

Epileptic seizure detection: a comparative study between deep and traditional machine learning techniques

Rekha Sahu¹, Satya Ranjan Dash², Lleuvelyn A Cacha³, Roman R Poznanski⁴ and Shantipriya Parida^{5,*}

¹*School of Computer Engineering, KIIT University, Bhubaneswar, Odisha, 751024, India*

²*School of Computer Application, KIIT University, Bhubaneswar, Odisha, 751024, India*

³*Faculty of Health Science, Universiti Sultan Zainal Abidin, Gong Badak Campus, Darul Iman, Terengganu, 21300, Malaysia*

⁴*Faculty of Informatics and Computing, Universiti Sultan Zainal Abidin, Besut Campus, Besut, Terengganu, 22200, Malaysia*

⁵*Idiap Research Institute, Centre du Parc, Rue Marconi 19, Martigny, CH-1920, Switzerland*

*Correspondence: shantipriya.parida@idiap.ch (Shantipriya Parida)

DOI: [10.31083/j.jin.2020.01.24](https://doi.org/10.31083/j.jin.2020.01.24)

This is an open access article under the CC BY 4.0 license (<https://creativecommons.org/licenses/by/4.0/>).

Electroencephalography is the recording of brain electrical activities that can be used to diagnose brain seizure disorders. By identifying brain activity patterns and their correspondence between symptoms and diseases, it is possible to give an accurate diagnosis and appropriate drug therapy to patients. This work aims to categorize electroencephalography signals on different channels' recordings for classifying and predicting epileptic seizures. The collection of the electroencephalography recordings contained in the dataset attributes 179 information and 11,500 instances. Instances are of five categories, where one is the symptoms of epilepsy seizure. We have used traditional, ensemble methods and deep machine learning techniques highlighting their performance for the epilepsy seizure detection task. One dimensional convolutional neural network, ensemble machine learning techniques like bagging, boosting (AdaBoost, gradient boosting, and XG boosting), and stacking is implemented. Traditional machine learning techniques such as decision tree, random forest, extra tree, ridge classifier, logistic regression, K-Nearest Neighbor, Naive Bayes (gaussian), and Kernel Support Vector Machine (polynomial, gaussian) are used for classifying and predicting epilepsy seizure. Before using ensemble and traditional techniques, we have preprocessed the data set using the Karl Pearson coefficient of correlation to eliminate irrelevant attributes. Further accuracy of classification and prediction of the classifiers are manipulated using k-fold cross-validation methods and represent the Receiver Operating Characteristic Area Under the Curve for each classifier. After sorting and comparing algorithms, we have found the convolutional neural network and extra tree bagging classifiers to have better performance than all other ensemble and traditional classifiers.

Keywords

Epilepsy; seizure; deep learning; artificial neural networks; neural sig-

nals; EEG signals; computer simulations

1. Introduction

The electroencephalography (EEG) recording of different channels shows the electrical activities of the brain and is also used to understand and elucidate brain functions in order to help us to diagnose neurological disorders. In particular, EEG record is an important tool for the diagnosis of neurological diseases, such as epilepsy. During the EEG test, the computer screen represents the brain's electrical signals into wavy lines, and these wavy lines are the track and record of the electrical activities of the brain. 256 electrodes are placed on the brain, which is recorded the signals from different areas of the brain. A channel is interpreted as one pair of electrodes, and a signal is a recording of the channel. Since the architectural features of the brain are uneven such as the cortical thickness and surface area.

Given that the cortex is functionally organized and movement diverges as a function, thus the EEG can vary and significantly differ depending on the topographic location of the recording electrodes. Sometimes the subject's EEG signals vary abnormally. These abnormalities of EEG signals are of two categories, one is abnormal epileptic signals, and another is non-epileptic abnormal signals. Spike and the sharp wavy line is the track and record of EEG signal pattern for epilepsy patients, whereas non-epilepsy abnormalities are characterized by the alternative of a normal and abnormal pattern of EEG signals (Siuly et al., 2016). The measurement of signals is represented in terms of Hz (frequencies in second). Thus, different signals of different locations of the brain are expressed in numerical forms. Higher frequencies in EEG channels are the symptoms of the abnormal state of the subject that may say suffering from epilepsy seizure.

Generally, from the EEG records' frequencies, the expert can diagnose the abnormalities. Automatic systems based on machine learning techniques can save hours of manually reviewing the EEG recording. Hence machine learning techniques are proposed and implemented. We have used an EEG dataset consists of five sets denoted as (A-E), and each containing 100 single-channel EEG

segments of 23.6 seconds EEG signals. Converging evidence suggests that machine learning techniques like support vector machine (SVM), random forest (RF), naive Bayes (NB), k-nearest neighbors (K-NN), and neural network (NN), are already implemented with it (Resque et al., 2019). We have considered the dataset to be implemented with different machine learning to compare their performances. Since it is a challenging job to develop an automated system to identify epileptic seizures to give an accurate medical treatment, so experiments with different ML techniques are to be conducted.

Recent studies have shown that deep learning models based on artificial neural networks (ANN) provide new avenues for solving the complex problems inherent in automatic seizure detection using EEG data (Thodoroff et al., 2016; Yuvaraj et al., 2018). Deep learning, convolution neural network (CNN) is giving an excellent performance in classifying the EEG data set. Although some traditional machine learning technique has given comparable accuracy, we have to merge other preprocessing techniques with it whereas CNN does not require preprocessing of reducing data (Avcu et al., 2019; Fukumori et al., 2019; Rahman et al., 2019; Resque et al., 2019; Wójcik et al., 2019).

The different experimental results showed that CNN classifies the EEG dataset with good accuracy of prediction. Multi-scale CNN algorithm learns features of EEG data and predicts the epilepsy seizure. Computer-Aided diagnosis is required to distinguish the class of the EEG signal automatically. Using 13-layers deep CNN, the classifier was trained and studied the accuracy of classification (Acharya et al., 2018). From the EEG signal, depression can be detected using CNN, followed by LSTM (long short-term memory). This CNN-LSTM was validated with 30 subjects, 15 normal and 15 depressed, and found an outstanding performance (Ay et al., 2019). The raw EEG signals are processed in the form of a Spatiotemporal representation. CNN classifies the Spatio-temporal representations of the EEG signals, and a fusion strategy based on a multilayer perceptron has given a good accuracy of classification (Alhussein et al., 2019). An automated detection algorithm for idiopathic generalized epilepsy (IGE) is implemented with CNN. CNN is trained with a dataset of over 6000 labeled events in a cohort of 103 patients. It was found automated computer-assisted review can increase speed and accuracy (Clarke et al., 2019). CNN's performance varies according to different datasets. The stacked sparse autoencoder (SSAE) is multiple layers of sparse autoencoders neural network, which is used as an unsupervised feature extraction method, and the Taguchi Method is employed for parameter optimization. This novel framework is tested with different experimental data sets like DDoS Detection, IDS Attack, Epileptic Seizure Recognition, and handwritten digit classification problem (Karim et al., 2018). There was a proposal of deep autoencoder architecture for medical data preprocessing, and for classification, the softmax classifier layer is trained. To evaluate the performance of the three datasets, i.e., epileptic seizure, SPECTF (single proton emission computed tomography) are implemented with it (Karim et al., 2019). Since CNN provides better performance in EEG data interpretation, implementation with it may be fruitful.

Compared with traditional machine learning techniques, CNN experiments with the EEG dataset through the survey have ob-

tained competitive results. For seizure onset detection, CNN has implemented two-channel recording and filter by the convolution layer. Compare to spectrum band power SVM, CNN experimented with 29 pediatric patients who have classified more accurately (Avcu et al., 2019). Different machine learning algorithms and preprocessing algorithms are used to classify the multiclass seizure type. For preprocessing EEG signals, Fast Fourier Transform (FFT) and correlation coefficients are used, and for classification K-NN, SGD classifier, XG boost, AdaBoost, and CNN are implemented. Event-related EEG signal of 70 patients are implemented with five classifiers boosted decision tree, classical neural network, Bayes point machine, and logistic regression and average perception supervised machine learning technique, whereas logistic regression and boost decision tree has given good accuracy of classification. In another report (Rahman et al., 2019), 3750 focal EEG segments and 3750 non-focal EEG segments of five subjects are studied using the ensemble stack classifier and found better classification accuracy. An ensemble deep learning-based CNN classifies the seizure type, which is comparatively robotic than traditional techniques. Paroxysmal spikes are the symptoms seen in patients with epileptic seizures, and these are confirmed with the EEG recording. SVM, RF, and CNN classified epilepsy spikes and non-epilepsy spikes from EEG recording. Characteristic spikes observed in the EEG detect epileptic spike. Feedforward CNN or recurrent neural network (RNN) classifies epileptic spike and non-epileptic spike, whereas SVM and RF with prefixed preprocessing achieved comparable scores (Fukumori et al., 2019).

For EEG signal classification, so many traditional machine learning techniques experimented with it (Nandy et al., 2019). Different machine learning techniques SVM, K-NN, Naïve Bayes classifier, random forest, and NN are implemented with our data set, and 90% above accuracy presented (Resque et al., 2019). Three machine learning techniques, neural networks, logistic regression, and linear integer model, are trained to predict seizures. 60% of the data set is for the fitting model, and 40% are for model evaluation and found out that the Integer model has given better performance (Struck et al., 2019). A mental illness that affects a person thinks acts or feels it is called major depressive disorder (MDD). EEG signals with alpha, alpha1, alpha2, beta, delta, and theta power and theta asymmetry were used as feature and classifiers SVM, logistic regression, naïve-Bayesian, and decision tree was used with SVM had the highest accuracy (Mahato and Paul, 2020).

2. Materials and methods

We carried out simulations with deep learning (CNN), ensemble machine learning (bagging, boosting, and stacking) and traditional machine learning (decision tree, random forest, extra tree, ridge classifier, logistic regression, KSVM, K-NN) techniques.

2.1 Machine learning techniques

Since our purpose is to analyze the EEG dataset with the performance of machine learning techniques, hence we first go through traditional machine learning techniques like decision tree, random forest, extra tree, ridge classifier, logistic regression, KSVM, K-NN, etc. The tree for classification or decision tree is constructed by selecting a specific feature from attributes as the root node using the criteria and split recursively (Mahato and Paul, 2020). Each non-leaf node is labeling with input attribute or feature. Each leaf

node is the output or class of the instance or a probability distribution over the different possible classes. Entropy and information gain (IG) are the formulas to select the feature to label the node. Entropy is used for manipulating IG from a feature, whereas IG is used to manipulate the information to gain splitting with the feature. The entropy is defined by

$$\text{entropy} = - \sum p(x) \log(p(x)) \quad (1)$$

where $p(x)$ stands for the probability of x , $\log(p(x))$ is logarithm of $p(x)$ and the expression for IG is

$$\text{IG}(x) = \text{entropy}(x) - (\text{weighted average} \times \text{entropy (children for feature)}) \quad (2)$$

where x is the feature & children for feature implies the next nodes or attributes may be considered as parent node to create children or subtree of decision tree.

Some reports also suggest random forest classifier. According to it, subsets of training data set are randomly selected, decision trees are built with it, and to find out the class of an object, we have to aggregate the votes from the all decision tree (Tavares et al., 2019). Thus, in a random forest classifier, the training data are randomly subdivided, and with each sample data, the decision tree is constructed. Extra tree classifier follows the same procedure that random forest classifier follows. The only difference is when splitting the node; we have considered a limited feature in the random forest, whereas, in the case of the extra tree, all features are considered to split a non-leaf node. Generally, it was found out that variance with decision tree classifier is more compared to the random forest, and the variance in the case of random forest classifier is more than an extra tree. The extension of linear regression classifier with minimization of error or loss function is called ridge regression classifier. The loss function can be modified using the Eqn. 3.

$$\text{Loss function} = \text{OLS} + \alpha \times \sum (\text{squared coefficient values}) \quad (3)$$

where OLS is the ordinary least square. We have to find out the value of alpha on which performance of the classifier depends. If the value of alpha is less, then it leads to overfitting. On the contrary, if the value of alpha is more, the ridge classifier performance is underfitting (Seifzadeh et al., 2017). Logistic regression can be defined as the probability that an instance with attributes values x_1, x_2, \dots, x_n is

$$p = 1 + \frac{1}{1 + e^{-(a_0 + a_1 x_1 + a_2 x_2 + \dots + a_n x_n)}} \quad (4)$$

where a_i $i = 0, 1, \dots, n$ are all constants.

From the training data set, which are labeled, we can calculate the constants a_i of the logistic regression equation, and the maximum likelihood estimation technique is used to get the values of constants. Prediction using logistic regression is an easy task, and if the coefficients are accurate, the prediction is robotic (Ilyas et al., 2016). From the mathematical point of view, the SVM classifier is a constrained minimization problem that is solved using the

Lagrange multiplier method (Lee et al., 2019). The dot products of support vectors those collected from the training data set are used to find out the classifier, which formulates the maximum gap between different classes of samples. Thus, the product is defined as

$$L = \sum_i \alpha_i - \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j \vec{x}_i \cdot \vec{x}_j \quad (5)$$

where α_i is the constant generated according to constraints.

To find out the class of a new object \vec{u} , the product of the support vector and the new sample has to use for a decision viz.,

$$\sum_i \alpha_i y_i \vec{x}_i \cdot \vec{u} + b \geq 0, b \text{ is the bias value.} \quad (6)$$

If data are not separated or labeled linearly, non-linearly, it is required to separate according to the target class of the instances. This may possibly use kernel tricks. We can generate a hyperplane for the non-linear separable dataset. So, we have to consider mapping function, which transforms the two-dimensional input space into three-dimensional output space. Then for maximizing the classifier SVM and form a decision rule, we have to do the dot product of mapping function for different samples, viz.

$$L = \sum_i \alpha_i - \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j \varphi(x_i) \varphi(x_j) \quad (7)$$

and the decision rule equation (6) is modified as

$$\sum_i \alpha_i y_i \varphi(x_i) \varphi(u) + b \geq 0 \quad (8)$$

We can define a function K as $K(x_i, x_j) = \varphi(x_i) \cdot \varphi(x_j)$. Thus, instead of the mapping function, we can use kernel function K , which reduces the complexity of deriving mapping function. Some of the Kernel functions are defined by

$$K(x_i, x_j) = (x_i \cdot x_j + 1)^p : \text{Polynomial kernel} \quad (9)$$

$$K(x_i, x_j) = e^{-\gamma(x_i - x_j)^2} : \text{RBF kernel} \quad (10)$$

$$K(x_i, x_j) = e^{\frac{-1}{2\sigma^2}(x_i - x_j)^2} : \text{Gaussian kernel} \quad (11)$$

The K-NN algorithm assumes that similar things exist near to each other. K-NN captures the idea of similarity with some mathematics like the distance between points on a graph. K-NN is easy to classify. The goal of any machine learning problem is to find a single model that will best predict our wanted outcome. Rather than making one model and hoping this model is the best/most accurate predictor with traditional machine learning, we can make, ensemble methods take a myriad of models into account, and average or vote those models to produce one final model. The ensemble machine learning techniques are bagging, boosting, and stacking. For implementing bagging classifiers, using the Bootstrapping sampling method, from the data set, random data subsets are formulated where each subset from the original data set is included all the features of the data set. A specified estimator

or base classifier, K-NN classifier, KSVM (gaussian) classifier, ridge classifier, logistic regression, decision tree classifier, GNB classifier, polynomial and RBF KSVM classifier, random forest classifier and extra tree classifier are fitted with it. Predictions from each model are combined with average or voting techniques and predict suitable classifiers. We have the weak classifiers that may say traditional classifiers. In the boosting algorithm, the weak classifier is converted into a robust classifier iteratively. Generally, in the case of a weak classifier, the accuracy of classification is low, and to make it high, we readjusted the weight, which is related to the accuracy of classification. The process of adjustment is as follows. The input data which are misclassified have to get more weight, and the inputs which classified correctly have to lose weights. Thus, the procedure emphasizes the misclassified instances.

Ada boost, grad boost, XG boost, etc. are different types of boosting algorithms. Ada boost uses the weak classifier as a decision tree. The weight of each training instance is recorded and accordingly modified as follows. The first weight of the training data x is assigned as $1/N$, where N is the total number of training data set. Then error or misclassified rate is calculated as $E = (R - N) / N$, where E is the misclassified rate, R is correctly classified training data, and N is the total number of training data set. The weight of the training data is modified as follows:

$$E = \sum (w_i \times t_i) / \sum w_i \quad (12)$$

where E is the misclassified rate, w_i is the weight, and t_i is predicted misclassified rate. $t_i = 1$ if misclassified and $t_i = 0$ if correctly classified. The parameter or stage value S is manipulated to modify the classifier as in follow: $S = \ln[(1-E) / E]$.

Finally, training weights are updated by giving more weight to incorrectly classified and less weight to correctly classified by the following manipulation. $w = w \times e^{(s \times t_j)}$, where w is the weight, e is the Euler number, s is the stage value which is used for weight prediction from the model manipulated using formula $s = \ln((1 - \text{error})/\text{error})$, $\ln()$ is the natural logarithm & misclassified error from the model, and t_j is the misclassified rate. The value of t_j manages the weight if the instances are correctly classified. Gradient Boosting (GBM) is also a boosting algorithm that iteratively interpreted a classification tree or decision tree. Suppose we have a decision tree $h_m(x)$ where x is a training instance and the number of leaves j_m . The tree divides the training data into j_m disjoint regions, i.e., $R_{1m}, R_{2m}, R_{3m}, \dots, R_{j_m m}$ and having a constant value b_{j_m} for each region R_{j_m} . Thus for instance x , the target is $h_m(x) = \sum_{j=1}^{j_m} b_{j_m} I_{R_{j_m}}(x)$. Hence the model is updated as follows:

$$F_m(x) = F_{m-1}(x) + \sum_{j=1}^{j_m} Y_{j_m} I_{R_{j_m}}(x) \quad (13)$$

$$Y_{j_m} = \underset{Y}{\operatorname{argmin}} \sum_{x_i \in R_{j_m}} L(y_i, F_{m-1}(x_i) + Y) \quad (14)$$

where $F_m(x)$ minimizes the expected value of loss function $L(Y, F_m(x))$. The indicator function of a subset A of a set X is a function

$I_A: X \rightarrow \{0, 1\}$. Y_{j_m} is the constant. The loss function is minimized by multiplying constant Y_{j_m} with b_{j_m} . XGBoost (extreme gradient boosting) is an optimized distributed gradient boosting library and uses the GBM framework at the core and does better than it.

The heterogeneous ensemble learning technique is stacking, which has become a commonly used technique for generating prediction. Using all data set, the base classifiers are trained, and on the next level, a meta classifier is trained, which is based on base classifiers. We can use the decision tree, GNB, K-NN, random forest, extra tree, and logistic regression as the base classifiers to train the dataset. Each model is fine-tuned using probability scores. At the final stage, the predictions of all the base models are combined using majority voting to create a final model called the meta-classifier. Thus, ensemble machine learners are the modified version of traditional machine learning techniques. In bagging, data are split, and a base classifier is implemented with each subset of data and the best one taken as meta-classifier. In the case of boosting, a base classifier is used with the whole dataset and modified the base classifier parameter to get a strong meta-classifier. In the case of stacking, more than one classifier implemented with the data set independently, and the classifier showing the best performance is taken as a meta-classifier. Hence ensemble learner performs better than traditional classifiers (Cacha et al., 2016; Parida et al., 2015).

In Fig. 1, we have summarized overall work in this paper. We have used the epilepsy dataset¹ freely available (see Andrzejak et al. (2001)). The data set recorded from 500 individuals. It is categorized into five sets A, B, C, D, and E. Each set has 100 files representing each file as one subject, and the recording activity of the brain is 23.6 seconds for each subject. In 23.6 s, 4096 sample data points are collected per subject. All details are summarized in Table 1.

Each 4096 sample data points are shuffled and made 23 chunks with 178 data points. Thus, we have 178 attributes and $23 \times 500 = 11500$ instances. Another attribute y is added to label the target to the instances. The last values fitted for last column y is 1, 2, 3, 4 and 5. 1 represents the recording of EEG signal at the time of epileptic seizure, 2 denotes recording signal where the tumor located, 3 denotes recording of EEG signal from healthy brain area when subject has tumor, 4 means the eyes are closed whereas 5 indicates eye is opened. Thus, we have 179 attributes where 178 are frequencies of EEG signals named X_1, X_2, \dots, X_{178} , and another designated as y as the target value. All subjects falling in classes 2, 3, 4, and 5 have no epileptic seizure, whereas subjects in class 1 have an epileptic seizure. Although there are five categories of subjects, we have in intension to classify into two categories, namely class 1 and class 0, where class 1 representing epileptic seizure and class 0 describing non- epileptic seizure. We have used the Pearson coefficient of correlation method to eliminate attributes. Among 178 attributes, we have selected the characteristics whose absolute correlations' coefficients with labeled attribute " y " have more than 0.025. Basing the coefficient's criteria, we have 31 attributes to consider. Again, all instances, cate-

¹ http://www.epileptologie-bonn.de/cms/front_content.php?idcat=193&lang=3

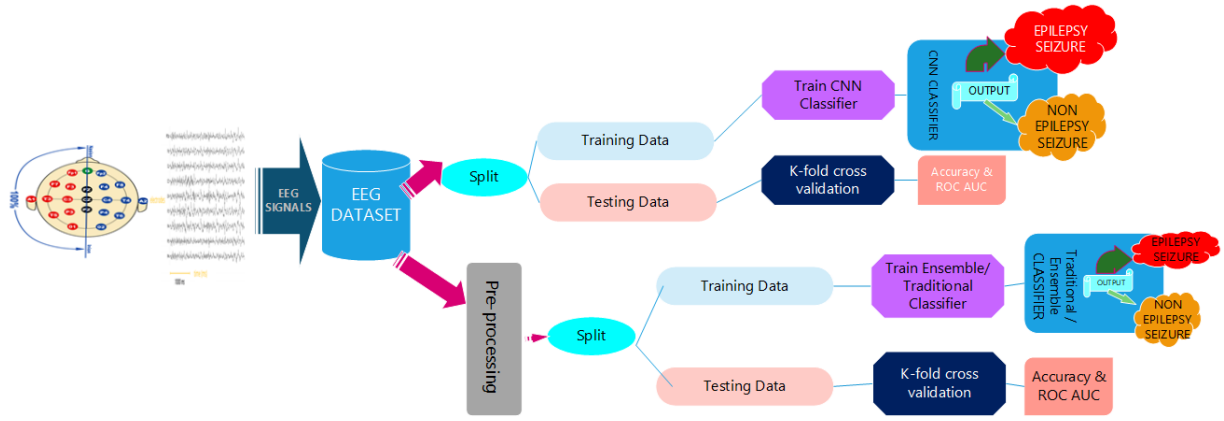


Figure 1. Overall work on EEG Data set with the implementation of CNN, ensemble, and traditional machine learning algorithms. The EEG dataset is preprocessed (except CNN model) to eliminate irrelevant features and split into train and test datasets. The training and test datasets are used to train the traditional, ensemble, and deep learning models and used to classify epilepsy or non-epilepsy Seizure.

Table 1. Summary of the epileptic EEG data. All Set (A-E) contains five healthy subjects.

Subjects	Set A 100 subjects	Set B 100 subjects	Set C 100 subjects	Set D 100 subjects	Set E 100 subjects
Patient's state	Epilepsy seizure	Having tumor	Healthy	Eye closed	Eye opened
Number of text files containing recording of EEG signals	100 with each file includes 4096 samples of one EEG time series.	100 with each file includes 4096 samples of one EEG time series.	100 with each file includes 4096 samples of one EEG time series.	100 with each file includes 4096 samples of one EEG time series.	100 with each file includes 4096 samples of one EEG time series.
Time duration (s)	23.6	23.6	23.6	23.6	23.6

gorized as five labels, are converted into two as epilepsy seizure and non-epilepsy seizure. The label of example is taken as 1 if it is a category of epilepsy seizure. Otherwise, the category is 0, non-epilepsy seizure.

Before applying the machine learning technique, we may require preprocessing to reduce attributes. Pearson's coefficient of correlation can be used to measure the coefficient of relationship between the attributes and the target class:

$$r = \frac{Cov(x, y)}{\sigma_x \sigma_y} \quad (15)$$

where $Cov(x, y)$ is the covariance and σ_x and σ_y stand for standard deviations. Higher the absolute value, more the degree of relationship between them.

To manipulate the performance of machine learning techniques cross-validation method or k-fold cross-validation method is suggested. According to the cross-validation procedure, the total data set are divided into two parts training set and testing set. The classifier is trained with the training data set, whereas the performance is tested with the testing data set. For proper evaluation, we can split the datasets into equally K parts, and the classifier is trained for K times. Each classifier is trained with K-1 parts, and another one part is used for testing. Finally, the average is taken of all accuracy predicted by K numbers of tests to get the final accuracy. Evaluation of classifiers is measured by finding accuracy and drawing the receiver operating characteristics' curve, called

the ROC curve. Accuracy can be measured as:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (16)$$

where TP, TN, FP, and FN stand for true positive, true negative, false positive, and false negative, respectively.

For representing the ROC curve of a classifier, the procedure is to be followed with the X-axis represents the false positive rate, and the Y-axis represents the true positive rate, and the threshold values are between 0.0 to 1.0. This fitting of the curve is useful to show the performance of the classifiers and compare them among the classifier performances. Again, the area under the curve summarized the classifier's skill and hence is a useful tool.

3. Results

CNN, a sequence of layers, has manipulated the dataset in the following ways. We have not pre-processed the data set, but instead of taking five classes of instances, two classes of instances (i.e., one is seizure '1,' and another is non-seizure '0') are categorized.

Input Layer: This layer holds the inputs of numeric values with width 11500, length 178, and height 1.

Convolution Layer: We have taken 9×1 filter and 1 stride. The output volume is the computing dot product between filter and patch. We used a total of 170 filters for this layer and got an output of dimension $11500 \times 170 \times 1$.

Pool Layer: We have used max-pooling to reduce the volume. The resultant volume is of dimension $11500 \times 19 \times 1$.

Table 2. Accuracy and Standard Deviation of different machine learning techniques.

Machine Learning Techniques	Accuracy	Standard Deviation
Decision tree	0.8886	+/- 0.0014
Random Forest classifier	0.9517	+/- 0.0009
Extra tree classifier	0.9435	+/- 0.0030
Kernel Support Vector Machine (polynomial)	0.9349	+/- 0.0010
Kernel Support Vector Machine (Gaussian)	0.942	+/- 0.0037
Naïve Bays Classifier	0.943	+/- 0.0011
Logistic regression	0.8048	+/- 0.0006
K-nearest neighbor classifier	0.9301	+/- 0.0015

Activation Function Layer: The softmax activation function is used for the output of the convolution layer.

We have implemented CNN with the dataset using Python programming with Sklearn², Keras³, and pandas⁴ libraries having platform Window 10. We have taken 50 epochs with filter size nine, and stride is one. It was found at 0.96 accuracies. Using CNN, we have increased the values of the attributes, for instance. The characteristics had more values gaining more importance. Hence, it is convenient to classify accurately. The ROC AUC representation of it is in Fig. 2, and the area under the curve is 0.99, which is an actual positive rate.

For classifying the dataset into two classes, i.e., epilepsy seizure and non-epilepsy seizure, we have used the decision tree with gini impurity. The accuracy of classification is 0.8886, with a standard deviation (+/- 0.0014). When the random forest classifier was used, it created a set of decision trees from a randomly selected subset of the training set and aggregated the votes from different decision trees to decide the final class of the test object. The number of trees for the random forest is taken as one hundred. The accuracy of using this method is 0.9517, with a standard deviation (+/- 0.0009). In an extra tree classifier, instead of trying to find an optimal cut-point for each one of the randomly chosen features at each node in a random forest, it selects a cut-point at random. This leads to producing the accuracy in classifying is 0.9435, and the standard deviation is (+/- 0.0030). Using a KSVM algorithm, we have the accuracy of classification 0.9420 with standard deviation (+/- 0.0037) using the Gaussian method, and accuracy of classification using polynomial kernel is 0.9349 with standard deviation (+/- 0.0010). The accuracy of classification using naïve bay's classifier is 0.9430, with a standard deviation (+/- 0.0011). Using logistic regression, we have an accuracy of 0.8048 with a standard deviation (+/- 0.0006). By implementing data set with a ridge classifier, we have found out the accuracy of 0.80 with standard deviation (+/- 0.00).

The non-parametric algorithm K-NN is implemented with the data set by using Euclidean distance, i.e., $d(P, Q) = \sqrt{(p_i - q_i)^2}$, where $P = (p_1, p_2, \dots, p_n)$ and $Q = (q_1, q_2, \dots, q_n)$, we have the accuracy of classification 0.9301 with standard deviation (+/- 0.0015). Thus, by using traditional machine learning techniques, we have classified into two categories, i.e., epilepsy seizure and non-epilepsy seizure with accuracy summarized in Table 2 and ROC, AUC are shown in Fig. 3.

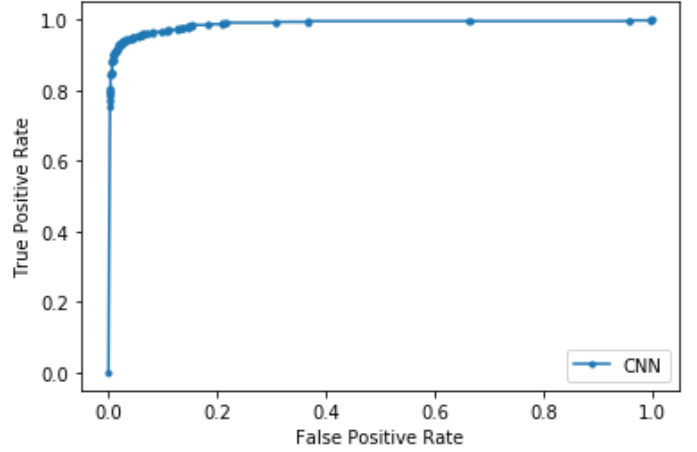


Figure 2. CNN model performance by depicting ROC AUC representation of CNN classifier. The area under the curve is 0.99, which is a valid positive rate.

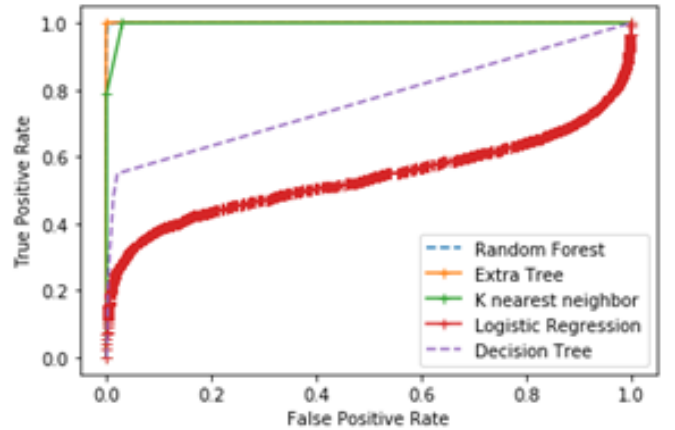


Figure 3. Representation of the ROC curve of traditional machine learning techniques. Random Forest: ROC, AUC = 1.000; Extra Tree: ROC, AUC = 1.000; K-NN: ROC, AUC = 0.997; Logistic Regression: ROC, AUC = 0.538; Decision Tree: ROC, AUC = 0.767.

² <https://scikit-learn.org/stable/>

³ <https://keras.io/>

⁴ <https://pandas.pydata.org/>

Table 3. Different Bagging Classifiers' accuracy.

Base Estimators for Bagging	Average Manipulation		Voting to estimators	
	Accuracy	Standard Deviation	Accuracy	Standard Deviation
K-nearest neighbors' classifier	0.9393	+/- 0.0005	0.93	+/- 0.00
Kernel Support Vector Machine (Gaussian)	0.9448	+/- 0.0015	0.94	+/- 0.01
Ridge Classifier	0.8000	+/- 0.0001	0.80	+/- 00
Logistic regression	0.8008	+/- 0.0003	0.80	+/- 00
Decision tree classifier	0.9019	+/- 0.0041	0.89	+/- 00
Naïve Bays Classifier (Gaussian)	0.9427	+/- 0.0017	0.94	+/- 00
Kernel Support Vector Machine (Polynomial)	0.9309	+/- 0.0019	-	-
Random Forest Classifier	0.9474	+/- 0.0014	0.95	+/- 0.00
Extra tree classifier	0.966	+/- 0.0007	0.95	+/- 0.00

Ensemble machine learning techniques bagging, boosting, and stacking are implemented with the dataset. In bagging technique, after using base classifiers as KNN classifier, KSVM (Gaussian) classifier, ridge classifier, logistic regression, decision tree classifier, GNB classifier, and polynomial and KSVM classifier in meta bagging classifier, again random forest bagging classifier and extra tree bagging classifier, we have made average and voting manipulation. The results are summarized in Table 3, and Fig. 4 is representing ROC AUC of bagging classifiers.

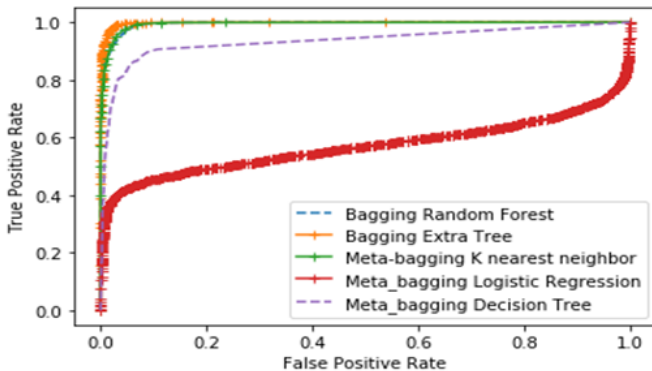


Figure 4. ROC AUC representation of bagging classifiers. Bagging Random Forest: ROC, AUC = 0.995; Bagging Extra Tree: ROC, AUC = 0.998; Meta-bagging K-NN: ROC, AUC = 0.994; Meta-bagging Logistic Regression: ROC, AUC = 0.570; Meta-bagging Decision Tree: ROC, AUC = 0.935.

In the boosting algorithm, we have considered models to the overall ensemble model sequentially. AdaBoost builds an additive logistic regression model by stage wise fitting. Using AdaBoost with the data set, we have an accuracy of 0.93. In a Gradient boosting algorithm, a decision tree is used as a weak learner and implementing with the data set, the accuracy of classification is 0.95. The accuracy of using XGBoost with the data set is 0.95. The summary of boosting algorithms' performance is presented in Table 4 and ROC, AUC of boosting classifiers are shown in Fig. 5.

The heterogeneous ensemble learning technique, stacking, is trained with the data set, and the meta-model is trained on the outputs of the base-level model. We have used six different base learn-

Table 4. Accuracy of Boosting algorithms implementation.

Boosting Methods	Accuracy	Standard Deviation
Ada Boost	0.93	+/- 0.00
Gradient boosting algorithm	0.95	+/- 0.00
XG Boost Algorithm	0.95	+/- 0.00

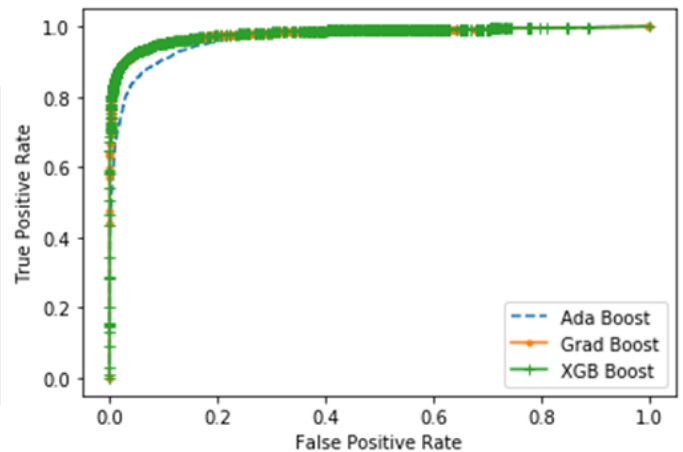


Figure 5. ROC AUC of Boosting classifiers. Ada Boost: ROC, AUC = 0.965; Grad Boost: ROC, AUC = 0.980; XGB Boost: ROC, AUC = 0.981.

ers, i.e., decision tree, GNB, K-NN, random forest, extra tree, and logistic regression, to train the dataset. Each model is fine-tuned using probability scores, and majority voting have got logistic regression. The accuracy is represented in Table 5 and Fig. 6, ROC, AUC description of stacking classifiers is presented.

4. Discussion

With the data set, we have used different machine learning techniques like CNN, bagging, boosting, stacking, and other traditional classifiers like KSVM, random forest, extra tree, ridge classifier, decision tree, K-NN, logistic regression, etc. From Table 2, we have got a traditional classifier; the random forest has given better accuracy as 0.95. Hence among all traditional classifiers, the random forest accuracy score is better. From Table 3, the extra tree

Table 5. Stacking implementation accuracy.

Base Estimators for Stacking	Voting to estimators	
	Accuracy	Standard Deviation
K-nearest neighbors classifier	0.9301	+/- 0.0015
Logistic regression	0.8048	+/- 0.0006
Decision tree classifier	0.8886	+/- 0.0014
Naïve BaysClassifier (Gaussian)	0.943	+/- 0.0011
Random forest classifier	0.947	+/- 0.0029
Extra tree classifier	0.9435	+/- 0.0030
Stack Classifier (second level classifier logistic regression)	0.951	+/- 0.0009

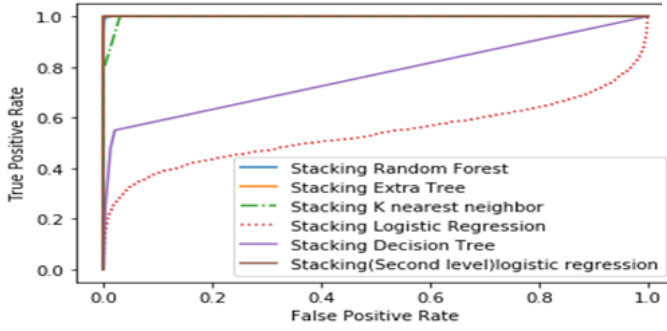


Figure 6. Representation ROC, AUC of stacking implementation. Stacking Random Forest: ROC, AUC = 1.000; Stacking Extra Tree: ROC, AUC = 1.000; Stacking K-NN: ROC, AUC = 0.997; Stacking Logistic Regression: ROC, AUC = 0.538; Stacking Decision Tree: ROC, AUC = 0.767; Stacking 2nd level classifier logistic regression: ROC AUC = 1.000.

bagging classifier has given the highest accuracy as 0.96. Hence extra tree bagging classifier has given better performance among all bagging classifiers. From Table 4, it is found both Gradient boosting, and XGBoosting has the highest accuracy 0.95. From Table 5, logistic regression as meta classifier, when implementing stacking, has accuracy 0.95. CNN's accuracy score is 0.96. All maximum scores of traditional classifiers, bagging classifiers, boosting, stacking, and CNN are summarized in Table 6, and ROC AUC is plotted in Fig. 7. Both CNN and extra tree bagging classifiers have shown the same and best accuracy in Table 6. ROC AUC of extra tree bagging is 1, whereas CNN is 0.99.

Table 6. Summary of optimal accuracy of classification for comparative study.

Classifiers	Accuracy	ROC AUC
CNN	0.96	0.99
Extra Tree Bagging (Average)	0.96	1.00
Gradient Boosting	0.95	0.98
XG Boosting	0.95	0.98
Stacking	0.95	1.00
Random Forest	0.95	1.00

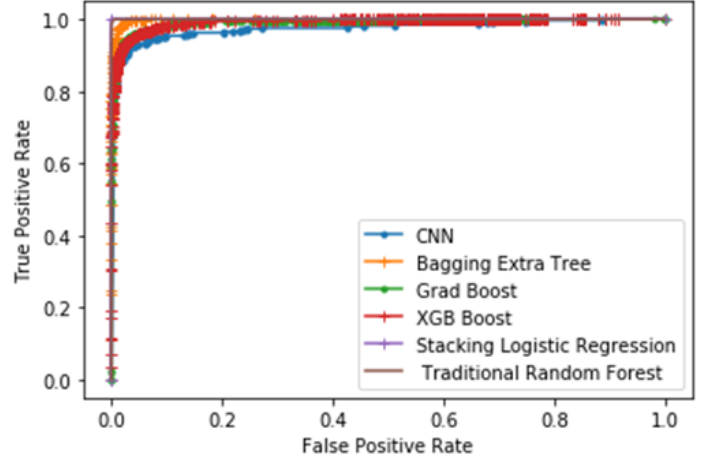


Figure 7. Performance summary by depicting ROC AUC of all the optimal classifiers (traditional, ensemble, and deep learning). CNN and Bagging Extra Tree outperforms as compared to classifiers based on the conventional machine learning approach.

Although both CNN and extra tree bagging classifier have the same accuracy ROC AUC of extra tree bagging classifier giving better, to implement extra tree bagging classifier first, we have eliminated some attributes using coefficient correlation. But in the case of CNN, it is not required. Different traditional machine learning techniques have used for classifying the epilepsy seizure from five categories of symptoms of EEG recording containing 11500 instances with 178 signal recording. Before using the techniques, the dataset is pre-processed to eliminate the attributes. Then ensemble machine learning algorithms are used to classify the data set after the same type of preprocessing, and ensemble classifiers have given better accuracy than traditional classifiers. Again, deep learning technique CNN is implemented with the data set without eliminating any attributes which have given approximately the same result with extra tree bagging classifier. The ensemble and deep learning models outperformed in comparison to traditional machine learning techniques and found effective in detecting epileptic seizures automatically.

Acknowledgment

We are thankful to the authors of the freely available dataset used in this paper.

Conflict of Interest

The authors declare no conflict of interest.

Submitted: February 03, 2020

Accepted: March 04, 2020

Published: March 30, 2020

References

- Acharya, U. R., Oh, S. L., Hagiwara, Y., Tan, J. H. and Adeli, H. (2018) Deep convolutional neural network for the automated detection and diagnosis of seizures using EEG signals. *Computers in Biology and Medicine* **100**, 270-278.
- Alhussein, M., Muhammad, G. and Hossain, M. S. (2019) EEG pathology detection based on deep learning. *IEEE Access* **7**, 27781-27788.

- Andrzejak, R. G., Lehnertz, K., Mormann, F., Rieke, C., David, P. and Elger, C. E. (2001) Indications of non-linear deterministic and finite-dimensional structures in time series of brain electrical activity: Dependence on recording region and brain state. *Physical Review E* **64**, 061907.
- Avcu, M. T., Zhang, Z. and Chan, D. W. S. (2019) 'Seizure detection using least EEG channels by deep convolutional neural network,' *ICASSP 2019 - 2019 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. Brighton, UK.
- Ay, B., Yildirim, O., Talo, M., Baloglu, U. B., Aydin, G., Puthankattil, S. D., and Acharya, U. R. (2019) Automated depression detection using deep representation and sequence learning with EEG signals. *Journal of Medical Systems* **43**, 205.
- Cacha, L., Parida, S., Dehuri, S., Cho, S. B. and Poznanski, R. R. (2016) A fuzzy integral method based on the ensemble of neural networks to analyze fMRI data for cognitive state classification across multiple subjects. *Journal of Integrative Neuroscience* **15**, 593-606.
- Clarke, S., Karoly, P., Nurse, E., Seneviratne, U., Taylor, J., Knight-Sadler, R., Kerr, R., Moore, B., Hennessy, P., Mendis, D., Lim, C., Miles, J., Cook, M. and Freestone, D. (2019) Computer-assisted EEG diagnostic review for idiopathic generalized epilepsy. *Epilepsy & Behavior*, 106556.
- Fukumori, K., Nguyen, H. T. T., Yoshida, N. and Tanaka, T. (2019) 'Fully data-driven convolutional filters with deep learning models for epileptic spike detection,' *ICASSP 2019 - 2019 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. Brighton, UK.
- Ilyas, M., Saad, P., Ahmad, M. and Ghani, A. (2016) 'Classification of EEG signals for brain-computer interface applications: Performance comparison,' *2016 International Conference on Robotics, Automation and Sciences (ICORAS)*. Ayer Keroh.
- Karim, A. M., Güzel, M. S., Tolun, M. R., Kaya, H. and Çelebi, F. V. (2018) A new generalized deep learning framework combining sparse autoencoder and Taguchi method for novel data classification and processing. *Mathematical Problems in Engineering* **2018**, 1-13.
- Karim, A. M., Güzel, M. S., Tolun, M. R., Kaya, H. and Çelebi, F. V. (2019) A new framework using deep auto-encoder and energy spectral density for medical wave-form data classification and processing. *Biocybernetics and Biomedical Engineering* **39**, 148-159.
- Lee, S. B., Kim, H., Lee, S., Kim, H. J., Lee, S. W. and Kim, D. J. (2019) 'Classification of the motion artifacts in near-infrared spectroscopy based on wavelet statistical feature,' *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*. Bari, Italy, 2019. IEEE.
- Mahato, S. and Paul, S. (2020) Classification of depression patients and normal subjects based on electroencephalogram (EEG) signal using alpha power and theta asymmetry. *Journal of Medical Systems* **44**, 28.
- Nandy, A., Alahe, M. A., Uddin, S. N., Alam, S., Nahid, A. A. and Awal, M. A. (2019) 'Feature extraction and classification of EEG signals for seizure detection,' *2019 International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST)*. Dhaka, Bangladesh.
- Parida, S., Dehuri, S., Cho, S. B., Cacha, L. and Poznanski, R. (2015) A hybrid method for classifying cognitive states from fMRI data. *Journal of Integrative Neuroscience* **14**, 355-368.
- Rahman, M. M., Bhuiyan, M. I. H. and Das, A. B. (2019) Classification of focal and non-focal EEG signals in VMD-DWT domain using ensemble stacking. *Biomedical Signal Processing and Control* **50**, 72-82.
- Resque, P., Barros, A., Rosário, D. and Cerqueira, E. (2019) 'An investigation of different machine learning approaches for epileptic seizure detection,' *2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC)*. Tangier, Morocco.
- Seifzadeh, S., Rezaei, M., Faez, K., and Amiri, M. (2017) Fast and efficient four class motor imagery electroencephalography signal analysis using common spatial pattern-ridge regression algorithm for the purpose of brain-computer interface. *Journal of Medical Signals and Sensors* **7**, 80-85.
- Siuly, S., Li, Y. and Zhang, Y. (2016) Injecting principal component analysis with the OA scheme in the epileptic EEG signal classification. In, Siuly, S. et al. (eds.) *EEG signal analysis and classification (pp. 141-144)*. Germany, CA: Springer.
- Struck, A. F., Rodriguez-Ruiz, A. A., Osman, G., Gilmore, E. J., Haider, H. A., Dhakar, M. B., Schrettnner, M., Lee, J. W., Gaspard, N., Hirsch, L. J., Westover M. B. and Critical Care EEG Monitoring Research Consortium (CCERMRC). (2019) Comparison of machine learning models for seizure prediction in hospitalized patients. *Annals of Clinical and Translational Neurology* **6**, 1239-1247.
- Tavares, G., San-Martin, R., Ianof, J. N., Anghinah, R. and Fraga, F. J. (2019) 'Improvement in the automatic classification of Alzheimer's disease using EEG after feature selection,' *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*. Bari, Italy.
- Thodoroff, P., Pineau, J. and Lim, A. (2016) 'Learning robust features using deep learning for automatic seizure detection,' *Machine Learning for Healthcare Conference (MLHC 2016)*. Los Angeles, USA.
- Wójcik, G. M., Kawiak, A., Kwasniewicz, L., Schneider, P. and Masiak, J. (2019) Azure machine learning tools efficiency in the electroencephalographic signal P300 standard and target responses classification. *Bio-Algorithms and Med-Systems* **15**, 1-8.
- Yuvaraj, R., Thomas, J., Kluge, T. and Dauwels, J. (2018) 'A deep learning scheme for automatic seizure detection from long-term scalp EEG,' *2018 52nd Asilomar Conference on Signals, Systems, and Computers*. Pacific Grove, CA, USA.