Detection and Prediction of Epileptic Seizure Using Different Machine Learning Classifiers

Nuna Tasnim
Department of CSE
BRAC University
Dhaka, Bangladesh
nuna.tasnim@g.bracu.ac.bd

Md. Ashikur Rahman
Department of CSE
BRAC University
Dhaka, Bangladesh
ashikurrahman1052@gmail.com

Subha Sumaiya B Jamal
Department of CSE
BRAC University
Dhaka, Bangladesh
subhasumaiya@gmail.com

Afnan Ahmed Crystal
Department of CSE
BRAC University
Dhaka, Bangladesh
afnan.ahmed.crystal@g.bracu.ac.bd

Dr. Amitabha Chakrabarty,
Dewan Ziaul Karim
Department of CSE
BRAC University
Dhaka, Bangladesh

***Abstract*—Analyzing neural signals produced by neurons in the brain, epilepsy can be diagnosed. An electroencephalogram (EEG) measures brain electrical activity, and studying EEG data in order to detect epileptic seizures in their early phases is an important aspect of epilepsy research. Despite optimal medication management, around one-third of epileptic patients continue to experience seizures. As a result, detecting epileptic seizures has become increasingly important in the field of research in recent years. It has been observed that machine learning has a revolutionary effect on classifying EEG data, seizure detection, and identifying sensible patterns without performance deterioration.This study provides a comprehensive summary of works on automated epileptic seizure recognition utilizing a variety of machine learning techniques, including SVC, Logistic Regression, Decision Tree Classifier, Random Forest Classifier, Gradient Boosting, and Multilayer Perceptron (MLP). Our research found that the Random Forest Classifier achieved a higher F1 score of 95.8967% compared to other classifiers when five groups of people were considered. Later, we implemented PCA and clustering to determine if we could improve the Random Forest Classifier's performance. After performing PCA for dimension reduction and K-means for clustering, we compare the F1 scores and cannot find any significant difference, which implies that our data set does not need any further clustering. Hence, the findings suggest that classification performance remains the same after implementing dimension reduction and clustering.**

***Index Terms*—EEG; Epilepsy; Seizure; Feature extraction; Classification; PCA**

I. Introduction

An epileptic seizure is a period of symptoms induced by abnormal or excessive neuronal activity in the brain, characterized by uncontrolled movements, shaking, and electrical motions in all or part of the body, as well as varied levels of consciousness or merely a short momentary loss of awareness [3]. According to research, about 3.5 out of every 10,000 people experience triggered seizures each year, while 4.2 out of every 10,000 experience unprovoked seizures. After a single seizure, the likelihood of another is approximately 50%. Also, about 1% of the population suffers from epilepsy at any one time, and 4% suffer from it at some point in their lives [2]. EEG refers to an electrophysiological examination of the brain's electrical activity. This electrical movement alluded to signs of brain function. Typically, the electrodes are put on the scalp to estimate voltage fluidity caused by ionic current inside the brain's neurons [5]. Either the event's potential outcomes or the intriguing content of EEG are usually the focus of specific segments. In this work, EEG motions in the frequency range are used to analyze the type of neural motion known as "brain waves." Any changes in those brain waves or signals indicate a problem. As a result, signal analysis is required to diagnose epilepsy since it identifies a person's brain signal that is comparable to epilepsy.

II. Related Works

Because epilepsy is a common neurological complication, programmatic classification of epileptic EEG events has become a focus of study in the past few years. There are several methodologies to approach machine learning. SVMs (support vector machines) and ANNs are two of the most widely used among them (artificial neural networks) [18]. Epilepsy and non-epileptic EEG are used to create models for neural network design, which can be used to categorize EEG signals. Preparatory data sets are used to gather features for model construction. The characteristics were selected for the objective of recording differences between epileptic and normal EEGs. Extracting features is critical to the success of ANN classification [1]. The SVM classifier is a general classifier that has shown outstanding results in identifying EEG signals based on data transfer capacity factors [4] [17] uses a Gaussian mixture model classifier, in which each class's probability density capacity is predicted as a combination of multidimensional Gaussian distributions. Our classifier outperforms regression trees and KNN (K-nearest neighbor). In [5], a number of classic machine

learning methods are used to look at the EEG dataset including logistic regression as a base learner/classifier. From the labeled training data, we can figure out what the constants are in the equation. If the parameters are accurate, the prediction will be mechanical. The training data is randomly segmented in a random forest classifier, and the decision tree is built with each sample of data. In the random forest, only a restricted feature is evaluated when splitting a non-leaf node, whereas in the additional tree, when dividing a non-leaf node, every characteristic is considered. In general, It was revealed that the deviation of a decision tree classifier is greater than that of a random forest classifier and greater than that of an extra tree. When the random forest classifier was used, it constructed a series of decision trees from a randomized piece of the training dataset and merged the scores from different decision trees to get the resulting class of the test object. The rest of this paper is organized in the following manner: The data and methods used in this study are explained in Section III. In section IV, we examined and then analyzed the data. Finally, in section V, we explored the challenges we faced in compiling the models throughout the thesis, as well as our future intentions.

III. Data Method

A. Dataset

Our work employed a preprocessed version of a commonly used epileptic seizure detection dataset from the UCI Machine Learning Repository [6]. The dataset consists of 500 patients and contains 4097 EEG readings for each subject over the course of 23.5 seconds. After that, the 4097 data points were split up into 23 unique chunks for each patient, and each of those pieces was turned into a single row in the dataset. Each row has 178 readings, which are then transformed into columns; to put it another way, one second's worth of EEG readings is 178 columns. There are a total of 11,500 rows and 179 columns, with the very last column indicating the patient's status, which indicates whether or not the patient is experiencing a seizure at the moment. The response variable, y, can be found in column 179, and the independent variables are X1,.., and X178. y includes a response variable, which can be written as y in 1, 2, 3, 4, or 5:

Value of response variable y	Description
5	When the EEG was being recorded, eyes were open
4	When the EEG signal was being recorded, eyes were closed
3	Found the position of the tumor within the brain, as well as the recorded EEG activity from the healthy part of the brain
2	The EEG is recorded at the tumor's location
1	Recordings of seizure activity

Table I represents the brief description about the dataset.

B. Dataset Preprocessing

Data preprocessing is a process that puts unprocessed data into a format that is easy to understand so that our dataset can be better understood. Initially, we eliminated the first column from the dataset because it was irrelevant to our machine learning algorithm. Since our main goal is to determine whether or not a patient is having a seizure using 178 EEG signals per second, we converted the multiclass dataset to a binary classification problem by replacing the y column values greater than 1 with 0 and leaving the 1 values unchanged. In this case, 1 denotes a patient who is experiencing a seizure, and 0 indicates that the patient is not having a seizure. We chose Stratified Shuffle Split as our binary classifier as it consists of 0 and 1, with a total of 9200 0s and 2300 1s. However, in this instance, implementing a random split could result in a bias toward 9200. To overcome this issue, we used Stratified Shuffle Split, which takes 80 percent of data from both class 0 and class 1, then combines them to create a combined 80 percent trained dataset that is optimal and unbiased because it is a blend of both classes. We performed dimension reduction to determine if the performance of our best classifier improved after dimension reduction. As our dataset is so huge, dimension reduction may be necessary. It is really difficult to manipulate this massive dataset. We used principal component analysis (PCA) to cut down on the number of dimensions as much as possible, hence removing ineffective columns.

C. Algorithms

Afterwards, we'll go through our model workflow, the machine learning classifiers we applied, and the testing, validation, and evaluation methods we used in our study on epileptic seizure detection and prediction. Our workflow includes data preprocessing, data splitting, classifier implementation, testing validation, and evaluation of performance analysis. It will provide a comprehensive summary of work on automated epileptic seizure detection using a variety of machine learning algorithms, including SVC, Logistic Regression, Decision tree, Random Forest, Gradient Boosting, and MLP.

SVC: The Support Vector Classifier, or SVC, is the most appropriate machine learning method for our case. SVC does not allow multiclass categorization by default. It enables binary classification and data item separation into two categories. A SVC's goal is to fit the data given to it, providing a "best fit" hyperplane that divides or categorizes it. After obtaining the hyperplane, we may input some characteristics into the classifier in order to get the "predicted" class. This makes this algorithm particularly ideal for our needs, although it may be used in a variety of circumstances. [14].It's a one-vs-one C-support vector classification that deals with multiclass support.In our dataset, we applied SVC as it lets us choose from a variety of non-linear classifiers. kernel RBF, c=0.025.

Logistic Regression: The categorical dependent variable is predicted using logistic regression, applying a set of independent factors. However, we used this algorithm to solve

classification problems in our model. To predict the outcome, a sigmoidal activation function is given a linear equation. [20] To evaluate the model's performance, we calculate the loss. The most common loss function is mean squared error. The mean squared error is not the best option because the output of logistic regression is a probability value between 0 and 1.

Decision Tree: Decision Tree is the most effective even among highly prominent classifiers, which have outstanding performance in several areas. This is a result of its capacity to construct a number of rational explanatory logic rules with excellent prediction accuracy. In this tree-structured classifier, input data comprises dataset characteristics and variables, branches indicate decision rules, and each leaf node delivers the output. A decision tree has two nodes: the decision node and the leaf node. Unlike leaf nodes, which are the results of decisions and have no further branches, Decision nodes can be used to make any kind of decision. The dataset is stored in the root node. While implementing a decision tree, we must pick the best characteristic for the root node and subnodes.

Random Forest: Random forest generates decision trees from numerous samples by using the majority vote for classification and the average for regression. This approach is used in regression and classification to handle data sets containing continuous and categorical variables. However, it produces higher classification accuracy. In our paper, we used it for classification to get our final output, which is based on majority voting [9].We created a decision tree, which uses a tree-like flowchart to represent the predictions that result from a series of feature-based splits. A Random Forest is a bagging method that makes predictions using a subset of the original dataset, which helps to overcome overfitting. A random forest creates many decision trees, each with a different collection of observations, rather than a single decision tree [7] [8].

Gradient Boosting: We chose gradient boosting as a classifier because our dataset comprises categorical variables and the cost function is log loss. The Gradient Boosting Classifier implements a loss function. This classifier can use a user-defined loss function, as well as a variety of standardized loss functions. The loss function, however, must be differentiable. In classification algorithms, the logarithmic loss is commonly utilized. In addition, we used logarithmic loss in our analysis [10]. Learning rate is one of the most important parameters for gradient boosting decision trees. The learning rate refers to how quickly the model picks up new information. The overall model is altered by each new tree. The learning rate measures the strength of the change. Primarily, we are concerned about the classifier's accuracy on the validation set, although it shows that a learning rate of 0.05 provides the best validation set performance as well as decent training set performance.

MLP: Since there are multiple layers of neurons, MLP is a deep learning technique that is used for solving problems in our computational neuroscience. There are three levels in the hypothesized MLP design: input, output, and a hidden layer. Layer 1 is the input layer, it conducts normalization on all numbers of features. In this study, we just employed one hidden layer to minimize classification time and hardware resource consumption [15].MLP trains with two arrays. Training samples are stored as floating-point feature vectors in Array X, while performance measures (class labels) for the training data are stored in Array Y (n samples, n classes) [16]. The controller parameters are described by a collection of neurons in the input layer. In the hidden layer, each neuron performs a weighted linear sum on the data from the previous layer, followed by a non-linear activation function such as the hyperbolic tan function. The output layer receives the values from the final concealed layer. Finally, it turns them into output values in the output layer. [11] [12].

K-means: K-Means Clustering is used in machine learning and data science to divide unlabeled datasets into groups, with each dataset belonging to only one group with similar qualities. The number of groups that must be formed during the process is set by K. The algorithm starts with an unlabeled dataset, divides it into k groups, and keeps doing this until no better groups can be found, resulting in each cluster comprising data points that share some characteristics but are isolated from the others. In this algorithm, k should always have the same value [13]. We do, however, have domain knowledge: these features correspond to 5 y-categories, so there should be 5 clusters. So, we used K=5.

IV. Result & analysis

We evaluated model accuracy, F1-score, recall, and precision, as well as the confusion matrix while analyzing the results. This section displays the results of classifying the epileptic seizure dataset using multiple classifiers.

A. Confusion Matrix Analysis:

The performance of our classification models is compared in detail using confusion matrices for two binary classes, 0 and 1, where 0 means no seizure and 1 means seizure. There are four possible classification. The four possible classification outcomes are True-Positive (TP), True-Negative (TN), False-Positive (FP), and False-Negative (FN). In the table, we have shown the True-Positive (TP), True-Negative (TN), False-Positive (FP), and False-Negative (FN) values of different classifiers.

Classifiers	TP	TN	FP	FN
Logistic Regression	174	1288	552	286
SVC	378	1812	28	82
Decision Tree Classifier	393	1780	60	67
Random Forest Classifier	430	1819	21	30
Gradient Boosting	399	1820	20	61
Multi-layer perceptron	344	1667	173	116

Following table II, an analysis of the confusion matrix, it is evident that the Random Forest Classifier fits best our dataset as it provides us with the least amount of error (51) compared to other used classifiers while predicting seizures.

B. ROC Curve Analysis:

In this section, the AUC, which is a measure of a classifier's ability to differentiate between classes and is used to summarize the ROC curve, is included.

Classifiers	Train Auc	Test Auc
Logistic Regression	0.596	0.495
SVC	0.992	0.990
Decision Tree Classifier	1.000	0.903
Random Forest Classifier	1.000	0.997
Gradient Boosting	0.999	0.994
Multi-layer Perceptron	0.998	0.848

From the table III, it is clear that AUC is good for all six classifiers except Logistic Regression. Random Forest has the highest train and tests AUC, which is 100 and 99 percent, and slightly higher than the Gradient Boosting Classifier's train and test AUC, which is 99.9 and 99.4 percent, respectively. In terms of ROC and confusion matrix, the Random Forest Classification performed impressively.

C. Result Comparison:

Different performance measures, including Accuracy, Recall, Precision, and F1-score were compared to illustrate the differences among classifiers.

Classifiers	Accuracy	Recall	Precision	F1 score
Logistic regression	63.5652%	53.9130%	52.8983%	53.9130%
SVC	95.2174%	90.3261%	94.3870%	90.3261%
Random forest classifier	97.6087%	95.8967%	96.5848%	95.8967%
Decision Tree Classifier	94.0000%	90.4620%	90.7283%	90.4620%
Gradient boosting classifier	96.4783%	92.8261%	95.9919%	92.8261%
Multilayer perceptron	87.6522%	82.9891%	80.3321%	82.9891%

Here, the table (IV) illustrates the comparison of performance among the six classifiers which we used in our research with respect to different performance metrics such as accuracy, recall, precision, and F1-score. Logistic regression has an accuracy rate of 63.5652%, which is the lowest accuracy in our findings with a recall rate of 53.9130%, a very low precision rate of 52.8983% and an F1 score of 53.9130%. It is because there are fewer observations than features. The MLP classifier performs better than logistic regression. We got an 87.6522.0% accuracy rate and a 80.3321% precision rate. The decision tree classifier obtains 94.0000% accuracy and a 90.7283% precision rate. The SVC classifier gives better results compared to logistic regression, MLP, and decision tree classifiers. It performs 95.2174% of accuracy and 90.3261% of precision, with an F1 score of 90.3261%. We achieved the best performance by using the random forest classifier in our findings. It gives 97.6087% accuracy with a high precision of 96.5848%, a recall rate of 95.8967% and an F1 score of 95.8967%. The very high values of the indicators might be due to the embedded feature selection of our model. The second best performance was obtained by the gradient boosting classifier. It gives a 96.4783% accuracy rate and a 95.9919% precision rate, which is also sufficiently good. We considered the F1 score as our key indicator because it compares precision and recall and works well with unbalanced datasets. Therefore, compared to the other five models, the random forest classifier is the model of preference in this study, with the highest F1 score for our unbalanced dataset. We compared the performance of our best-fit classifier, Baseline Random Forest, to that of PCA with RF and clustering to see if clustering could increase performance. We used k-means clustering and hierarchical clustering to show the difference.

Classifiers	Accuracy	Recall	Precision	F1 score
Logistic regression	63.5652%	53.9130%	52.8983%	53.9130%
SVC	95.2174%	90.3261%	94.3870%	90.3261%
Random forest classifier	97.6087%	95.8967%	96.5848%	95.8967%
Decision Tree Classifier	94.0000%	90.4620%	90.7283%	90.4620%
Gradient boosting classifier	96.4783%	92.8261%	95.9919%	92.8261%
Multilayer perceptron	87.6522%	82.9891%	80.3321%	82.9891%

Table V shows that the F1 score of the random forest increased after performing dimension reduction. But if we compare the F1 scores of three different clusters, clustering makes a slight difference here, but the effect is negligible because the performance is already maxed out. So we can come to the conclusion that our dataset was preprocessed, so it doesn't need any further clustering.

V. Conclusion

Reliable and accurate detection is becoming more common, so epilepsy is becoming more necessary nowadays. In this study, we thoroughly studied and examined a wide range of machine learning classifiers for seizure detection.Therefore, we come to the conclusion that the 'Random Forest' classifier comes out to be the most successful in our study. It gives 97.6087% accuracy with a precision of 96.5848% and F1 score of 95.8967%. We considered model accuracy, model loss, recall, and precision, as well as the confusion matrix when evaluating performance. The results suggest that by implementing dimension reduction and clustering, classification performance can be slightly improved. For example, dimension reduction improved the F1 score of the random forest classifier by 4%, which is 99.89%, but clustering couldn't bring any changes to

the performance as the dataset was already preprocessed and clustered. Our future goal for this research is to implement hyperparameter tuning of our random forest model. We hope that it will solve the bias and variance performance of the model. Developing ensemble models may be of benefit. although our performance is nearly excellent at this moment.

REFERENCES

[1] Ocak, H. (2009). Automatic detection of epileptic seizures in eeg using discrete wavelet transform and approximate entropy. Expert Systems with Applications, 36:2027–2036.

[2] Jouny, C. C., Franaszczuk, P. J., Bergey, G. K. (2011). Improving early seizure detection. Epilepsy behavior : E&B, 22 Suppl 1(Suppl 1), S44–S48. https://doi.org/10.1016/j.yebeh.2011.08.029

[3] U. Rajendra Acharya, Filippo Molinari, S. Vinitha Sree, Subhagata Chattopadhyay, Kwan-Hoong Ng, Jasjit S. Suri, Automated diagnosis of epileptic EEG using entropies, Biomedical Signal Processing and Control, Volume 7, Issue 4, 2012, Pages 401-408, ISSN 1746-8094, https://doi.org/10.1016/j.bspc.2011.07.007

[4] Bajaj, V. and Pachori, R. B. (2012). Classification of seizure and nonseizure eeg signals using empirical mode decomposition. IEEE Transactions on Information Technology in Biomedicine, 16:1135–1142.

[5] Teplan, M. (2002). Fundamentals of EEG measurement. Measurement science review, 2(2), 1-11.

[6] Q. Wu and E. Fokoue, Epileptic seizure recognition data set, May 2017. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/Epileptic+Seizure+ Recognition.

[7] Donos, C., Dümpelmann, M., Schulze-Bonhage, A. (2015). Early Seizure Detection Algorithm Based on Intracranial EEG and Random Forest Classification. International Journal of Neural Systems, 25(05), 1550023. https://doi.org/10.1142/s0129065715500239

[8] Siddiqui, M. K., Morales-Menendez, R., Huang, X., Hussain, N. (2020). A review of epileptic seizure detection using machine learning classifiers. Brain Informatics, 7(1). https://doi.org/10.1186/s40708-020-00105-1

[9] R, S. E. (2021, June 24). Random Forest — Introduction to Random Forest Algorithm. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2021/06/understanding-random-forest/

[10] Nelson, D. (2019, August 8). Gradient Boosting Classifiers in Python with Scikit-Learn. Stack Abuse. https://stackabuse.com/gradient-boosting-classifiers-in-python-with-scikit-learn/#:%7E:text=Gradient%20boosting%20classifiers%20are%20a,used%20when%20doing%20gradient%20boosting.

[11] Sriraam, N., Raghu, S., Tamanna, K., Narayan, L., Khanum, M., Hegde, A. S., Kumar, A. B. (2018). Automated epileptic seizures detection using multi-features and multilayer perceptron neural network. Brain Informatics, 5(2). https://doi.org/10.1186/s40708-018-0088-8

[12] NN - Multi-layer Perceptron Classifier (MLPClassifier) - Michael Fuchs Python. (2021, February3). Michael-Fuchs-Python.Netlify.https://michael-fuchs-python.netlify.app/2021/02/03/nn-multi-layer-perceptron-classifier-mlpclassifier/?fbclid=IwAR0vcjnHrYh4-XioVa2myysCshz55IO9WPBPTeRl3OzF-5jWiIxPxEako3Ymlpclassifier-for-binary-classification

[13] Understanding K-Means Clustering Algorithm - Analytics Vidhya. (2022). Retrieved 11 May 2022, from https://www.analyticsvidhya.com/blog/2021/11/understanding-k-means-clustering-in-machine-learningwith-examples/: :text=The%20K%2Dmeans%20clustering%20algorithm,K'%20in%20K%2Dmeans

[14] Support Vector Machines. (2007–2022). Scikit-Learn. https://scikit-learn.org/stable/modules/svm.htmlsvm-classification

[15] Gaikwad, N. B., Tiwari, V., Keskar, A., Shivaprakash, N. C. (2019). Efficient FPGA Implementation of Multilayer Perceptron for Real-Time Human Activity Classification. IEEE Access, 7, 26696–26706. https://doi.org/10.1109/access.2019.2900084

[16] Peixoto, F. (2020, December 18). A Simple Overview of Multilayer Perceptron (MLP) Deep Learning. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2020/12/mlp-multilayer-perceptron-simple-overview/

[17] U. Rajendra Acharya, S. Vinitha Sree, A. P. C. A. J. S. S. (2012). Use of principal component analysis for automatic classification of epileptic eeg activities in wavelet framework. Expert Systems with Applications, 39:9072–9078.

[18] M. Kemal Kiymik, Abdulhamit Subasi, H. R. O. (2004). Neural networks with periodogram and autoregressive spectral analysis methods in detection of epileptic seizure. Journal of Medical Systems, 28:511–522.

[19] Sahu, R., Dash, S. R., Cacha, L. A., Poznanski, R. R., Parida, S. (2020). Epileptic seizure detection: a comparative study between deep and traditional machine learning techniques. Journal of integrative neuroscience, 19(1), 1-9

[20] H. Bonthu, An introduction to logistic regression, Jul. 2021. [Online]. Available: https://www.analyticsvidhya.com/blog/2021/07/an-introduction-to-logisticregression/: :text=Logistic%20Regression%20is%20a%20%E2%80%9CSupervised,used%20for%20Binary%20classification%20problems