36	68.57	5.15
+STN and +OPT	89.50	90.00	90.00	39.26	87.78	94.55	77.14	0.19	61.11	100.00	50.00	5.44

Table 4.7: Exp.5 : (Healthy and Epileptic) vs Seizure

Raw EEG												
	ANN				**CNN**				**CNN-LSTM**			
	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)
-STN, -OPT	72.50	50.00	79.35	0.60	82.50	62.50	62.50	5.12	94.00	90.25	96.64	10.91
+STN	90.00	57.14	100.00	0.56	93.33	86.96	86.96	4.71	92.00	83.03	98.32	10.47
+OPT	74.17	64.29	77.17	0.68	80.83	58.93	58.93	4.75	94.00	86.68	99.16	9.81
+STN and +OPT	90.00	57.14	100.00	0.71	93.33	85.71	85.71	5.00	95.33	89.90	99.16	10.61
DERV												
	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)
-STN, -OPT	84.00	80.65	84.87	2.45	89.33	80.16	80.16	21.46	90.00	78.19	78.19	42.52
+STN	92.00	61.29	100.00	2.32	96.00	91.52	91.52	21.69	94.00	92.64	92.64	40.39
+OPT	82.00	83.87	81.51	3.06	90.67	83.38	83.38	21.43	95.33	91.10	91.10	41.35
+STN and +OPT	87.33	38.71	100.00	3.01	94.67	88.29	88.29	21.57	84.00	62.48	62.48	336.74
WT												
	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)
-STN, -OPT	68.89	60.32	88.89	0.41	81.11	98.18	54.29	0.18	70.00	50.00	100.00	10.50
+STN	94.44	90.74	81.48	0.33	87.78	90.91	82.86	3.38	70.00	50.00	100.00	10.17
+OPT	62.22	70.90	92.59	0.36	90.00	98.18	77.14	3.64	70.00	50.00	100.00	0.81
+STN and +OPT	92.22	87.04	74.07	0.59	93.33	98.1	85.7	3.42	67.78	53.70	88.89	0.82

Table 4.8: Exp.6 : Healthy vs Epileptic vs Seizure

Raw EEG												
	ANN				**CNN**				**CNN-lstm**			
	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)
-STN , -OPT	65.83	65.12	82.06	0.74	80.83	76.77	88.23	4.46	93.33	92.73	96.39	9.74
+STN	75.00	70.88	86.23	0.77	93.33	91.45	95.83	4.91	85.33	85.55	92.01	9.19
+OPT	67.50	71.02	84.29	0.88	75.83	68.25	86.40	4.55	83.33	83.75	91.52	9.37
+STN and +OPT	79.17	79.17	50.67	0.85	89.17	85.41	92.95	4.63	88.67	87.30	93.82	9.72
DERV												
	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)
-STN , -OPT	67.33	69.02	83.29	2.44	80.83	76.77	88.23	4.46	83.33	80.71	91.07	40.06
+STN	75.83	74.36	87.22	2.22	93.33	91.45	95.83	4.91	89.33	86.46	94.08	40.77
+OPT	62.50	61.25	80.24	2.46	75.83	68.25	86.40	4.55	84.00	81.34	91.41	39.04
+STN and +OPT	69.33	67.25	83.75	0.95	89.17	85.41	92.95	4.63	74.00	76.34	86.63	40.35
WT												
	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)	ACC	Recall	SPC	T (min)
-STN , -OPT	80.83	76.77	88.23	4.47	93.33	91.45	95.83	4.91	80.00	78.98	80.02	3.84
+STN	93.33	91.45	95.83	4.91	80.83	76.77	88.23	4.47	84.44	86.67	84.78	3.95
+OPT	75.83	68.25	86.40	4.55	75.83	68.25	86.40	4.55	72.22	74.29	71.70	1.47
+STN and +OPT	89.17	85.41	92.95	4.63	89.17	85.41	92.95	4.63	84.44	85.15	84.68	0.19