COMPARATIVE STUDY OF EEG SIGNAL CLASSIFICATION USING DEEP LEARNING FOR BIOMEDICAL APPLICATIONS

MAJOR PROJECT
SUBMITTED in PARTIAL FULFILLMENT of
THE REQUIREMENTS FOR THE AWARD OF THE DEGREE of

BACHELOR OF TECHNOLOGY

(COMPUTER SCIENCE & ENGINEERING)

SUBMITTED BY

Deepanshi Dabas, Ayushi & Mehak Lakhani – 35196302716, 40596302716, 35496302716

Under the Guidance of

(Dr. Bharti Sharma, Assistant Professor)

Department of Computer Science & Engineering
MAHARAJA SURAJMAL INSTITUTE OF TECHNOLOGY, JANAKPURI DELHI-58
GURU GOBIND SINGH INDRAPRASTHA UNIVERSITY
DELHI, INDIA

JAN 2020 – MAY 2020

DECLARATION

We the undersigned solemnly declare that the project report titled "Comparative study of EEG signal classification using deep learning for biomedical applications" is based on our own work carried out during the course of our study under the supervision of Dr. Bharti Sharma, Assistant Professor (Department of Computer Science & Engineering, 2nd shift). We assert the statements made and conclusions drawn are an outcome of our research work. We further certify that the work has not been submitted to any other Institution for any other degree/diploma/certificate in this university or any other University of India or abroad. We have followed the guidelines provided by the university in writing the report. All the materials (data, theoretical analysis, and text) from other sources, have been given due credit in the text of the report and their details have been mentioned in the references.

Deepanshi Dabas, Ayushi, Mehak Lakhani

(35196302716, 40596302716, 35496302716)

Date:

CERTIFICATE

This is to certify that the project entitled "Comparative study of EEG signal classification using deep learning for biomedical applications" is a bonafide work carried out by Ms. Deepanshi Dabas, Ms. Ayushi and Ms. Mehak Lakhani under our guidance and supervision and submitted in partial fulfillment of B.Tech degree in Computer Science & Engineering of Maharaja Surajmal Institute of Technology affiliated by Guru Gobind Singh Indraprastha University, Delhi. The work embodied in this project has not been submitted for any other degree or diploma.

Dr. Bharti Sharma
(Assistant Professor)

Dr. Naveen Dahiya
HOD, Computer Science & Engineering, 2^{nd} Shift MSIT,
Delhi

ACKNOWLEDGEMENT

We truly acknowledge the cooperation and help made by Dr. Bharti Sharma, Assistant Professor Department of Computer Science and Engineering, Maharaja Surajmal Institute of Technology, C-4 Janakpuri, Delhi, Guru Gobind Singh Indraprastha University, Delhi. She has been a constant source of guidance throughout the course of this project. We are also thankful to our friends and family whose silent support led us to complete our project.

Deepanshi Dabas, Ayushi and Mehak Lakhani
(35196302716), (40596302716), (35496302716)

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

EEG is widely used in the research involving biomedical engineering (e.g. brain computer interfaces, BCI), neural engineering, neuroscience, seizure detection and sleep analysis because of its high temporal resolution, non-invasiveness, and relatively low financial cost. EEG is becoming increasingly important in the diagnosis and treatment of mental and brain related diseases and abnormalities. This study aims to perform a comparative study of different deep learning algorithms and implementing them on the datasets of two disorders – Epilepsy Seizure and Schizophrenia. Early detection of these disorders can facilitate a wider scope of treatments that can come of aid to the patient. Using EEG as a tool for fast detection of the diseases at an early stage and integrating this with deep learning tools can provide better diagnosis. Recently, deep learning has shown a significant role in this field due to its capacity to learn good feature representations from raw data and could be extended further for a generalized detection of neural disorders.

CHAPTER 1 - INTRODUCTION

As the power of modern computers grows alongside our understanding of the human brain, we move ever closer to making some pretty spectacular science fiction into reality. Imagine transmitting signals directly to someone's brain that would allow them to see, hear or feel specific sensory inputs. Consider the potential to manipulate computers or machinery with nothing more than a thought. Training the patients suffering from neural disabilities, to initiate brain signals for controlling external devices, provides scope for extensive future development. The electrical physiological activity in the brain can be recorded using a non-invasive monitoring method, EEG (Electroencephalography). EEG measures voltage fluctuations within the neurons of the brain. Recently, deep learning has aroused wide interest in machine learning fields. Deep learning is a multilayer perceptron artificial neural network algorithm and it has the advantage of approximating the complicated function and alleviating the optimization difficulty associated with deep models [1]. Machine learning has demonstrated truly life-impacting potential in healthcare – particularly in the area of biomedical science. The breadth, complexity, and rapidly expanding size of biomedical data have stimulated the development of novel deep learning methods, and application of these methods to biomedical data have led to scientific discoveries and practical solutions.[2]

1.1 EEG SIGNAL CLASSIFICATION

An electroencephalogram is a test used to evaluate the electrical activity in the brain. Brain cells communicate with each other through electrical impulses. An EEG can be used to help detect potential problems associated with this activity.
An EEG (Electroencephalography) tracks and records brain wave patterns. Small metal discs with thin wires (electrodes) are placed on the scalp, and then send signals to a computer to record the results. EEG is widely used in research involving biomedical engineering (e.g. brain computer interfaces, BCI), neural engineering, neuroscience, seizure detection and sleep analysis because of its high temporal resolution, non-invasiveness, and relatively low financial cost.

The electrical impulses in an EEG recording look like wavy lines with peaks and valleys. These lines allow doctors and researchers to quickly assess whether there are abnormal patterns. Any irregularities may be a sign of seizures or other brain disorders.

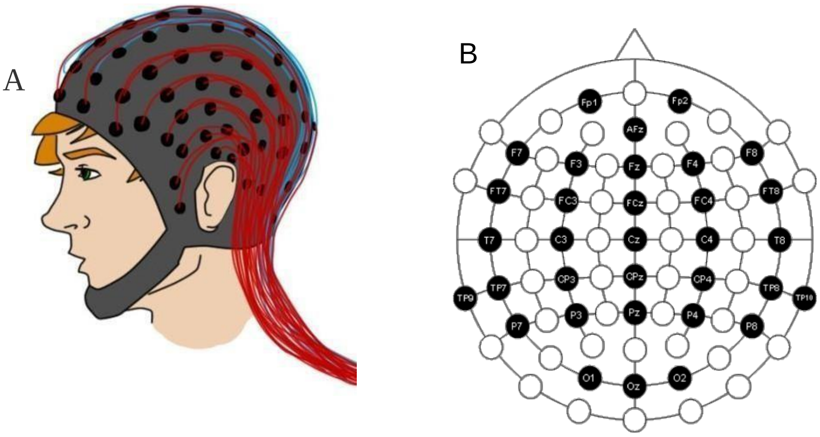


Figure 1.1 EEG Brain cap illustration (A) the placement of the brain cap on a human operator (B) placement of the electrode on a human scalp to measure physiological changes

An EEG is used to detect problems in the electrical activity of the brain that may be associated with certain brain disorders. The measurements given by an EEG are used to confirm or rule out various conditions, including:

- Seizure disorders
- Epilepsy
- Schizophrenia
- Brain tumor
- Encephalopathy
- Stroke
- Sleep Disorders

1.2 DEEP LEARNING MODELS

Deep learning allows computational models that are composed of multiple processing layers to learn representations of data with multiple levels of abstraction [3]. **Deep Learning** is a subfield of machine learning concerned with algorithms inspired by the structure and function of the brain called **artificial neural networks.**

The core of deep learning is that we now have fast enough computers and enough data to actually train large neural networks. Deep learning methods aim at learning feature hierarchies with features from higher levels of the hierarchy.

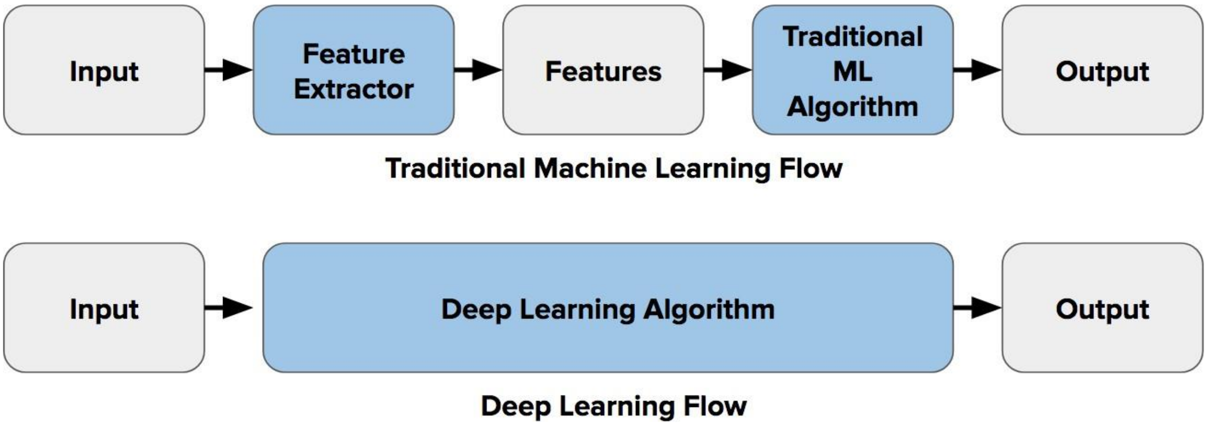


Figure 1.2 Comparison of Traditional Machine Learning and Deep Learning flow

Deep Learning comprises various algorithms. Due to implementation of these algorithms, various fields like biomedical sciences, armed forces, educational sector and many more can be benefitted and can be made advanced. Deep learning algorithms perform feature extraction in an automated way, which allows researchers to extract discriminative features with minimal domain knowledge and human effort. [4]

Different types of models used in Deep Learning:

Supervised Models

- Classic Neural Networks
- Convolutional Neural Networks (CNNs)
- Recurrent Neural Networks (RNNs)

Unsupervised Models

- Self-Organizing Maps (SOMs)
- Boltzmann Machines
- AutoEncoders

In deep learning models, data is filtered through a cascade of multiple layers, with each successive layer using the output from the previous one to inform its results. Deep learning models can become more and more accurate as they process more data, essentially learning from previous results to refine their ability to make correlations and connections.

In artificial neural networks (ANNs), the basis for deep learning models, each layer may be assigned a specific portion of a transformation task, and data might traverse the layers multiple times to refine and optimize the ultimate output. These "hidden" layers serve to perform the mathematical translation tasks that turn raw input into meaningful output.

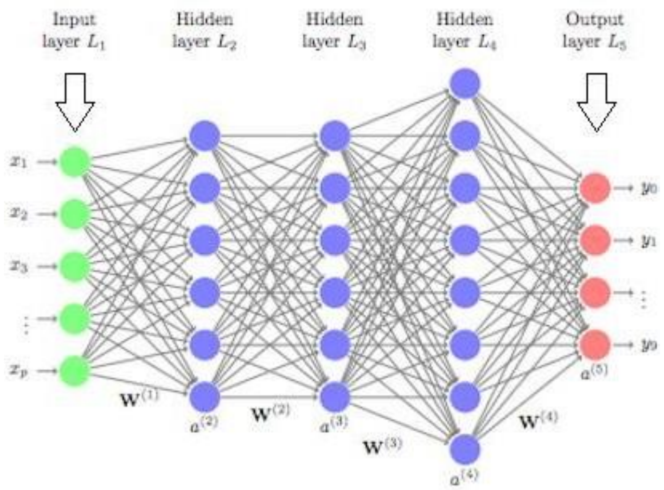


Figure 1.3 Input, Output and Hidden layers of ANN

For building a deep learning model, a procedure is being followed which include various steps. First, the datasets are collected. If the data is clean then we can proceed further but if not, then pre-processing of the dataset is performed to make it clean and free of errors. Second, read and understand the data used as input for the model. Different software like Pandas is used to achieve the goal. Third, a model is built by the functions present in the software. Fourth is to compile the model. Compiling the model took two basic parameters: Optimizer (controls the learning rate) and Loss (calculate the average squared difference between actual and predicted values). Last is to train the data and use the model to make predictions on new data.

1.3 DEEP LEARNING IN BIOMEDICAL SCIENCES

With a massive influx of multimodality data, the role of data analytics in biomedical health informatics has grown rapidly in the last decade. This has also prompted increasing interests in the generation of analytical, data driven models based on machine learning in this field. Deep learning, a technique with its foundation in artificial neural networks, is emerging in recent years as a powerful tool for machine learning, promising to reshape the future of artificial intelligence. [5]

Biomedical sciences are a set of sciences applying portions of natural science or formal science, or both, to knowledge, interventions, or technology that are of use in healthcare or public health [6]. As such the biomedical sciences have a much wider range of academic and research activities and economic significance than that defined by hospital laboratory sciences. It is yielding a large amount of data daily from research and development (R&D), physicians and clinics, patients, caregivers etc. These data can be used as synchronizing the information and using it to improve healthcare infrastructure and treatments.

Applications of Deep Learning in Biomedical Science-

- Disease Identification/Diagnosis
- Personalized Treatment/Behavioral Modification
- Drug Discovery/Manufacturing
- Clinical Trial Research
- Smart Electronic Health Records
- Epidemic Outbreak Prediction
- Radiology and Radiotherapy

In our project, we focused on brain-related diseases. Due to deep learning algorithms present, early detection of such diseases can be made possible. This results in fast improvement in the health of the person and so it can contribute in this field on a large scale.

The first applications of deep learning to clinical data were on image processing, especially on the analysis of brain Magnetic Resonance Imaging (MRI) scans to predict Alzheimer disease and its variations [7]. Several works applied deep learning to predict diseases from the patient clinical status.

Liu et al. used a four-layer CNN to predict congestive heart failure and chronic obstructive pulmonary disease and showed significant advantages over the baselines. [8]. RNNs with long short-term memory (LSTM) hidden units, pooling and word embedding were used in DeepCare, an end-to-end deep dynamic network that infers current illness states and predicts future medical outcomes [9]. Miotto et al. proposed to learn deep patient representations from the EHRs using a three-layer Stacked Denoising Autoencoder (SDA) [10].

Deep learning methods are powerful tools that complement traditional machine learning and allow computers to learn from the data, so that they can come up with ways to create smarter applications. These approaches have already been used in a number of applications, especially for computer vision and natural language processing. It can open the way toward the next generation of predictive health care systems, which can scale to include billions of patient records and rely on a single holistic patient representation to effectively support clinicians in their daily activities

CHAPTER 2 -
LITERATURE SURVEY

S.No	AUTHOR	YEAR	TITLE	ALGORITHM	DESCRIPTION
1	Yurun Shen et al	2017	Classification of Motor Imagery EEG Signals with Deep Learning Models	Deep Forest Algorithm	Ensembles method with a cascade structure which enables the model to do representation learning
2	Antonio Maria Chiarelli et al	2018	Deep Learning for hybrid EEG-fNIRS Brain-Computer Interface: application to Motor Imagery Classication	Deep Artificial Neural Networks (DNNs)	Able to learn representations of data with multiple levels of abstraction
3	Syed Umar Amin et al	2019	Deep Learning for EEG motor imagery classification based on multi-layer CNNs feature fusion	Multi-layer CNNs feature fusion	Utilizes different convolutional features to capture spatial and temporal features from raw EEG data
4	Xiao Zheng et al	2019	Decoding human brain activity with deep learning	Long short-term memory-convolutional neural network (LSTM-CNN)	It helps to decode the human brain activity by using an image-EEG-image transformation
5	S.Ramkumar et al	2019	Performance Analysis of EEG Signals using Conventional and Hybrid Artificial Neural Network	Feed forward neural network with wolf Grey optimization algorithm (FFNNWGOA)	22 features extracted from the variance features was trained and tested with ten hidden neurons using hybrid Grey wolf Optimization algorithm based neural network classifier
6	K. Saranya, S. Jayanthy	2018	BCI based EEG Signals for Emotion Classification	Quadratic Discriminant Analyser Classifier	QDA is closely related to LDA, where it is assumed that the measurements are normally distributed

7	Sarmiento, L. C.,Cortes, C. J.	2019	Brain Computer Interface (BCI) with EEG Signals for Automatic Vowel Recognition based on Articulation Mode	nonlinear SVM	Classifying a non-linearly separable dataset using a SVM – a linear classifier: ... However, it can be used for classifying a non-linear dataset
8	Anh Kha Vo	2019	Subject-Independent ERP-Based Brain-Computer Using Adaptive and Ensemble Learning	Domain Adversarial Neural Network	Consisting of an input layer, a sigmoid function output layer, and three 1000-node hidden layers with hyperbolic tangent activation function
9	Anh Kha Vo	2019	Subject-Independent ERP-Based Brain-Computer Using Adaptive and Ensemble Learning	GANN	Key principle is that regulatory regions are composed of features such as consensus strings, characterized binding sites, and DNA structural properties
10	Anh Kha Vo	2019	Subject-Independent ERP-Based Brain-Computer Using Adaptive and Ensemble Learning	DCGANN Deep Convolutional Generative Adversarial Neural Network	Composed of ConvNets in place of multi-layer perceptrons
11	Anurag Nishad et al	2020	Classification of epileptic electroencephalogram signals using tunable-Q wavelet transform based filter-bank	Tunable-Q wavelet transform (TQWT)	Different Q-factor values generate various mother wavelets, which are suitable to analyse different EEG signals
12	S Raghu et al	2020	EEG based multi-class seizure type classification using convolutional neural network and transfer learning	Convolution neural network (CNN) and Transfer Learning	CNN has been found to be an ideal for image-based classification due to its self-feature learning capability

13	Yu-Cheng Liu et al	2020	Design exploration predicts designer creativity: a deep learning approach	Long short-term memory-convolutional neural network (LSTM-CNN)	LSTM networks, and their mixtures were examined and the results of applying visualisation methods revealed variations in brain activity
14	D. Jude Hemanth	2020	EEG signal based Modified Kohonen Neural Networks for Classification of Human Mental Emotions	Kohonen neural network	Kohonen neural network is a single layer, unsupervised, feed forward artificial neural
15	Meysam Golmohammadi et al	2019	Automatic Analysis of EEGs Using Big Data and Hybrid Deep Learning Architectures	Stacked denoising Autoencoder (SdA)	Composed of multiple layers of denoising autoencoders in a way that the input to each layer
16	Kaushalya Kumarasinghe et al	2019	Deep learning and deep knowledge representation in Spiking Neural Networks for Brain-Computer Interfaces	Brain-Inspired Spiking Neural Network (BI-SNN)	Learn and reveal deep in time-space functional and structural patterns from spatio-temporal data
17	Kemal Akyol et al	2020	Stacking ensemble based deep neural networks modeling for effective epileptic seizure detection	Stacking ensemble approach	Performs a multi-class ensemble learning following the two-stage procedure
18	Parikshat Sirpal et al	2019	fNIRS improves seizure detection in multimodal EEG-fNIRS recordings	Recurrent neural networks (RNNs)	Success of RNNs in these domains lead to applying LSTM-RNNs for human seizure activity detection in multimodal EEG-fNIRS recordings

19	Kemal Akyol	2019	Comparing of deep neural networks and extreme learning machines based on growing and pruning approach	Deep Neural Networks and Extreme Learning Machines	Determine the ideal parameters i.e. optimum hidden layer number, optimum hidden neuron number and activation function for DNN
20	M. Shamim Hossain et al	2019	Applying Deep Learning for Epilepsy Seizure Detection and Brain Mapping Visualization	Deep Convolutional Neural Network (CNN) model	Evaluating the power of a deep CNN to learn robust features from raw Electroencephalogram (EEG) data to detect seizures
21	Xuebin Tang et al	2019	A Novel Classification Algorithm for MI-EEG based on Deep Learning	Convolutional Neural Network (CNN) and Stacked AutoEncoders (SAE)	Deep learning end-to-end classification model which is combined with CNN and stacked autoencoders (SAE)
22	Ihsan Ullah et al	2018	An automated system for epilepsy detection using EEG brain signals based on deep learning approach	Pyramidal one-dimensional convolutional neural network (P-1D-CNN) models	Works on the concept of refinement approach
23	Mona Nasseri et al	2019	Semi-supervised training data selection improves seizure forecastingin canines with epilepsy	Seizure forecasting algorithm and Hierarchical clustering method	Compared the performance of a seizure forecasting algorithm with and without hierarchical clustering
24	Shu Lih Oh et al	2019	Deep Convolutional Neural Network Model for Automated Diagnosis of Schizophrenia Using EEG Signals	Convolutional neural network (CNN) model	Collected EEG signals from 14 healthy subjects and 14 Schizophrenia(SZ) patients and developed an eleven-layered CNN model to analyze the signals.

25	Srivathsan Srinivasagopalan et al	2019	A deep learning approach for diagnosing schizophrenic patients	Deep neural network (DNN)	Analyzing the dataset for the existence of schizophrenia using traditional machine learning approaches such as LR, SVM, and random forest

Table 2.1 Literature Survey

CHAPTER 3 - PROPOSED SOLUTION

3.1 PROBLEM STATEMENT

EEG is becoming increasingly important in the diagnosis and treatment of mental and brain related diseases and abnormalities. Using EEG as a tool for fast detection of the diseases at an early stage and integrating this with machine learning tools can provide better diagnosis. Recently, deep learning has shown a significant role in this field due to its capacity to learn good feature representations from raw data and could be extended further for a generalized detection of neural disorders.

- For early detection of neural disorders like Epilepsy, Schizophrenia, Parkinson's etc. and facilitating a wider scope of treatments that can come of aid to the patient.
- To perform a comparative analysis on the use of EEG methodology for detection of neural disorders and their accurate diagnosis using machine learning tools.
- To understand different algorithms and train a deep learning model for the classification.
- Implement the model on the given data sets and observe various parameters like accuracy, recall rate and precision provided by different models and adapt the one with highest accuracy.
- Various algorithms have to be implemented and their accuracies have to be observed to select the most efficient and accurate deep learning algorithm.

3.2 METHODOLOGY

This study aims to perform a comparative analysis on the use of EEG methodology for detection of neural disorder and their accurate diagnosis using machine learning tools. The timeline of this study comprises of:

3.2.1 Data Acquisition

This is the first paradigm of the timeline. Datasets are gathered or acquired in data acquisition step. For our study, we have gathered the datasets of two neural disorders from the online sources. The datasets are for:

A. Schizophrenia Dataset :

Schizophrenia is a serious mental disorder in which people interpret reality abnormally. Schizophrenia may result in some combination of hallucinations, delusions, and extremely disordered thinking and behavior that impairs daily functioning, and can be disabling.

B. Epilepsy Seizure Recognition Dataset :

Epilepsy is a central nervous system (neurological) disorder in which brain activity becomes abnormal, causing seizures or periods of unusual behavior, sensations, and sometimes loss of awareness.

3.2.2 Data Preprocessing

Data preprocessing is a data mining technique that involves transforming raw data into an understandable format. Real-world data is often incomplete, inconsistent, and/or lacking in certain behaviors or trends, and is likely to contain many errors. Data preprocessing is a proven method of resolving such issues. Data preprocessing prepares raw data for further processing.

For Epilepsy:

EEG data set has acquired different age groups in this study. They are known epileptic with uncontrolled seizures and are admitted to the neurology department of the Medical Faculty Hospital of Dicle University1. For this system LabView programming language has been used and the EEG data used in 400 people who received 200 of them are epilepsy and with 200 of them are normal. Data set represents signals belonging to several healthy and epileptic patients. The EEG signals that are contained by PCI-MIO 16E DAQ card system that provides real time processing and is a data bus of computer, signal processor and personal computer. EEG signals ensure the accuracy of diagnosing disease that usually is taken 8-10 hours in the form of records. EEG signals are used in section and 23.6 seconds, 173 Hz sampling frequency is illustrated with. International 10–20 electrode placement system according to the data collected, 12-bit analog-digital conversion after the samples are recorded subsequently. Data can be passed through the filter of 0.53–40 Hz band–pass, the EEG in the presence of clinical interest for focusing range is provided. The EEG data used in our study were downloaded from 24-h EEG recorded from both epileptic patients and normal subjects.

For Schizophrenia :

The original dataset from the reference consists of 5 different folders, each with 100 files, with each file representing a single subject/person. Each file is a recording of brain activity for 23.6 seconds. The corresponding time-series is sampled into 4097 data points. Each data point is the value of the EEG recording at a different point in time. So we have a total of 500 individuals with 4097 data points for 23.5 seconds. We divided and shuffled every 4097 data points into 23 chunks, each chunk contains 178 data points for 1 second, and each data point is the value of the EEG recording at a different point in time. So now we have 23 x 500 = 11500 pieces of information (row), each information contains 178 data points for 1 second (column), the last column represents the label y {1, 2, 3, 4, 5}. The response variable is y in column 179, the Explanatory variables X1, X2, ..., X17; y contains the category of the 178-dimensional input vector. Specifically y in {1, 2, 3, 4, 5}

3.2.3 Data Modeling

Data modeling is the process of creating a data model for the data to be stored in a database. It helps in the visual representation of data and enforces business rules, regulatory compliances, and government policies on the data. After the two datasets have been preprocessed, we have applied four deep learning algorithms on them to find which algorithm has the highest accuracy and is best suited for our purpose. The detailed study of the algorithms used are as follows:

- ANN (Artificial Neural Networks)

  Artificial Neural Network (ANN) uses the processing of the brain as a basis to develop algorithms that can be used to model complex patterns and prediction problems. It is a computing system vaguely inspired by biological neural networks. An ANN is based on a collection of connected units or nodes called neurons, which loosely model the neurons in a biological brain. Each connection can transmit a signal to other neurons.

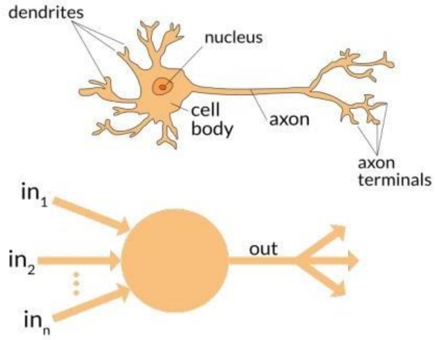


Figure 3.1 Representation of ANN w.r.t neurons

The artificial neural network (ANN)–a machine learning technique inspired by the human neuronal synapse system–was introduced in the 1950s. [36]

ANNs have the ability to learn and model non-linear and complex relationships, which is really important because in real-life, many of the relationships between inputs and outputs are non-linear as well as complex.

After learning from the initial inputs and their relationships, it can infer unseen relationships on unseen data as well, thus making the model generalize and predict on unseen data. Unlike many other prediction techniques, ANN does not impose any restrictions on the input variables (like how they should be distributed). Additionally, many studies have shown that ANNs can better model data with high volatility and non-constant variance, given its ability to learn hidden relationships in the data without imposing any fixed relationships in the data.

Artificial Neural Network is capable of solving the complex problem which cannot be solved by the traditional machine learning problems. Neural networks has many applications like object recognition, Automatic speech recognition, machine translation, image captioning, video classification, translating text and so on.

- CNN (Convolutional Neural Networks)

Convolutional Neural Networks (CNN) is one of the variants of neural networks used heavily in the field of Computer Vision. It derives its name from the type of hidden layers it consists of. The hidden layers of a CNN typically consist of convolutional layers, pooling layers, fully connected layers, and normalization layers. Here it simply means that instead of using the normal activation functions defined above, convolution and pooling functions are used as activation functions.

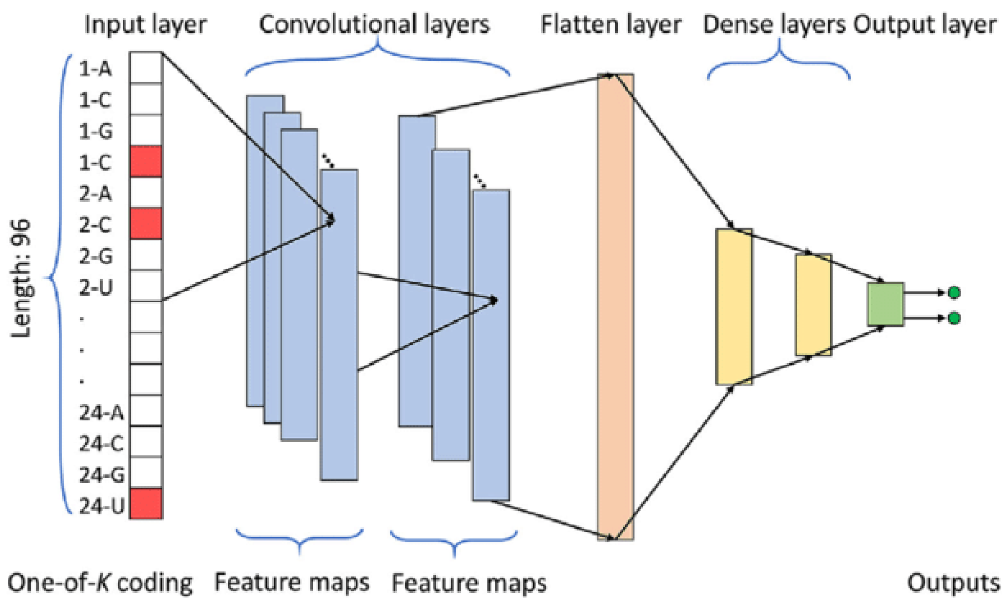


Figure 3.2 Architecture of CNN

Convolutional networks are simply neural networks that use convolution in place of general matrix multiplication in at least one of their layers. CNN is a specialized type of neural network for processing data that has a grid-like topology, such as image data, which can be regarded as a two-dimension (2D) grid of pixels [33]. The performance of CNN increases with more effective training methods and breakthroughs in hardware technology, and more and more people used CNN for medical image-related research [34]. Convolution operates on two signals (in 1D) or two images (in 2D): you can think of one as the "input" signal (or image), and the other (called the kernel) as a "filter" on the input image, producing an output image (so convolution takes two images as input and produces a third as output). In layman terms it takes in an input signal and applies a filter over it, essentially multiplies the input signal with the kernel to get the modified signal. Mathematically, a convolution of two functions f and g is defined as

$$(f * g)(i) = \sum_{j=1}^{m} g(j) \cdot f(i - j + m/2)$$

which, is nothing but dot product of the input function and a kernel function

Following steps can be followed:

- Provide input image into convolution layer
- Choose parameters, apply filters with strides, padding if requires. Perform convolution on the image and apply ReLU activation to the matrix.
- Perform pooling to reduce dimensionality size
- Add as many convolutional layers until satisfied
- Flatten the output and feed into a fully connected layer (FC Layer)
- Output the class using an activation function (Logistic Regression with cost functions) and classifies images.

- RNN (Recurrent Neural Networks)

Recurrent Neural Networks or RNN as they are called in short, are a very important variant of neural networks heavily used in Natural Language Processing. In a general neural network, an input is processed through a number of layers and an output is produced, with an assumption that two successive inputs are independent of each other. RNN is the first algorithm that is able to rememers its inputs, due to its internal states that makes RNN perfect for machine learning problems that involve sequential data. Another way to think about RNNs is that they have a “memory” which captures information about what has been calculated so far. In theory, RNNs can make use of information in arbitrarily long sequences, but in practice, they are limited to looking back only a few steps. RNN has shown to be hugely successful in natural language processing especially with their variant LSTM, which are able to look back longer than RNN.

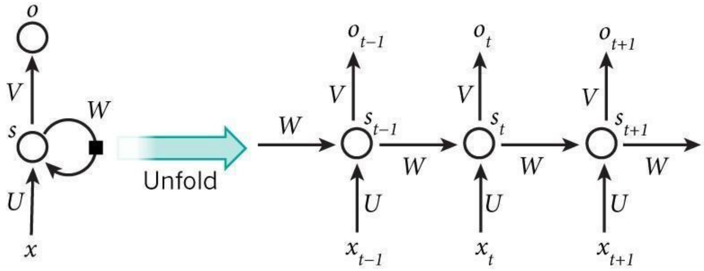


Figure 3.3 Illustration of RNN

Figure : Architecture wise, an RNN looks like this. One can imagine it as a multilayer neural network with each layer representing the observations at a certain time t.

Because of their internal memory, RNN's are able to remember important things about the input they received, which enables them to be very precise in predicting what's coming next. This is the reason why we preferred them for sequential data like time series, speech, text, financial data, audio, video, weather and much more because they can form a much deeper understanding of a sequence, compared to other algorithms. Augmenting a high-level structure with learning capability of RNNs leads to a powerful tool that has the best of both worlds. [35]

In RNN, the information cycles through a loop. When it makes a decision, it takes into consideration the current input and also what it has learned from the inputs it received previously.

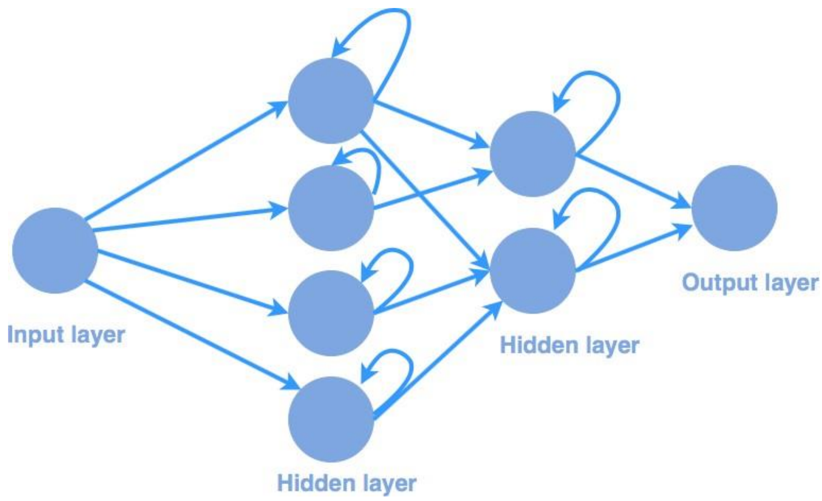


Figure 3.4 Information flow in RNN

- LSTM (Long-Short term Memory)

A LSTM cell is a type of RNN which stores important information about the past and forgets the unimportant pieces. In this way, when gradient back-propagates, it won't be consumed by unnecessary information.

LSTM enables RNN to remember its inputs over a long period of time. This is because

LSTM contains its information in a memory that is much like the memory of a computer because the LSTM can read, write and delete information from its memory. So because of LSTM, now RNN can analyze text and answer questions, which involves keeping track of long sequences of words.

This memory can be seen as a gated cell, where gated means that the cell decides whether or not to store or delete information (e.g if it opens the gates or not), based on the importance it assigns to the information. The assigning of importance happens through weights, which are also learned by the algorithm. This simply means that it learns over time which information is important and which not.

In LSTM, you have three gates: input, forget and output gate. These gates decide whether or not to let new input in (input gate), delete the information because it isn't important (forget gate) or to let it effects the output at the current time step (output gate). You can see an illustration of a RNN with its three gates below:

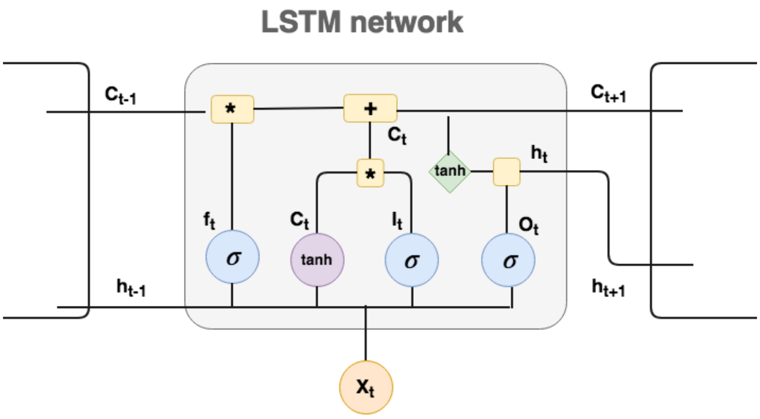


Figure 3.5 A LSTM network

LSTM's learning algorithm is local in space and time; its computational complexity per time step and weight is O(1). It solves complex long-time lag tasks that have never been solved by previous RNN algorithms. (37)

There are several applications where LSTM is highly used. Applications like time series, finance, music composition, speech recognition, handwriting recognition, image captioning and in my current research of human mobility and travel predictions.

3.2.4 Comparative Analysis

All the algorithms mentioned above are performed on the Epilepsy and Schizophrenia dataset. After implementation of the code performed on the dataset, various parameters are considered such as accuracy, precision value, recall rate and F1 score. The results obtained from these algorithms helped us in comparing and understanding which algorithm is better. The algorithm with highest accuracy can then be used for the treatment purpose further and can help us in achieving our goal. Deep learning can help in early detection of these disorders which can be beneficial in the treatment process.

CHAPTER 4 - IMPLEMENTATION

4.1 SOFTWARE SPECIFICATIONS

The following project in its entirety has been implemented using the following software platforms,

→ MATLAB

MATLAB combines a desktop environment tuned for iterative analysis and design processes with a programming language that expresses matrix and array mathematics directly. It includes the Live Editor for creating scripts that combine code, output, and formatted text in an executable notebook. MATLAB toolboxes are professionally developed, rigorously tested, and fully documented. MATLAB apps let you see how different algorithms work with your data. Iterate until you've got the results you want, then automatically generate a MATLAB program to reproduce or automate your work. EEGLAB is an interactive Matlab toolbox for processing continuous and event- related EEG. EEGLAB runs under Linux, Unix, Windows, and Mac OS X. EEGLAB provides an interactive graphic user interface (GUI) allowing users to flexibly and interactively process their high-density EEG and other dynamic brain data using independent component analysis (ICA) and/or time/frequency analysis (TFA), as well as standard averaging methods. EEGLAB also incorporates extensive tutorial and help windows, plus a command history function that eases users' transition from GUI-based data exploration to building and running batch or custom data analysis scripts.

→ JUPYTERLAB

JupyterLab is a web-based interactive development environment for Jupyter notebooks, code, and data. JupyterLab is flexible: configure and arrange the user interface to support a wide range of workflows in data science, scientific computing, and machine learning. JupyterLab is extensible and modular: write plugins that add new components and integrate with existing ones. You can arrange multiple documents and activities side by side in the work area using tabs and splitters. JupyterLab also offers a unified model for viewing and handling data formats. JupyterLab understands many file formats (images, CSV, JSON, Markdown, PDF, Vega, Vega-Lite, etc.) and can also display rich kernel output in these formats.

→ MICROSOFT EXCEL

Microsoft Excel has the basic features of all spreadsheets, using a grid of cells arranged in numbered rows and letter-named columns to organize data manipulations like arithmetic operations. It has a battery of supplied functions to answer statistical, engineering and financial needs. In addition, it can display data as line graphs, histograms and charts, and with a very limited three-dimensional graphical display. It allows sectioning of data to view its dependencies on various factors for different perspective. In a more elaborate realization, an Excel application can automatically poll external databases and measuring instruments using an update schedule, analyze the results, make a Word report or PowerPoint slide show, and e-mail these presentations on a regular basis to a list of participants. Excel was not designed to be used as a database.

4.2 DATASETS

For this study, datasets have been acquired from online sources. Link has been mentioned in the section 2.2.1.

4.2.1 Schizophrenia Dataset

Schizophrenia is a chronic mental illness that affects about 1% of people across the globe. One possible explanation for some of the symptoms of schizophrenia is that one or more problems with the corollary discharge process in the nervous system make it difficult for patients to differentiate between internally and externally generated stimuli. Therefore, studying this process and its relationship to symptoms in the illness might allow us to better understand abnormal brain processes in patients with this diagnosis.

EEG data from 22 controls and 36 patients with schizophrenia have been combined with 10 controls and 13 patients.

	subject	Fz	FCz	Cz	FC3	FC4	C3	C4	CP3	CP4	group
0	1	1.85745789473684	2.66335157894737	3.05191789473684	1.78901052631579	1.9610652631578900	4.9499010526315800	2.28283578947368	4.66476210526316	3.42579157894737	0
1	1	1.76969157894737	2.60173368421053	3.00564842105263	1.8684284210526300	1.7744536842105300	5.02446210526316	2.19453684210526	4.73727684210526	3.27253052631579	0
2	1	1.76479263157895	2.5421936842105300	3.00281157894737	1.8734	1.80001894736842	4.9576536842105300	2.15280210526316	4.70050842105263	3.17171157894737	0
3	1	1.8849800000000000	2.63246	3.19882105263158	1.98757789473684	2.02819368421053	5.09169473684211	2.33792105263158	4.91442	3.38507684210526	0
4	1	1.59151578947368	2.3694642105263200	2.96531789473684	1.86897894736842	1.85311684210526	4.89432947368421	2.26296842105263	4.73030315789474	3.3353852631578900	0
5	1	1.26641368421053	2.10466842105263	2.56489684210526	1.63683578947368	1.55818736842105	4.61721263157895	2.02600631578947	4.35392631578947	3.043252631578950 0	0
6	1	1.03354421052632	1.8988547368421100	2.45406105263158	1.52338210526316	1.46166526315789	4.552207368421050	1.96629263157895	4.26984842105263	2.97469473684211	0
7	1	0.6826810526315790	1.5454336842105300	2.13274631578947	1.27269157894737	1.12402315789474	4.22836421052632	1.66714736842105	3.9251	2.64553684210526	0
8	1	0.380174736842105	1.31245157894737	1.89597368421053	1.12740526315789	0.828367368421053	3.8514673684210500	1.40796315789474	3.57313894736842	2.3424263157894700	0
9	1	0.376265263157895	1.30062736842105	1.9482547368421100	1.05204105263158	0.8421305263157890	3.91238947368421	1.43986842105263	3.53397368421053	2.39066631578947	0
10	1	0.741969473684211	1.5783063157894700	2.23281473684211	1.2971389473684200	1.1414421052631600	4.1439705263157900	1.71629157894737	3.77417684210526	2.72960947368421	0
11	1	1.13786105263158	1.86214105263158	2.44058105263158	1.5276989473684200	1.52753789473684	4.3446547368421000	2.05196947368421	4.04179684210526	3.01270315789474	0
12	1	1.28401684210526	1.9782147368421100	2.49624526315789	1.64807157894737	1.64369894736842	4.47822210526316	2.26590315789474	4.08130947368421	3.1595178947368400	0
13	1	1.4153452631578900	2.0895568421052600	2.5799957894736800	1.75748421052632	1.84854842105263	4.63903263157895	2.5035757894736800	4.04207368421053	3.39619368421053	0
14	1	1.50551578947368	2.15808315789474	2.64949578947368	1.8112642105263200	2.00535157894737	4.69916105263158	2.62447789473684	4.05883157894737	3.5158336842105300	0
15	1	1.29966947368421	1.9597315789473700	2.4062136842105300	1.58247263157895	1.71426421052632	4.53126315789474	2.34649684210526	3.88984105263158	3.2358600000000000	0
16	1	1.14356631578947	1.69397473684211	2.18717473684211	1.3203452631578900	1.47190526315789	4.30164842105263	2.0592600000000000	3.50229157894737	2.90369578947368	0
17	1	1.0587600000000000	1.59443368421053	2.11788315789474	1.16775368421053	1.41516736842105	4.23390526315789	1.9342073684210500	3.4036473684210500	2.73489578947368	0
18	1	0.862311578947369	1.45008842105263	2.1140021052631600	1.05787263157895	1.3120263157894700	4.23247157894737	1.81642842105263	3.41543578947368	2.6368042105263200	0
19	1	0.901377894736842	1.52435894736842	2.15626526315789	1.12382631578947	1.4111494736842100	4.17056526315789	1.80165263157895	3.447552631578950 0	2.61699368421053	0
20	1	1.07970105263158	1.65227263157895	2.15881789473684	1.11918947368421	1.47748631578947	4.26903684210526	1.92371157894737	3.49434842105263	2.7888736842105300	0
21	1	1.3906852631578900	1.86504736842105	2.2832873684210500	1.2322600000000000	1.7186694736842100	4.41635684210526	2.1960705263157900	3.6103642105263200	2.97966315789474	0
22	1	1.34338	1.86916315789474	2.37760736842105	1.25924210526316	1.81774947368421	4.32695894736842	2.2084378947368400	3.7471789473684200	2.9560473684210500	0
23	1	0.8454757894736840	1.47514	2.01287578947368	0.834471578947368	1.4500863157894700	3.96542105263158	1.90520631578947	3.38754315789474	2.7229957894736800	0
24	1	0.6484178947368420	1.2759452631578900	1.72387894736842	0.5745231578947370	1.17894105263158	3.69527052631579	1.6040231578947400	3.0010084210526300	2.40017684210526	0
25	1	0.782385263157895	1.37606105263158	1.8014947368421100	0.6771463157894740	1.2249610526315800	3.75227157894737	1.61478315789474	3.00903473684211	2.3585347368421100	0
26	1	1.07558526315789	1.63267789473684	2.15051684210526	1.10582842105263	1.5927273684210500	4.10152210526316	2.04241684210526	3.29798631578947	2.77751684210526	0
27	1	1.37798526315789	1.9311705263157900	2.4281989473684200	1.4274610526315800	1.9771357894736800	4.44640842105263	2.4861642105263200	3.7350536842105300	3.29699473684211	0
28	1	1.41491157894737	2.05731684210526	2.53229789473684	1.42283578947368	2.18159578947368	4.63592736842105	2.62629789473684	3.88036315789474	3.50046526315789	0
29	1	1.38381368421053	2.22937473684211	2.78563052631579	1.37847263157895	2.37420526315789	4.82590842105263	2.7632452631578900	3.9690368421052600	3.6389642105263200	0
30	1	1.39414736842105	2.21356421052632	2.89466105263158	1.4459905263157900	2.37253684210526	4.88997157894737	2.8162221052631600	4.01749263157895	3.70801368421053	0
31	1	1.4499084210526300	2.34263157894737	2.99046210526316	1.5639778947368400	2.43234	4.89751578947368	2.83654631578947	4.0914968421052600	3.72341473684211	0
32	1	1.6521821052631600	2.4579336842105300	3.09917368421053	1.66021263157895	2.5910357894736800	4.96511894736842	2.88009578947368	4.24794526315789	3.8215021052631600	0
33	1	1.7346336842105300	2.47320631578947	3.13184947368421	1.7373042105263200	2.59471894736842	5.0435105263157900	2.91378315789474	4.24238105263158	3.87802210526316	0
34	1	1.47913263157895	2.2717473684210500	2.96779368421053	1.6886347368421100	2.30734105263158	4.94282210526316	2.67824315789474	4.07664947368421	3.64460631578947	0
35	1	1.36244842105263	2.1849968421052600	2.7669494736842100	1.62167368421053	2.13640210526316	4.92353473684211	2.44209473684211	3.96052210526316	3.445652631578950 0	0
36	1	1.25354631578947	2.13833578947368	2.61177157894737	1.46674	2.06008631578947	4.88614526315789	2.38475473684211	3.90004736842105	3.36676315789474	0

Figure 4.1 Schizophrenia Dataset

4.2.2 Epilepsy Seizure Recognition Dataset

Epilepsy is a serious brain illness that is an endemic neurological disorder all over the world. It is a clinical result that occurs with abnormal neurological electrical discharging of brain. Epileptic seizures represent the most common positive signs and symptoms of brain disturbance, and epilepsy is one of the most common primary brain disorders.

The frequency components of the EEG are extracted by using the discrete wavelet transform (DWT) and parametric methods based on autoregressive (AR) model. Both these two feature extraction methods are applied to the input of machine learning algorithms.

Unnamed	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	X14	X15	X16	X17	X18	X19	X20	X21	X22
X21.V1.79	135	190	229	223	192	125	55	-9	-33	-38	-10	35	64	113	152	164	127	50	-47	-121	-138	-125
X15.V1.92	386	382	356	331	320	315	307	272	244	232	237	258	212	2	-267	-605	-850	-1001	-1109	-1090	-967	-746
X8.V1.1	-32	-39	-47	-37	-32	-36	-57	-73	-85	-94	-99	-94	-96	-104	-103	-92	-75	-69	-69	-53	-37	-14
X16.V1.60	-105	-101	-96	-92	-89	-95	-102	-100	-87	-79	-72	-68	-74	-80	-83	-73	-68	-61	-58	-59	-64	-79
X20.V1.54	-9	-65	-98	-102	-78	-48	-16	0	-21	-59	-90	-103	-84	-43	-9	3	-21	-60	-96	-103	-75	-29
X14.V1.56	55	28	18	16	16	19	25	40	52	66	81	98	111	122	105	85	66	51	34	19	16	8
X3.V1.191	-55	-9	52	111	135	129	103	72	37	0	-38	-77	-113	-128	-121	-105	-71	-27	13	44	60	64
X11.V1.27	1	-2	-8	-11	-12	-17	-15	-16	-18	-17	-19	-18	-16	-15	-14	-21	-19	-24	-24	-24	-17	-20
X19.V1.87	-278	-246	-215	-191	-177	-167	-157	-139	-118	-92	-63	-39	-11	14	36	60	70	78	79	69	27	-45
X3.V1.491	8	15	13	3	-6	-8	-5	4	25	41	48	44	34	16	-2	-11	-24	11	33	43	48	42
X3.V1.6	-5	15	28	28	9	-29	-41	-19	14	30	22	-6	-30	-40	-42	-48	-50	-55	-58	-66	-49	-20
X21.V1.72	-167	-230	-280	-315	-338	-369	-405	-392	-298	-140	27	146	211	223	214	187	167	166	179	192	190	168
X7.V1.162	92	49	0	-32	-51	-65	-37	-19	-25	-29	-52	-62	-85	-107	-97	-69	-46	-37	-48	-59	-58	-61
X1.V1.211	15	12	0	-17	-28	-31	-39	-51	-44	-35	-20	1	16	24	22	26	27	22	16	14	26	34
X1.V1.615	-24	-15	-5	-1	4	3	6	10	11	7	8	12	10	10	5	-1	-11	-13	-24	-39	-44	-52
X22.V1.24	-135	-133	-125	-118	-111	-105	-102	-93	-94	-90	-82	-75	-71	-69	-69	-69	-61	-59	-57	-64	-66	-65
X1.V1.863	39	41	41	42	43	43	46	47	49	50	52	52	53	59	58	63	62	64	59	57	55	50
X9.V1.302	9	4	-5	-10	-22	-30	-33	-43	-41	-40	-42	-46	-47	-52	-50	-51	-43	-34	-23	-6	4	10
X7.V1.541	-21	-5	1	7	19	20	13	2	-1	-3	-3	-14	-18	-21	-2	17	39	56	65	58	31	19
X9.V1.915	4	24	51	76	92	102	104	101	90	80	53	32	9	5	17	42	72	94	103	106	107	106
X23.V1.96	410	451	491	541	581	641	736	757	692	435	61	-387	-823	-1107	-1188	-1110	-947	-765	-600	-471	-376	-301
X1.V1.614	-24	-27	-23	-28	-34	-40	-47	-43	-38	-23	-1	7	18	7	11	-1	-18	-22	-21	-2	15	35
X11.V1.13	-264	-189	-117	-45	20	70	111	143	161	179	194	200	193	164	128	92	67	57	38	-21	-141	-239
X18.V1.54	-4	40	78	123	149	185	197	189	167	141	129	140	167	183	182	154	124	102	85	63	34	1
X19.V1.29	593	328	88	-106	-456	-732	-921	-782	-522	-248	-68	89	221	342	336	219	82	-32	-83	-114	-134	-134
X21.V1.80	-16	-15	-19	-16	-14	-5	0	-1	-3	-5	-7	-6	-6	4	15	16	25	32	32	33	33	34
X2.V1.72	-20	-38	-53	-58	-66	-66	-69	-77	-87	-84	-82	-72	-58	-47	-50	-65	-81	-98	-105	-102	-97	-95
X3.V1.744	-340	-381	-376	-336	-275	-204	-131	-70	-16	20	46	60	68	76	80	85	87	88	86	77	63	44
X12.V1.73	-30	15	61	80	72	41	-11	-31	-47	-63	-53	-48	-41	-49	-51	-48	-47	-52	-57	-60	-81	-82

Figure 4.2 Epilepsy Dataset - A

X161	X162	X163	X164	X165	X166	X167	X168	X169	X170	X171	X172	X173	X174	X175	X176	X177	X178	y
-85	-109	-98	-72	-65	-63	-11	10	8	-17	-15	-31	-77	-103	-127	-116	-83	-51	4
123	136	127	102	95	105	131	163	168	164	150	146	152	157	156	154	143	129	1
-11	-39	-44	-42	-45	-48	-42	-6	29	57	64	48	19	-12	-30	-35	-35	-36	5
-42	-51	-68	-71	-69	-69	-74	-74	-80	-82	-81	-80	-77	-85	-77	-72	-69	-65	5
-2	4	18	27	27	14	15	11	10	4	2	-12	-32	-41	-65	-83	-89	-73	5
-76	-79	-64	-40	-25	0	9	12	-6	-12	-31	-42	-54	-60	-64	-60	-56	-55	5
141	129	95	41	-21	-77	-117	-135	-137	-125	-99	-79	-62	-41	-26	11	67	128	4
-110	-110	-109	-104	-118	-111	-102	-80	-67	-79	-91	-97	-88	-76	-72	-66	-57	-39	2
248	349	418	419	291	73	-152	-311	-386	-400	-379	-336	-281	-226	-174	-125	-79	-40	1
2	-5	-6	13	41	66	72	68	65	49	31	11	-5	-17	-19	-15	-15	-11	4
-9	2	11	24	29	16	-8	-36	-51	-38	-4	25	16	-16	-74	-101	-89	-49	5
-258	-168	-32	140	277	366	408	416	415	423	434	416	374	319	268	215	165	103	1
-23	-39	-43	-32	-18	-30	-51	-72	-80	-56	-41	-40	-43	-32	-13	-1	-7	-44	3
129	121	112	100	83	63	41	-2	-51	-88	-102	-97	-77	-45	-19	13	44	68	4
-13	-4	3	15	19	21	25	29	33	32	35	36	34	32	26	23	18	20	2
-35	-34	-39	-37	-48	-58	-64	-64	-57	-49	-39	-35	-29	-10	4	21	31	37	3
54	50	48	49	47	48	47	42	42	43	41	41	43	43	40	41	41	49	2
-8	-8	2	10	18	28	27	28	28	34	27	22	18	15	13	9	9	3	3
28	6	-8	-4	8	27	40	54	54	43	28	25	19	30	35	26	5	-13	4
4	5	13	13	14	17	11	7	2	3	5	10	19	31	36	40	43	36	2
181	237	278	315	342	366	385	399	409	415	428	463	510	562	607	667	748	763	1
-25	-22	-34	-44	-74	-86	-103	-108	-92	-81	-51	-38	-11	-12	4	5	-4	-3	2
3	-18	-31	-40	-48	-79	-121	-172	-227	-231	-221	-248	-321	-444	-530	-548	-536	-486	1
-14	-28	-31	-49	-70	-93	-118	-134	-162	-189	-214	-226	-224	-203	-171	-129	-85	-40	1
169	173	176	173	172	191	217	248	271	312	360	421	445	413	310	177	41	-71	1
31	28	22	20	26	23	37	38	48	56	56	62	54	53	53	50	57	54	3
-45	-38	-16	-7	-4	7	5	-3	-12	-26	-45	-55	-55	-46	-45	-43	-42	-42	5
-13	9	48	92	150	219	268	283	234	114	-39	-185	-293	-351	-379	-380	-350	-308	1
-13	-10	4	14	38	32	22	21	9	-9	-21	-17	-4	5	7	-6	-17	-16	2

Figure 4.3 Epilepsy Dataset - B

4.3 SCREENSHOTS

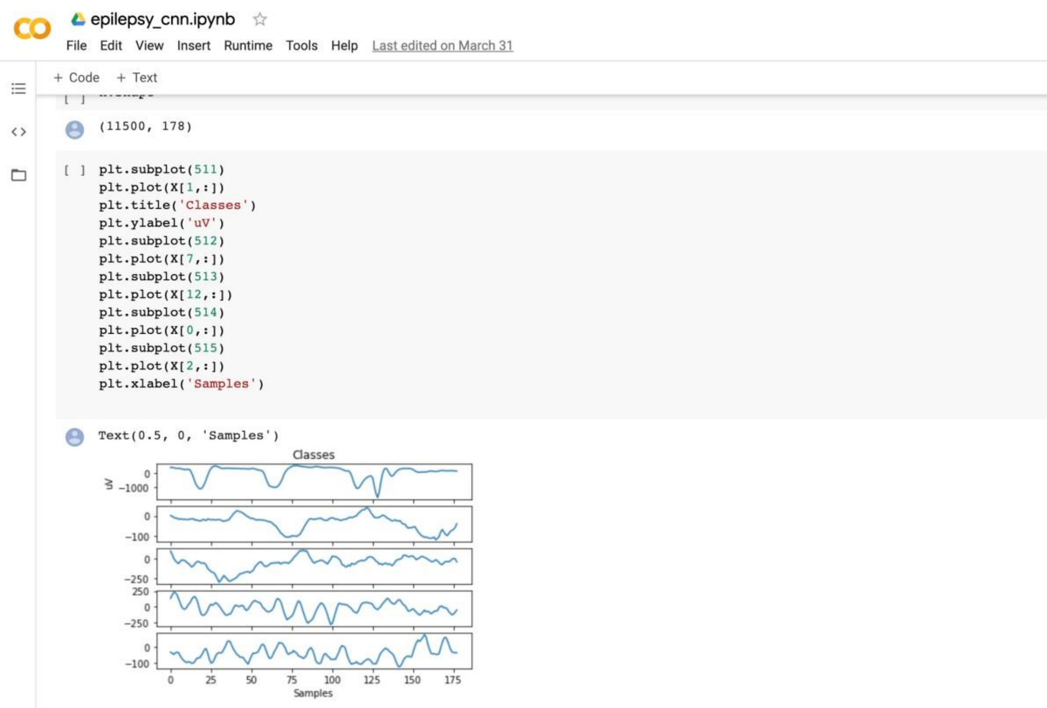


Figure 4.4 Plot of EEG signals for epilepsy task for different classes

```
schizo_cnn.ipynb
File Edit View Insert Runtime Tools Help Last edited on April 14
+ Code + Text

[ ] sc = StandardScaler()
    X_train = sc.fit_transform(X_train)
    X_test = sc.transform(X_test)

[ ] X_train.shape

    (199065, 10)

[ ] trainX = np.reshape(X_train, (X_train.shape[0], 1, X_train.shape[1]))
    testX = np.reshape(X_test, (X_test.shape[0], 1, X_test.shape[1]))

[ ] model_m = keras.Sequential()
    model_m.add(keras.layers.Conv1D(80, 1, activation='relu', strides=10,input_shape=(None, 10), kernel_initializer= 'uniform'))
    model_m.add(keras.layers.GlobalAveragePooling1D())
    model_m.add(keras.layers.Dropout(0.5))
    model_m.add(keras.layers.Dense(2, activation='softmax'))
    print(model_m.summary())

    Model: "sequential_18"
    _________________________________________________________________
    Layer (type)                 Output Shape              Param #
    =================================================================
    conv1d_25 (Conv1D)           (None, None, 80)          880
    _________________________________________________________________
    global_average_pooling1d_17  (None, 80)                0
    _________________________________________________________________
    dropout_16 (Dropout)         (None, 80)                0
    _________________________________________________________________
    dense_17 (Dense)             (None, 2)                 162
    =================================================================
    Total params: 1,042
    Trainable params: 1,042
    Non-trainable params: 0
    _________________________________________________________________
    None
```

Figure 4.5 Building Convolutional 1D Model

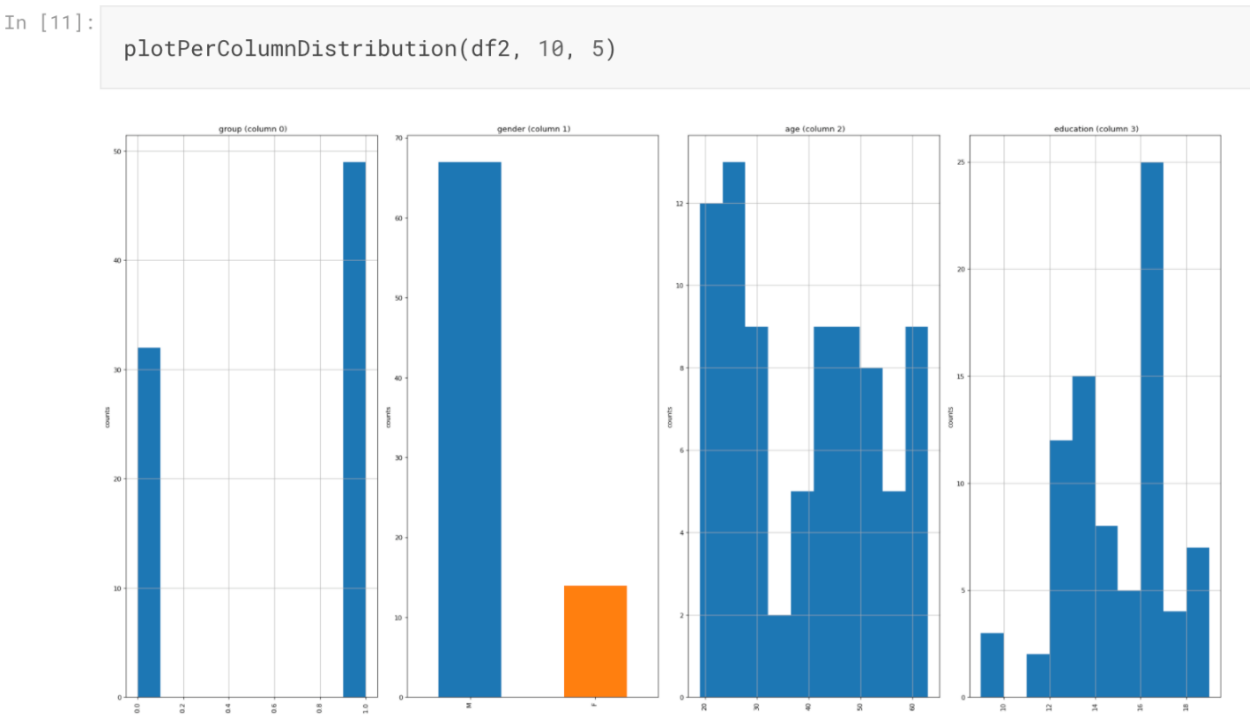


Figure 4.6 Distribution of subjects across demographic parameters

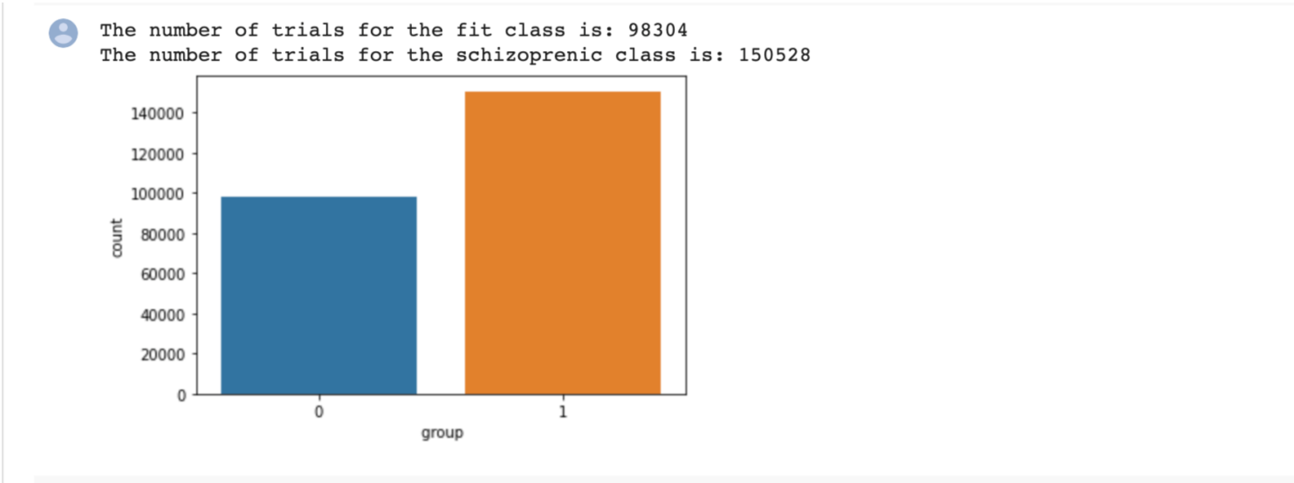


Figure 4.7 Schizophrenia distribution of trials

Figure 4.8 Epilepsy dataset distribution of trials

```
In [17]:  plotCorrelationMatrix(df3, 8)
```

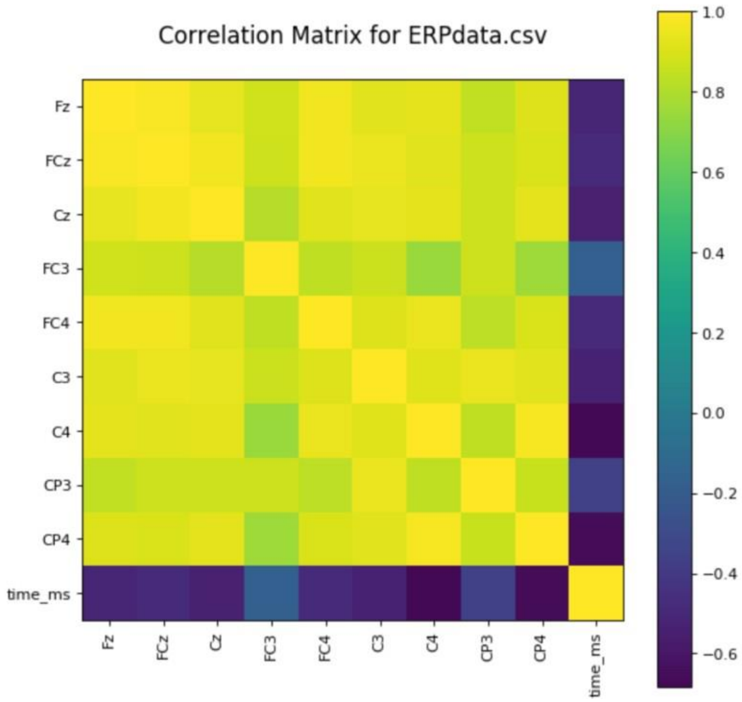


Figure 4.9 Correlation matrix of channel signals in schizophrenia charts

```
+ Code   + Text

import pandas as pd
import seaborn as sn
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split, cross_val_score
from sklearn.preprocessing import StandardScaler
import tensorflow as tf
from tensorflow import keras
from sklearn.neural_network import MLPClassifier
# from tensorflow.python.keras.layers import Dense
# from tensorflow.python.keras import Sequential
import numpy as np
from sklearn.metrics import accuracy_score
from sklearn.metrics import precision_score
from sklearn.metrics import recall_score
from sklearn.metrics import f1_score
from sklearn.metrics import cohen_kappa_score
from sklearn.metrics import roc_auc_score
from sklearn.metrics import confusion_matrix
print(tf.__version__)

2.2.0-rc2

[ ] ERP = pd.read_csv('../final_schizo.csv')

[ ] ERP
```

	Unnamed: 0	subject	Fz	FCz	Cz	FC3	FC4	C3	C4	CP3	CP4	group
0	0	1	1.857458	2.663352	3.051918	1.789011	1.961065	4.949901	2.282836	4.664762	3.425792	0
1	1	1	1.769692	2.601734	3.005648	1.868428	1.774454	5.024462	2.194537	4.737277	3.272531	0
2	2	1	1.764793	2.542194	3.002812	1.873400	1.800019	4.957654	2.152802	4.700508	3.171712	0

Figure 4.10 Model building – Part A

+ Code + Text Connect Editing

```
[ ] ERP.describe()
```

	Unnamed: 0	subject	Fz	FCz	Cz	FC3	FC4	C3	C4	CP3	CP4	
count	248832.000000	248832.000000	248832.000000	248832.000000	248832.000000	248832.000000	248832.000000	248832.000000	248832.000000	248832.000000	248832.000000	248
mean	124415.500000	41.000000	0.548699	1.039918	1.253373	0.840808	0.513989	1.146055	0.579773	0.925381	0.518643	
std	71831.755429	23.380951	2.141285	2.366516	2.494080	2.059043	1.825502	2.227648	1.884145	2.225112	1.921775	
min	0.000000	1.000000	-9.238550	-9.176126	-11.313430	-9.714905	-9.437540	-7.977966	-10.623550	-8.570679	-11.704520	
25%	62207.750000	21.000000	-0.761931	-0.455954	-0.330233	-0.482438	-0.575475	-0.252759	-0.587142	-0.439472	-0.657320	
50%	124415.500000	41.000000	0.408099	0.796265	0.898430	0.584198	0.391141	0.870537	0.460987	0.747819	0.408181	
75%	186623.250000	61.000000	1.713991	2.321995	2.586085	1.991961	1.551555	2.388095	1.677817	2.195711	1.626807	
max	248831.000000	81.000000	22.861889	16.870169	14.813741	20.009208	18.866536	18.436541	16.277234	16.881090	13.735362	

```
[ ] X = ERP.iloc[:,1:11].values
    X.shape

(248832, 10)

[ ] X[0]

array([1.        , 1.85745789, 2.66335158, 3.05191789, 1.78901053,
       1.96106526, 4.94990105, 2.28283579, 4.66476211, 3.42579158])

[ ] y = ERP.iloc[:,11].values
    y

array([0, 0, 0, ..., 1, 1, 1])
```

Figure 4.11 Model building – Part B

+ Code + Text Connect Editing

```
[ ] classifier = keras.Sequential()

[ ] classifier.add(keras.layers.Dense(12, activation = 'relu', input_dim = 10))
    classifier.add(keras.layers.Dense(8,  activation = 'relu'))
    classifier.add(keras.layers.Dense(1 ,activation = 'sigmoid'))
    classifier.compile(optimizer = 'adam', loss = 'binary_crossentropy', metrics = ['accuracy'])

    #Fitting the ANN to the training set
    classifier.fit(X_train, y_train, batch_size = 10, nb_epoch = 50)

19907/19907 [==============================] - 27s 1ms/step - loss: 0.0438 - accuracy: 0.9923
<tensorflow.python.keras.callbacks.History at 0x7fac5dc776d8>

[ ]
    # y_pred = classifier.predict(X_test, verbose =0)
    # y_pred = (y_pred > 0.5)
    yhat_probs = classifier.predict(X_test, verbose=0)
    # predict crisp classes for test set
    yhat_classes = classifier.predict_classes(X_test, verbose=0)
    # reduce to 1d array
    yhat_probs = yhat_probs[:, 0]
    yhat_classes = yhat_classes[:, 0]

    # accuracy: (tp + tn) / (p + n)
    accuracy = accuracy_score(y_test, yhat_classes)
    print('Accuracy: %f' % accuracy)
    # precision tp / (tp + fp)
    precision = precision_score(y_test, yhat_classes)
    print('Precision: %f' % precision)
    # recall: tp / (tp + fn)
    recall = recall_score(y_test, yhat_classes)
    print('Recall: %f' % recall)
```

Figure 4.12 Model building – Part C

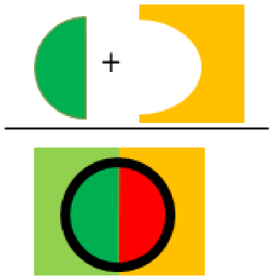
CHAPTER 5 - RESULTS

5.1 Evaluation Metric

- **Accuracy**

  In measurement of a set, accuracy refers to closeness of the measurements to a specific value. The most common metric for classification is accuracy, which is the fraction of samples predicted correctly as shown below:

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



Fraction predicted correctly

Figure 5.1 Illustration of calculation of accuracy for a machine learning model where TP is True positives and TN is true negatives

- **Recall Rate**

  Recall can be defined as the ratio of the total number of correctly classified positive examples divide to the total number of positive examples. High Recall indicates the class is correctly recognized (a small number of FN). It basically refers to the percentage of total relevant results correctly classified by the specified algorithm. It actually calculates how many of the Actual Positives the model captures through labeling it as Positive (True Positive). Also, it can be defined as the model metric that is used to select the best model when there is a high cost associated with False Negative.

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}$$

$$= \frac{\text{True Positive}}{\text{Total Actual Positive}}$$

Figure 5.2 Formulae for Recall Rate

- **Precision Value**

  To get the value of precision we divide the total number of correctly classified positive examples by the total number of predicted positive examples. High Precision indicates an example labeled as positive is indeed positive (a small number of FP). Precision talks about how precise/accurate the model is out of those predicted positive, how many of them are actual positive. It is a good measure to determine, when the costs of False Positive is high. Precision means the percentage of the results which are relevant.

$$\textbf{Precision} = \frac{\text{True Positive}}{\text{Actual Results}} \quad \text{or} \quad \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}}$$

Figure 5.3 Formulae of Precision Value

- **F1 Score**

  There is a simpler metric which takes into account both precision and recall, and therefore, you can aim to maximize this number to make your model better. This metric is known as F1-score, which is simply the harmonic mean of precision and recall. F1 Score is needed when we want to seek a balance between Precision and Recall.

$$\textbf{F1 Score} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

Figure 5.4 Mathematical formulae of F1 Score

5.2 Comparative Results

A confusion matrix is a table that is often used to describe the performance of a classification model (or "classifier") on a set of test data for which the true values are known. It allows the visualization of the performance of an algorithm. Following are the confusion matrices obtained for each of the classification algorithms implemented in this study.

- EPILEPSY SEIZURES RECOGNITION

ANN (Artificial Neural Networks)

	FP	FN
FP	1795	32
FN	32	441

Table 5.1 Confusion Matrix for ANN of Epilepsy Dataset

Accuracy: 0.972174 Recall: 0.932347

F1 score: 0.932347 Precision: 0.932347

RNN (Recurrent Neural Networks)

	FP	FN
FP	1795	35
FN	71	399

Figure 5.2 Confusion Matrix for RNN of Epilepsy Dataset

Accuracy: 0.953913

Recall: 0.848936

F1 score: 0.882743

Precision: 0.919355

CNN (Convolutional Neural Networks)

	FP	FN
FP	1774	88
FN	256	182

Table 5.3 Confusion Matrix for CNN of Epilepsy Dataset

Accuracy: 0.850435

Recall: 0.415525

F1 score: 0.514124

Precision: 0.674074

LSTM (Long-Short term Memory)

	FP	FN
FP	1074	772
FN	240	214

Table 5.4 Confusion Matrix for LSTM of Epilepsy Dataset

Accuracy: 0.560000

Recall: 0.471366

F1 score: 0.297222

Precision: 0.217039

Epilepsy Dataset Results

	ANN	CNN	RNN	LSTM
ACCURACY	0.972	0.85	0.953	0.56
PRECISION	0.932	0.674	0.919	0.217
F1 SCORE	0.932	0.514	0.882	0.297
RECALL	0.932	0.415	0.84	0.471

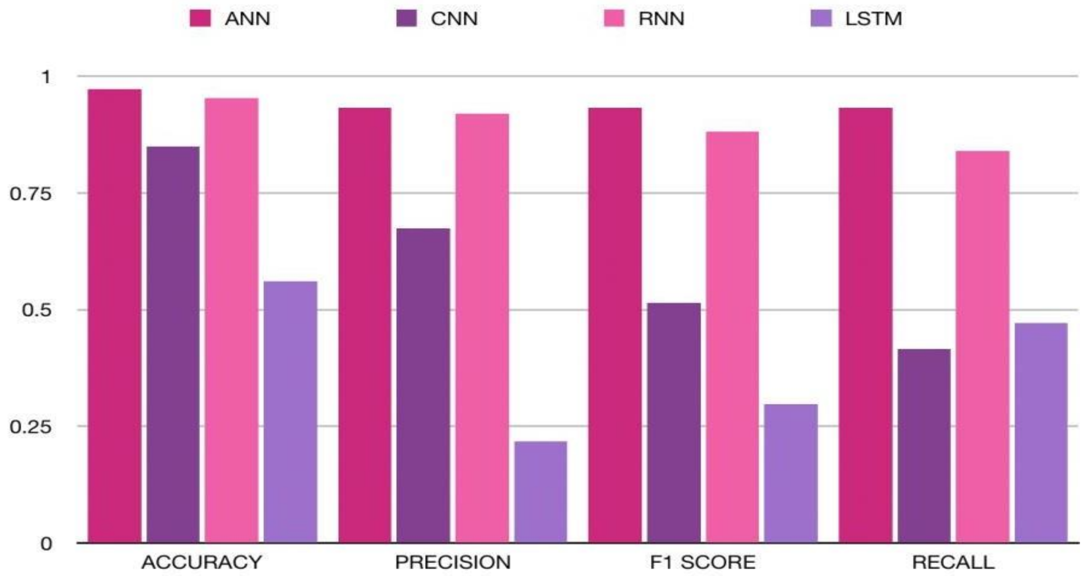


Figure 5.5 Comparative Analysis of four Deep Learning algorithms on Epilepsy Seizures Dataset

- SCHIZOPHRENIA

ANN (Artificial Neutral Networks)

	FP	FN
FP	19506	143
FN	4	30114

Table 5.5 Confusion Matrix for ANN of Schizophrenia Dataset

Accuracy: 0.997046 Recall: 0.999867

F1 score: 0.997565 Precision: 0.995274

RNN (Recurrent Neural Networks)

	FP	FN
FP	19621	0
FN	0	30146

Table 5.6 Confusion Matrix off RNN of Schizophrenia dataset

Accuracy: 1.000000 Recall: 1.000000

F1 score: 1.000000 Precision: 1.000000

CNN (Convolutional Neural Networks)

	FP	FN
FP	11830	7887
FN	266	29784

Table 5.7 Confusion Matrix for CNN of Schizophrenia Dataset

Accuracy: 0.836177

Recall: 0.991148

F1 score: 0.879609

Precision: 0.790635

LSTM (Long-Short term Memory)

	FP	FN
FP	13811	5938
FN	2970	27048

Table 5.8 Confusion Matrix of Schizophrenia dataset

Accuracy: 0.821006

Recall: 0.901059

F1 score: 0.858612

Precision: 0.819984

Schizophrenia Dataset Results

	ANN	CNN	RNN	LSTM
ACCURACY	0.997	0.836	0.999	0.821
PRECISION	0.995	0.790	0.999	0.819
F1 SCORE	0.997	0.879	0.999	0.858
RECALL	0.999	0.991	0.999	0.901

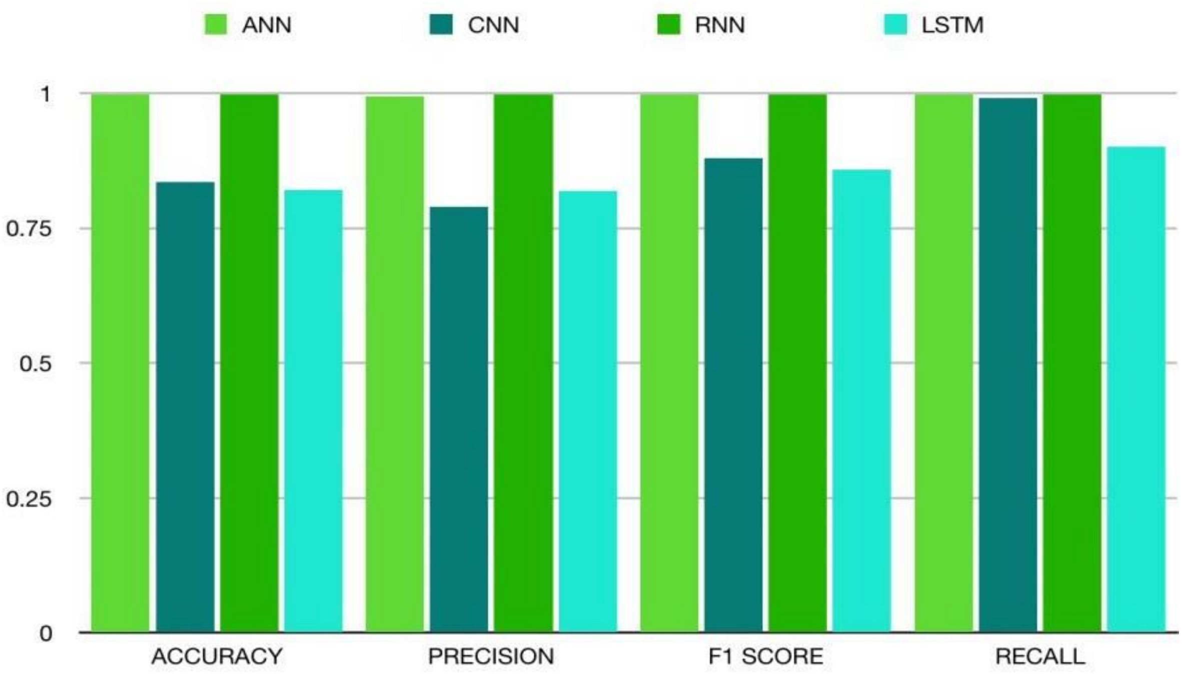


Figure 5.6 Comparative Analysis of four Deep Learning algorithms on Schizophrenia Dataset

CHAPTER 6 -

CONCLUSION AND FUTURE SCOPE

The electrical physiological activity in the brain can be recorded using a non-invasive monitoring method, EEG (Electroencephalography). EEG measures voltage fluctuations within the neurons of the brain. Recently, deep learning has shown a significant role in this field due to its capacity to learn good feature representations from raw data and could be extended further for a generalized detection of neural disorders. The aim of this project was to perform the comparative analysis using deep learning in the field of biomedical science and its applications and implementing various algorithms on the given dataset of brain disorders and check which algorithms fits with highest accuracy and can be used further. Study of various research papers on EEG based classification and Deep Learning Models and Algorithms has been done and a literature survey has been maintained for the same. Therefore, we can say now that using EEG as a tool for fast detection of the diseases at early stage and integrating this with machine learning tools can provide better diagnosis.

According to the literature survey, ANN (Artificial Neural Networks), CNN (Convolutional Neural Networks), RNN (Recurrent Neural Networks) and LSTM (Long-Short Term Memory) are the four deep learning algorithms that can be used in the biomedical sciences field in order for the early detection of the brain disorders. The data was gathered from the online sources and after preprocessing and cleaning of the data sets, the algorithms are applied on it and the one with highest accuracies is adapted as the best one and can be further used for the detection of the diseases. In this study, RNN was found with the highest accuracy and can be used for the future purpose of diagnosing the disorder at the early stage. Training the patients suffering from neural disabilities, to initiate brain signals for controlling external devices, provides scope for extensive future development.

REFERENCES

1. Craik, Alexander, Yongtian He, and Jose L. Contreras-Vidal. "Deep learning for electroencephalogram (EEG) classification tasks: a review." *Journal of neural engineering* 16.3 (2019): 031001.

2. Baldi, Pierre. "Deep learning in biomedical data science." *Annual review of biomedical data science* 1 (2018): 181-205.

3. LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton. "Deep learning." *nature* 521.7553 (2015): 436-444.

4. Maryam M. Najafabadi, Flavio Villanustre, Taghi M. Khoshgoftaar, Naeem Seliya, Randall Wald, and Edin Muharemagic. 2015. Deep learning applications and challenges in big data analytics. Journal of Big Data 2, 1 (2015), 1–21

5. Ravì, Daniele, et al. "Deep learning for health informatics." *IEEE journal of biomedical and health informatics* 21.1 (2016): 4-21.

6. *"The Future of the Healthcare Science Workforce. Modernising Scientific Careers: The Next Steps". 26 November 2008. p. 2. Retrieved 1 June 2011.*

7. Miotto, Riccardo, et al. "Deep learning for healthcare: review, opportunities and challenges." *Briefings in bioinformatics* 19.6 (2018): 1236-1246.

8. Liu C, Wang F, Hu J, et al. Risk prediction with electronic health records: a deep learning approach. In: ACM International Conference on Knowledge Discovery and Data Mining, Sydney, NSW, Australia, 2015, 705–14.

9. Pham T, Tran T, Phung D, et al. DeepCare: a deep dynamic memory model for predictive medicine. arXiv 2016. https:// arxiv.org/abs/1602.00357.

10. Miotto R, Li L, Kidd BA, et al. Deep patient: an unsupervised representation to predict the future of patients from the electronic health records. Sci Rep 2016;6:26094.

11. Shen, Yurun, Hongtao Lu, and Jie Jia. "Classification of motor imagery EEG signals with deep learning models." *International Conference on Intelligent Science and Big Data Engineering*. Springer, Cham, 2017.

12. Chiarelli, Antonio Maria, et al. "Deep learning for hybrid EEG-fNIRS brain– computer interface: application to motor imagery classification." *Journal of neural engineering* 15.3 (2018): 036028.

13. Amin, Syed Umar, et al. "Deep Learning for EEG motor imagery classification based on multi-layer CNNs feature fusion." *Future Generation Computer Systems* 101 (2019): 542-554.

14. Zheng, Xiao, et al. "Decoding human brain activity with deep learning." *Biomedical Signal Processing and Control* 56 (2020): 101730.

15. Ramkumar, S., et al. "Performance Analysis of EEG Signals using Conventional and Hybrid Artificial Neural Network."

16. Sarmiento, L. C., et al. "Brain computer interface (BCI) with EEG signals for automatic vowel recognition based on articulation mode." *5th ISSNIP-IEEE Biosignals and Biorobotics Conference (2014): Biosignals and Robotics for Better and Safer Living (BRC)*. IEEE, 2014.
17. Vo, Anh Kha. *Subject-independent ERP-based brain-computer interface using adaptive and ensemble learning*. Diss. 2019.

18. Nishad, Anurag, and Ram Bilas Pachori. "Classification of epileptic electroencephalogram signals using tunable-Q wavelet transform based filter- bank." *Journal of Ambient Intelligence and Humanized Computing* (2020): 1-15.

19. Raghu, S., et al. "EEG based multi-class seizure type classification using convolutional neural network and transfer learning." *Neural Networks* 124 (2020): 202-212.

20. Liu, Yu-Cheng, and Chaoyun Liang. "Design exploration predicts designer creativity: a deep learning approach." *Cognitive Neurodynamics* (2020): 1-10.

21. Hemanth, D. Jude. "EEG signal based Modified Kohonen Neural Networks for Classification of Human Mental Emotions." *Journal of Artificial Intelligence and Systems* 2 (2020): 1-13.

22. Golmohammadi, Meysam, et al. "Automatic analysis of EEGs using big data and hybrid deep learning architectures." *Frontiers in human neuroscience* 13 (2019): 76.

23. Kumarasinghe, Kaushalya, Nikola Kasabov, and Denise Taylor. "Deep learning and deep knowledge representation in Spiking Neural Networks for Brain-Computer Interfaces." *Neural Networks* 121 (2020): 169-185.

24. Akyol, Kemal. "Stacking ensemble based deep neural networks modeling for effective epileptic seizure detection." *Expert Systems with Applications* 148 (2020): 113239.

25. Sirpal, Parikshat, et al. "fNIRS improves seizure detection in multimodal EEG-fNIRS recordings." *Journal of biomedical optics* 24.5 (2019): 051408.

26. Akyol, Kemal. "Comparing of deep neural networks and extreme learning machines based on growing and pruning approach." *Expert Systems with Applications* 140 (2020): 112875.

27. Hossain, M. Shamim, et al. "Applying deep learning for epilepsy seizure detection and brain mapping visualization." *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 15.1s (2019): 1-17.

28. Tang, Xuebin, et al. "A novel classification algorithm for MI-EEG based on deep learning." *2019 IEEE 8th Joint International Information Technology and Artificial Intelligence Conference (ITAIC)*. IEEE, 2019.

29. An automated system for epilepsy detection using EEG brain signals based on deep learning approach

30. Nasseri, Mona, et al. "Semi-supervised training data selection improves seizure forecasting in canines with epilepsy." *Biomedical Signal Processing and Control* 57 (2020): 101743.

31. Oh, Shu Lih, et al. "Deep convolutional neural network model for automated diagnosis of schizophrenia using EEG signals." *Applied Sciences* 9.14 (2019): 2870.

32. Srinivasagopalan, Srivathsan, et al. "A deep learning approach for diagnosing schizophrenic patients." *Journal of Experimental & Theoretical Artificial Intelligence* 31.6 (2019): 803-816.

33. Wang, Danshi, et al. "Modulation format recognition and OSNR estimation using CNN-based deep learning." *IEEE Photonics Technology Letters* 29.19 (2017): 1667- 1670.

34. Chin, Chiun-Li, et al. "An automated early ischemic stroke detection system using CNN deep learning algorithm." *2017 IEEE 8th International Conference on Awareness Science and Technology (iCAST)*. IEEE, 2017.

35. Jain, Ashesh, et al. "Structural-rnn: Deep learning on spatio-temporal graphs." *Proceedings of the ieee conference on computer vision and patternrecognition*. 2016.

36. Lee, June-Goo, et al. "Deep learning in medical imaging: general overview." *Korean journal of radiology* 18.4 (2017): 570-584.

37. Gers, Felix A., Jürgen Schmidhuber, and Fred Cummins. "Learning to forget: Continual prediction with LSTM." (1999): 850-855.