Integrating Deep Learning and Ensemble Techniques for Improved Epileptic Seizure Detection

Abstract:

Introduction: Automated seizure detection from EEG data is crucial for improving the lives of individuals with epilepsy. In order to identify seizures, we investigated various deep learning and machine learning models, aiming to identify the most accurate and efficient approach.

Problem Statement: Existing seizure detection methods often struggle with balancing accuracy, computational efficiency, and generalizability across diverse EEG datasets. This study addresses this challenge by evaluating various models on two distinct datasets.

Methodology: Several models were trained and evaluated on two EEG datasets: a novel Voting Classifier ensemble (combining SVM, Random Forest, and XGBoost), CNN, DWT-based DNN, SVM, Random Forest, XGBoost, and MDBCN. Accuracy, precision, recall, F1-score, and computation time were used to evaluate performance.

Results: The Voting Classifier demonstrated outstanding performance across both datasets, achieving perfect accuracy, precision, recall, and F1-scores (100%) with competitive computation times of 9.63 seconds on Dataset 1 and 15 seconds on Dataset 2. In comparison, other models such as DWT-based DNN, CNN, MDBCN, SVM, Random Forest, and XGBoost showed strong but varied performance, with accuracies ranging from 88% to 99% and significantly different computational times. The MDBCN, in particular, struggled with low seizure recall on Dataset 2 (39%), while the CNN exhibited high computational demands. The results emphasize the Voting Classifier's ability to balance accuracy and efficiency, making it highly effective for seizure detection tasks.

Conclusion: The Voting Classifier consistently outperformed other models, demonstrating its potential as a highly accurate and efficient solution for automated seizure detection. The DWT-based DNN also emerged as a compelling option, especially for applications requiring rapid processing. Future research will focus on optimizing computationally intensive models like CNN, exploring hybrid models, validating performance on larger and more diverse real-world datasets, and integrating these models into real-time monitoring systems for improved patient care.

Keywords: Seizure Detection, Deep Learning, Multi-Dimensional Bayesian Convolutional Network, Deep Neural Network, Voting Classifier, Discrete Wavelet Transform**.**

1. Introduction

According to the World Health Organization (WHO), epilepsy is a chronic neurological illness that affects over 50 million individuals worldwide and is characterized by recurring seizures.[1] [2]. Unpredictable occurrences brought on by aberrant brain electrical activity, seizures can have a major negative impact on a person's and their family's quality of life. The impact goes beyond the immediate discomfort and disruption of daily routines, often leading to long-term challenges in various aspects of life. Seizures can lead to injuries from falls, cognitive impairment, social isolation, and, in severe cases, sudden unexpected death in epilepsy (SUDEP) [3]. The unpredictable nature of seizures necessitates continuous monitoring and rapid intervention to mitigate these risks. Current methods of seizure detection, largely reliant on manual review of electroencephalography (EEG) data by trained professionals, are laborious, time-consuming, subjective, and lack the scalability required for continuous, real-time monitoring of large populations [4]. This limitation underscores the urgent need for more accurate, efficient, and readily deployable seizure detection systems. The development of such systems holds the potential to revolutionize epilepsy care, enabling timely interventions, personalized treatment strategies, and potentially even seizure prediction and prevention, ultimately improving patient outcomes and reducing the overall burden of this debilitating condition [5] [6].

Electroencephalography (EEG) continues to be the gold standard for epilepsy diagnosis and surveillance. Electrodes are applied to the scalp using this non-invasive neurophysiological method to capture the brain's electrical activity. These recordings provide a detailed representation of brainwave patterns, offering invaluable insights into the underlying neurological processes. **Figure 1** illustrates a typical EEG recording system. The characteristic EEG patterns associated with epileptic seizures differ significantly from those observed during interictal periods (between seizures).

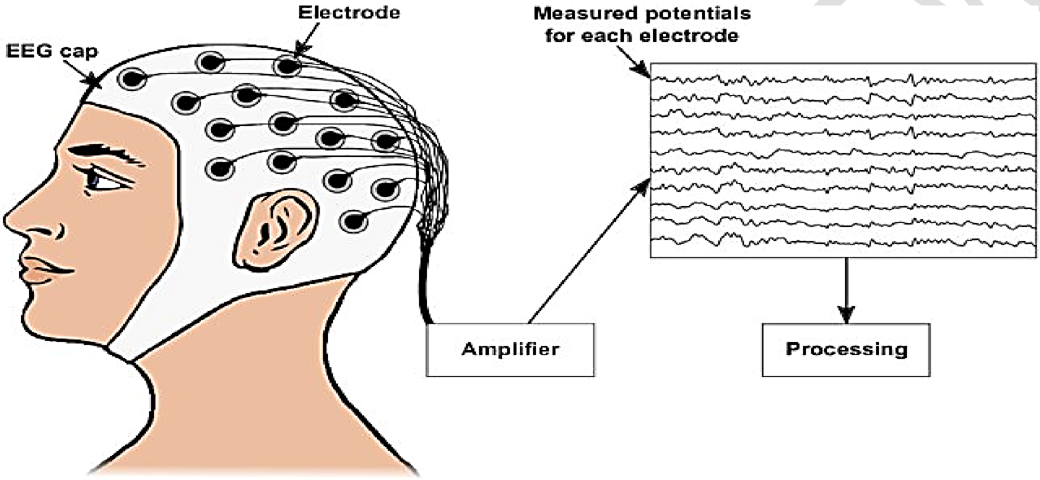


Figure 1: Illustration of EEG Recording Process [7].

Several distinct EEG patterns are recognized [8]**:**

- **Interictal Spikes**: Brief bursts of high-frequency electrical activity between seizures, often signaling a higher risk of seizures and helping identify epileptogenic brain zones. Detecting these transient, low-amplitude spikes is challenging.
- **Ictal Activity**: Abnormal electrical patterns during a seizure. These patterns vary by seizure type and brain location, making automated detection a complex task requiring advanced algorithms.
- **Post-ictal Slowing**: Low-frequency waves following a seizure, reflecting brain recovery. Their duration and characteristics offer valuable diagnostic insights.
- **High-Frequency Oscillations (HFOs)**: Fast oscillations (>80 Hz) linked to interictal spikes and epileptogenic activity. HFOs are promising biomarkers for epilepsy but demand sophisticated signal-processing techniques for detection.

These EEG patterns' complexity underscores the limitations of manual analysis, highlighting the need for advanced computational tools to identify subtle but critical diagnostic features accurately.

1.1 Machine Learning: A Paradigm Shift in Seizure Detection:

Seizure detection traditionally relies on neurologists visually analyzing EEG traces, a subjective and time-consuming process prone to variability. With the large volume of EEG data from continuous monitoring, there's a growing need for faster and more consistent solutions. Machine learning (ML) and deep learning (DL) now enable automated analysis of EEG data to identify seizure patterns and provide objective, data-driven evaluations. This shift transforms epilepsy management from reactive (responding to seizures) to proactive, enabling early warnings and personalized prevention strategies.[9].

The application of ML to EEG-based seizure detection has several key advantages [10]:

- Automation: ML algorithms can automate the process of seizure detection, freeing up clinicians' time and allowing for continuous monitoring of large patient populations.
- Objectivity: ML eliminates the subjectivity inherent in manual analysis, leading to more consistent and reliable seizure detection.
- Scalability: ML-based systems can easily handle the large volumes of data generated by continuous EEG monitoring.
- Real-time processing: Advanced ML architectures allow for real-time or near real-time seizure detection, enabling prompt interventions.
- Potential for prediction: Some ML models show promise in predicting seizure onset, enabling preventative measures.

1.2 Existing Challenges and Research Gaps:

Despite advances in ML for seizure detection, several challenges persist:

- **Data Variability**: EEG results vary due to factors like electrode placement, noise, and individual brain differences. This requires algorithms that can handle noisy and inconsistent data.
- **Computational Complexity**: Deep learning models need significant processing power, making them hard to implement in resource-limited settings.
- **Interpretability**: Many deep learning models are "black boxes," making their predictions hard to understand, which can limit their acceptance in medical use.
- **Dataset Limitations**: The lack of large, high-quality, annotated EEG datasets hinders the development of accurate models. Issues like class imbalance and data inconsistencies affect current datasets.

1.3 The Proposed Approach: Integrating Deep Learning and Ensemble Methods:

This study presents a novel approach combining deep learning and ensemble methods to address key challenges in seizure detection. Deep learning models like the Multi-Dimensional Bayesian Convolutional Network (MDBCN) and Deep Neural Networks (DNNs), along with Discrete Wavelet Transform (DWT), capture the complex spatiotemporal properties of EEG signals. To combat overfitting and complexity, ensemble techniques such as Support Vector Machines (SVMs), Random Forests, and XGBoost classifiers are used. A Voting Classifier combines the outputs of these models to improve accuracy, robustness, and generalizability. DWT enhances feature extraction by breaking down EEG signals into frequency sub-bands, capturing both high and low-frequency components essential for accurate seizure detection.

1.4 Objectives and Contributions:

The goal of the proposed study is to create a reliable and accurate system for detecting epileptic seizures. Using deep learning methods like Multi-Dimensional Bayesian Convolutional Networks (MDBCNs), [11] and Deep Neural Networks (DNNs) [12], with ensemble methods like Support Vector Machines (SVMs), Random Forests (RF), and XGBoost (XGB), the system aims to capture complex spatiotemporal features in EEG signals and improve overall performance.

This study introduces the use of Discrete Wavelet Transform (DWT) to improve feature extraction by breaking EEG data into frequency sub-bands, capturing both low- and high-frequency components linked to seizures. A voting classifier combines predictions from multiple models, enhancing the system's resilience and reducing errors. This ensemble approach helps address EEG data variability and minimizes overfitting, improving generalization and real-time detection. Focused on binary classification of seizure vs. non-seizure events, the study aims to advance automated seizure detection for more accurate and reliable patient monitoring and timely intervention.

1.5 Outline of the Study:

This manuscript's remaining sections are organized as follows: A thorough analysis of relevant research in the area of EEG-based seizure detection is provided in Section 2. The methodology is described in Section 3, which also includes model designs, data pretreatment methods, performance evaluation criteria, and a description of the dataset. Section 4 presents the results of the experiments, providing a detailed analysis of the performance of each model and the proposed ensemble system. Section 5 discusses the findings, limitations, and potential directions for future research. Finally, Section 6 concludes the study and highlights potential future avenues for exploration.

2. Related Work

EEG-based automated seizure detection has emerged as a key area of study in deep learning (DL) and machine learning (ML). Traditional ML approaches have typically relied on handcrafted feature extraction, while recent DL methods allow for learning directly from EEG data with minimal preprocessing. This section reviews notable advancements in seizure detection, focusing on ML and DL methodologies applied across various datasets with different evaluation metrics.

Feature-Based Machine Learning Approaches

Feature-based ML techniques have shown success in seizure detection by leveraging the statistical and geometric properties of EEG signals. Support Vector Machines (SVM), Naive Bayes, and k-nearest Neighbors (k-NN) are popular classifiers in this context:

- **SVM** classifiers utilizing covariance matrices with Riemannian geometry achieved high accuracy (99.87%) and sensitivity (99.91%) on the CHB-MIT dataset, using 2-second segments without overlap and validated via 10-fold cross-validation (CV).
- **Naive Bayes** models with 10 geometric features extracted across frequency bands (θ, β, δ, α) demonstrated 94.54% accuracy on the CHB-MIT dataset using 20-second segments with 15-second overlap.
- **Fuzzy k-NN** classifiers applied to GNMF-decomposed SSTFT maps achieved high accuracy (98.99%) and sensitivity (99.27%) on both the CHB-MIT and Bonn datasets.
- **K-NN combined with Random Forests (RF)** classifiers, applied to weighted degree and clustering coefficients, achieved an F1 score of 86.69% on the CHB-MIT and Siena datasets using 4-second segments.
- **SVM** classifiers utilizing kurtosis, skewness, and line length features with PCA dimensionality reduction achieved 96.67% accuracy on the CHB-MIT and Siena datasets with 1-second segments and a 0.5-second overlap.

Deep Learning Approaches

Deep learning has enabled major advancements in seizure detection, as DL models can learn representative features from raw EEG data directly, thereby reducing reliance on manual feature engineering.

- **CNNs in combination with classifiers** like ANN, LR, RF, SVM, GB, k-NN, SGD, and ensembles achieved ensemble accuracies of up to 97% on the CHB-MIT and Bonn datasets with 5-second, non-overlapping segments.
- **BERT-based large language models (LLMs)** reached approximately 77% accuracy on the TUSZ dataset with 1-second segments, indicating the potential for LLM applications in sequence data.
- **Graph-generative neural Networks (GGNs)** trained on the TUH dataset achieved 91% accuracy with a 70-30 train-test split and a 5-second segment length.
- **CNNs combined with RNNs** achieved 96.23% accuracy on the CHB-MIT, Bonn, and Bern-Barcelona datasets with 8-fold cross-validation.
- **Attention-enhanced CNNs** reached 86% accuracy and an F1 score of 81% on the TUH dataset with leave-one-out validation, using 3-second, non-overlapping segments.
- **CNN with CBAM and GRU layers** applied on the CHB-MIT dataset achieved 91.73% accuracy with 88.09% sensitivity using a 30-second segment with a 1-second overlap.
- **Standard CNN architectures** applied on the CHB-MIT dataset reached 97.57% accuracy and 98.90% sensitivity with 5-second segments and 1-second overlap.
- **Scalp Swarm Algorithm (SSA) and LSTM** classifiers achieved 99.2% accuracy on the TUSZ dataset, supported by high sensitivity (98.99%) and specificity (99.01%) using an 80-20 train-test split with 1-second segments.

  Table 1 gives an overview of deep learning and machine learning techniques for EEG-based seizure detection.

Table 1: Deep Learning and Machine Learning Methods for EEG-Based Seizures

Classifier / Model	Features / Architecture	Dataset(s)	Performance Metrics	Validation Method	Segment Length, Overlap	Year	Reference
SVM	Covariance matrices (Riemannian geometry)	CHB-MIT	Acc: 99.87%, Sens: 99.91%, Spec: 99.82%	10-fold CV	2s, no overlap	2022	[13]
Naive Bayes	Geometric features (θ, β, δ, α bands)	CHB-MIT	Acc: 94.54%	10-fold CV	20s, 15s overlap	2022	[14]
Fuzzy k-NN	GNMF-decomposed SSTFT maps	CHB-MIT, Bonn	Acc: 98.99%, Sens: 99.27%, Spec: 98.53%	10-fold CV	1s, no overlap	2023	[15]
k-NN, RF	Weighted degree, clustering coefficient	CHB-MIT, Siena	F1: 86.69%, AUC: 84.33%, Acc: 84.83%, Prec: 85.6%, Sens: 87.81%	5-fold CV	4s	2023	[16]
SVM	Kurtosis, skewness, line length (PCA)	CHB-MIT, Siena	Acc: 96.67%, Spec: 95.62%, Sens: 97.72%	Bootstrap	1s, 0.5s overlap	2023	[17]
Ensembles, CNN, ANN, LR, RF, SVM, GB, k-NN, SGD	Feature extraction CNN	CHB-MIT, Bonn	Ensembles Acc: 97%	10-fold CV	5s, no overlap	2022	[18]
BERT	BERT LLM	TUSZ	Acc: ~77%	-	1s	2022	[19]

Classifier / Model	Features / Architecture	Dataset(s)	Performance Metrics	Validation Method	Segment Length, Overlap	Year	Reference
GGN	Graph-Generative Network	TUH	Acc: 91%	Train-test (70-30)	5s	2022	[20]
CNN and RNN	CNN-RNN combination	CHB-MIT, Bonn, Bern-Barcelona	Acc: 96.23%	8-fold CV	N/A	2023	[21]
CNNs with Attention	Attention mechanism	TUH	Acc: 86%, F1: 81%	LOO	3s, no overlap	2023	[22]
CNN-CBAM, GRU	CNN with CBAM, GRU	CHB-MIT	Acc: 91.73%, Sens: 88.09%, FPR: 0.053/h	10-fold CV	30s, 1s overlap	2023	[23]
CNN	Standard CNN	CHB-MIT	Acc: 97.57%, Sens: 98.90%, FPR: 2.13%	LOO	5s, 1s overlap	2023	[24]
SSA, LSTM	SSA for feature selection, LSTM	TUSZ	Acc: 99.2%, Sens: 98.99%, FDR: 98.43%, F1: 97.54%	Train-test (80-20)	1s	2024	[25]

Effective contributions to EEG-based seizure identification are made by both ML and DL techniques. While DL models give greater accuracy by managing temporal patterns in EEG signals and learning features directly from the data, feature-based machine learning techniques offer interpretability and high performance. The dataset's properties, the intended segment length, and the target accuracy all influence the method selection.

2.1 Preliminaries

Table 2 provides a detailed description of the individual machine-learning models employed in this study. These models, selected for their complementary strengths and proven effectiveness in classification tasks, form the foundational components of our proposed ensemble approach for epileptic seizure detection. Each model's description includes its underlying principles, training methodology, advantages, and potential limitations. This comparative overview facilitates a comprehensive understanding of the individual model contributions within the context of the overall ensemble framework.

Table 2: Characteristics and Properties of Individual Machine Learning Models for Seizure Detection

Model	Description	Steps	Pros	Cons
MDBCN [11]	MDBCN is a deep learning model that improves classification performance, especially in complicated datasets, by fusing multi-scale feature extraction with the concepts of convolutional neural networks (CNNs). It learns the hierarchical representations of input data using a multi-layer architecture.	1. **Data Preparation**: Preprocess the dataset, including normalization and augmentation. 2. **Feature Extraction**: Apply convolutional layers with different kernel sizes to capture multi-scale features. 3. **Pooling Layers**: Use pooling layers to reduce dimensionality while retaining essential features. 4. **Training**: Utilize a suitable optimizer (e.g., Adam) to train the model with a labeled dataset. 5. **Fine-tuning**: Adjust hyperparameters and layers to improve accuracy. 6. **Evaluation**: Use metrics such as accuracy, precision, and recall on a validation dataset to gauge the model's performance.	- Effective at capturing multi-scale features from data. - Improved classification accuracy for complex datasets. - Hierarchical representation learning enhances generalization. - Suitable for various applications, including image and time series classification.	- Computationally intensive and requires significant resources for training. - If improperly regularized, it could be prone to overfitting on small datasets. - Complexity in model design and tuning hyperparameters can be challenging. - needs a lot of labeled data in order to train well.
CNN [12]	Convolutional Neural Network (CNN) One kind of deep learning model that works very well for picture and sequence data processing. It captures spatial hierarchies by applying filters to input data in a grid-like topology.	1. Data Preparation: Preprocess data, including normalization and augmentation for images. 2. Convolutional Layers: To extract spatial characteristics, use convolutional filters. 3. Pooling Layers: Use max or average pooling layers to reduce spatial dimensions. 4. Training: Use optimizers (e.g., Adam) to update weights based on training data. 5. Fine-tuning: Adjust filter sizes, number of layers, and regularization parameters. 6. Evaluation: Use measures like as accuracy and validation data loss to gauge performance.	- Highly effective in image classification and pattern recognition. - Reduced need for extensive preprocessing due to feature extraction layers. - Adaptable to various data types, such as time series and speech.	- Prone to overfitting if not regularized properly. - needs a lot of labeled data in order to train well. - Computationally demanding, especially with deep architectures.
DWT-based DNN [26]	A deep neural network (DNN) that incorporates a Discrete Wavelet Transform (DWT) for pre-processing. The DWT decomposes the EEG signal into different frequency sub-bands, enhancing feature extraction for the subsequent DNN.	1. DWT decomposition of EEG signal 2. Feature extraction from DWT coefficients3. DNN training and evaluation	- Improved feature extraction through multi-resolution analysis - Potentially enhanced performance with noisy data - Reduced computational cost compared to some pure deep learning approaches	- Performance depends on DWT parameters and DNN architecture - May not capture all relevant features effectively

Model	Description	Steps	Pros	Cons
SVM [27]	A popular supervised machine learning approach for classification and regression problems is the Support Vector Machine (SVM). In order to successfully divide the data into discrete categories, it finds the best hyperplane in the feature space that optimizes the gap between classes. When working with non-linear data, SVM may transfer the input data into higher-dimensional spaces using kernel functions. This allows the algorithm to choose the optimum hyperplane to divide the classes in the converted space. Because of its adaptability, SVM works incredibly well for challenging classification issues.	1. **Data Preparation**: Begin by preprocessing the dataset, which includes normalizing the data to ensure consistent scaling and handling missing values to maintain data integrity. 2. **Choosing the Kernel**: Depending on the type of data, choose the right kernel function. A radial basis function (RBF) kernel for more intricate, non-linear patterns, a polynomial kernel for capturing non-linear connections, or a linear kernel for linearly separable data are among the available options. 3. **Training the Model**: Fit the SVM model to the training data, optimizing the hyperplane that maximizes the margin between the different classes. 4. **Parameter Tuning**: Fine-tune hyperparameters like the regularization parameter (C) and the kernel-specific parameter (gamma) to improve model performance. Techniques like GridSearchCV or RandomizedSearchCV can be used to identify the best parameter combinations. 5. **Prediction**: After training the model, use it to make predictions on new or test data, classifying the input into the appropriate categories. 6. **Evaluation**: Evaluate the model's performance in classification tasks using a variety of measures, including accuracy, precision, recall, and F1-score.	1. **Data Preparation**: Preprocess the dataset, including normalization and handling missing values. 2. **Choosing the Kernel**: Select an appropriate kernel function (linear, polynomial, RBF, etc.) based on the data characteristics. 3. **Training the Model**: Fit the SVM model to the training data by optimizing the hyperplane that separates classes. 4. **Parameter Tuning**: Adjust hyperparameters (like C and gamma) using techniques such as GridSearchCV for better performance. 5. **Prediction**: Use the trained model to predict classes on new or test data. 6. **Evaluation**: Evaluate the model's performance with measures like F1-score, recall, accuracy, and precision.	- Computationally intensive, particularly for large datasets. - Performance is heavily dependent on the choice of kernel and parameters. - Less effective on very large datasets or noisy data. - Requires careful tuning of hyperparameters to achieve optimal results.
Random Forest [28]	During training, many decision trees are constructed using the Random Forest ensemble	1. **Data Preparation**: To prepare the data for modeling, start by preprocessing the dataset. This involves encoding categorical variables	**Handles Large Datasets with Higher Dimensionality Well**: Random Forest can effectively manage large and high-	• Slower to Predict: Due to its ensemble nature, Random Forest can be slower at making

Model	Description	Steps	Pros	Cons
	learning approach, which then aggregates the output to generate predictions. It determines the mean forecast of all the trees for regression tasks and the modal (most common class) of each tree's predictions for classification tasks. The technique makes use of bagging (bootstrap aggregating), in which a random subset of the data, drawn with replacement, is used to train each tree. By lowering variance and avoiding overfitting, this method increases model resilience and improves prediction accuracy.	using methods like one-hot encoding or label encoding and addressing missing values using imputation or removal. 2. **Bootstrap Sampling**: Create many subsets of the original dataset using replacement sampling and random sampling. To ensure variation among the trees in the forest, a decision tree will be trained from each subgroup 3. **Building Decision Trees**: For each subset, construct a decision tree by selecting a random subset of features at each split, helping to reduce correlation between trees and increasing model robustness. 4. **Aggregating Results**: For classification tasks, combine the predictions of all trees by taking a majority vote. For regression tasks, aggregate the predictions by calculating the average value from all the trees. 5. **Evaluation**: Use measures like accuracy, precision, recall, and F1-score to evaluate the model's performance on a test dataset. These metrics assist determine how effectively the model generalizes to data that hasn't been seen before.	dimensional datasets due to its ensemble approach, which processes multiple decision trees in parallel and leverages a variety of features. **Reduces the Risk of Overfitting**: By averaging the predictions of multiple trees, Random Forest reduces the likelihood of overfitting, which is a common issue with individual decision trees that may model noise in the data. **Provides Feature Importance**: By identifying the most significant variables in the dataset through feature significance scores, Random Forest provides insightful information that can be helpful for feature selection and model interpretation . **Robust to Noise and Handles Missing Values Well**: Random Forest is resistant to noise, as it relies on multiple trees and aggregates their predictions. It also has built-in mechanisms for handling missing data, such as using surrogate splits when certain features are unavailable.	predictions compared to individual decision trees, as it requires aggregating the outputs of multiple trees for each prediction. • Requires More processing Resources: In contexts with limited resources, the model may be a constraint due to its higher processing power and memory requirements, especially when utilizing a large number of trees. • Less Interpretable: Unlike a single decision tree, which provides a clear and understandable decision path, Random Forest is more complex and harder to interpret, making it less transparent and more difficult to explain the reasoning behind its predictions. • Performance Degradation with Too Many Trees: While adding more trees can improve the model's performance to a certain extent, beyond a certain point, the performance can plateau or even degrade, leading to diminishing returns in terms of prediction accuracy.
XGB **Classifier [29]**	Extreme Gradient Boosting, or XGBoost, is a fast and efficient gradient boosting framework implementation that is very scalable and adaptable. Decision trees are constructed in a sequential fashion using this ensemble learning	**Data Preparation**: To prepare the data for the model, start by preprocessing the dataset. This involves encoding categorical variables using techniques like one-hot encoding or label encoding and addressing missing values using imputation or removal. . **Feature Selection**: Select the most relevant	• **High Predictive Accuracy and Performance**: XGBoost offers excellent predictive accuracy due to its ensemble approach and advanced optimization techniques, making it a top choice for many machine learning tasks. • **Regularization Capabilities**: It has built-in regularization (L1 and L2) that penalizes	• **Complex to Tune**: XGBoost has a large number of hyperparameters that need to be carefully tuned, which can make the model optimization process complex and time-consuming. • **Interpretability Challenges**: It can be challenging to interpret the

Model	Description	Steps	Pros	Cons
	technique, with each new tree aiming to fix the mistakes of the one before it. Over time, this iterative process increases the predicted accuracy of the model. Because of its great success in a range of machine learning competitions and real-world applications, as well as its high efficiency and resilience, XGBoost is frequently employed for both classification and regression problems.	features for model training. While XGBoost can handle large feature sets, choosing the right features can improve model performance, reduce computation time, and avoid overfitting. **Model Initialization**: Establish the model's hyperparameters, including the number of estimators (trees), maximum tree depth, learning rate, and other factors like regularization terms to prevent bias and overfitting. **Training**: Iteratively construct decision trees to fit the model to the training data. To increase prediction accuracy, each new tree is trained to fix the mistakes caused by the ones that came before it. The model is then adjusted depending on the residual errors. **Evaluation**: Use measures like accuracy, precision, recall, and F1-score to assess the model's performance on a test dataset after training. These measures aid in gauging the model's efficacy in classification tasks as well as its ability to generalize to new data.	too complicated models, reducing overfitting and improving generalization. • **Handles Missing Data Internally**: XGBoost can handle missing values during training without requiring explicit data imputation, simplifying the preprocessing step. • **Highly Efficient in Terms of Speed and Memory Usage**: XGBoost is designed for efficiency, with fast training times and optimized memory usage, especially when working with large datasets. • **Supports Parallel and Distributed Computing**: Because XGBoost facilitates distributed computing and parallel processing, training is accelerated, making it appropriate for complicated models and huge datasets.	model due to its ensemble structure and decision tree complexity, which makes it more difficult to explain the model's conclusions than simpler models like logistic regression or decision trees. • **Higher Computational Resources**: Compared to more straightforward models (like logistic regression), XGBoost uses more memory and processing resources, particularly when working with big datasets or a lot of trees. • **Sensitive to Noisy Data and Outliers**: Because outliers and noisy data might impair the model's performance, XGBoost may be susceptible to them. To lessen these problems, proper data preparation and outlier treatment are essential.

3. Methodology

3.1 Dataset (1) Description

A pre-processed and restructured version of a popular epileptic seizure detection dataset that is accessible on Kaggle is utilized in this work [https://www.kaggle.com/datasets/harunshimanto/epileptic-seizure-recognition/data] [30]. The original dataset includes 500 participants' EEG recordings and was provided from the UCI Machine Learning Repository. Each 23.6-second recording is split into 4097 data points that represent the EEG signal's amplitude at different periods in time.

The original data is organized into five folders, each containing 100 files corresponding to individual subjects. Each file represents a 23.6-second EEG recording with 4097 data points. These recordings have been pre-processed for this study by segmenting and shuffling the data.

Each 4097-data point sequence is divided into 23 chunks, each 178 data points long, representing one second of brain activity. This results in 11,500 (23 x 500) one-second EEG segments. Each segment is labeled according to the following scheme:

1: Seizure activity
2: EEG data obtained from the region of the brain containing the tumor
3: EEG from a healthy part of the brain in tumor-bearing patients
4: Eyes closed
5: Eyes open

This study employs binary classification, differentiating between seizure activity (class 1) and non-seizure activity (classes 2–5), although the original dataset included five classes. While tackling the fundamental issue of seizure detection, our binary classification method streamlines the analysis and is consistent with the standard practice in the literature. Applying different machine-learning models is made easier by the restructure, which makes data access and manipulation simpler. The response variable (y, in column 179) shows the class name, whereas the explanatory variables (X1 to X178) show the EEG signal levels inside each one-second segment. This dataset and its pre-processing are based on the work of Andrzejak et al. (2001) [31], who investigated nonlinear deterministic structures in brain electrical activity. Their research highlighted the dependence of these structures on the recording region and brain state, providing a valuable foundation for EEG-based seizure detection studies.

Figure 2 represents a decision tree visualization that depicts the structure and flow of a classification or regression model. The tree begins at the root node, displayed at the top, which represents the first decision point based on a specific feature and its threshold value. As the tree branches out, each node represents further splits based on feature thresholds, dividing the dataset into subsets with shared characteristics.

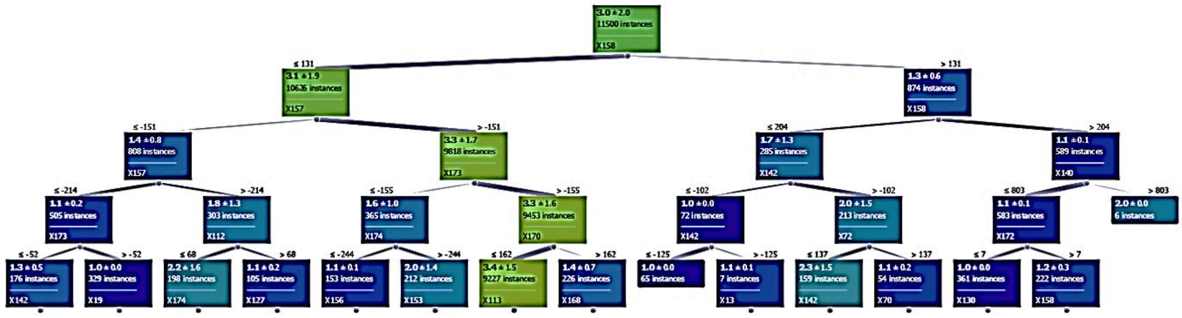


Figure 2 shows the correlation between the dataset features.

3.2 Dataset (2) Description

The dataset, which includes 500 people's EEG records, was made available by Bonn University's Epileptology Department. There are 4096 uniformly spaced data points (sampled every 0.0057 seconds) in each 23.5-second recording. These recordings are classified into one of five labeled classes, as described below:

- **Set A - Class 4**: EEG recording of an awake patient without epilepsy who has their eyes open.
- **Set B - Class 3**: EEG recording of an awake, non-epileptic subject with both eyes open and closed.
- **Set C - Class 2**: Electrodes inserted in the brain's epileptogenic zone are used to record an epileptic patient's EEG during a seizure-free interval.
- **Set D - Class 1**: EEG recording from the hippocampus formation of the opposite hemisphere of the brain from Set C of an epileptic patient during a seizure-free period.
- **Set E - Class 0**: EEG recording of a patient having an epileptic seizure in progress.

These diverse classes provide a comprehensive dataset for analyzing and classifying EEG signals associated with both epileptic and non-epileptic states.

Set A - Class 4: EEG recording of an awake patient without epilepsy who has their eyes open

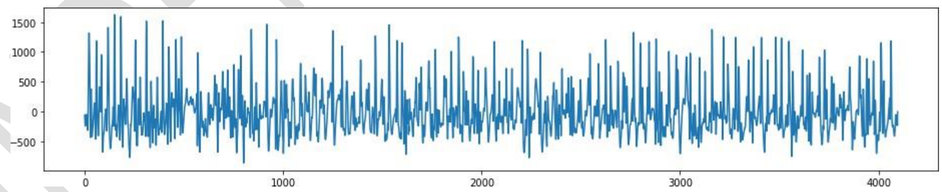


Figure 3: Example of Class 4 EEG [32].

Set B - Class 3: EEG recording of an awake, non-epileptic subject with both eyes open and closed

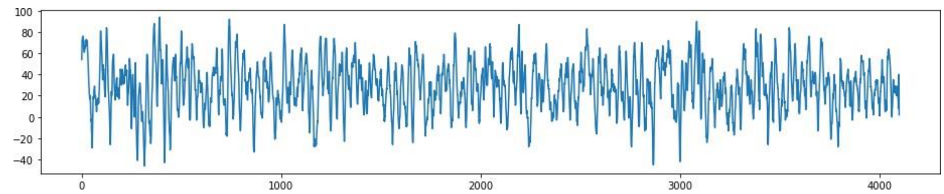


Figure 4: Example of Class 3 EEG [32].

Set C - Class 2: Electrodes inserted in the brain's epileptogenic zone are used to record an epileptic patient's EEG during a seizure-free interval.

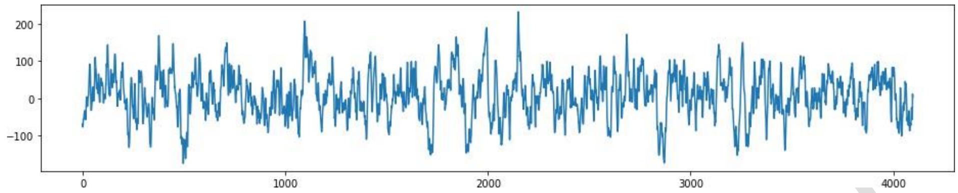


Figure 5: Example of Class 2 EEG [32].

Set D - Class 1: EEG recording from the hippocampus formation of the opposite hemisphere of the brain from Set C of an epileptic patient during a seizure-free period.

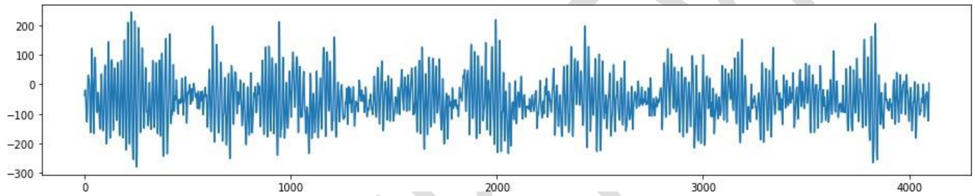


Figure 6: Example of Class 1 EEG [32].

Set E - Class 0: EEG recording of a patient having an epileptic seizure in progress

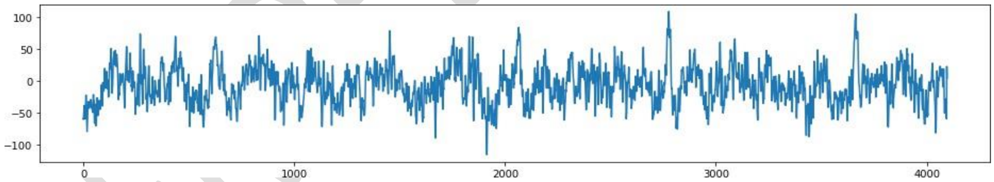


Figure 7: Example of Class 0 EEG [32].

The dataset is balanced and clean, with 100 recordings per category. Since EEG patterns are subtle and crucial for identifying epileptic activity, no modifications, over-sampling, or synthetic data augmentation were applied to maintain data integrity.

The dataset is publicly accessible through the Department of Epileptology at Bonn University at this link: https://tinyurl.com/yylxbzfj. All data points have been pre-processed, aggregated, and labeled in the file `all_data_epileptic_seizures.csv`, available on GitHub: (https://github.com/jkuypers93/LSTM-Epileptic-Seizure-Recognition) [32].

3.3 Data Pre-processing Phase

To get the EEG dataset ready for precise and trustworthy seizure detection modeling, important procedures are carried out during the data pre-processing phase. In this stage, data integrity is guaranteed, class labeling is optimized, and the data is organized for efficient training and assessment. The following is an overview of the steps:

I. Handling Missing Values:

- Challenge: Missing data can compromise model accuracy and reliability.
- Solution: Either remove entries with missing values or apply imputation techniques to fill them with appropriate values, ensuring data consistency and completeness.

II. Class Labeling:

- Binary Classification Approach: Although the original dataset contains five classes, this study focuses on binary classification, distinguishing between seizure activity (class 1) and non-seizure activity (classes 2-5). This binary restructuring is widely used in literature, simplifying the analysis and zeroing in on the key objective of seizure detection.
- Explanatory and Response Variables: The EEG signal values within each one-second segment are represented by explanatory variables (X1 to X178), and the response variable (y, in column 179) indicates the class label.
- Target Variable Labeling: The target variable, "Diagnosis," is set as binary, with "0" indicating no seizure (unaffected) and "1" indicating seizure presence (affected), enabling clear and accurate seizure event classification.

III. Data Splitting and Evaluation:

- Training Set (80%): The model is trained on 80% of the dataset, learning patterns, and relationships to minimize errors and enhance predictive accuracy.
- Validation Set (20%): The remaining 20% is used to assess model generalization to new data, identifying any overfitting and ensuring real-world applicability. Validation is further supported by 5-fold cross-validation, reinforcing the reliability of the model's performance.

3.4 Model Architecture (Voting Classifier)

A voting classifier is a type of ensemble machine learning technique that enhances overall performance by combining the predictions of several models. To arrive at a final judgment, it aggregates the predictions from several classifiers (such as logistic regression, support vector machines, and decision trees). [33].

Three primary categories of voting classifiers exist:

1. **Hard Voting:** A class is selected based on the number of votes each model casts for it.

How it Works:

A class label (such as "Class 1" or "Class 0") is predicted by each model in the ensemble. The final forecast is made for the class that receives the most votes from the models. A random class is chosen from among the tied classes in the event of a tie (i.e., an equal number of votes for several classes).

The majority class that each individual model in the ensemble predicts determines the final prediction in a hard voting process. The mathematical expression for hard voting can be written as:

$$\hat{y} = \arg\max_{c \in C} \sum_{i=1}^{N} \mathrm{II}(h_i(\mathrm{x}) = \mathrm{c}) \quad (1)$$

Where:

- $\hat{y}$: Final predicted class.
- C: Set of all possible classes.
- N: Total number of classifiers.
- h_i: Prediction of the i-th classifier for input x.
- II: Indicator function, which is 1 if $h_i(\mathrm{x}) = \mathrm{c}$, otherwise 0.

2. Soft Voting: The class chosen is the one with the highest average projected probability across all models.
How it Works:
The likelihood that an instance will belong to each class is predicted by each model.
All models' probabilities are averaged for each class.
The class with the highest average probability is picked as the final forecast.
Each model produces the probability distribution across classes in soft voting, and the average of these probabilities serves as the basis for the final forecast. The class chosen is the one with the highest average probability. The mathematical equation for soft voting is:

$$\hat{y} = \arg\max_{c \epsilon C} \frac{1}{N} \sum_{i=1}^{N} P_i\,(c|x) \qquad (2)$$

Where:

- $P_i(c|x)$: Probability of class ccc predicted by the i-th classifier for input x
- $\hat{y}$: Final predicted class.

Key Points:
- Advantages: Improves accuracy, reduces overfitting, and is easy to implement.
- Disadvantages: Requires diverse models, can be computationally expensive, and may not outperform more advanced techniques like boosting.

3. Weighted Voting
3.1. Weighted Soft Voting
In **weighted soft voting**, classifiers are assigned weights ω_i based on their importance or accuracy. The final prediction is determined by the weighted average of probabilities.
Equation:

$$\hat{y} = \arg\max_{c \epsilon C} \frac{\sum_{i=1}^{N} \omega_i \cdot P_i(c|x)}{\sum_{i=1}^{N} \omega_i} \qquad (3)$$

Where:

- ω_i: Weight assigned to the i-th classifier.

3.2. Weighted Hard Voting
In **weighted hard voting**, classifiers cast votes that are weighted by their importance.
Equation:

$$\hat{y} = \arg\max_{c \epsilon C} \sum_{i=1}^{N} \omega_i \cdot \mathrm{II}(h_i(x) = \mathrm{c}) \qquad (4)$$

Voting Classifiers are effective for both classification and regression tasks, especially when combining diverse models.

The Voting Classifier offers several benefits, including improved accuracy by combining predictions from multiple models, which reduces overfitting and increases resilience to noise. Its flexibility allows it to handle both classification and regression tasks, and it's easy to deploy. However, its effectiveness depends on the diversity of the base models, as a lack of variety can reduce performance. It may not always outperform more advanced methods like boosting or stacking, and hard voting can lead to suboptimal decisions by ignoring model confidence. Additionally, it's computationally expensive and can be complex to tune. Despite these drawbacks, the Voting Classifier is a practical and balanced option for many machine learning tasks, especially for reducing overfitting and handling diverse datasets.

3.5 Model Architecture

The proposed architecture for epileptic seizure detection leverages a combination of deep learning techniques and ensemble methods to achieve robust and accurate results as shown in **Figure 8.**

Data preprocessing is a crucial initial step to ensure data quality. This process involves removing null cells, normalizing the data, and balancing the dataset to address any class imbalances. After preprocessing, the data is labeled as either "seizure" or "non-seizure" to facilitate the classification task. There are several different models used to extract pertinent characteristics from the EEG data. Complex spatiotemporal patterns in the EEG data are captured using Multi-Dimensional Bayesian Convolutional Networks (MDBCN), which makes it possible to identify subtle seizure features. Deep neural networks based on discrete wavelet transform (DWT-DNN) break down EEG data into distinct frequency sub-bands, enabling the extraction of characteristics from both high- and low-frequency components that are essential for precise seizure identification. In order to detect localized patterns linked to seizures, it is crucial to identify spatial elements in the EEG data using Convolutional Neural Networks (CNN).

To further examine the collected features for classification, strong algorithms like Random Forest, XGBoost, and Support Vector Machine (SVM) are employed. These models can increase seizure detection accuracy and handle high-dimensional, complicated data. A Voting Classifier is used in order to improve the system's overall performance and resilience. This ensemble approach improves generality by combining the predictions of several models, so lessening the influence of mistakes from any one model. Training and testing sets make up the dataset. To determine the underlying patterns in the EEG signals, the models are trained using the training data. The testing set is used to assess the models' generalizability to new data. Several criteria are used to evaluate the system's performance, such as accuracy, precision, recall, F1 score, time required, and sensitivity, which give a thorough picture of how well the system detects seizures.

By integrating these techniques and employing a rigorous evaluation framework, this architecture aims to provide accurate and reliable seizure detection, enabling early diagnosis and intervention for individuals with epilepsy.

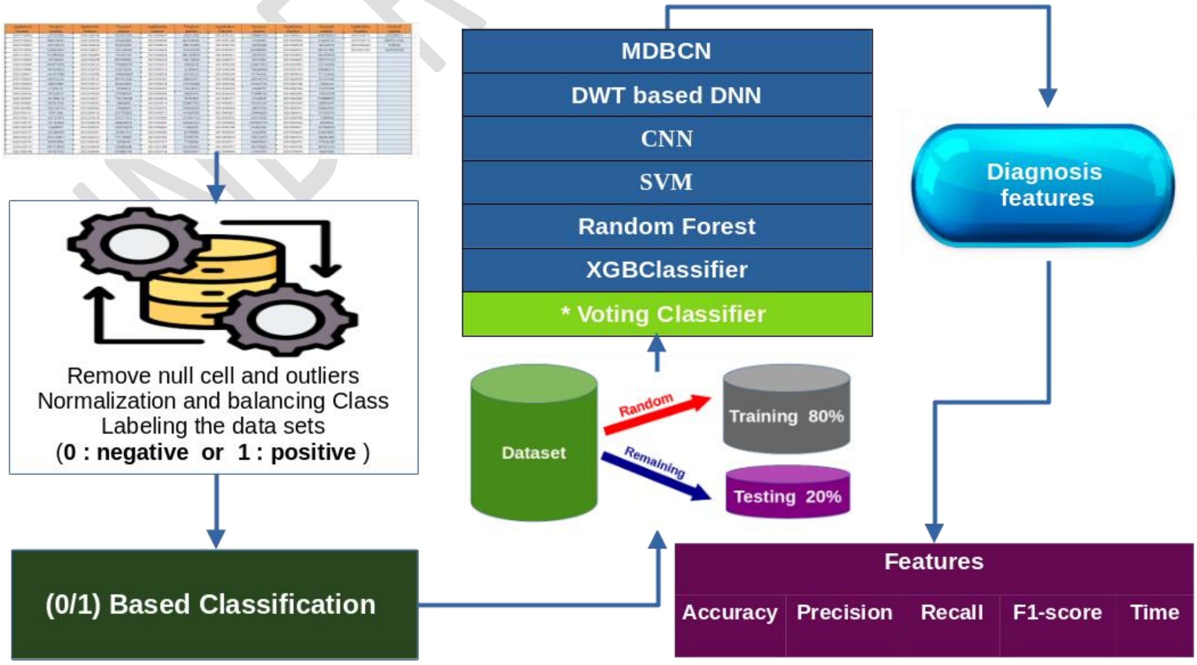


Figure 8: Seizure Detection Framework.

Figure 9 illustrates the comprehensive workflow followed in this binary classification task to detect seizures from the dataset. The process begins with setting up the environment, including fixing the random seed for reproducibility and suppressing warnings. The dataset is then loaded and preprocessed by separating the features and target variable while converting the target into binary classes: Seizure (1) and No Seizure (0).

Following data preprocessing, the class distribution and balance are shown using a bar chart. To create a soft-voting ensemble classifier, several machine learning models are used, such as Random Forest, Logistic Regression, and Extra Trees classifiers. After assessing the model's performance through cross-validation, the entire dataset is used for training and prediction generation. Using sensitivity analysis and a classification report, the model's performance is evaluated. Calculations are made for important measures including sensitivity, F1-score, recall, and accuracy. Sensitivity is the ratio of true positives to the sum of false negatives and true positives. Plotting the confusion matrix and ROC curve allows for a more thorough analysis of the model's performance.

Finally, the execution time for cross-validation is recorded, providing an overview of the time efficiency of the process. This figure encapsulates the steps taken from data loading and preprocessing to model evaluation and performance visualization**.**

1. Set Up Environment:
  - Set random seed for reproducibility (for numpy and random).
  - Suppress warnings to avoid unnecessary outputs.

2. Load Dataset:
  - Read dataset from 'data_1.csv' into a DataFrame.
  - Drop the 'Unnamed' column from the DataFrame.
  - Split the DataFrame into features (X) and target variable (y).

3. Preprocess Data:
  - Convert target variable (y) to binary classes: 1 for Seizure and 0 for No Seizure.

4. Visualize Data:
  - Count the occurrences of Seizure and No Seizure classes.
  - Plot the distribution of the Seizure vs. No Seizure classes using a bar chart.

5. Set Up Models:
  - Initialize a Logistic Regression model.
  - Initialize a Random Forest Classifier.
  - Initialize an Extra Trees Classifier.
  - Combine all models into a Voting Classifier using soft voting.

6. Cross-Validation Setup:
  - Set up Stratified K-Fold Cross-Validation with 5 splits.

7. Evaluate Models Using Cross-Validation:
  - Start the timer.
  - Perform cross-validation using the Voting Classifier.
  - Calculate accuracy scores and print the results:
    - Print individual accuracy scores.
    - Print the mean and standard deviation of the accuracy scores.
  - Measure the time taken for cross-validation.

8. Train on Full Dataset:
  - Fit the Voting Classifier on the full dataset (X and y_binary).
  - Make predictions using the trained model.

9.. Generate Classification Report:
  - Print the classification report (precision, recall, F1-score, etc.) using the Voting Classifier.

10. ROC Curve Plotting:
  - Define a function to draw the ROC curve:

- Calculate False Positive Rate (FPR) and True Positive Rate (TPR).
- Compute the Area Under the Curve (AUC).
- Plot the ROC curve with labels and AUC value.

11. Confusion Matrix Plotting:
- Define a function to plot the confusion matrix:
- Compute the confusion matrix.
- Plot it as a heatmap with annotations for each cell.
12. Plot ROC and Confusion Matrix:
- Get predicted probabilities from the Voting Classifier.
- Call the function to plot the ROC curve.
- Call the function to plot the confusion matrix.
13. Print Execution Time:
- Print the time taken for cross-validation

Figure 9: Overview of the Binary Classification Workflow for Seizure Detection.

3.6 Performance Metrics

The performance of each model is evaluated using several key metrics to gain insights into its classification capabilities, particularly in scenarios with imbalanced datasets. These metrics provide a deeper understanding of the model's strengths and weaknesses, going beyond a simple accuracy score.

1. **Confusion Matrix:**
   - The confusion matrix offers a detailed summary of the model's predictions by categorizing them into four key components:
     - **True Positives (TP):** Malignant cases correctly classified as "Malignant."
     - **False Positives (FP):** Benign cases incorrectly classified as "Malignant."
     - **True Negatives (TN):** Benign cases correctly classified as "Benign."
     - **False Negatives (FN):** Malignant cases incorrectly classified as "Benign."
   - This tool highlights the model's misclassifications and provides a comprehensive breakdown of its performance, aiding in pinpointing specific areas for improvement.

2. Accuracy:

Accuracy [34] represents the overall proportion of correct predictions and is calculated as:

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

Although accuracy provides a broad sense of model performance, it might be deceptive in cases when the dataset is unbalanced since it fails to take the severity of misclassifications into consideration.

3. Precision:

Precision [35] measures the proportion of correct positive predictions among all instances predicted as positive:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

A high accuracy means a low false positive rate, which is important when false positives (like misdiagnosing cancerous patients) are expensive.

4. Recall (Sensitivity or True Positive Rate):

Recall [36] [37]evaluates the model's ability to correctly identify all actual positive instances. It is calculated as:

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

High recall is essential when it's critical to find as many positive examples as possible, such in medical diagnosis, where failing to identify a malignant case (false negative) might have detrimental effects.

5. F1 Score (F-measure):

The F1 score [38] [39] is the harmonic mean of Precision and Recall, offering a balanced metric that considers both false positives and false negatives. It is calculated as:

$$\text{F1} - \text{score} = 2 * \frac{(\text{Precision} \times \text{Recall})}{(\text{Precision} + \text{Recall})}$$

Due to its ability to balance precision and recall, the F1 score is particularly useful when there is an unequal distribution of classes. In situations where reducing false positives and false negatives is equally crucial, this statistic is essential for a more impartial assessment of the model's efficacy.

4 Results and Analysis

The outcomes of our tests to assess the suggested model's performance are shown in this section. A 3 GHz Intel CPU with 4 GB of RAM and 64-bit Windows 10 were part of the experimental configuration. Python was used to implement each experiment. We present a thorough examination of the outcomes of tests carried out on two different datasets. Evaluating and contrasting how well different machine learning models classified seizure and non-seizure events was the main objective. Key performance indicators including accuracy, precision, recall, F1-score, and execution time were used to evaluate performance.

4.1 Results and Analysis (Dataset 1)

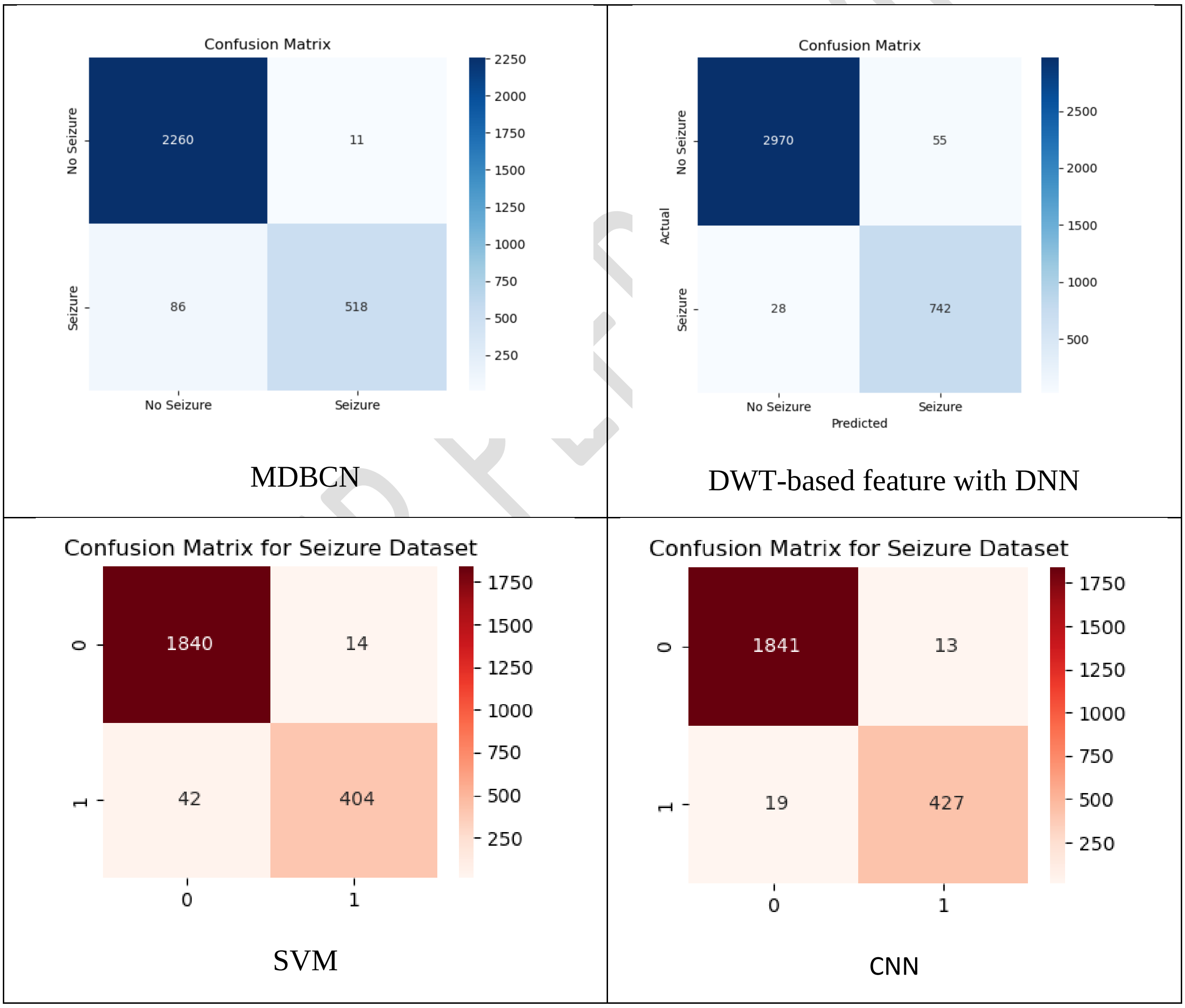
The first dataset reveals the performance of seven different models: Multi-Dense Block Concatenation Network (MDBCN), DWT-based feature with DNN, CNN, SVM, Random Forest, XGBClassifier, and the proposed Voting Classifier. The results are summarized in **Table 3.**

Table 3: Performance Metrics of Various Machine Learning and Deep Learning Models on Seizure Detection for Dataset (1).

Model	Accuracy	Precision	Recall / Sensitivity	F1-Score	Time (seconds)
Multi-Dense Block Concatenation Network (MDBCN)	0.97	0.97 (No Seizure) / 0.98 (Seizure)	1.00 (No Seizure) / 0.88 (Seizure)	0.98 (No Seizure) / 0.93 (Seizure)	351.10
DWT-based feature with DNN	0.98	0.99 (No Seizure) / 0.93 (Seizure)	0.98 (No Seizure) / 0.97 (Seizure)	0.99 (No Seizure) / 0.95 (Seizure)	12.40
CNN	0.99	0.99 (No Seizure) / 0.98 (Seizure)	1.00 (No Seizure) / 0.95 (Seizure)	0.99 (No Seizure) / 0.97 (Seizure)	531.76
SVM	0.98	0.98 (No Seizure)	0.99 (No Seizure)	0.99 (No Seizure)	11.23

Model	Accuracy	Precision	Recall / Sensitivity	F1-Score	Time (seconds)
		/ 0.97 (Seizure)	/ 0.91 (Seizure)	/ 0.94 (Seizure)	
Random Forest	0.98	0.99 (No Seizure) / 0.95 (Seizure)	0.99 (No Seizure) / 0.95 (Seizure)	0.99 (No Seizure) / 0.95 (Seizure)	19.96
XGBClassifier	0.98	0.98 (No Seizure) / 0.97 (Seizure)	0.99 (No Seizure) / 0.91 (Seizure)	0.99 (No Seizure) / 0.94 (Seizure)	4.24
Voting Classifier (Proposed)	1.00	1.00 / 1.00	1.00 / 1.00	1.00 / 1.00	9.63

The confusion matrices for all the models and techniques used in the study are shown in **Figure 10** for Dataset (1). This figure provides a detailed comparison of the performance of each model. Each confusion matrix illustrates the model's ability to correctly classify data points into their respective categories, providing insights into their accuracy, sensitivity, and overall effectiveness on the first dataset used in the analysis.



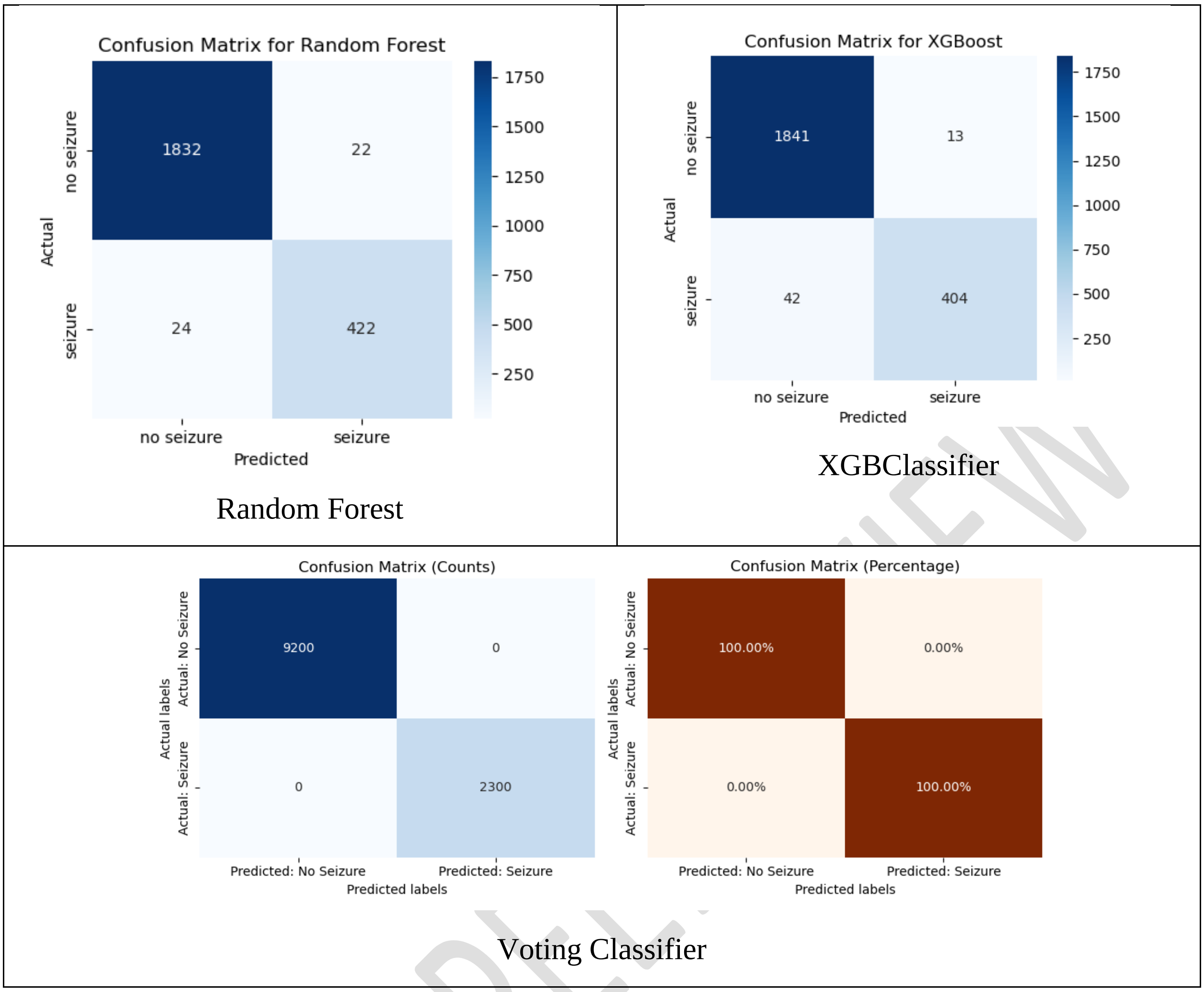


Figure 10: Confusion Matrices for All Models and Techniques on Dataset (1).

Key Observations for Dataset (1):

- The **Voting Classifier (Proposed)** performs best across all metrics (accuracy, precision, recall, and F1-score) for both seizure and no seizure predictions, achieving perfect scores of 1.00 in all categories. It also has a reasonable computation time of 9.63 seconds.
- **CNN** stands out with high accuracy (0.99) and overall good performance but at the cost of significantly higher computation time (531.76 seconds).
- **DWT-based features with DNN** and **SVM** have comparably good performance metrics but with much faster computation times, 12.40 seconds and 11.23 seconds respectively, making them efficient choices.
- **MDBCN** has good precision and F1-scores but lower recall for seizures (0.88) and a higher computation time of 351.10 seconds.
- Both **Random Forest** and **XGBClassifier** show good efficiency with competitive performance metrics and relatively short processing times.

In conclusion, while all models show strong performance, the **Voting Classifier** outperforms others in accuracy, recall, precision, and F1-score across both classes and does so with a relatively short computation time.

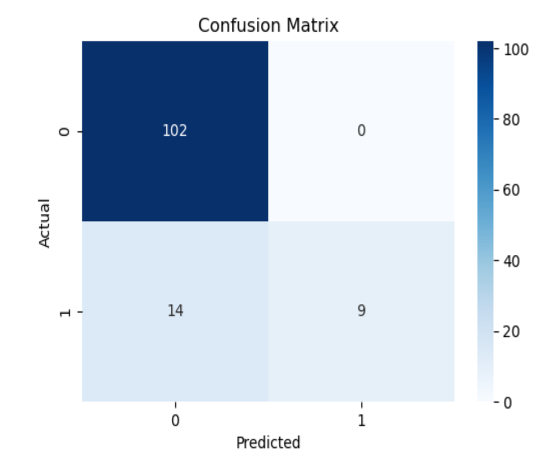
4.2 Results and Analysis (Dataset 2)

The second dataset provides a comparative analysis of the same models under different conditions. The metrics evaluated include accuracy, precision, recall, F1-score, execution time, and sensitivity. The results are presented in **Table 4**.

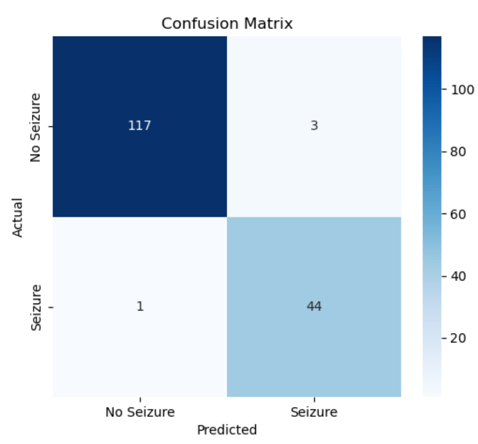
Table 4: Performance Metrics of Various Machine Learning and Deep Learning Models on Seizure Detection for Dataset (2).

Model	Accuracy	Precision	Recall / Sensitivity	F1-Score	Time (seconds)
Multi-DenseBlock Concatenation Network (MDBCN)	0.88	0.88 (No Seizure) / 1.00 (Seizure)	1.00 (No Seizure) / 0.39 (Seizure)	0.94 (No Seizure) / 0.56 (Seizure)	49.76
DWT-based feature with DNN	0.98	0.99 (No Seizure) / 0.94 (Seizure)	0.97 (No Seizure) / 0.98 (Seizure)	0.98 (No Seizure) / 0.96 (Seizure)	0.62
CNN	0.96	0.96 (No Seizure) / 0.93 (Seizure)	0.99 (No Seizure) / 0.82 (Seizure)	0.98 (No Seizure) / 0.88 (Seizure)	407.40
SVM	0.95	0.96 (No Seizure) / 0.88 (Seizure)	0.98 (No Seizure) / 0.82 (Seizure)	0.97 (No Seizure) / 0.85 (Seizure)	0.68
Random Forest	0.96	0.96 (No Seizure) / 0.93 (Seizure)	0.99 (No Seizure) / 0.82 (Seizure)	0.98 (No Seizure) / 0.88 (Seizure)	1.11
XGBClassifier	0.94	0.96 (No Seizure) / 0.82 (Seizure)	0.96 (No Seizure) / 0.82 (Seizure)	0.96 (No Seizure) / 0.82 (Seizure)	6.04
Voting Classifier (Proposed)	1.00	1.00 / 1.00	1.00 / 1.00	1.00 / 1.00	15.00

Figure 11 presents the confusion matrices for each model evaluated on Dataset (2), providing a visual representation of their performance. These matrices illustrate the ability of each model to correctly classify data points into their respective categories, and facilitate a comprehensive comparison of their: Accuracy, Sensitivity, and Overall effectiveness.



MDBCN



DWT-based feature with DNN

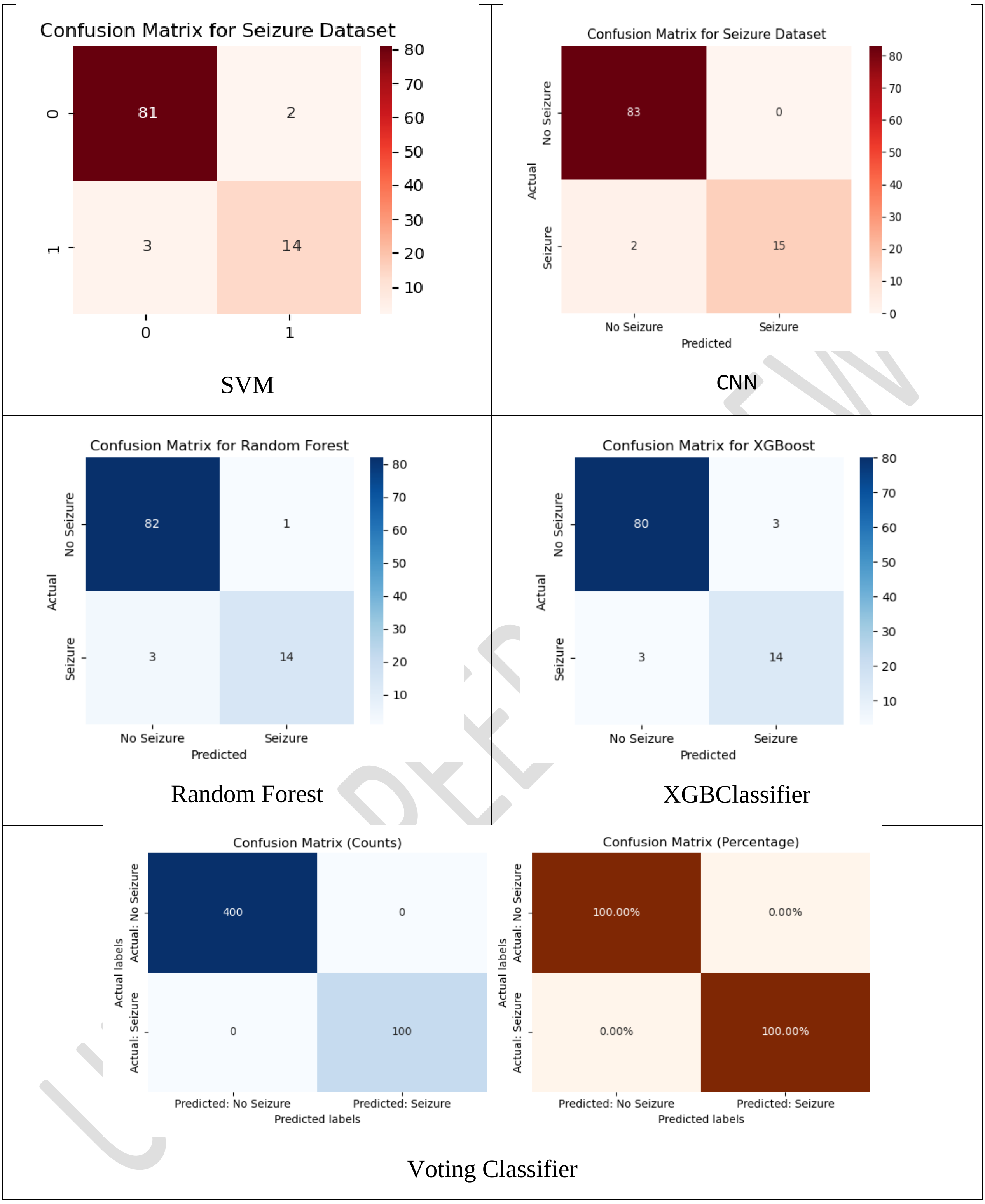


Figure 11: Model Performance Comparison using Confusion Matrices Dataset (2).

Key Observations for Dataset (2):

- The **Voting Classifier (Proposed)** emerges as the clear winner with perfect scores of 1.00 across all metrics (accuracy, precision, recall, and F1-score) for both seizure and no seizure predictions, with a moderate computation time of 15.00 seconds.

- **DWT-based feature with DNN** also performs exceptionally well with high metrics across the board (Precision, Recall, and F1-score), demonstrating very efficient computation time at just 0.62 seconds.
- **CNN** has high accuracy (0.96) and good performance metrics but requires significantly more computation time (407.40 seconds), which could be a limitation.
- **SVM** and **Random Forest** show competitive performance with high accuracy and good precision, recall, and F1-scores, and they both have relatively low computation times (0.68 and 1.11 seconds respectively).
- **XGBClassifier** has lower performance compared to others in this list but still maintains a solid 0.94 accuracy, with balanced precision, recall, and F1-scores, and a moderate computation time (6.04 seconds).
- **MDBCN** has the lowest accuracy (0.88) and performs poorly in recall and F1-score for seizure predictions, although its precision for seizures is high (1.00). The computation time is relatively high at 49.76 seconds, indicating inefficiency.

In conclusion, the **Voting Classifier** outshines others with its perfect scores in all categories, making it the best choice among the listed models, followed by the **DWT-based feature with DNN** for its strong performance coupled with extremely efficient computation time.

5 Discussion and Limitations

The quest for accurate and reliable epilepsy diagnosis through EEG signal analysis has spurred a wealth of research employing a diverse array of methodologies. Several studies have leveraged frequency domain techniques, such as the Fourier transform, to extract diagnostically relevant features. Tzallas et al. [40] demonstrated the efficacy of a Fourier-based approach combined with an artificial neural network (ANN) for epilepsy classification, highlighting the significance of fractional energy features. Similarly, Peker et al. [41] explored the utility of the dual-tree complex wavelet transform, demonstrating the effectiveness of this method coupled with complex-valued neural networks in differentiating epileptic patients based on statistical features derived from the wavelet coefficients.

Time-frequency (TF) analysis has also emerged as a powerful tool in epilepsy research. Alcin et al. [42] combined the Grey-Level Co-occurrence Matrix (GLCM) texture descriptor with Fisher vector encoding to extract features from TF images of EEG signals, achieving superior diagnostic performance. Li et al. [43] further refined TF analysis by developing a multiscale radial basis function method to generate high-resolution TF images, followed by GLCM feature extraction and Fisher vector encoding based on frequency sub-bands. These studies underscore the value of capturing both temporal and spectral information in EEG data for enhanced epilepsy diagnosis.

Wavelet transforms, particularly the discrete wavelet transform (DWT) and stationary wavelet transform (SWT), have been widely employed for their ability to provide multi-resolution analysis of EEG signals. Sharmila [44] successfully implemented a DWT-based framework with linear and nonlinear classifiers for seizure detection in both normal and epileptic individuals. Islamet et al. [45] demonstrated the promising performance of an SWT algorithm for seizure detection. Furthermore, Hassan et al. [46] proposed a system utilizing the tunable wavelet transform and bagging techniques, achieving encouraging results in epilepsy diagnosis. The versatility of wavelet transforms in capturing both transient and sustained features in EEG signals makes them a valuable tool in this domain.

Beyond signal processing techniques, researchers have explored various feature extraction and classification methods. Wang et al. [47] employed coherence analysis to extract information flow features from EEG signals, demonstrating their utility for seizure detection. Jaiswal et al. [48] introduced novel feature extraction methods based on sub-patterns and correlations of Principal Component Analysis (PCA), coupled with a Support Vector Machine (SVM) classifier. Yuan et al. [49] proposed a weighted extreme learning machine (ELM) method utilizing wavelet packet analysis and time series complexity features, achieving accurate classification.

Ensemble methods, such as random forests, have also been investigated to enhance diagnostic accuracy. Raghu et al. [50] demonstrated the efficacy of combining DWT-derived features with a random forest classifier for epileptic classification. Mursalin et al. [51] further explored the application of random forests by employing an improved correlation feature selection technique to identify crucial features from the time, frequency, and entropy domains of EEG signals. Optimization techniques, such as genetic algorithms and particle swarm optimization, have also been applied to fine-tune classifier parameters, as demonstrated by Subasi et al. [52] in their hybrid SVM approach. Finally, Chen et al. [53] focused on characterizing the dynamic behavior of EEG signals using the autoregressive average method, emphasizing the importance of time series characteristics in epilepsy diagnosis.

The literature reveals a diverse landscape of approaches for EEG-based epilepsy diagnosis, encompassing a range of signal-processing techniques, feature extraction methods, and classification algorithms. This breadth of research reflects the complexity of epilepsy and the ongoing efforts to develop more accurate and reliable diagnostic tools. From frequency domain analysis to time-frequency representations and wavelet decompositions, researchers continue to explore innovative methods to unlock the hidden information within EEG signals and improve the lives of individuals affected by epilepsy.

The integration of deep learning and ensemble techniques provides a sophisticated and effective approach to epileptic seizure detection, addressing the inherent complexities of analyzing EEG signals. Deep learning models, including the Multi-Dimensional Bayesian Convolutional Network (MDBCN) and Deep Neural Networks (DNNs), play a pivotal role in capturing intricate spatiotemporal features. These models excel at learning hierarchical representations, which reduces the reliance on manual feature engineering and enhances the detection of subtle seizure-related patterns. By incorporating Discrete Wavelet Transform (DWT), the system further enriches feature extraction, enabling the decomposition of EEG signals into distinct frequency bands that reveal both high-frequency and low-frequency components essential for identifying seizures.

To strengthen the system's performance, ensemble methods such as Support Vector Machines (SVMs), Random Forests, and XGBoost classifiers are employed. These techniques mitigate overfitting and variability issues often associated with individual deep-learning models. By aggregating the predictions of these diverse models through a Voting Classifier, the system capitalizes on their complementary strengths, enhancing robustness and generalization. The combined use of deep learning for automatic feature learning, DWT for detailed signal decomposition, and ensemble strategies for prediction fusion ensures a multi-faceted and reliable detection framework.

This approach underscores its capability to advance the state of the art in seizure detection by uniting advanced feature extraction methods, powerful learning algorithms, and robust ensemble strategies. The result is a system with significant potential for clinical applications and real-time monitoring, offering improved accuracy and resilience in distinguishing between seizure and non-seizure events.

Despite the promising results, this study is subject to several limitations that warrant consideration. Firstly, the reliance on publicly available datasets, while facilitating reproducibility and comparison, may not fully capture the heterogeneity and complexity of real-world clinical scenarios. Datasets such as those from Kaggle and Bonn University, though widely used, may not encompass the diverse range of seizure types, patient populations, and recording conditions encountered in clinical practice. Consequently, the generalizability of the findings to broader populations and varied clinical settings may be limited. Additionally, the computational demands of deep learning models, particularly the MDBCN, pose challenges for deployment in resource-constrained environments. The training and execution of these models require substantial computational power and memory, potentially limiting their accessibility in settings with limited infrastructure. Furthermore, while ensemble methods enhance robustness, they can introduce increased complexity and potentially obscure the interpretability of the final predictions. The "black box" nature of deep learning models, coupled with the aggregation of multiple model outputs, can make it challenging to decipher the underlying factors influencing the system's decisions. This lack of interpretability can hinder clinical acceptance and trust, as clinicians may be hesitant to rely on a system whose decision-making process is not transparent. Finally, the binary classification approach, while practical and widely adopted, simplifies the complexity of seizure detection. Real-world seizure events are characterized by diverse patterns and temporal dynamics that may not be adequately captured by a binary distinction between seizure and non-seizure states. Future research should explore multi-class classification approaches and consider the temporal evolution of seizure activity to provide more nuanced and clinically relevant insights. Addressing these limitations through further research and development will be crucial for translating the promising findings of this study into practical and impactful clinical applications.

6 Conclusions and Future Work

The study evaluated various machine learning and deep learning models for seizure detection across two datasets. The Voting Classifier excelled, achieving perfect scores (accuracy, precision, recall, F1-score: 1.00) with short computation times, making it the most effective and efficient model. The CNN showed high accuracy but required more computation time, limiting its real-time applicability. Models like DWT with DNN and SVM offered strong performance and faster computation, making them viable for time-sensitive applications. The MDBCN model performed well in precision but struggled with seizure recall and had long computation times. Random Forest and XGBClassifier demonstrated solid performance with quick processing times, suitable for practical use.

On the second dataset, the Voting Classifier again outperformed others, while DWT with DNN maintained high performance and speed. MDBCN showed lower seizure recall but high precision, and CNN's computation time remained a drawback.

Future work should focus on improving computational efficiency for models like CNN, developing hybrid models for better precision and speed, and exploring real-world clinical applications. Expanding datasets and integrating models with edge computing and real-time monitoring systems could enhance seizure detection and patient care.

Declarations

Data and code availability

The dataset (1) of this study is available **at** **https://www.kaggle.com/datasets/harunshimanto/epileptic-seizure-recognition/data.**

The dataset (2) of this study is available at: https://github.com/jkuypers93/LSTM-Epileptic-Seizure-Recognition

Declaration of competing interest

No conflicts of interest exist, according to the authors. They affirm that there are no known conflicting financial interests or personal ties that may have influenced the work described in this publication.

Ethical Statement: "This article does not contain any studies with human participants or animals performed by any of the authors."

References

[1] A. Ahmed *et al.*, "Seizures and Epilepsy: An Overview for UK Medical Students," *Cureus*, vol. 16, no. 10, p. e70654, Oct. 2024, doi: 10.7759/cureus.70654.

[2] "World health statistics 2024: Monitoring health for the SDGs, sustainable development goals." Accessed: Nov. 16, 2024. [Online]. Available: https://www.who.int/publications/i/item/9789240094703

[3] "SUDEP (Sudden Unexpected Death in Epilepsy)," CURE Epilepsy. Accessed: Nov. 16, 2024. [Online]. Available: https://www.cureepilepsy.org/understanding-epilepsy/epilepsy-basics/sudep-sudden-unexpected-death-in-epilepsy/

[4] H. Mukhtar, "Chapter 11 - EEG signal processing with deep learning for alcoholism detection," in *Artificial Intelligence and Multimodal Signal Processing in Human-Machine Interaction*, A. Subasi, S. M. Qaisar, and H. Nisar, Eds., in Artificial Intelligence Applications in Healthcare and Medicine. , Academic Press, 2025, pp. 211–226. doi: 10.1016/B978-0-443-29150-0.00009-3.

[5] M. A. AbuAlrob and B. Mesraoua, "Harnessing artificial intelligence for the diagnosis and treatment of neurological emergencies: a comprehensive review of recent advances and future directions," *Front. Neurol.*, vol. 15, Oct. 2024, doi: 10.3389/fneur.2024.1485799.

[6] A. Aziz *et al.*, "An Unobtrusive and Lightweight Ear-worn System for Continuous Epileptic Seizure Detection," *ACM Trans. Comput. Healthcare*, Nov. 2024, doi: 10.1145/3703164.

[7] S. Nath, N. Kumar, K. Guha, K. L. Baishnab, and K. S. Rao, "Novel design of a low power neural amplifier using split push pull balanced high swing OTA for brain-machine interface," *Microsyst Technol*, vol. 30, no. 2, pp. 197–207, Feb. 2024, doi: 10.1007/s00542-023-05588-6.

[8] S. Nazish *et al.*, "Utility of Various Activation Procedures in Provoking Ictal and Interictal Patterns, during Routine Electroencephalogram (rEEG) Recording," *Annals of African Medicine*, vol. 23, no. 4, p. 688, Dec. 2024, doi: 10.4103/aam.aam_64_24.

[9] A. M. Affandi and F. A. Talmees, "Comprehensive Survey of Machine Learning Techniques for Seizure Detection and Prediction: Challenges and Future Directions," *Journal of Advances in Mathematics and Computer Science*, vol. 39, no. 11, Art. no. 11, Oct. 2024, doi: 10.9734/jamcs/2024/v39i111938.

[10] A. S., S. K. S., N. Mohan, and S. K.p., "EEG based automated detection of seizure using machine learning approach and traditional features," *Expert Systems with Applications*, vol. 251, p. 123991, Oct. 2024, doi: 10.1016/j.eswa.2024.123991.

[11] C. Bielza, G. Li, and P. Larrañaga, "Multi-dimensional classification with Bayesian networks," *International Journal of Approximate Reasoning*, vol. 52, no. 6, pp. 705–727, Sep. 2011, doi: 10.1016/j.ijar.2011.01.007.

[12] T. K. Murugan and A. Kameswaran, “Employing convolutional neural networks and explainable artificial intelligence for the detection of seizures from electroencephalogram signal,” *Results in Engineering*, vol. 24, p. 103378, Dec. 2024, doi: 10.1016/j.rineng.2024.103378.
[13] A. Shariat, A. Zarei, S. A. Karvigh, and B. M. Asl, “Automatic detection of epileptic seizures using Riemannian geometry from scalp EEG recordings,” *Med Biol Eng Comput*, vol. 59, no. 7–8, pp. 1431–1445, Aug. 2021, doi: 10.1007/s11517-021-02385-z.
[14] R. Wang, H. Wang, L. Shi, C. Han, and Y. Che, “Epileptic Seizure Detection Using Geometric Features Extracted from SODP Shape of EEG Signals and AsyLnCPSO-GA,” *Entropy*, vol. 24, no. 11, Art. no. 11, Nov. 2022, doi: 10.3390/e24111540.
[15] M. Li, W. Chen, and M. Xia, “GNMF-based quadratic feature extraction in SSTFT domain for epileptic EEG detection,” *Biomedical Signal Processing and Control*, vol. 80, p. 104274, Feb. 2023, doi: 10.1016/j.bspc.2022.104274.
[16] Y. Xiong *et al.*, “Seizure detection algorithm based on fusion of spatio-temporal network constructed with dispersion index,” *Biomedical Signal Processing and Control*, vol. 79, p. 104155, Jan. 2023, doi: 10.1016/j.bspc.2022.104155.
[17] L. Jiang, J. He, H. Pan, D. Wu, T. Jiang, and J. Liu, “Seizure detection algorithm based on improved functional brain network structure feature extraction,” *Biomedical Signal Processing and Control*, vol. 79, p. 104053, Jan. 2023, doi: 10.1016/j.bspc.2022.104053.
[18] F. Hassan, S. F. Hussain, and S. M. Qaisar, “Epileptic Seizure Detection Using a Hybrid 1D CNN-Machine Learning Approach from EEG Data,” *Journal of Healthcare Engineering*, vol. 2022, no. 1, p. 9579422, 2022, doi: 10.1155/2022/9579422.
[19] S. Davidson *et al.*, “Seizure Classification Using BERT NLP and a Comparison of Source Isolation Techniques with Two Different Time-Frequency Analysis,” in *2022 IEEE Signal Processing in Medicine and Biology Symposium (SPMB)*, Dec. 2022, pp. 1–7. doi: 10.1109/SPMB55497.2022.10014769.
[20] Z. Li, K. Hwang, K. Li, J. Wu, and T. Ji, "Graph-generative neural network for EEG-based epileptic seizure detection via the discovery of dynamic brain functional connectivity," *Sci Rep*, vol. 12, no. 1, p. 18998, Nov. 2022, doi: 10.1038/s41598-022-23656-1.
[21] M. Varlı and H. Yılmaz, “Multiple classification of EEG signals and epileptic seizure diagnosis with combined deep learning,” *Journal of Computational Science*, vol. 67, p. 101943, Mar. 2023, doi: 10.1016/j.jocs.2023.101943.
[22] A. Einizade, S. Nasiri, M. Mozafari, S. H. Sardouie, and G. D. Clifford, “Explainable automated seizure detection using attentive deep multi-view networks,” *Biomedical Signal Processing and Control*, vol. 79, p. 104076, Jan. 2023, doi: 10.1016/j.bspc.2022.104076.
[23] H. Ji *et al.*, “An effective fusion model for seizure prediction: GAMRNN,” *Front. Neurosci.*, vol. 17, Aug. 2023, doi: 10.3389/fnins.2023.1246995.
[24] M. Shen, P. Wen, B. Song, and Y. Li, “Real-time epilepsy seizure detection based on EEG using tunable-Q wavelet transform and convolutional neural network,” *Biomedical Signal Processing and Control*, vol. 82, p. 104566, Apr. 2023, doi: 10.1016/j.bspc.2022.104566.
[25] T. J. Rani and D. Kavitha, “Effective Epileptic Seizure Detection Using Enhanced Salp Swarm Algorithm-based Long Short-Term Memory Network,” *IETE Journal of Research*, vol. 70, no. 2, pp. 1538–1555, Feb. 2024, doi: 10.1080/03772063.2022.2153090.
[26] S. Khandelwal, N. Salankar, S. M. Qaisar, J. Upadhyay, and P. Pławiak, “ECG based apnea detection by multirate processing hybrid of wavelet-empirical decomposition Hjorth features extraction and neural networks,” *PLOS ONE*, vol. 18, no. 11, p. e0293610, Nov. 2023, doi: 10.1371/journal.pone.0293610.
[27] P. Srinivas, M. Arulprakash, M. Vadivel, N. Anusha, G. Rajasekar, and C. Srinivasan, “Support Vector Machines Based Predictive Seizure Care using IoT-Wearable EEG Devices for Proactive Intervention in Epilepsy,” in *2024 2nd International Conference on*

Computer, Communication and Control (IC4), Feb. 2024, pp. 1–5. doi: 10.1109/IC457434.2024.10486581.

[28] M. A. Mohammed Vishar, Hepsiba. D, R. Prasanna, and L. D. Vijay Anand, "Machine Learning-Based Early Seizure Detection: A Random Forest Classifier Approach for Pre-Ictal Stage Prediction in Epilepsy," in *2024 Tenth International Conference on Bio Signals, Images, and Instrumentation (ICBSII)*, Mar. 2024, pp. 1–5. doi: 10.1109/ICBSII61384.2024.10564015.

[29] B. Kapoor and B. Nagpal, "Gradient Boosting for Predicting the Relation Between Bio-medical Signals and Seizures Using LGBM and XGBoost," *Journal of Mobile Multimedia*, pp. 359–390, Mar. 2024, doi: 10.13052/jmm1550-4646.2025.

[30] "Epileptic Seizure Recognition." Accessed: Nov. 16, 2024. [Online]. Available: https://www.kaggle.com/datasets/harunshimanto/epileptic-seizure-recognition

[31] R. G. Andrzejak, G. Widman, K. Lehnertz, C. Rieke, P. David, and C. E. Elger, "The epileptic process as nonlinear deterministic dynamics in a stochastic environment: an evaluation on mesial temporal lobe epilepsy," *Epilepsy Res*, vol. 44, no. 2–3, pp. 129–140, May 2001, doi: 10.1016/s0920-1211(01)00195-4.

[32] J. Kuypers, *jkuypers93/LSTM-Epileptic-Seizure-Recognition*. (Jun. 25, 2024). Python. Accessed: Nov. 16, 2024. [Online]. Available: https://github.com/jkuypers93/LSTM-Epileptic-Seizure-Recognition

[33] S. Hadhri, M. Hadiji, and W. Labidi, "A voting ensemble classifier for stress detection," *Journal of Information and Telecommunication*, vol. 8, no. 3, pp. 399–416, Jul. 2024, doi: 10.1080/24751839.2024.2306786.

[34] A. Ameen, I. E. Fattoh, T. Abd El-Hafeez, and K. Ahmed, "Advances in ECG and PCG-based cardiovascular disease classification: a review of deep learning and machine learning methods," *J Big Data*, vol. 11, no. 1, p. 159, Nov. 2024, doi: 10.1186/s40537-024-01011-7.

[35] E. H. I. Eliwa, A. M. E. Koshiry, T. A. El-Hafeez, and A. Omar, "Secure and Transparent Lung and Colon Cancer Classification Using Blockchain and Microsoft Azure," *Advances in Respiratory Medicine*, vol. 92, no. 5, p. 395, Oct. 2024, doi: 10.3390/arm92050037.

[36] E. Hassan, T. Abd El-Hafeez, and M. Y. Shams, "Optimizing classification of diseases through language model analysis of symptoms," *Sci Rep*, vol. 14, no. 1, p. 1507, Jan. 2024, doi: 10.1038/s41598-024-51615-5.

[37] A. M. E. Koshiry, E. Eliwa, T. A. El-Hafeez, and A. Omar, "Classification of University Excellence: A Multi-dimensional Exploration of Ranking Criteria Using Data Science and Visualization Technology," in *Proceedings of the 10th International Conference on Advanced Intelligent Systems and Informatics 2024*, A. E. Hassanien, A. Darwish, M. F. Tolba, and V. Snasel, Eds., Cham: Springer Nature Switzerland, 2024, pp. 209–220. doi: 10.1007/978-3-031-71619-5_18.

[38] A. Omar and T. Abd El-Hafeez, "Optimizing epileptic seizure recognition performance with feature scaling and dropout layers," *Neural Comput & Applic*, vol. 36, no. 6, pp. 2835–2852, Feb. 2024, doi: 10.1007/s00521-023-09204-6.

[39] T. Abd El-Hafeez, M. Y. Shams, Y. A. M. M. Elshaier, H. M. Farghaly, and A. E. Hassanien, "Harnessing machine learning to find synergistic combinations for FDA-approved cancer drugs," *Sci Rep*, vol. 14, no. 1, Art. no. 1, Jan. 2024, doi: 10.1038/s41598-024-52814-w.

[40] A. T. Tzallas, M. G. Tsipouras, and D. I. Fotiadis, "Epileptic Seizure Detection in EEGs Using Time–Frequency Analysis," *IEEE Transactions on Information Technology in Biomedicine*, vol. 13, no. 5, pp. 703–710, 2009.

[41] M. Peker, B. Sen, and D. Delen, "A Novel Method for Automated Diagnosis of Epilepsy Using Complex-Valued Classifiers," *IEEE Journal of Biomedical and Health Informatics*, vol. 20, no. 1, pp. 108–118, 2016.

[42] Ö. Alçin, S. Siuly, V. Bajaj, Y. Guo, and Y. Zhang, “Multi-category EEG signal classification developing time-frequency texture features based Fisher Vector encoding method,” *Neurocomputing*, vol. 218, pp. 251–258, 2016.
[43] Y. Li, W.-G. Cui, H. Huang, Y.-Z. Guo, K. Li, and T. Tan, “Epileptic seizure detection in EEG signals using sparse multiscale radial basis function networks and the Fisher vector approach,” *Knowledge-Based Systems*, vol. 164, pp. 96–106, 2019.
[44] A. Sharmila and P. Geethanjali, “DWT based detection of epileptic seizure from EEG signals using naive Bayes and k-NN classifiers,” *Ieee Access*, vol. 4, pp. 7716–7727, 2016.
[45] M. K. Islam, A. Rastegarnia, and Z. Yang, “A wavelet-based artifact reduction from scalp EEG for epileptic seizure detection,” *IEEE Journal of Biomedical and Health Informatics*, vol. 20, no. 5, pp. 1321–1332, 2015.
[46] A. R. Hassan, S. Siuly, and Y. Zhang, “Epileptic seizure detection in EEG signals using tunable-Q factor wavelet transform and bootstrap aggregating,” *Computer methods and programs in biomedicine*, vol. 137, pp. 247–259, 2016.
[47] G. Wang, Z. Sun, R. Tao, K. Li, G. Bao, and X. Yan, “Epileptic seizure detection based on partial directed coherence analysis,” *IEEE Journal of biomedical and health informatics*, vol. 20, no. 3, pp. 873–879, 2015.
[48] A. K. Jaiswal and H. Banka, “Epileptic seizure detection in EEG signal using machine learning techniques,” *Australas Phys Eng Sci Med*, vol. 41, no. 1, pp. 81–94, Mar. 2018, doi: 10.1007/s13246-017-0610-y.
[49] Q. Yuan *et al.*, “Epileptic seizure detection based on imbalanced classification and wavelet packet transform,” *Seizure*, vol. 50, pp. 99–108, 2017.
[50] S. Raghu, N. Sriraam, A. S. Hegde, and P. L. Kubben, “A novel approach for classification of epileptic seizures using matrix determinant,” *Expert Systems with Applications*, vol. 127, pp. 323–341, 2019.
[51] E. M. Senan, M. E. Jadhav, T. H. Rassem, A. S. Aljaloud, B. A. Mohammed, and Z. G. Al-Mekhlafi, “Early diagnosis of brain tumour MRI images using hybrid techniques between deep and machine learning," *Computational and Mathematical Methods in Medicine*, vol. 2022, 2022, Accessed: Oct. 05, 2023. [Online]. Available: https://www.hindawi.com/journals/cmmm/2022/8330833/
[52] A. Subasi, J. Kevric, and M. A. Canbaz, “Epileptic seizure detection using hybrid machine learning methods,” *Neural Comput. Appl.*, vol. 31, pp. 317–325, 2019.
[53] Z. Chen, G. Lu, Z. Xie, and W. Shang, “A unified framework and method for EEG-based early epileptic seizure detection and epilepsy diagnosis,” *IEEE Access*, vol. 8, pp. 20080–20092, 2020.