University of Bahrain

College of Engineering



Department of Electrical and Electronics Engineering

Enhancing Electroencephalography Classifications and Decoding Accuracy of Emotion Recognition Using Machine Learning

Submitted In Partial Fulfillment of The Requirements For The Degree of PhD In Electrical Engineering

Submitted By

Mohammed Nabeel AlSallout

20150117

Supervised by

Ebrahim Abdulla Mattar

Professor

University of Bahrain

Kingdom of Bahrain

2023 May

UNIVERSITY OF BAHRAIN

College of Engineering



Department of Electrical and Electronics Engineering

Enhancing Electroencephalography Classifications and Decoding Accuracy of Emotion Recognition Using Machine Learning

Submitted In Partial Fulfillment of The Requirements For The Degree of PhD In Electrical Engineering

Submitted by:
Mohammad Nabeel AlSallout
20150117

Supervised by
Ebrahim Abdulla Mattar
Professor
University of Bahrain

Kingdom of Bahrain
May 2023

Defense Committee's Approval

University of Bahrain
College of Engineering
Department of Electrical and Electronics Engineering
Doctor of Philosophy in Electrical Engineering

Student Mohammed Nabeel AlSallout ID number 20150117 has defended his thesis by which he has fulfilled the requirement for the degree of Doctor of Philosophy in Electrical Engineering entitled (Enhancing Electroencephalography Classifications and Decoding Accuracy of Emotion Recognition Using Machine Learning) on Wednesday, May 3, 2023.

The committee recommends awarding of PhD Degree with the grade of: A

(Defense Committee)

Name		Signature
Professor Ebrahim A Mattar	Supervisor	Ebrahim Mattar
Professor Hasan Al-Nashash	External examiner	Hasan Al-Nashash
Professor Mohab A. Mangoud	Internal examiner	M Mangoud

الهندسة الكهربائية والإلكترونية
ELECTRICAL & ELECTRONIC ENGINEERING
UNIVERSITY OF BAHRAIN

Academic department's stamp

Abstract

This thesis explores various machine learning and deep learning methods for building real-time emotion recognition systems based on Electroencephalogram (EEG) signals. We aim to enhance the classification and decoding of EEG signals at a large scale and to bridge the gap between basic EEG research and real-world applications. We use some of the most popular EEG datasets for emotions and investigate how the data randomization affects the results. We also propose a novel adaptive channel selection method that uses different filtering techniques and recursive performance ranking with transfer learning to find the top-N ranked channels. We introduce a new feature extraction and selection approach that uses transfer learning to overcome overfitting problems caused by using local data. We divide features into centralized and non-centralized groups, where the centralized ones are more accurate but more resource-intensive. We compare several supervised and unsupervised classification methods, such as Decision Tree, Nearest Neighbor, Support Vector Machine, Naive Bayes, Hierarchical Clustering, and K-mean Clustering. We also evaluate several deep learning methods, such as artificial and deep neural networks. We design our model to have simple convolutions that work well with small samples and to have consistent input and output formats for real-time knowledge transfer and weight forwarding. We leverage the power of Convolutional Neural Networks (CNN) and Recurrent Neural Networks for automatic feature extraction and selection in both centralized and decentralized models. These models can handle different input data formats depending on the available resources and the decision constraints, leading to different levels of accuracy and performance. In summary, we develop and test a robust emotion recognition solution that can utilize modern application architecture, large data flow, and communication capabilities of different network components. This opens up new possibilities for research that is relevant to industry.

Keywords: electroencephalogram, EEG Classification, brain-computer interfaces, BCI applications, emotion detection, signal processing, filtering, feature extraction, classification, clustering.

Dedication

I dedicate this thesis to my family, who have supported me every step of the way. Their unwavering love, encouragement, and sacrifices have been my driving force throughout this journey. I would also like to dedicate this work to my mentors, colleagues, and friends who have shared their knowledge, expertise, and passion for research with me. Thank you all for your invaluable support and guidance.

Acknowledgements

First and foremost, I would like to express my deepest gratitude to Allah, the Most Gracious and the Most Merciful, for blessing me with the strength and perseverance to complete this PhD journey. I seek His forgiveness for any shortcomings and pray that He accepts this humble effort.

I would also like to express my heartfelt thanks to my supervisor, Professor Ebrahim, for his unwavering support and guidance. His expertise and insights have been invaluable in shaping my research and helping me grow as a scholar. I am truly grateful for the countless hours he has spent mentoring me and providing feedback on my work. Thanks, Prof. Ebrahim. Thanks to other university of Bahrain colleagues with whom we cooperated.

I am deeply grateful to my parents, Zainab and Nabeel. My father has always encouraged me to pursue my dreams and has been a constant source of support throughout my academic journey. His wisdom and guidance have been invaluable in helping me navigate the challenges of graduate school. My mother's prayers have given me strength and comfort during the most challenging times. Her unwavering faith in me has been a source of inspiration and motivation.

To my wife, Eman, thank you for your love and support. You have been my rock through the ups and downs of this journey, and I could not have done it without you. Your unwavering belief in me has given me the strength to keep pushing forward even when the going got tough. You have been a constant source of encouragement and support, always there to listen and offer advice when I needed it most.

To my children Nabeel, Adam and Tala, thank you for being a constant source of joy and inspiration. You have given me the motivation to keep pushing forward even when the going got tough. Your smiles and laughter have brightened even the darkest days. Watching you grow and learn has been one of the greatest joys of my life, and I am so proud of each and every one of you.

I'm happy to have an autistic son like Nabeel who empowers me in this life and maybe the next. He motivated me every time he greeted me home, every time he looked me in the eye and reminded me that we as humans have just scratched the surface of

knowledge, knowledge that can make a difference in people's lives and provide hope to those who need it most.

My family has been an incredible source of support throughout this journey. They have stood by me through thick and thin, offering love, encouragement, and guidance every step of the way. I am so grateful for their unwavering belief in me and for their constant presence in my life.

In conclusion, I would like to express my heartfelt thanks to everyone who has supported me on this journey. This PhD has been one of the most challenging but rewarding experiences of my life, and I could not have done it without the love and support of so many wonderful people. May Allah bless you all for your kindness and generosity.

Table of Contents

Abstract ..... *B*

Publications ..... *C*

Dedication ..... *D*

Acknowledgements ..... *E*

Table of Contents ..... *G*

List of Tables ..... *I*

List of Figures ..... *J*

List of Abbreviations ..... *Q*

Chapter One: Introduction *18*

1.1. Motivation: **20**

1.2. Research Questions: **22**

1.3. Aim of the Thesis **24**

1.4. Thesis Organization **26**

Chapter Two: Research Background *29*

2.1. Human Emotions **30**

2.1.1. Emotion Models 30

2.2. Brain and EEG Signals **35**

2.2.1. Brain Functional Structure 35

2.2.2. EEG Signals 37

2.2.3. Signal Processing 39

2.2.4. Artifact Removal 40

2.2.5. Feature Extraction and Selection 40

2.2.6. Classification Techniques 41

2.3. Available Data and Tools **43**

2.3.1. EEG Datasets 43

2.3.2. EEG Hardware 45

2.3.3. EEG Software 48

2.4. Related Research **49**

2.4.1. Modern Applications 49

2.4.2. Emotional Detection Area 50

2.4.3. Transfer Knowledge 51

Chapter Three: Methodology *54*

3.1. Pre-Classification Methodologies: 56
3.1.1. Overall Processing Methodology 60
3.1.2. Datasets Acquisition 76
3.1.3. Signal Preprocessing: 88
3.1.4. Features Evaluation and Extraction: 100
3.2. Classification Methodologies: 114
3.2.1. Supervised Classifiers 117
3.2.2. Deep Learning Classification Models: 120
3.2.3. Classifiers Learning Transfer 141
Chapter Four: Results and Discussion 147
4.1. Results 148
4.1.1. Signal Processing 148
4.1.2. Feature Processing 150
4.1.3. Classifications Methods 156
4.1.4. Convolutional Neural Networks 162
4.1.5. Recurrent Neural Networks 168
4.2. Discussions 172
Chapter Five: Conclusions and Future Works 174
5.1. Research Contributions and Conclusions 174
5.2. Future Work 180
Bibliography: 181
Appendix A: Used ML/AI Python Language Libraries 197
Machine Learning using SciKit-Learn 197
Deep Learning Models using TenserFlow and Keras 198
Data Migration From MATLAB using SciPy: 199
Feature Extraction using eeglib: 200
PyEEGLTE (Python Library for EEG Load, Transform and Extract) 201
Appendix B: Data Preparation 203
Data Representation 203
Appendix C: Unsupervised Clustering 210
Hierarchical Clustering: 211
Clustering with K-means: 212
ملخص 213

List of Tables

Table 1: Examples of commercial non-invasive systems. 47

Table 2: Cattell's sixteen factors of personality (16PF). 68

Table 3: The SEED Dataset three categories for emotions triggering experiments. 77

Table 4: Comparing DEAP and SEED, two important EEG emotion datasets. 78

Table 5: Sample of Dataset Gap Analysis of SEED Dataset, low and high gap can impact expermint performance. 81

Table 6: Some implemented Hybrid Multilevel Features. 108

Table 7: Comparing newly proposed deep learning classification performance with other classification methods. 167

Table 8: Classification performance based on processing and signal format. 172

Table 9: lists a performance comparison between state-of-the-art models and our proposed method for EEG emotion categories. 173

Table 10: Summary of the features that can be extracted through eeglib by default. 201

Table 11: EEGLAB dataset event configuration. 206

List of Figures

Figure 1: 2D Distribution of affective stimuli in the two-dimensional affective-valence space, Source: adapted from (Koelstra et al., 2012). ...32

Figure 2 :Plutchik's 3D model "wheel of emotions" depicting fundamental and derived emotions, Source: (Nielek et al., 2017). 33

Figure 3: Validity (top), arousal (middle), and dominance (bottom) SAM (Self-Assessment Manikin), Source: from (Gatti et al., 2018).....34

Figure 4: Basic breakdown of the structure and function of this structure, Source: adapted from (Cudlenco et al., 2020).....36

Figure 5: A time-domain EEG graph with 64 channels, created by MATLAB. 37

Figure 6: EEGLAB frequency-domain power spectrum for all (64 channels).....38

Figure 7: Categories of existing EEG datasets. 44

Figure 8: Modern BCI application areas. 49

Figure 9: EEG processing methodology and working areas, (Image by Author). 57

Figure 10: Outcomes of comparing the computation time (in μs) of verity of features in MATLAB (Arnin et al., 2018). 59

Figure 11: Duration of entropy feature extraction computation (Arunkumar et al., 2018; Z. Chen et al., 2020; Krishnan & Balasubramanian, 2016). 59

Figure 12: Overall proposed processing methodology.....62

Figure 13: Data sources enhancements covered areas. 66

Figure 14: Recursive data sources, contains different transfer learning data sources. 67

Figure 15: Two level of feature extraction, on distributed and centralized units. 70

Figure 16: Multiple selection methodology can be choose based on system stability. 71

Figure 17: Classification method has a fixed recursive learning transfer data to be used for feature and channel selection actions.....72

Figure 18: Proposed multi-stage recursive classification model, combines CNN-AE with RNN. 73

Figure 19: Terminal decode emotion instantly with the help of central classifier, based on the real-time adaptive decoding algorithm that is controlled by the terminal and defined accuracy acceptance level. 75

Figure 20: Analyze SEED dataset's experiments count per month, can impact practitioner performance. 80

Figure 21: Analyze SEED dataset's experiments count per day give better look on workload.....80

Figure 22: Overview of multi-layer transfer knowledge.....82

Figure 23: Signal analysis for the same subject where second experiment held after four months (negative emotion). The signal was sampled at 200 Hz. 85

Figure 24: Signal analysis for the same subject where second experiment held after four months (neutral emotion). 86

Figure 25: Signal analysis for the same subject where second experiment held after four months (positive emotion).....87

Figure 26: Raw EEG signals in the three different emotion states, used channels are fp1, fpz, fp2, af3, af4. 89

Figure 27: Spectral analysis with spectrogram for time-frequency of the raw EEG signals at the three states, (EEG signal from SEED dataset, with 200 Hz sampling rate). 90

Figure 28: Before and after applying band-pass filter from 0.1 Hz to 65 Hz (EEG signal from SEED dataset, with 200 Hz sampling rate).....91

Figure 29: Before and after applying notch filter on bands 50Hz and 60Hz (EEG signal from SEED dataset). The signal was sampled at 200 Hz. ...92

Figure 30: Before and after applying signal down-sampling by half from 200Hz to 100Hz using MATLAB resample() function, note that signal maintain the general characteristics (EEG signal from SEED dataset, with sampling rate 200 Hz). 94

Figure 31: Constant time segments with a fixed window size were applied on 12,000 samples with a length of 60 seconds, and the window sizes were 3, 15, and 30 seconds (SEED dataset, sampled at 200 Hz).....95

Figure 32: Change point detection segments with variable window sizes, applied on 12000 sample of 60 second duration (EEG signal from SEED dataset, sampled at 200 Hz).....97

Figure 33: Increasing change point detection sensitivity by reduce minimum window size (1000) and increase allowed number of windows

(10) with variable window sizes, applied on 12000 sample of 60 second duration (EEG signal from SEED dataset, sampled at 200 Hz). 97

Figure 34: Adaptive channel selection techniques using learning transfer weights 99

Figure 35: The outcome of doing signal analysis on all 62 channels across the spectrum domain and time domain 105

Figure 36: On the same 62 channels, in both the temporal and spectrum domains, we use a function that smooths signals by a factor of three. ..105

Figure 37: Adaptive feature extraction method using auto switch technique based on learning transfer weights 109

Figure 38: Filter method action flow 110

Figure 39: Wrapper method action flow. 111

Figure 40: Embedded method action flow. 112

Figure 41: Overview of machine learning 115

Figure 42: General autoencoder structure. 124

Figure 43: Proposed multi-stage recursive classification model, combines CNN, AE and RNN. 126

Figure 44: High level CNN architecture. 128

Figure 45: Sample CNN model, saved in h5 Keras model format and visualized by netron app (Ganesh, n.d.; Roeder, n.d.). 131

Figure 46: Sample CNN model designed by NN-SVG (LeNail, 2019). 131

Figure 47: Three spectrogram samples of EEG signal representation, with 200 Hz sampling rate 133

Figure 48: Some topographies for different EEG time windows. 134

Figure 49: Overall used deep learning model, including data formatting , preparation and overall classification methods structure. 135

Figure 50: Proposed CCN model. 137

Figure 51: Proposed CCN AE model. 139

Figure 52: Proposed CCN LSTM model 140

Figure 53: CNN (IMAGE) classification accuracy in real-time micro training (1-2 minutes). in the left without knowledge transfer, in the middle with applying for knowledge transfer from different experiments, and finally, on the right with applying for knowledge transfer between inter-subjects. 142

Figure 54: RNN (IMAGE) classification accuracy in real-time micro training (1-2 minutes). in the left without knowledge transfer, in the middle with applying for knowledge transfer from different experiments, and finally, on the right with applying for knowledge transfer between inter-subjects. 143

Figure 55: CNN (Signal Flow) classification accuracy in real-time micro training (1-2 minutes). in the left without knowledge transfer, in the middle with applying for knowledge transfer from different experiments, and finally, on the right with applying for knowledge transfer between inter-subjects. 144

Figure 56: RNN (Signal Flow) classification accuracy in real-time micro training (1-2 minutes). in the left without knowledge transfer, in the middle with applying for knowledge transfer from different experiments, and finally, on the right with applying for knowledge transfer between inter-subjects. 145

Figure 57: Feature extraction time based on number of channels counts. 148

Figure 58: Accuracy of sample classification methods based on number of channels..... 149

Figure 59: Example plot of averaged standard deviation feature of 1,000 reads on 200 Hz. 150

Figure 60: Example plot of averaged third order moments (skewness) feature of 5000 reads on 200 Hz. 151

Figure 61: Example plot of averaged fourth order moments (kurtosis) feature of 5000 reads on 200 Hz. 151

Figure 62: Example plot of absolute second difference feature of 5000 reads on 200 Hz. 152

Figure 63: Computation time of entropy feature in distributed mode (Intel Core i7 920) and centralized mode (ARM Cortex A53). 153

Figure 64: Computation time in centralized mode (ARM Cortex A53) for different type of features. 153

Figure 65: Computation time decentralized mode (ARM Cortex A53) for different type of features. 154

Figure 66: Estimated delay of feature extraction and getting ready for classification in different processing modes. 155

Figure 67: Estimated bandwidth usage of features data transformation to the system for classification in different processing modes. 155

Figure 68: The categorization algorithms were tested on different counts of standard deviation features.....157

Figure 69: The categorization algorithms were tested on different counts of PCA generated features those are based on standard deviation features. 157

Figure 70: The categorization algorithms were tested on different counts of 2nd differences features. 158

Figure 71: The categorization algorithms were tested on different counts of PCA generated features those are based on 2nd differences features. 158

Figure 72: The categorization algorithms were tested on different counts of 4th-order moments features. 159

Figure 73: The categorization algorithms were tested on different counts of PCA generated features those are based on 4th-order moments features. 159

Figure 74: The categorization algorithms were tested on different counts of Shanoon entropy features.....160

Figure 75: The categorization algorithms were tested on different counts of PCA generated features those are based on Shanoon entropy features. 160

Figure 76: The categorization algorithms were tested on different counts of differential entropy features. 161

Figure 77: The categorization algorithms were tested on different counts of PCA generated features those are based on differential entropy features. 161

Figure 78: Proposed CNN accuracy result; the x-axis displays the accuracy per iteration, the orange line represents the training results, and the blue line represents the validation accuracy.....162

Figure 79: Proposed CNN loss result; the x-axis displays the loss per iteration, the orange line represents the training results, and the blue line represents the validation accuracy.....162

Figure 80: Trained classification model is tested with new data, each row of the confusion matrix represents the target class, and each column represents the predicted class that a classifier outputs. 163

Figure 81: Proposed CNN accuracy result; the x-axis displays the accuracy per iteration, here we used continuous raw features as an input

to the model instead of images, the orange line represents the training results, and the blue line represents the validation accuracy.....164

Figure 82: Proposed CNN loss result; the x-axis displays the loss per iteration, here we used continuous raw features as an input to the model instead of images, the orange line represents the training results, and the blue line represents the validation accuracy.....164

Figure 83: Trained classification model is tested with new data, here we used continuous raw features as an input to the model, each row of the confusion matrix represents the target class, and each column represents the predicted class that a classifier outputs. 165

Figure 84: Proposed CNN-AE accuracy result; the x-axis displays the accuracy per iteration, the orange line represents the training results, and the blue line represents the validation accuracy.....166

Figure 85: Proposed CNN-AE loss result; the x-axis displays the loss per iteration, the orange line represents the training results, and the blue line represents the validation accuracy.....166

Figure 86: Trained CNN classification model is tested with new data, each row of the confusion matrix represents the target class, and each column represents the predicted class that a classifier outputs. 167

Figure 87: Proposed CNN-LSTM accuracy result; the x-axis displays the accuracy per iteration, the orange line represents the training results, and the blue line represents the validation accuracy.....168

Figure 88: Proposed CNN-LSTM accuracy result; the x-axis displays the accuracy per iteration, here we used continuous raw features as an input to the model instead of images, the orange line represents the training results, and the blue line represents the validation accuracy.....169

Figure 89: Proposed CNN-LSTM loss result; the x-axis displays the loss per iteration, the orange line represents the training results, and the blue line represents the validation loss.....169

Figure 90: Proposed CNN-LSTM loss result; the x-axis displays the loss per iteration, here we used continuous raw features as an input to the model instead of images, the orange line represents the training results, and the blue line represents the validation loss. 170

Figure 91: Trained CNN-LSTM classification model is tested with new data, each row of the confusion matrix represents the target class, and each column represents the predicted class that a classifier outputs.....171

Figure 92: Trained CNN-LSTM classification model is tested with new data, here we used continuous raw features as an input to the model

instead of images, each row of the confusion matrix represents the target class, and each column represents the predicted class that a classifier outputs. 171

Figure 93: Summary of algorithms enabled in the python machine learning library (scikit-learn)....198

Figure 94: Mind map of most popular python library applying machine learning algorithms....199

Figure 95: Plotting different signal representation using matplotlib python library. 200

Figure 96: PyEEGLTE (Python Library for EEG Load, Transform and Extract), project page in GitHub. 202

Figure 97: Available EEGLAB import formats. 204

Figure 98: EEGLAB import raw data dialogue....204

Figure 99: Our SEED EEG dataset settings. 204

Figure 100: The international 10-20 system's electrode placements for EEG recording (Rojas et al., 2018) (right), EEGLAB channel location displayed for SEED dataset (right)....205

Figure 101: Channel activities for the first event. 207

Figure 102: Channel activities for the second event. 207

Figure 103: Channel activities for the third event. 208

Figure 104: Channel activities for the last event. 208

Figure 105: Sample of EEGLAB artifact detection tool. 209

Figure 106: Visualizing of a two-dimensional dataset containing four distinct blobs before applying K-Means. 211

Figure 107: Visualizing four distinct blobs clustering after applying K-Means. 212

List of Abbreviations

EEG ..... Electroencephalogram

ERP ..... Event-related potentials

EOG ..... Electrooculography

EKG ..... Electrocardiography

EMG ..... Electromyogram

EP ..... Evoked Potentials

MRI ..... Magnetic Resonance Imaging

EMU ..... Epilepsy Monitoring Unit

ICA ..... Independent component analysis

PCA ..... Principal Component Analysis

WT ..... Wavelet Transform

K-NN ..... K-nearest neighbor

NN ..... Neural network

BCI ..... Brain Computer Interface

ECG ..... Electrocardiograms

PDF ..... Probability Density Function

STD ..... Standard deviation

SVM ..... Support Vector Machines

SNR ..... Signal-to-Noise Ratio

MI ..... Motor Imagery

CNN ..... Convolutional Neural Network

Chapter One:
Introduction

This thesis addresses the problem of emotion detection using EEG recordings. The fundamental goal is to develop and compare several procedures that enhance efficiency significantly for analyzing EEG signals for a specific purpose. These methods aim to optimize the trade-off between accuracy and computational cost by reducing the required resources, lowering the transmission bandwidth and speeding up the classification process.

These procedures are designed to evaluate the data collection method clinically or during an experiment, then extract the information from those signals, evaluate it, display the results, and take a predefined action on the final observations; all these activities are optimized for fewer data processing, faster and more accurate results.

Contrary to previous works on the emotional field, this thesis analyzes less highlighted areas, like the clinical experiment environment, the personal level differences, and regular researching fields related to the EEG signal, such as noise filtering and features.

The first part of this research covers the datasets, the clinical experiments behind them, and a comparison of those datasets, and the experiment selection. Also, signal description has been covered, such as data density and complexity.

The second part focuses on EEG signal filtering, selection, and representation on frequency and time domains, then selecting and extracting standard features methods for all signal channels and applying simple comparison on the resulting data.

The novel part of the methods used in this thesis is the multilevel feature extraction, where the selection and extraction of features are convolutional and features at each level feed to the next level of feature extraction, which leads to producing more optimum feature.

Getting optimum features is essential to speed up getting results, but it also impacts the classification outcome, where feature complexity and detection accuracy are measured with a different

type of features and different quantities of features and components.

Representing the classification is the most crucial part since it represents the actual application and the encoding process. Here, we use the result to rebuild the personal profile centralized database, which is used to enhance accuracy and reduce errors.

The third part deals with features preprocessing methods, especially EEG signal segmentation and detection of various artifacts. The subsequent section provides a parametric description of EEG. It presents many different feature extraction and feature selection methods, and extracts information from EEG and from other signals that can be measured together with EEG.

An essential part of this work is designing and implementing appropriate algorithms for classifying automated neurophysiological states. In this part we compare various classification methods according to the accuracy and robustness of the classification. In most cases, the automated method with visual analysis is a fundamental criterion for its acceptance.

The proposed techniques provide a better understanding of EEG signal and expand the ability to reduce its complexity during extracting features and open a new window for using a neural network with the help of the clinical and update personal profile. Furthermore, a method could be used for trending technology such as wearable technology, IoT, 5G network, and upcoming technology.

1.1. Motivation:

The field of research is expanding quickly to meet the ever-more complex needs of human experience as a result of the unceasing progress of science and technology. Despite this, there are still a lot of unanswered questions regarding the human body. Researchers are increasingly focusing on a particular discussion regarding the relationship between brain signals and human emotion. The study of the human brain has gone on without pause over the years, but the technical barriers to acquiring brain signals have significantly slowed down the rate at which new findings in brain science can be made.

The development of techniques like wearable EEG sensors, magnetic resonance imaging (MRI), and functional magnetic resonance imaging (fMRI) in recent years has laid the groundwork for the advancement of brain science. These methods have made it possible to study the brain in real time and have opened up new avenues for research, leading to a better understanding of the brain's functions and disorders. As technology continues to advance, it is likely that even more sophisticated tools will be developed, further advancing our knowledge of the human brain (Mullen et al., 2015).

The electroencephalogram (EEG) is one of these techniques because it is relatively easy to record and analyze the signal. Because of this, it is not impractical to represent brain activity using an EEG signal in order to study the relationship between human emotion and brain activity. The results of earlier studies are being used more and more in numerous significant facets of daily life as the field of emotional research advances. EEG-based Emotion classification research is a promising field that can help diagnose depression and mental disorders. By analyzing the brain signals of patients, it can provide valuable insights to doctors in making accurate treatment decisions. This attention is driven by its potential to support doctors. (Alarcao & Fonseca, 2019)

When treating patients with emotional disorders or depression in clinical settings, it is crucial to understand their emotions. In the field of engineering, the most desired indicator to achieve brain-computer interaction is emotion as a metric of humanity (BCI). Understanding the emotions of patients with emotional disorders or depression can help clinicians provide effective treatment and support. Similarly, the use of emotion as a metric of humanity in

brain-computer interaction can lead to more natural and intuitive communication between humans and machines (Tangermann et al., 2012).

As a result of the quick development of AI technology, EEG-based emotional researchers get a quick push to process the EEG signal and enhance its trustworthiness and clarity, making EEG-based emotional research highly applicable. The development of the application required extensive research on EEG-based emotion recognition, and this research is crucial for the future expansion of scientific inquiry. In a nutshell, the main objective of the research is to determine the framework that is most effective and appropriate for the creation of high-performance EEG-based emotion recognition systems.

The ultimate goal is to create system-based applications that can be used to address problems that arise in the real world. One of the most challenging methods to use, EEG-based emotion recognition, has a high research value because it has the potential to produce significant results. EEG-based emotion recognition involves evaluating brain signals to determine a person's emotional state, and it can be used in various fields such as healthcare, marketing, and entertainment. Despite its challenges, this method has the potential to reshape the way we interact with technology and improve our understanding of human emotions.

1.2. Research Questions:

EEG-based emotion recognition technologies have matured in recent years as a result of advanced developments in bioelectric sensor technology. In comparison to other relatively quiet emotion recognition areas, it is undeniable that EEG-based emotion recognition research is still in its infancy due to high noise and non-linear and non-stationary nature. However, the development of the key technology for EEG-based emotion recognition has a significant impact on future technological development. Most of the trials were conducted in the computing unit, away from the clinic or lab, and researchers were still investigating how beneficial their online extensions were.

To begin, online emotional computing necessitates the timely generation of EEG data computations, limiting computation speed. To optimize for real-time needs, simpler mathematical models that reduce computation costs while maintaining sufficient classification accuracy are preferred.

A time-domain approach, for example, requires the processing time of an FFT while providing equivalent precision. Despite the enormous potential of online EEG categorization, the number of electrodes, power supply, computing speed, and data transfer speed all influence portable reader deployment.

As a result, such constraints should drive the development of novel data processing, transmission, and handling strategies. Furthermore, in some cases, processing duty may be split between an edge terminal unit (cap) and a central processing and storage unit, resulting in a hybrid method that maximizes the value of online EEG engagement apps and cloud-based infinite resources.

These strategies should aim to optimize the efficiency and accuracy of EEG data analysis while also addressing issues related to privacy and security. Additionally, the hybrid method can provide a more flexible and scalable approach to EEG data processing, allowing for real-time analysis and personalized feedback in online engagement apps.

In order to accomplish these goals, it is necessary to find answers to the following questions:

1. How can we improve the EEG signal processing methodology for real-time emotion recognition in a complex

and dynamic global environment, taking into account the inter-subject emotional sensitivity and differences, the experiential adjustments of the participants, and the unspecified situations that may arise during the data collection and analysis?

2. How can we implement multilayer feature extraction and selection methods that optimize the accuracy and performance of EEG-based emotion recognition classifiers in real-time, by combining the strengths of various feature types and overcoming the weaknesses of existing methods, and by exploring the impact of different classification methods and dimensions on the emotion recognition results?
3. How can we apply transfer learning and cloud computing to leverage historical data and user feedback for EEG-based emotion recognition using wearable and IoT devices, by investigating the feasibility and effectiveness of transferring knowledge between two unrelated topics or domains, and by addressing the challenges and limitations of new technology such as transmission, processing, and memory capacities?

1.3. Aim of the Thesis

This thesis addresses the problem of continuous online EEG readings of emotional subject status. And come up with new methodology optimized to classify then decode EEG signals and achieving below research goals:

1. Designing end-to-end data processing: There are a lot of research done on emotional detection; however, one common thing missed in these works is an end-to-end data evaluation, which has been covered in this research, including but not limited to experiment environment analysis. We started with our methodology of acquiring and evaluating data, after that preprocessing and reformatting the signals, then apply a variety of recursive methods for complexity reduction. Finally, we evaluated multi-segmenting and detection techniques.

2. Design and test a new EEG signal processing and multilayer feature extraction method: EEG signal processing is challenging due to the data density and complexity. This research proposes a new method to optimize the signal processing for unspecified situations and reduce the signal dimension. It also applies a multilayer feature extraction technique that uses the features from one method as the input for another. The final features are evaluated and compared in classifier accuracy.

3. Comparing different classification methods: General speaking, collected features can be classified with supervision and trained input or with no supervision using clustering techniques; different methods have been tested and compared through similar features for better performance and accuracy. In addition, the classifications have been tested on different dimensions, feature types, and feature quantity.

4. Enable Transfer Learning: One uniqueness of this thesis is the focus to match the era of the internet of things and wearable technology and focusing on optimizing the process

cost, using an additional layer of transfer learning including offline data of personality profile and online data from previous trials which help optimizing problem complexity and data density which is suitable with limitation of new technology such as transmission, processing, and memory resources.

1.4. Thesis Organization

This thesis follows a knowledge flow that goes from broad to deep in the area of processing EEG signals. Chapter one introduces the topic and the objectives of the thesis, which are to explore, analyze and classify EEG signals based on emotional states.

Chapter two explores EEG by providing an overview of common EEG signals and data processes, such as filtering, feature extraction and dimensionality reduction. It also reviews existing tools for ETL (Extract, Transform and Load) of EEG signals, and methods for visualizing useful information. Moreover, it gives an overview of popular EEG emotional datasets that will be discussed in more detail in the following chapters.

Chapter three has two sections that cover the main steps of our proposed methodology. The first section 3.1 covers the pre-classification process, which includes data preparation, feature selection and clustering. The second section 3.2 covers the classification process, which involves applying different supervised and unsupervised methods to label the EEG signals based on emotions.

As part of section 3.1, we cover the pre-classification process in four subsections. Section 3.1.1 explains our overall processing methodology at a high level, including all the processing layers that we applied to the EEG signals. Section 3.1.2 provides more in-depth information about data acquisition, which is a vital part of designing the experiment. It describes the two main emotional datasets that we used in our study, namely The SJTU Emotion EEG Dataset (SEED) and the Dataset for Emotion Analysis Using Physiological Signals (DEAP).

Section 3.1.3 shows the different methods of preparing and representing the data, and how we optimized it for further analysis. It also details the data processing steps that we performed, such as signal collection, filtering, reformatting and information extraction. Section 3.1.4 discusses the previous work on features and introduces a list of features that we selected and extracted from the EEG signals. These features belong to different groups of features, such as statistical features and entropy-based features.

Section 3.2 completes the data life cycle by making a decision and assigning a label to the action that was performed based on the

EEG signals. This label can then be used to inform appropriate actions in the future, such as providing feedback or intervention. We have experimented with various classification methods in this section, each with its own strengths and weaknesses, and compared their performance and accuracy. Section 3.2.1 discussed the most common supervised classification methods that we applied and assessed in our study, such as logistic regression, decision trees and support vector machines. These methods are based on learning from labeled data and applying rules or functions to new data.

Section 3.2.2 explained the convolutional neural network method that we used for classification, which is a type of deep learning method that can learn from complex and high-dimensional data. It included the network structure, the number of inputs, layers and outputs in detail, as well as the training and testing steps that we followed. Similarly, Section 3.2.3 described our recurrent neural network design in detail, which is another type of deep learning method that can learn from sequential data. It included the recurrence mechanism that enables the network to capture temporal patterns in the data, as well as the network parameters and evaluation metrics.

In chapter four, we present and discuss the results of our experiments on EEG signal processing, feature processing and classification methods. We apply and evaluate different methods and techniques on the SEED dataset, which contains EEG signals of emotional states. We also analyze the effects of various parameters and factors on the results, such as signal length, feature selection, network architecture and training settings. We show that our proposed methods of CNN and RNN can achieve better results than the traditional supervised methods and can capture the temporal and spatial patterns of EEG signals more effectively.

In the last chapter, we summarize the main contributions and findings of our thesis. We also discuss the limitations and challenges that we faced during our study and suggest some directions for future work. We conclude that our proposed methods of CNN and RNN can effectively process and classify EEG signals based on emotional states and can provide useful insights for applications such as emotion recognition, affective computing and brain-computer interfaces. We also highlight the potential of using deep learning methods for EEG analysis, and

the need for more research and development in this field. One of our main focuses is on real-time applications, where we aim to improve the speed and accuracy of our methods, and to explore more scenarios and use cases where they can be applied.

Chapter Two:
Research Background

This thesis focuses on utilizing readily available biomedical data, i.e., EEG signals, to understand better subjects' status in specific areas such as emotion in our case. Broader umbrella of applications utilizing brain data is known as a brain-computer interface (BCI) which is recognized as a system that translates the brain activity generated signals into usable data to analyze the subject current emotional status. This activity is executed and processed by the BCI system (Clerc et al., 2016).

The concept of translating brain signals into instructions that control an effector is central to non-invasive EEG-based BCI, which can be characterized both by the method that is used to measure brain activity and by the technique that brain signals are translated into instructions—for example, controlling the remote controller to change home electronic devices for disabled individuals. Similar non-motor imagery subject matters are amenable to using methods in a similar vein.

This research has a significant approach of covering the entire flow of data, starting from comparing and selecting between various datasets, analyzing the experiments' environment, comparing, and selecting significant features, classification, visualization, and optimization of the complexity of each mentioned process. We worked with the most prevalent clinical emotional datasets. Since there is much research and publications out there using basic and hybrid techniques for emotional classification, we will not mention all of them. However, the first part of this chapter will mention only work or applications related to emotional detection topics. In the second part of the chapter, we will mention closer research and techniques used in emotional detection.

2.1. Human Emotions

In order to survive, all creatures, including single-cell organisms, must be able to recognize and respond to significant stimuli. For instance, bacteria seek nutrition and avoid toxic substances (Macnab & Koshland, 1972).

Aristotle stated, "Emotions are all sentiments that so transform persons as to influence their judgments and are accompanied by either pain or pleasure." These include wrath, pity, fear, and similar emotions with opposites (Dow, 2011).

Fear and fury can promote an organism's survival by causing it to escape to safety or fight to protect itself, for example. It is possible to construct more complex or compound emotions by combining the elements that make up fundamental emotions (Ekman, 1992). Despite the widespread acceptance of the notion of basic emotions among psychologists, there is no unanimity regarding the exact number of fundamental emotions. Robert Plutchik postulated eight fundamental emotions and placed them on a color wheel: anger, fear, sorrow, disgust, surprise, anticipation, trust, and happiness. Ekman initially recommended seven primary emotions: fear, anger, joy, sadness, contempt, disgust, and surprise; however, he altered it to six: fear, rage, joy, sorrow, contempt, and disgust. (Plutchik, 1991)

2.1.1. Emotion Models

Complex states of human emotion manifest themselves in outwardly visible ways, such as in the shape of a person's face, the way they move their body, and the tone of their voice. In order to understand human emotions, we need emotion models.

In the field of emotional research, establishing and collecting real emotional labels presents a significant challenge. Factors such as historical background, geographical location, cultural background, and personal characteristics all have a role in shaping an individual's emotional state. Depending on their own interests and expertise, researchers may construct a variety of emotional models.

Three emotion models are introduced here.

Discrete Model

Discrete models are based on universally present fundamental emotions and are differentiated from one another by means of distinct physiological hypotheses. Happiness, surprise, wrath, fear, sorrow, and disgust are some examples of emotions that have been recognized.

The development of this classification was predicated on the premise that these emotions exist in every culture globally. (Hammal et al., 2007)(Qiang Ji et al., 2006)

The labelling scheme based on categories is quite natural and corresponds with people's experiences, as a result. Nevertheless, separate lists of emotions fall short of describing the vast array of emotions seen in everyday communication circumstances.

Some preliminary efforts identify non-basic emotional states from purposely shown facial expressions, including exhaustion (Qiang Ji et al., 2006) and mental states like agreement, concentration, curiosity, thought, confusion, and frustration. A collection of feelings, however, is only a list of labels with no true connection. It does not reflect a three-dimensional space and lacks algebra: every emotion must be investigated and identified separately.

Dimensional model

To address the limitations of the discrete model above, scholars like Whissell (Whissell, 2009) and Plutchik (Zangeneh Soroush et al., 2017) prefer to see emotional states as structurally interrelated. They view emotions as a continuous 2D/3D space with appraisal and activation being the dimensions. Dimensional representations are appealing because they may be used to describe a vast array of emotional experiences. In the real world, feelings don't just change from one universal emotion to another on their own. They change throughout time and traverse various intermediary feelings. Dimensional techniques are much better at dealing with different levels of emotions and changes in how people feel over time. See Figure **1** (adapted from Calvo & D'Mello, 2010) for an example of the 2D distribution of affective stimuli.

However, relatively few researchers have selected dimensional or descriptive emotional leveling, and those have been primarily

concerned with the creation of facial synthetics, data analysis, or psychological investigations rather than emotion recognition. Additionally, the descriptive potential of 2D/3D space is lost when the problem is reduced to a two-class classification (happiness vs. sadness or waking vs. sleeping), a four-class (quadrants of 2D/3D space), or other multiclass (where classes reflect distinct affective space sub-areas). (Gunes et al., 2011)

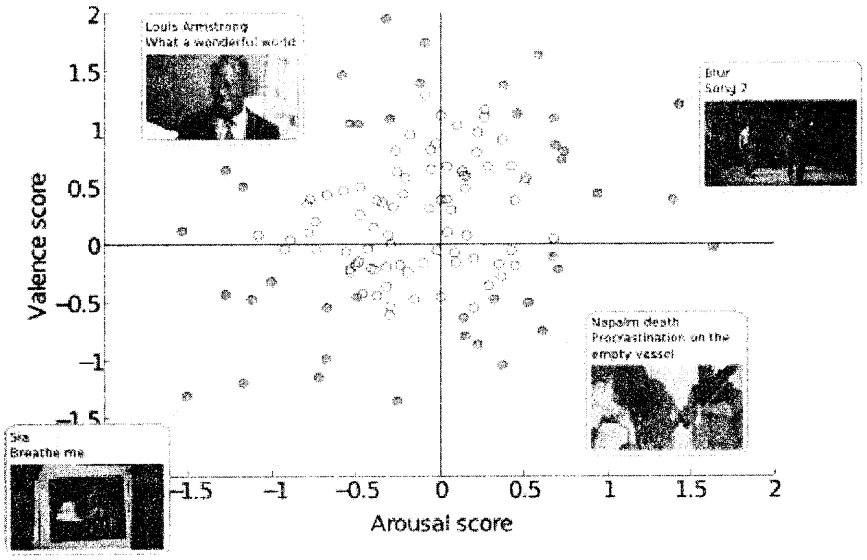


Figure 1: 2D Distribution of affective stimuli in the two-dimensional affective-valence space, Source: adapted from (Koelstra et al., 2012).

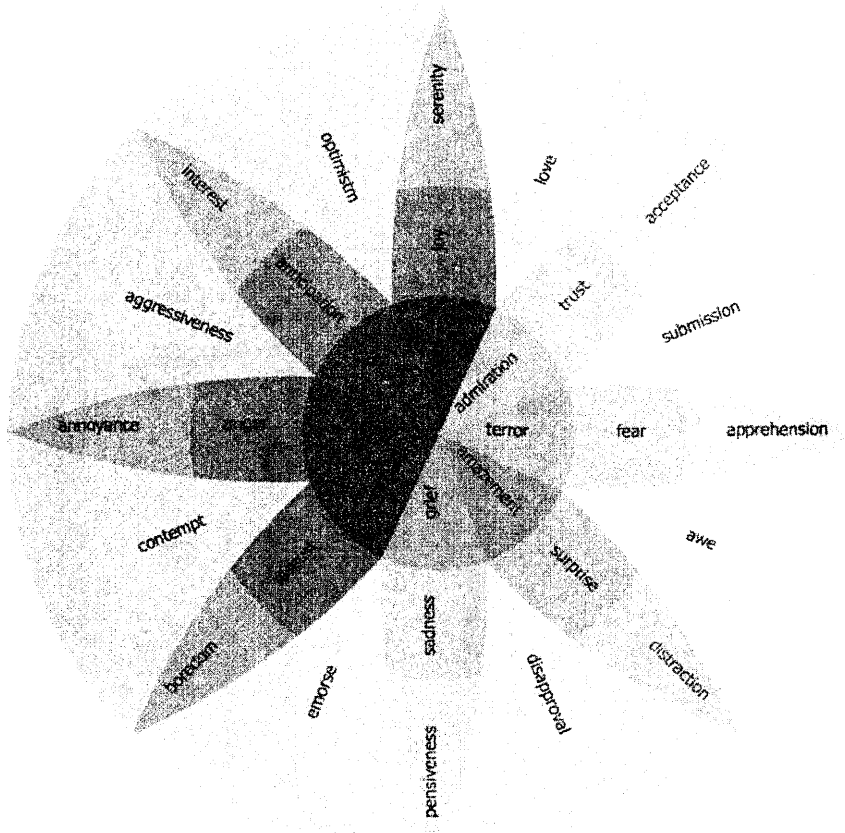


Figure 2 :Plutchik's 3D model "wheel of emotions" depicting fundamental and derived emotions, Source: (Nielek et al., 2017).

Scale Model

The Self-Assessment Manikin (SAM) is a picture-based questionnaire designed to quantify an individual's emotional reaction. (Bradley & Lang, 1994) The SAM questionnaire measures valence, arousal, and dominance, the three components of the 3D emotion paradigm. The questionnaire's symbols are meant to be straightforward and self-explanatory, reducing the impact of the individuals' diverse cultural backgrounds.

To be precise, single-item gauges evaluate reaction valence, perceived arousal, and dominance perceptions.

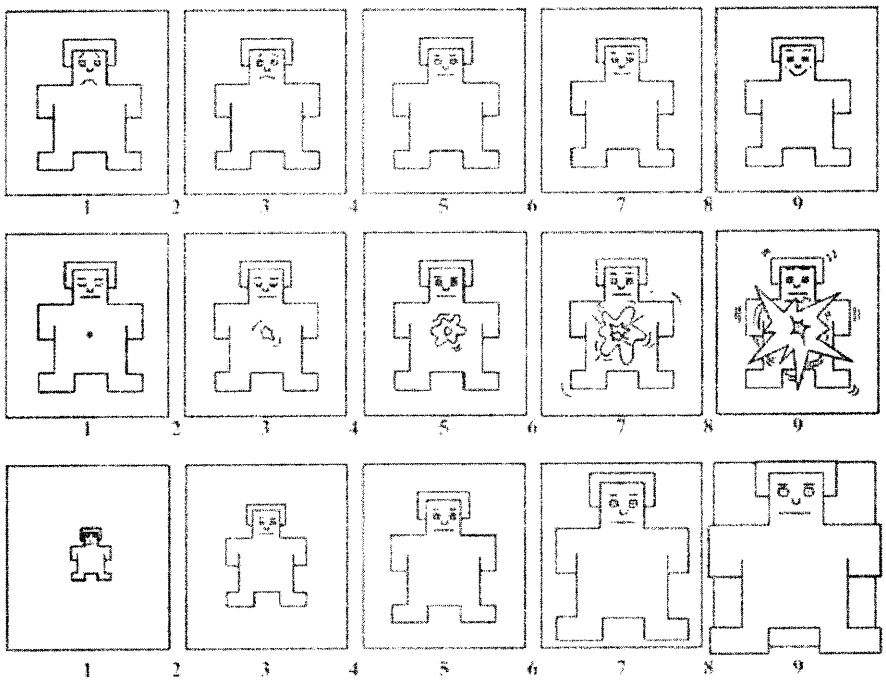


Figure 3: Validity (top), arousal (middle), and dominance (bottom) SAM (Self-Assessment Manikin), Source: from (Gatti et al., 2018).

2.2. Brain and EEG Signals

EEG refers to monitoring and storing the brain's natural electrical activity over time. Physiologists start recording EEG of animals by the end of the 18th century; later, in 1924, Hans Berger - a German physiologist - innovated the electroencephalogram and recorded the first human EEG, considered one of the most remarkable innovations in clinical neurology's history (Stone & Hughes, 2013).

1930s medical practitioners started analyzing EEG waves to find patterns of different clinical status; one of the first clinical observations of EEG was epileptiform spikes. In 1934, Fisher and Lowenbach the next year, Gibbs, Davis, and Lennox described a clinical pattern of seizures. Later, Gibbs and Jasper pattern of epilepsy. Even though many explorations were done, The American EEG Society was found by the end of the 1940s.

The first robotics application controlled by EEG was proposed by Stevo Bozinovski, Mihail Sestakov, and Liljana Bozinovska in 1988. The scientist went further than receiving the signal from the brain as a form of EEG signal to a new era where the brain could be a transceiver. In 2019, L. Jiang, A. Stocco, D. M. Losey, et al. designed Brain-to-Brain Interface was tested to calibrate thought between the group, which showed a very high success rate (Jiang et al., 2019)(Chang et al., 2016).

2.2.1. Brain Functional Structure

The human brain is the most complicated portion of any organ and serves as the top of the food chain for all neurological systems. It comprises the left and right hemispheres, in addition to the endplate located at the front of the third ventricle and serves as a connection between the hemispheres. The cerebral cortex is a relatively thin layer that may be found on the surface of the brain. With the establishment of linked giant fibers between the left and right hemispheres, cooperation between the two halves of the brain may be achieved. The topography of the cerebral hemisphere is irregular and riddled with a variety of sulci (Raichle et al., 2001).

The term "lateral fissure" refers to an oblique sulcus that can be seen on the dorsal side of the either cerebral hemisphere. Each of the hemispheres is further subdivided into four lobes by these

sulci and fissures. The frontal lobe is the biggest of the four lobes, and it is located above the central sulcus and the lateral fissure. It also occupies approximately one-third of the cerebral hemisphere. Figure 4 provides a basic breakdown of the structure and function of this structure.

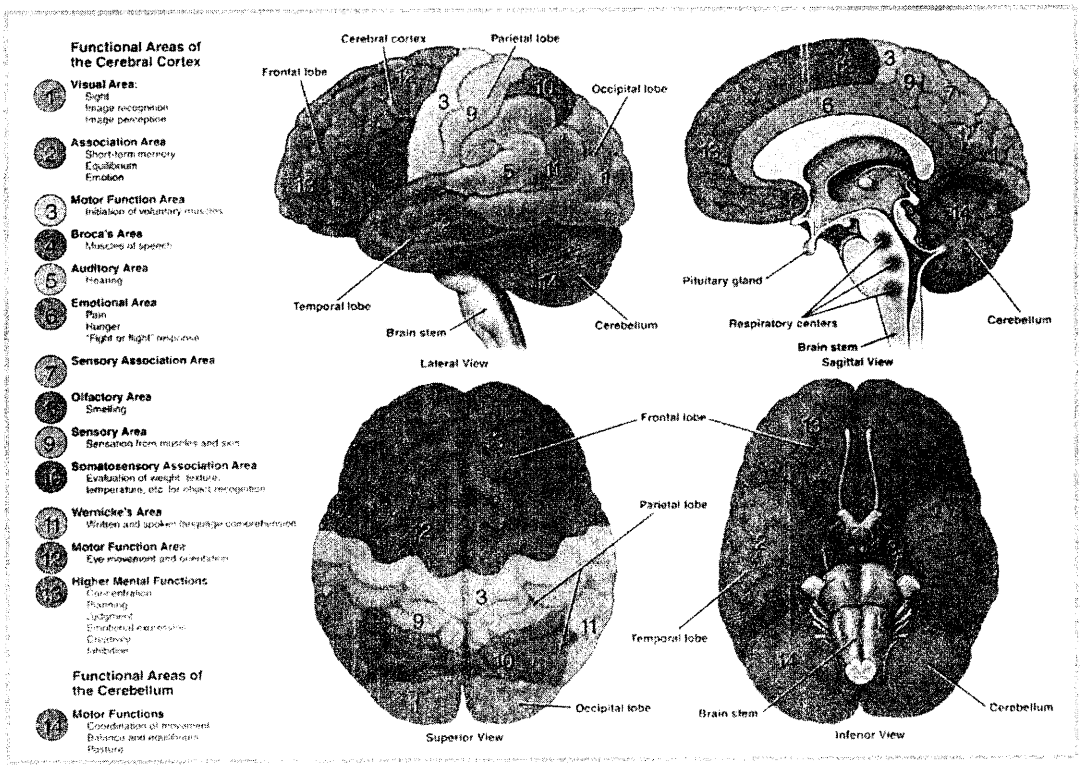


Figure 4: Basic breakdown of the structure and function of this structure, Source: adapted from (Cudlenco et al., 2020).

The temporal lobe can be found immediately underneath the lateral fissure. The parietal lobe may be found in the portion of the brain that is found immediately following the central sulcus and directly above the lateral fissure. The occipital lobe is the third lobe of the brain, located after the parietal lobe and the temporal lobe and above the cerebellum. In every section of the brain, there are numerous nerve centers as well as large neurons.

As a result, various functions for the activities are controlled by various regions of the brain, which together constitute a divisional functional structure of the cerebral cortex.

2.2.2. EEG Signals

Electroencephalography (EEG) measures voltage variations produced by the ionic current of the brain's neurons. Clinically speaking, after placing multiple electrodes on the scalp, electrodes will start record signals known as EEG. Electroencephalography (EEG) measures voltage variations produced by the ionic current of the brain's neurons, sample of signal representation non-digital and digital can be seen in Figure 5.

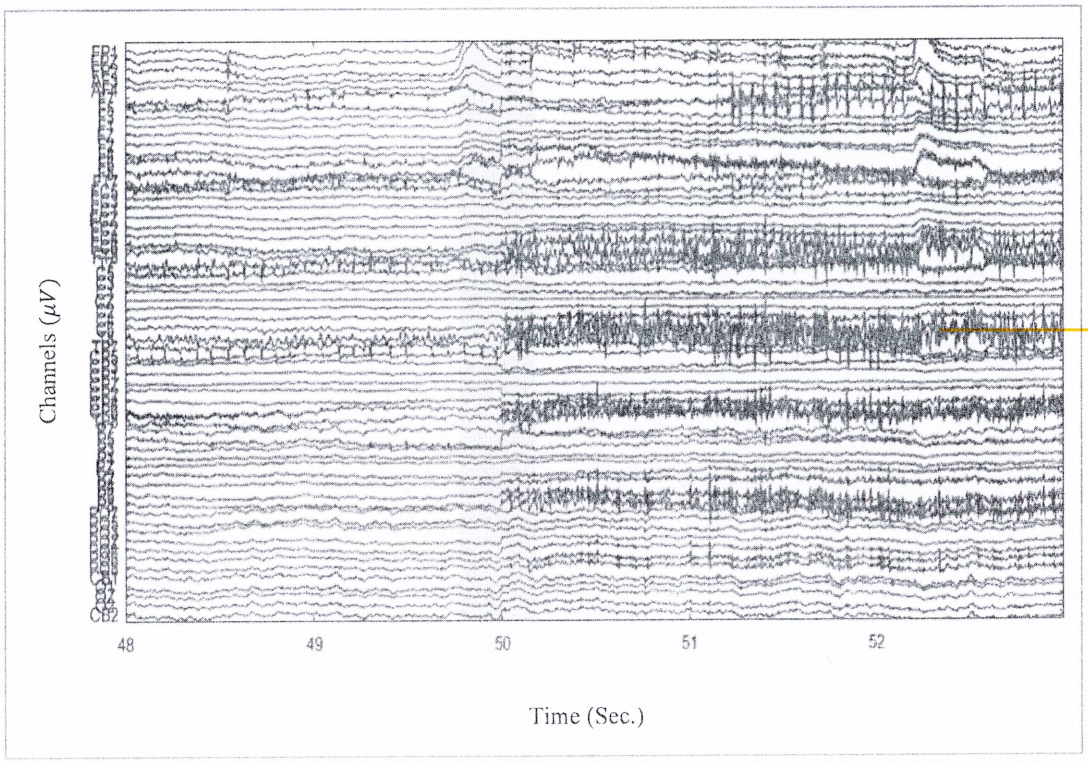


Figure 5: A time-domain EEG graph with 64 channels, created by MATLAB.

Clinically speaking, after placing multiple electrodes on the scalp, electrodes will start recording signals known as EEG. The practitioner usually prepares the scalp area by light abrasion and adds conductive gel to increase electrodes sensitivity. There are many EEG measurement systems. When the system has a low number of electrodes, they are attached directly one by one to the wires. However, if the system has many electrodes, it uses caps or nets with pre-attached electrodes. (Sallout & Mattar, 2022)(Mattar et al., 2018)

As seen in Figure 6, As clearly seen in Figure 6, the frequency spectrum during the emotion-triggered state has noticeable power changes at (8.0 Hz, 15.0 Hz, 22.0 Hz, 29.0 Hz, 36.0 Hz). These frequencies correspond to the affected band and the related brain area, which can have different effects on emotional stability or instability outcomes. (Hajare & Kadam, 2021)

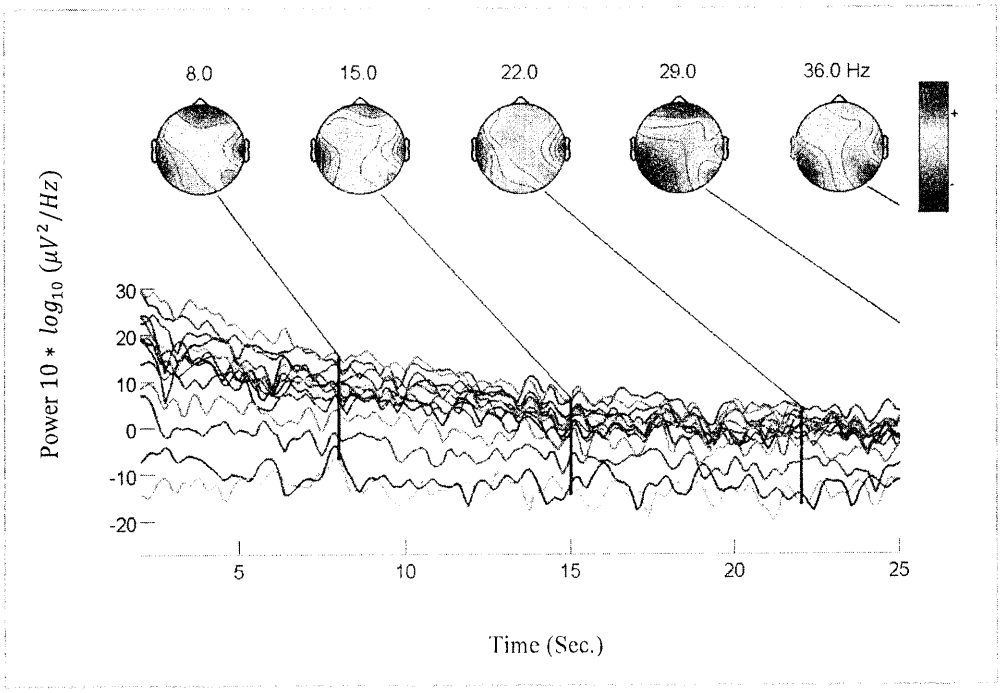


Figure 6: EEGLAB frequency-domain power spectrum for all (64 channels).

The International (10–20) system is the most know system for clinical and research applications. In this system, Electrode locations and names are defined to unify experiment design across laboratories.

Even though names and locations are defined, the number of electrodes may differ based on the application, from 19 electrodes for clinical applications to 64, 128, or 256 electrodes for higher-density applications.

2.2.3. Signal Processing

Manual Signal Processing Techniques

By visually examining them, clinical practitioners usually monitor EEG measures for any abnormalities. They use this method for all biopotentials measures, such as ECG, EEG, EMG, and EOG. This is a simple form of signal processing that can get detailed information from these signals.

However, this examination depends on the time and frequency domains and the practitioners' skill to identify specific patterns based on their learning and experience. This is a very basic task that only an expert eye can perform. To uncover more information beyond the known pattern recognition, signal processing techniques and computer-aided analyses are required. (Motamedi-Fakhr et al., 2014)

The analysis of the EEG signals is slightly challenging due to the signal nature. The raw signals are naturally inconstant, with a very high nonlinearity relation between the different source channels in the time domain, in addition to the presence of disturbing noise in addition to poor signal-noise ratio. The key target of designing signal processing algorithms is to improve the signal quality before further data modeling.

Modern Signal Processing Techniques

Manual evaluation of EEG reads could be a very productive method for simple applications. However, with higher density EEG read, new hurdles come up, where signals by nature are unpredictable with high interference; in other words, frequency, and power observations of all those channels would not be practical, and processing patches over time will give sight for the larger image. Therefore, more advanced hardware and techniques are needed to store, retrieve, and process more signal channels and for a longer duration. Besides frequency and power, more research topics are explored to optimize EEG signal complexity with minimal data loss.

For new applications, historical data plays a pivotal role in signal evaluation; therefore, it is crucial to have savvy methods that combine any convenient data from biometric or nonbiometric sources to be capable of coming up with practical solutions that

can be assembled, prepared, stored, evaluated and fed back to the system.

The rest of this unit will cover the modern EEG processing methods, which increase daily with increasing attention to new smart applications. With a quick look on related publications done on the topic. Since EEG evaluation includes a long processing flow that can be an area of interest, most of the publications solve a specific area without contributing offering a complete solution that can be used without grasping the particular problem context. So, to be able to solve a problem or present a solution, the EEG signal evaluation process must be designed to cover all stages.

2.2.4. Artifact Removal

EEG is known to have a high temporal resolution with low SNR and habitually gets interfered with by a wide range of noise, which will end in numerous artifacts. Thinking of the brain as the main processor of actions, senses and thoughts of the human, we end up with endless possibilities of artifacts causes, externally as the experiment environment and laboratory tools, or internal events like eye activities, muscle activities, and cardiac activities are known as physiological artifacts, which may interfere with original neural information and lead to misleadingly predictions for applications.

Accordingly, it is vital to utilize the preprocessing stage punctually in the laboratory or clinic by defining possible artifacts and removing them. For this purpose, many techniques can be used here, such as Regression, Blind Source Separation (BSS), Empirical-mode Decomposition (EMD) and Wavelet Transform algorithm.

2.2.5. Feature Extraction and Selection

As we have seen before, EEG signals have a lot of complexity, especially when they are used for new applications that rely on machine learning models. Most machine learning models are supervised, so they need to reduce the inputs to the minimum for better model training optimization; That is why feature dimension reduction plays a key role in enhancing the training machine learning models. Hereafter, feature selection and extraction come into sight, which is usually joint as a single step since both have

the same target to reducing dimensionality and, as a result, decreasing model complexity and overfitting, even though each has different functionality. (Gannouni et al., 2021; Wang & Wang, 2021)

From a high-level perspective, feature selection is to specify the working domain with only a portion of the original features to reduce complexity, increase the computing speed and reduce the out-of-sample error caused by noise. There are many feature selection techniques such as the Decision Tree Algorithm, Random Forest algorithm, and Greedy search algorithms. Unlike feature selection, Feature extraction aims to extract new features from the original features, which mostly have different characteristics as a form of compression or transformation of the features. The extracted feature covers most of the original information with less data density and complexity. There are many of feature extraction methods where most of which are enhancement of main extraction methods such as statistical methods, Wavelet Transform (WT), Fourier Transform (STFT), and Power Spectral Density (PSD). In addition to those methods, research commonly uses reduction methods to develop new hybrid methods such as Principal Component Analysis (PCA). (J. Li et al., 2018; Venkatesh & Anuradha, 2019)

2.2.6. Classification Techniques

The last step of the EEG evaluation journey is the classification process, which evaluates and categorizes EEG signals using Supervised or unsupervised classification methods. The researcher uses a lot of famous individual classifiers, some of which are specialized on a specific type of data or problem and show superiority. Recently researchers have started to investigate numerous mixtures of those individual classifiers to increase the accuracy and overcome individual classifier limitations.

Nowadays, some shallow machine learning algorithms are popularly practiced in EEG emotion classification, such as the Support Vector Machine (SVM), k-Nearest Neighbor (KNN), Random Forest (RF), and Naive Bayes (NB). Lately, with more complex situations and needs comes up, more attention of researchers goes to deep learning methods have been, such as Recurrent Neural Network (RNN), Convolutional Neural Network (CNN), Artificial Neural Network (ANN), and Long- and Short-

Term Memory (LSTM) which can result with better accuracy than the shallow models. (J. Liu et al., 2020; Moussa et al., 2022; Topic & Russo, 2021)

2.3. Available Data and Tools

2.3.1. EEG Datasets

The collection of EEG signals from the patient is the first stage in the process of analyzing EEG signals. This phase might be difficult not because it requires reading the data but rather because EEG is a basic two-dimensional signal. On the other hand, managing the experiment's environment ensures that it is identical and consistent throughout all trials. If this were not the case, the outcome would not be trustworthy, and the conclusion may be erroneous. As a result, being able to perform EEG signal collection requires a high degree of clinical expertise as well as specialized training and qualifications.

Researchers often have access to complete information on the experimental design, laboratory setting, intended subject, and even procedural details through the use of databases and datasets that are already in existence.

Because of the factors that have been discussed above, several datasets have been released that record EEG. These datasets are often meant to represent EEG activity in a preset setting, such as Motor-Imagery or Emotionally linked datasets. The primary EEG experiment setup categories are described in detail in Figure 7 below. These datasets are often compiled in order to cover a topic that is widely studied in the field of EEG.

In this context, researchers are aiming to get more recent datasets in order to check and compare their findings across a greater number of those datasets. Since EEG prediction is closely connected to robotics and smart applications, motor imagery is regarded as the dataset's oldest and most available topic. This is because it results from the brain simulating a physical action. Psychological health topics, such as Alzheimer's disease, Autism, and Epilepsy topics, as well as the Slow-Cortical Potentials in schizophrenia patients, are all examples of neural disorders that hold a significant portion of the available EEG datasets.

This is due to the fact that these topics explain questions that standard clinical techniques have yet to answer. The topic of emotional health is now trending due to the increased attention that it has received in recent years, particularly in light of the significant societal shifts and lifestyle changes that have taken place in the community.

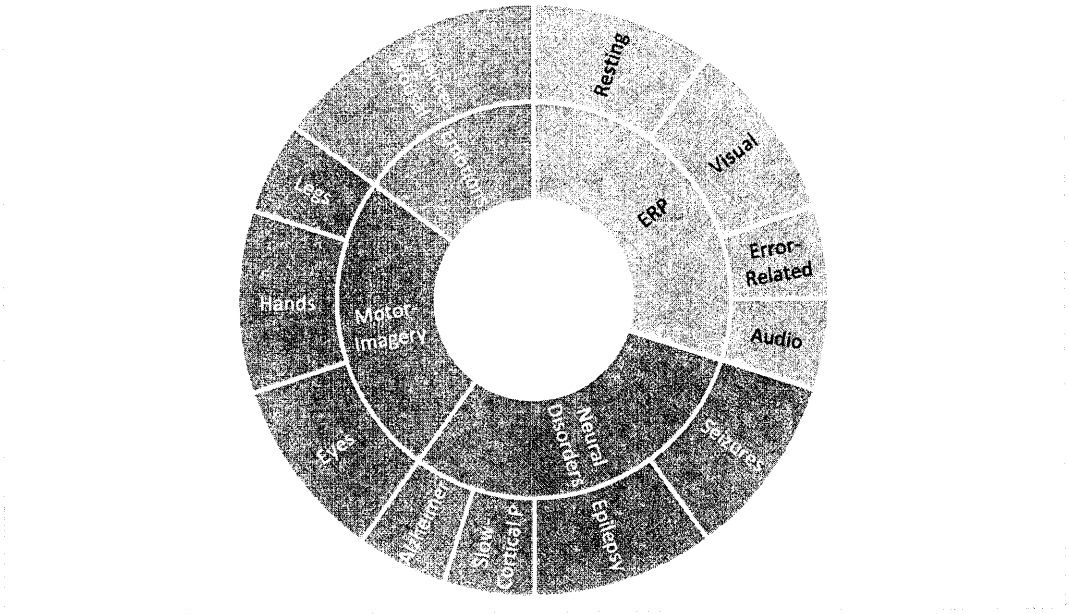


Figure 7: Categories of existing EEG datasets.

There are databases that are specialized on biomedical datasets, such as PhysioNet and OpenNeuro. Some of the current biomedical datasets are accessible to the public and are available without charge. PhysioNet is a database that is administered by the MIT Laboratory for Computational Physiology. It is a repository of medical research data that is freely available to the public, and it has tens of datasets that capture biological signals (Zangeneh Soroush et al., 2017). OpenNeuro is another free and open platform for sharing EEG, ECoGMRI, MEG, PET, and ASL data. At the moment, it contains more than 600 datasets including information on more than 20,000 individuals (Markiewicz et al., 2021).

2.3.2. EEG Hardware

In the remaining sections of this chapter, we will cover the existing hardware and software used to collect and process EEG signals, which vary from advanced equipment that require high technical skills to use up to wearable and mobile tools which can be used out of the books and individuals require only to watch short videos or read simple tutorials or manuals.

In general, EEG recording may be divided into two types: invasive and non-invasive. Instruments read signals from electrodes inserted within the skull during invasive EEG recording, which is often done therapeutically and under practitioner ongoing care. Because the implanted electrodes in invasive EEG are closer to the brain, the invasive approach interferes with brain signals with smaller spatial scales, greater amplitudes, and less noise and other artefacts, such as eye blinks.

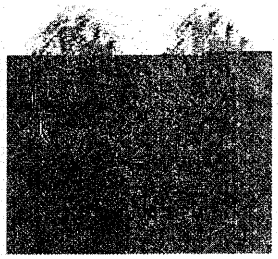
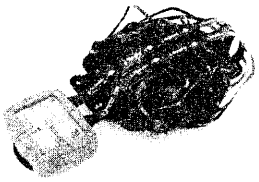
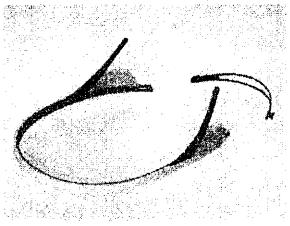
However, using an invasive EEG needs more specialized knowledge and higher clinical risks, such as infections and other brain damage, in addition to the time-consuming procedure of placing electrodes. Non-invasive EEG recordings, on the other hand, are obtained externally from electrodes connected directly to the scalp surface, and in this case, the environment requires less preparation and practitioner engagement, and it can always be set up easily and quickly; thus, it can be repeated easily with no harm to the subject.

Even though non-invasive EEG shows less accuracy compared to invasive EEG, it gives advantages for newer applications that require more online and real-time reading in addition to applications that require mobility or real-life readings such as robotics, word processing, or fatigue monitoring. The current non-invasive systems design differs from capturing signals, such as motor imagery, event-related synchronization (ERS), steady-state visual evoked potentials (SSVEP), slow cortical potentials (SCPs), and P300 of event-related potentials (ERPs).

In contracst to grade level EEG equipment which can cost more than 20,000$ such as, NeurOptimal® Neurofeedback Business bundle. The new non-invasive systems are more affordable and effective. Many non-invasive EEG devices are available in the market for less than 500 us dollars and they are gaining popularity because of their efficiency. Table 1 shows some examples of those pieces of equipment.

Because of the complicated behavior of EEG signals and the high rate of signal frequency, it may be impossible to recognize single changes by looking at the signal waves representation, especially for short periods of time, as shown in Figure 26. Those phenomena of EEG signals make it difficult to extract information from them using only the naked eye.

Table 1: Examples of commercial non-invasive systems.

Image	Brand	Min Price	Max No. of channels
	Neuroelectrics	1000$	32
	Gtech	1000$	32
	Muse	300$	4
	Emotive	300$	5
	Flowtime	200$	2

2.3.3. EEG Software

In this section, we will cover the existing hardware and software used to collect and process EEG signals, which vary from advanced equipment that require high technical skills to use up to wearable and mobile tools which can be used out of the books and individuals require only to watch small videos or read simple tutorials or manuals.

After preparing datasets, the time comes to analyze and visualize EEG signals; with a simple search, you can find tens of proprietary software that many commercial EEG kits bundle, especially in the medical field like imotions, Neuroelectrics, gtech, and emotiv that designed to support the products and hardware that usually sold by the same companies. There are some other general purposes and comprehensive software available that you can use to visualize EEG data, where a significant portion of it is based on the MATLAB platform.

EEGLab is a well-known software built on top of MATLAB that can compute and visualize continuous and event-related EEG data. BIOSIG is an open-source software toolbox for MATLAB (and Octave) that offers biomedical signal visualization. The EDF browser supports most of the common EEG file formats then converts them to EDF. It offers signal visualization and filtering on power and frequency bands. One interesting feature is the ability of data streaming and online visualization.

There are also a lot of other toolboxes like Brainstorm, Tempo, LORETA (low-resolution brain electromagnetic tomography), and PSGlab. Recently with the raise of Python new libraries and tools are exists such as PyEEG that was originally produced for epilepsy detection and enable EEG features extracted (Gerla et al., 2010; Schlogl & Brunner, 2008; Swartz Center for Computational Neuroscience, 2017; Vidaurre et al., 2011)(Bao et al., 2011).

2.4. Related Research

Quantitative computation of medical applications has lately received a lot of attention due to its applicability to a wide spectrum of health-related problems, for example, patients who are experiencing physical, mental or even emotional difficulties (Lopez-Bernal et al., 2022). It is essential to assess the level of mental distress experienced by a large number of patients who are immobile in their bodies and have trouble communicating. A high-level summary of existing BCI applications may be seen in the below figure.

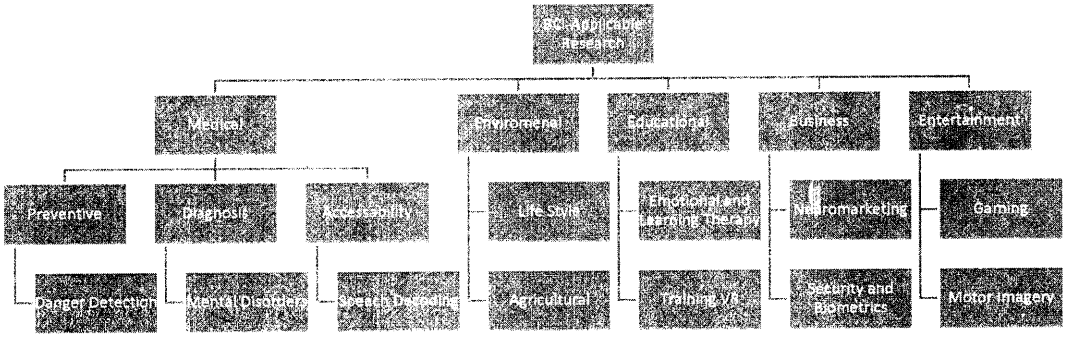


Figure 8: Modern BCI application areas.

2.4.1. Modern Applications

Using BCI, we think that shortly, new technology may offer a way out for patients who cannot communicate verbally. Those who are intubated, semi-anesthetized following an operation or surgery, and especially patients whose Glasgow coma scale (GCS) scores are low, cannot communicate the pain they are experiencing.

The GCS is a well-known measure of the level of awareness. An example of one of the difficulties faced by anesthesiologists is the task of monitoring and regulating the patient's level of discomfort throughout lengthy operations (such as orthopedic or neurosurgical procedures), which might last for several hours. (Wright, 2016).

Pain Pattern Recognition is another example related to pain. Various noninvasive medical imaging equipment have been used to study human pain (e.g., PET and fMRI which stand for positron emission tomography and functional magnetic resonance imaging, respectively). Electroencephalogram (EEG) devices, according to researchers, are the finest neuroimaging technology for pain

detection due to their high temporal resolution and ability to capture real-time, dynamic changes inside the brain. Many studies, for example, sought to construct an experiment that would objectively assess pain feeling using EEG and could be employed in a therapeutic environment (Lopes da Silva et al., 2010).

Another application area is biometric identification EEG signals decoding, which made remarkable improvement and is attracting growing attention in recent years (Yang & Deravi, 2017). Recently, the intriguing biometric potential of EEG has been studied for building personality identification and authentication systems. Researchers even examining the most effective biometric feature to be used for biometric recognition, i.e., the comparison of the power spectral density (PSD) properties of different EEG bands during a resting state. Thomas and Vinod show that the gamma band (30–50 Hz) has higher performance in biometric verification than the delta, theta, alpha, and beta bands. (Thomas & Vinod, 2018)

2.4.2. Emotional Detection Area

Apart from straightforward industrial applications that rely on human-robot interaction, recognizing an individual's emotional state has significant promise in the health arena, particularly in psychology and neuroscience, for example, to identify early on-sets of depression (Ackermann et al., 2016).

Focuses on the algorithmic identification of human emotions, such as displeasure or pleasure, from a range of inputs, including voice, facial movements, and, more recently, EEG signals. Nowadays, most pattern recognition and machine learning processes, in addition to BCIs, employ a classifier and use feature extraction and selection techniques to represent signals in a compact form. (Urigüen & Garcia-Zapirain, 2015) (Nakisa et al., 2018)

Before extracting features from the generated signals, BCI filters EEG signals in three domains: time-domain (band-pass filter), power-domain (spatial filter), and time-power domain.

Much researches have proven that EEG can detect emotional activity (Vijayakumar et al., 2017) (Arendt-Nielsen & Chen, 2003). Studies have also been conducted to demonstrate the activation of certain frequency bands in the pain scores, primarily the alpha band and, more recently, the gamma band (Nir et al.,

2010). Similarly, to the emotion domain, there is an increasing number of research that use the tonic thermal stimulation to imitate pain-related reactions (Huishi Zhang et al., 2016; Schulz et al., 2015).

2.4.3. Transfer Knowledge

This section analyses works on BCI employing the most important EEG signals and transfer learning approaches for our intended investigation. We presented research on inter-subject, inter-database, and EEG channel selection for emotion categorization.

Because raw data is already recorded and preserved locally, decreasing raw data is not difficult and attracts less attention than processing features; nevertheless, in the event of a new real-time application, optimization should be done as soon as feasible. Wrapper-based methods aim to match the channel selection step to the target model in the inter-subject case; however, they require multiple times retraining of the model across various subject and channel subsets, which frequently results in an unacceptably high computational cost, particularly in the case of deep neural networks (Topic et al., 2022).

Khalaf and Akcakaya proposed a transfer learning paradigm. They employed the Bhattacharyya distance to compare the conditional distributions of new and old subjects. In addition, the Wilcoxon rank-sum was utilized to determine the relevant EEG characteristics (Daoud & Bayoumi, 2019).

PCA is the base of most component analysis methods; researchers use many transfer learning techniques based on PCA or what is known as kernel-based versions of PCA known as principal component analysis (KPCA) (Schölkopf et al., 1998), which is an extended version of PCA that uses kernel techniques. KPCA was not initially designed for domain adaptation, however it has been incorporated for comparison with domain adaptation approaches such as Geodesic Flow Kernel (GFK) by Gong et al., Transfer Component Analysis (TCA) by Pan et al., and Fernando et al.'s method called Subspace Alignment (SA) (Pan et al., 2011; Yan et al., 2018a; W.-L. Zheng et al., 2015; W.-L. Zheng & Lu, 2016). Fortunately, many of those algorithms are already implemented and they are available online for free to be used with familiar software (Yan et al., 2018b).

Because of the domain feature augmentation that it employs, Maximum Independence Domain Adaptation (MIDA) is the only approach that is capable of managing numerous source domains. These methods are relatively similar, with only a few key distinctions. For instance, whereas TCA seeks to minimize the gap between the empirical means of the source domain and the target domain; however, SA seeks to minimize the Frobenius norm of the gap between the bases of the source domain and the target domain. Information-Theoretical Learning (ITL) works to reduce the amount of mutual information that exists between data samples and domain labels, while simultaneously working to increase the amount of mutual information that exists between target data and their estimated labels. (Shojaei & Razzazi, 2019; Y. Zhang et al., 2023)

Another form of operation that falls under the feature reduction category is feature selection. This kind of action is what has been employed to extract the pivot features. The pivotal features are those exhibit the same pattern of behavior across a variety of contexts. Because of the consistency of these features, they are able to serve as a bridge for the transmission of information (Khalaf & Akcakaya, 2020).

Hossain, Khosravi, and Nahavandhi came up with an example of instance-based transfer learning that they named enhanced "Active Transfer Learning (ATL)." The goal of this example was to reduce the discrepancies that exist across subjects that do not remain the same. The primary idea behind active transfer learning is that only the samples that include the most relevant information are labelled in such a way that it is anticipated that the learning performance of a new subject or activity would increase. In addition to this, they offered two ATL extensions to the method.

The first one was named Selective Instance Transfer with Active Learning (SITAL), and its objective was to choose examples from source domains that had distributions that were comparable to those of the destination domain. The second method was given the name Selective Informative Instance Transfer with Active Learning (SIITAL), and it is the one that chooses samples from the source domain based on their high classification accuracy and normalized entropy (Hossain et al., 2016).

We have identified some works that are connected to this topic. Jayaram et al. (2016) developed a model for transfer learning in

EEG-based BCIs. This model takes the use of various subjects and trials to boost performance. They proved that their approach was beneficial in reducing the amount of time used and that it could also outperform similar methods on identical datasets (Jayaram et al., 2016).

Shin et al. (2016) used a pre-trained deep neural network to solve difficulties using computer-aided detection and then fine-tuned the network (Shin et al., 2016).

Rodrigues et al. (2019) describe a transfer learning strategy that addresses the problem of dealing with statistical variances in EEG data gathered from various subjects throughout various sessions. Their research offered to make simple geometric adjustments to data points, the Procrustes analysis method may be used to provide a statistically convincing match between simulated and experimental data sets (Rodrigues et al., 2019).

Cimtay and Ekmekcioglu (2020) conducted research on weighted NN models that were built using the SEED emotional dataset. They then evaluated these models on the DEAP dataset, which resulted in satisfactory mean prediction accuracy (Cimtay & Ekmekcioglu, 2020).

Chapter Three:
Methodology

At the beginning of using EEG for diagnosing health-related findings, the experts rely on visual analysis of the EEG signal on the frequency or time domain. Most existing studies of EEG processing focus even on clinical experiments or on raw data processing, where authors, based on their experience, focus on solving a very narrow area of the problem. Even though because of the challenge of finding or creating the required data set, researchers mostly use a small data set, and sometimes they use simulated datasets, which of course, adds more doubts about the reality of those studies.

So usually, studies even focus on clinical part of the problem or the signal processing part; the clinical-centric studies are concerned with the clinical environment, the participant understanding, and the clinical problem definition and expected solution impact. On the other hand, the processing-centric studies concentrate on the signal shape and optimization method of processing the signal ending, putting their eye on the final classification output regardless of the previous stages' performance which ends up with theoretical results that may not get reflected on real-life problems. Therefore, we have studied the EEG signal classification to cover the widespread problem with a strong scientific methodology designed for real-life scenarios that align with the market trend of the internet of things and wearable technologies.

A review of the existing research reveals that many published studies on EEG signal evaluation belong to a small domain of work that concentrates on resolving specific problem topics from a very narrow technical angle. Furthermore, these publications often rely on small or simulated data due to the limitation of the available dataset. On the contrary, much work has been done on final classification methods, which can result in misleading conclusions that are not consistent with real-life scenarios because of bad experiment design and wrong expectations.

This chapter will talk about how we studied the whole journey of signal processing and designed and tested a detailed methodology that can be used for similar biomedical signal processing

applications and research, as per our research, which is novel and original in its approach and implementation, where every stage consists of different methods on multilayer architect that enhance auto-processing. The strength of auto-processing occurs clearly in the case of a lack of clinical information where clustering can segment signals accordingly and could be used for automating feature selection and extraction.

Despite the fact that much existing processing has been proposed and is available publicly in the domain, the majority of them can still belong in the primary sections. Figure 9 highlights the basic processing procedures, which will be detailed in greater detail later in this chapter. Our ultimate goal is to develop generic processing stages capable of evaluating any emotional dataset.

3.1. Pre-Classification Methodologies:

Challenges of Real Time Applications:

Detecting emotions using EEG has the advantage of moving toward real-world practical applications more quickly than other detection techniques due to affordable signal streaming of EEG signal information that fit with the continues human emotional changes. In addition to the availability of low-cost and high-portability devices, which are fundamentally important for real-time smart applications, it comes with new challenges.

The collection of EEG signals is the initial step in the development of an EEG-based emotional intelligence system. Recently, aimed at consumers that are both affordable and easy to transport were introduced to the market. With a few semi-dry electrodes and a lightweight cap, setting up these gadgets does not require the assistance of a specialist and can be done relatively effortlessly, and applying them on a larger scale can be a game changer, where limitless dataset options will be available related to size, update, subjects, and more individual base labels.

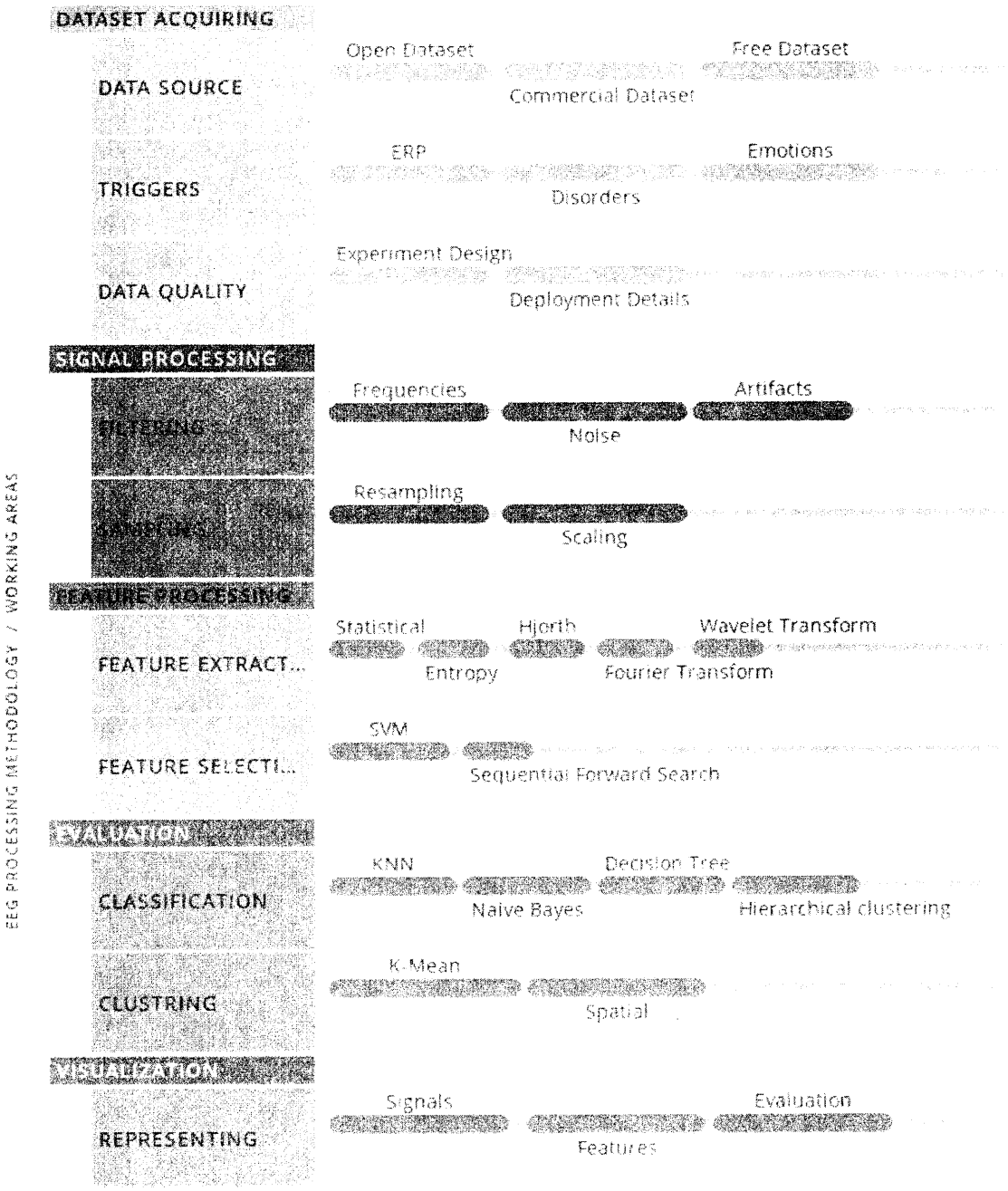


Figure 9: EEG processing methodology and working areas, (Image by Author).

Limitations on Portable Resources

Despite the enormous potential of online EEG categorization, deploying portable readers is restricted by hardware capabilities such as the number of electrodes, power supply, computing speed, and data transfer speed (G. Li & Chung, 2015). As a result, such constraints should drive the development of novel strategies for processing, sending, and dealing with data streams.

Furthermore, in some situations, processing duty may be allocated between the edge terminal unit (cap) and a centralized processing and storage unit, resulting in a hybrid method that optimizes the value of online EEG engagement applications and cloud-based infinite resources (Nahum, 2015). A new processing flow should be allocated to overcome this challenge, where a good classification can be achieved with a practical balance with online application requirements.

The Live Stream Computing Delay

The majority of the trials were carried out away from the clinic or lab to the computing unit, and researchers were still looking into how beneficial their online extensions were. For starters, online emotional computing demands the timely generation of EEG data computations, which limits computation speed. Simpler mathematical models that decrease computation costs while maintaining sufficient classification accuracy are preferred to optimize for real-time needs. For example, a time-domain approach needs processing time of an FFT while delivering equivalent precision.

In a study done by Arnin JKahani DLakany H et al. (Arnin et al., 2018), they evaluated Different Signal Processing Methods in Time and Frequency Domain and find out that there is a big difference in processing time as can be seen in Figure 10. In another research, V. Venkataraman , K. Ram Kumar, and N. Arunkumar computed extraction time of different entropy feature where a noticeable difference has been found as in Figure 11. As a result, it would be better for researchers to regard computing time, in addition to accuracy, as an important outcome and to indicate computation time in their studies (Arunkumar et al., 2018).

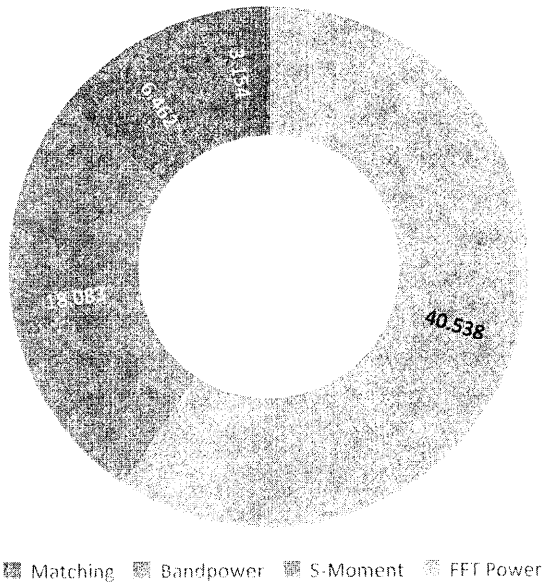


Figure 10: Outcomes of comparing the computation time (in μs) of verity of features in MATLAB (Arnin et al., 2018).

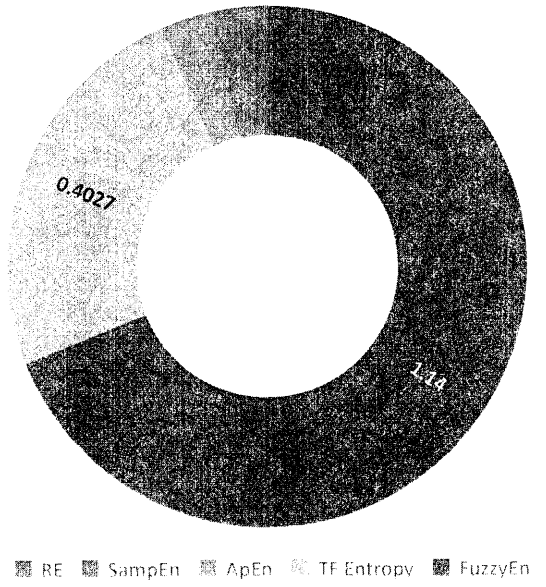


Figure 11: Duration of entropy feature extraction computation (Arunkumar et al., 2018; Z. Chen et al., 2020; Krishnan & Balasubramanian, 2016).

Model Generalization and Individual Variations

Emotional experiences are thought to be extremely personal. As a result, most EEG-based emotional computing investigations have been conducted subject-by-subject, with affective labels from each subject being utilized for training individual-based computer models.

However, such an approach greatly limits its widespread implementation because training and using individual-based models takes time and resources. A paradigm easily adaptable to the general public would significantly boost the appeal of EEG-based affective computing.

Because pursuing across-subject robust features is thought to be a very difficult goal to achieve, so a novel distributed and centralized technique can be used; for example, a self-personalized profile or personality categorization and label can be applied on edge; centralized transfer learning approaches can be used. Individual differences might be addressed using case-specific strategies.

The primary principle of this strategy is to selectively adjust a model to data or knowledge from existing subjects and apply them to a new subject to reduce individual differences. Furthermore, big data can attain equivalent performance to subject-dependent models when large enough datasets are available.

3.1.1. Overall Processing Methodology

This section will attempt to provide a high-level explanation of how we went about developing our adaptive online EEG emotion adaptive classification solution. We developed this solution in order to address the problems that were discussed earlier.

The term "Internet of Things" (IoT) was first used by Kevin Ashton in 1999 during a presentation at Procter & Gamble. Since then, the academic community and the community of ICT professionals have studied the fundamental idea underlying IoT solutions in great details. These things have the capacity to gather enormous amounts of unprocessed data, process it, and then convert it into information that is pertinent (Tripathy & Anuradha, 2020).

This ecosystem typically adopts preventative measures, reducing the amount of human involvement (or at least attempting to do so). Despite appearing straightforward, this scenario actually raises a number of unanswered questions. What kind of structure ought to be used in its place? Which of the requirements, compared to the others, is the most important? Which currently implemented communication standards should be used so that various device types can cooperate? What kind of application programming interface (API) needs to be created in order for a sensor to seamlessly integrate into an ecosystem? Numerous platforms and solutions were proposed for the Internet of Things.

Adaptive Processing Structure

We present an overall solution diagram that illustrates the primary components of the solution, such as the various data states, data resources, and data processing transformations. The layers of this solution are organized vertically according to the data form or processing stage, and the layers of this solution are organized horizontally according to the processing location and processing technique see Figure 12.

This process began with gathering data from various sources, sometimes known as "data acquisition," and was followed by the selection of data. Following that, we will discuss the extraction, selection, and reduction of features, and then we will conclude by discussing advanced classification algorithms. In a subsequent part of this article, we will talk about each layer in greater detail and some of the internal approaches. (Maksimenko et al., 2018)(Cattell & Mead, 2008)

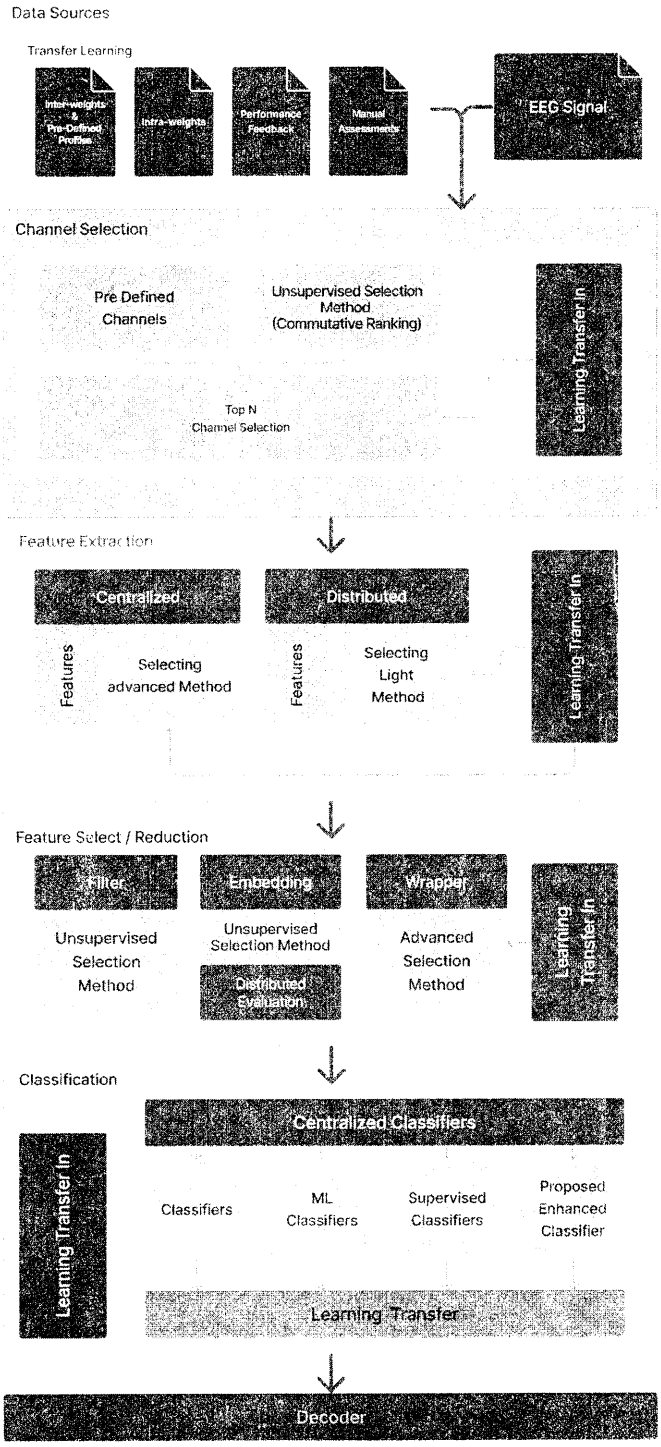


Figure 12: Overall proposed processing methodology.

Recursive Data Sources

Improving the ability to classify emotions experienced online is the primary focus of this research. Most bio-classifiers related to emotions center their attention on EEG data as the primary data source, with facial expressions serving as the secondary data source (Mehdizadehfar et al., 2020)(Moshirian Farahi et al., 2019). In contrast to the EEG signal, which can be obtained relatively easily, collecting data on facial expressions might be challenging, particularly in open environments and during normal daily activities (Black et al., 2017)(Brenner et al., 2014).

Datasets Acquisition

One technological problem in emotion research is triggering individuals' goal emotions and capturing appropriate signals in laboratory settings. Currently, there are three basic methods of emotional elicitation: Participants in intuitive emotions are exposed to the current world and experience spontaneous feelings influenced by their surroundings.

Individuals use mental images and memories in a controlled lab environment to place themselves in a certain emotional state. A particular emotional stimulus elicits emotions.

There are several stimuli for eliciting emotions, including pictures, text, music, and video. Furthermore, to eliminate individual base variations, researchers employ the evoked emotion type, which may be elicited by various stimuli, including pictures, text, music, and video. Videos, among these modalities, are more successful in maintaining steady emotional processing for longer periods and expose people to more realistic circumstances.

Even though the chosen method of emotion trigger yet utilized video clips might differ from an emotional effect standpoint, there were IAPS when making standardized visual clips. The International Affective Picture System (IAPS) (Bradley & Lang, 2017) is a collection of emotionally evocative color photos of events and things experienced in the human experience that men and women have scored on aspects of pleasure, arousal, and dominance. IAPS stimuli are frequently employed in human emotion research all over the world.

The International Affective Picture System (IAPS) is a collection of photographic pictures portraying a variety of typical and

uncommon human experiences that have been graded on scales assessing perceived pleasantness, arousal, and dominance, providing researchers with credible metrics for selecting stimuli for scientific experiments on emotion.

The IAPS stimuli are standardized based on pleasure and arousal ratings derived from emotion theory and data. Wundt (Wundt, 1894) was the first in the nineteenth century to suggest that effect emerges from differences in fundamental characteristics of pleasure and arousal. In a later study using factor analysis to evaluate evaluative language, Osgood and colleagues (e.g., Osgood (Osgood et al., 1957); see also Mehrabian and Russell (Mehrabian & Russell, 1980); Russell (Russell, 1980)) found that the hedonic valence dimension, which runs from unpleasant (angry and upset, annoyed, despairing, etc.) to pleasant, taken into account for the most variability in semantic judgments (glad, pleased, optimistic), while the arousal dimension, which runs from unaroused (calm, relaxed, drowsy, etc.) to high arousal, accounted for the least (excited, stimulated, wide awake).

This standard has been widely utilized in the creation of a large number of video stimulus databases. The following are some of the most well-known databases (Al-Shargie et al., 2019):

- SEED (Y. Li et al., 2019): Zheng chose 15 Chinese video snippets from notable Chinese films, each lasting four minutes. No explanation is required to stimulate three fundamental emotions (positive, neutral, and negative); the database has been tested on 15 Chinese individuals.
- DEAP (Chaudhary & Jaswal, 2021): For a subset of 40 music videos, Koelstra and colleagues chose 120 one-minute music videos classified by arousal, valence, and dominance, as well as physiological records with face films.
- HUMAINE (Douglas-Cowie et al., 2011): created by Douglas-Cowie and colleagues. HUMAINE comprises 50 movies ranging from 5 to 180 seconds in length, each with a variety of labels (emotionality states, contextual labels, relevant events, emotion expressions), as well as frame-based assessment (degree of excitement, domination, and predictability).
- FilmStim (Schaefer et al., 2010): created by Schaefer and colleagues. It was made up of 824 emotional film clips and 70 film excerpts that were 1 to 7 minutes long. There were 24 ways to classify them: qualitative arousal, negative and positive affect,

negative and positive affect rankings from the Variance Emotions Level, six emotion discreteness scores, and 15 mixed feelings scores. 364 people rated them.

- MAHNOB-HCI (Haamer et al., 2018)(Petridis et al., 2013): S. Petridis, B. Martinez, and M. Pantic recorded 20 film snippets with extended emotional keywords, arousal, valence, dominance, and predictability, together with face videos, EEG, audio, gaze, and peripheral physiological data.
- Emotional Movie Database (EMDB) (Carvalho et al., 2012): Sandra Carvalho, Jorge Leite, and colleagues produced the EMDB containing 52 silent film clips of 40 seconds each and global scores for the induced arousal, valence, and dominance dimensions.
- LIRIS-ACCEDE (Baveye et al., 2015): created by Baveye and colleagues; the database contains 9,800 clips ranging from 8 to 12 seconds in length, with extensive scores for arousal and valence dimensions and tested on hundreds of volunteers worldwide.

As we can see, a lot of emotional datasets exist, and to be able to evaluate our new classification methodology and get stable results, selecting a stable data set plays a crucial factor in it.

A significant weak point of most previous studies is the lack of evaluation of the datasets, especially in the emotion classification area since the availability is very limited. However, each of those datasets has been executed for a long time; for example, the SEED (SJTU Emotion EEG Dataset) has been collected for 24 months which is a very long time to collect a single dataset from a social point of view. (Sallout & Mattar, 2021)

This makes each of those datasets look like a group of datasets with varieties rather than an identical set of exercises, which we consider. So, the first stage we do is to evaluate global differences of datasets before we dig deep into the selected dataset, evaluate it is sub-experiment environment and pick the ideal set of experiments and sub-dataset.

Moreover, we evaluated many signal-processing techniques, including signal filtering, signal sampling and segmenting. Since there is no unified source format of EEG data, researchers with less technical skills find it very hard to consume the dataset to their available tool; that is why we programmed a complete

library set at the top of the MATLAB toolbox to simplify, unify and speed up the overall signal processing.

This library set includes functions for data loading and filtering, data optimizing, dimension reshaping, feature extraction and selection, in addition to data exporter to other tools like python and related artificial intelligence libraries such as TensorFlow, summarize covered areas can be seen in Figure 13.

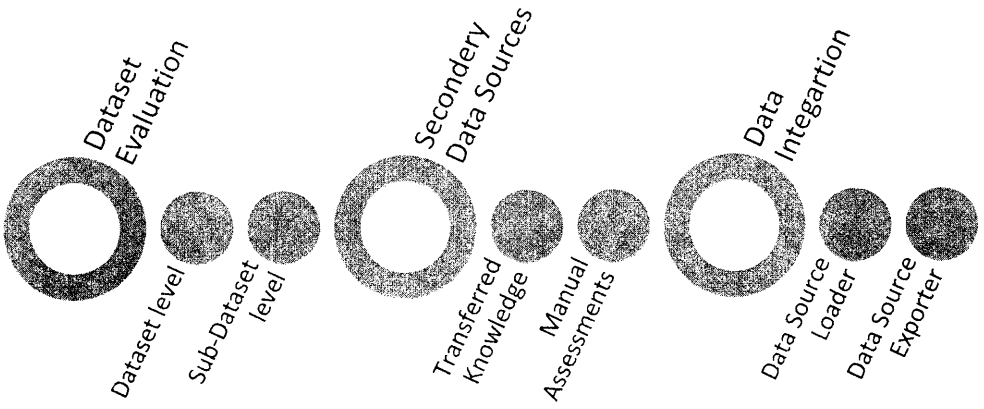


Figure 13: Data sources enhancements covered areas.

Our methodology starts from the earliest stage possible to evaluate available datasets on the area of interest. The methodology involves experiment design evaluations, data randomization, total experiment trials, labels complexity and data representation. These evaluation procedures will be discussed further in Chapter 5. In addition to evaluating selective emotional datasets, note that based on the existing datasets and the topic, different performance matrices may be selected; those highlights level of data compliant with the scientific research standards, the quality of the data, and simplicity of consuming those datasets.

Secondary Data Sources

Because of this, in this study, two additional secondary data sources from transferred knowledge of previous classification trials are used as a preference for tuning processing methodologies, depending on the subject's personality. The second transfer of knowledge data is performance feedback, which helps efficiently select between different signals, features, or processing methods. Another secondary data source is manual

assessment, which allows for the setting or modifying subjects' personalities on an individual basis; based on these 16 personality variables see Table **2**, an assessment can be performed using a well-known test devised by Cattell (Cattell & Mead, 2008). The exam is known as the 16PF Personality Questionnaire and is still widely used today, particularly in employment and marriage coaching.

This manual assessment can be used to initiate the solution settings, assist in categorizing global transfer knowledge, and improve the solution's overall performance. This information can be used to initiate the solution settings, assist in categorizing global transfer knowledge, and improve the solution's overall performance. In Figure 14, we illustrate the different types of data sources, depending on the nature of the data, which can be grouped into biometric data (EEG Signals), manual information (assessments), and transfer knowledge information. (Maksimenko et al., 2018)

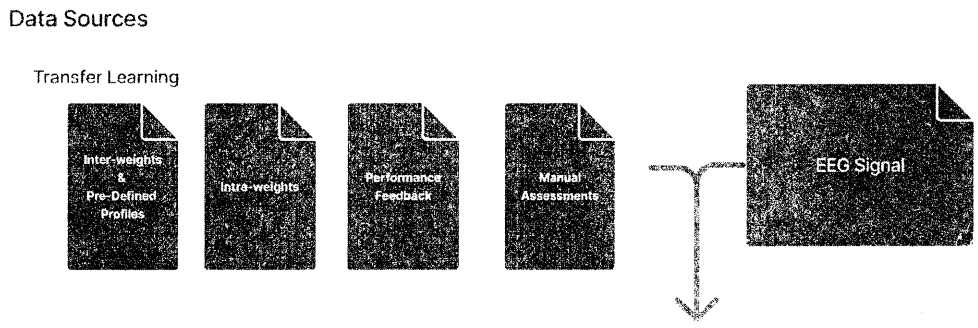
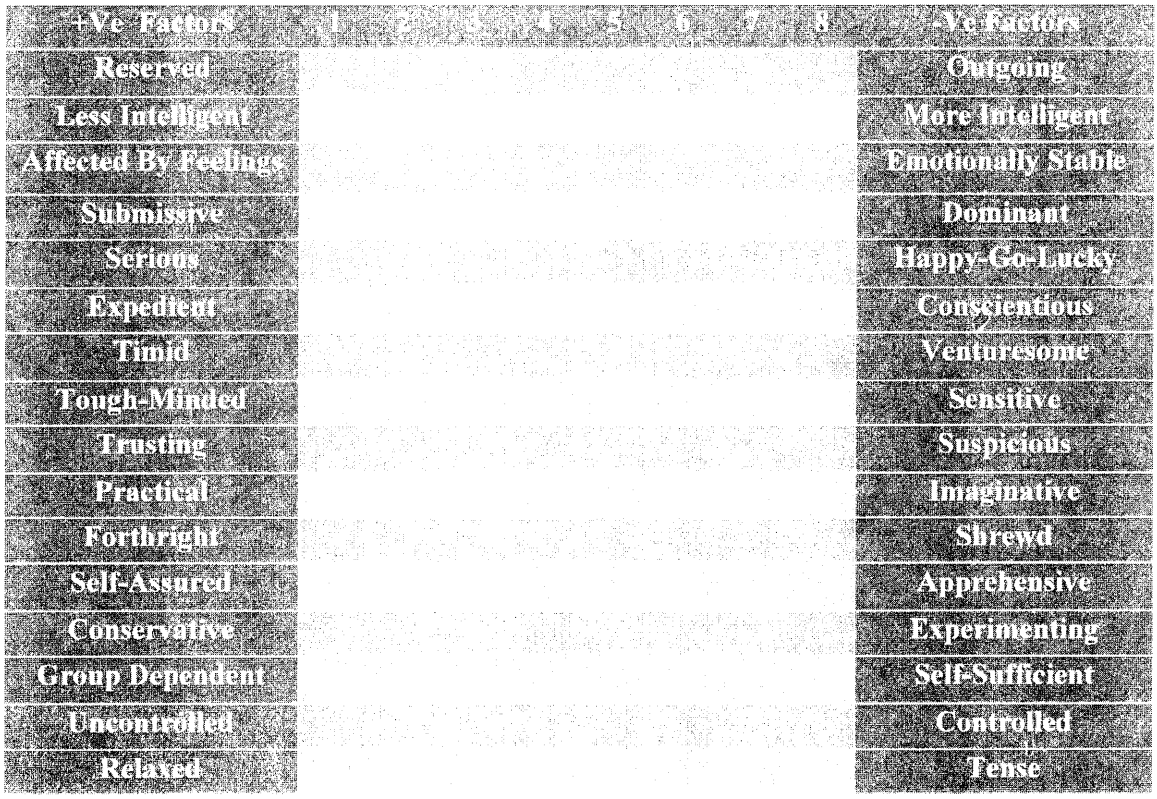


Figure 14: Recursive data sources, contains different transfer learning data sources.

Table 2: Cattell's sixteen factors of personality (16PF).



Adaptive Channel Selection

Signal preprocessing is an essential step that plays a significant role in improving the performance and accuracy of subsequent stages. We assessed the efficacy of various preprocessing methods and suggested automatically eliminating methodologies in order to increase the feasible classification success rates. We merged several methods, including the artefact contents check, signal filtering, signal resampling, signal segmentation, and numerous strategies for averaging signals together.

In this section, we will focus on enhancing the solution from the electrode or channel count perspective, directly reducing overall solution complexity; according to Mühl et al. (Mühl et al., 2014), a relatively small number of frontal lobe channels are all that is required for human emotion recognition, and this does not greatly hinder performance.

In fact, each area of the brain is usually responsible for different functions, so selecting appropriate brain regions for EEG signal acquisition is crucial. For instance, measuring across the parietal

and central-parietal regions is preferable to measuring over the occipital and frontal regions, as per et al. (Jenke et al., 2014).

Most researchers have selected various frequency bands and EEG electrode attachment locations. Some research, such as (Lin & Jung, 2014; Matiko et al., 2014), utilized a small number of channels. Even in (Quesada-Tabares et al., 2017) author used a single electrode as a signal source, but others, such as (H. Li et al., 2020; Ponz et al., 2014), utilizing up to 64 electrodes selected for signal capture. In addition, numerous study channels exceed 64 electrodes; for example, Deweese and his team used over 250 EEG channels for collecting signals (Deweese et al., 2014).

We came up with an adaptive channel selection method that is similar to selecting features, to minimize channel data transmission to the centralized unit. The system uses two ways to choose channels: filter and wrapper. When the system as a whole or a subject is in its first state, we use filtering methods like predefined or unsupervised selection methods, which don't require any previous trials or feeds. Based on previous studies, we set up a pool of default channels that will always be there. Predefined channels will be chosen later based on weight feed from transfer learning and from previous trailers. We suggest using different filter methods to find the top-N ranked channels based on local computation with one or more ranking methods, such as maximum variance and highest entropy. The selection method can be chosen based on the live computation limitations.

The other type of selection is called "wrapper selection." We used wrapper methods to find the minimum number of channels that met the required accuracy. This is done to avoid over-fitting, which can happen when cross-validation is used to measure the accuracy of predictions. We also used multiple filter methods to find the top-N ranked channels based on recursive performance ranking through transfer learning. Unlike other researchers, we used the ranking value to rank channels locally, but we used recursive learning to pick the right number of channels.

Adaptive Feature Extraction

A noteworthy amount of work has been done on examining various feature extraction methods. The output of EEG signal segmentation using a distinct subset of obtained from the segmentation process is represented by a collection of characteristics (features) that may contain unnecessary and repetitive features, rendering them unproductive for applications of machine learning. Selecting relevant features in addition to giving greater generalization also reduces the processing time to train the classification model.

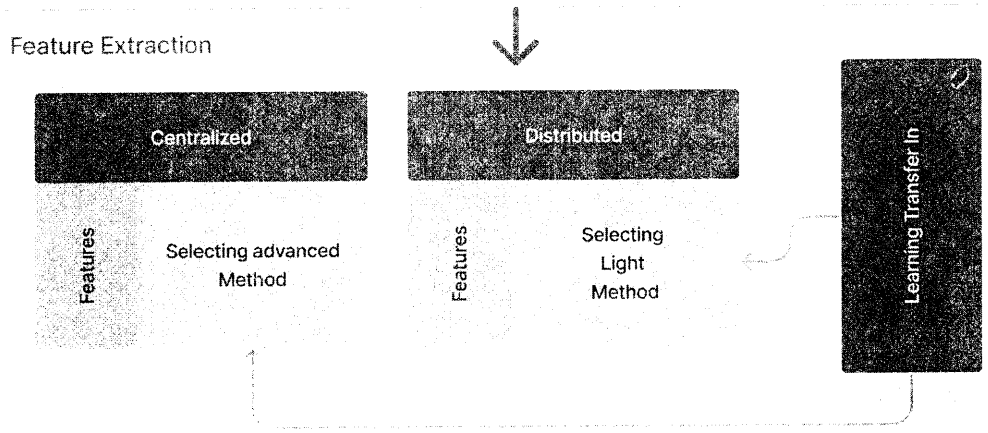


Figure 15: Two level of feature extraction, on distributed and centralized units.

We are reminded of real-time applications limiting reader resources such as communication, computation, and memory. Those resources can vary based on the real action. So, proposing an adaptive feature extraction methodology makes total sense. Our methodology can be adjusted based on the actual status to complete the feature extraction in two modes: a centralized unit or a distributed unit extraction. Each of those modes uses different techniques.

Since we focused on inter-subject transfer learning, we presumed that our approach is used to domains that are either identical or quite similar to one another. This is because we focus on inter-subject transfer learning. As a consequence of this, we are exempt from the obligation to carry out any kind of transferring component analysis, such as the Transfer Component Analysis (TCA).

As a result, we employed the tactic of transferring previously featured selected performances from the same subject or another subject that is equivalent, see Figure **15**. On the other hand, we can read the performance of feature subsets from various sources of the same topic or of another subject whose profile is comparable. In either scenario, however, we used weighted transfer learning, which indicates that each subset is influenced in a manner that is determined by the degree to which it has at least one minimal profile variation with the destination subject.

Adaptive Feature Selection and Reduction

We proposed adaptive filtering, wrapper, and embedding methodologies like feature selection based on computation mode. We employed the filtering approach with lightweight computing methods, such as maximum variance for feature extraction and PCA for feature reduction, in distributed computation mode. Similarly, wrapper and embedded approaches are utilized for distributed computation.

We came up with new advanced methods for feature selection for centralized systems, such as ReliefF (Q. Liu et al., 2021; J. Zhang et al., 2016), minimum redundancy maximum relevance (mRMR) (Baig et al., 2020; Vikram Shenoy & Vinod, 2014) and recursive feature elimination (RFE). We utilized PCA to reduce the number of features in all modes. In contrast, for the embedding approach, we recommend using a simple feature evaluation method for distributed computation, such as decision trees and SVM, and a more sophisticated method for centralized computation, such as Ensemble or RNN classifiers, see Figure **16**.

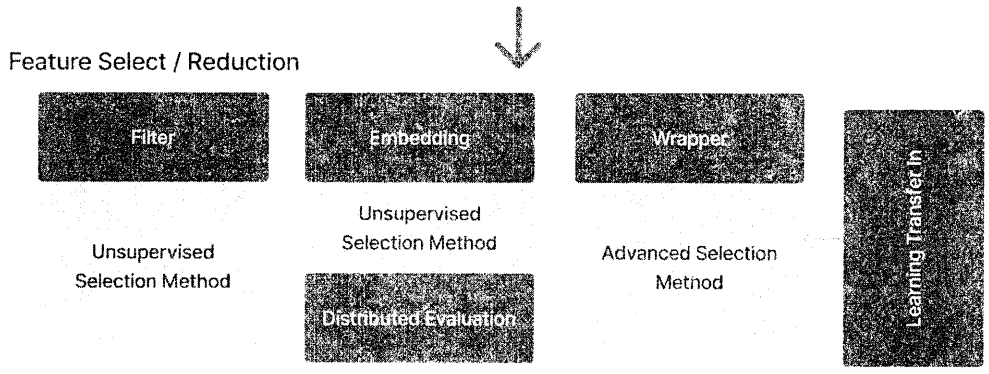


Figure 16: Multiple selection methodology can be choose based on system stability.

Enhanced Classification

The final stage of EEG signal processing work is signal classification, where the final observation occurs and based on that, decoding and corresponding action to be taken. Since this part is the part where most of the application takes place, usually, this work is shining and takes most of the attention. In the classification area, we put accuracy as the main goal; however, taking into consideration the real-time health application nature where the data size and processing time can play a critical role in considering practical solutions.

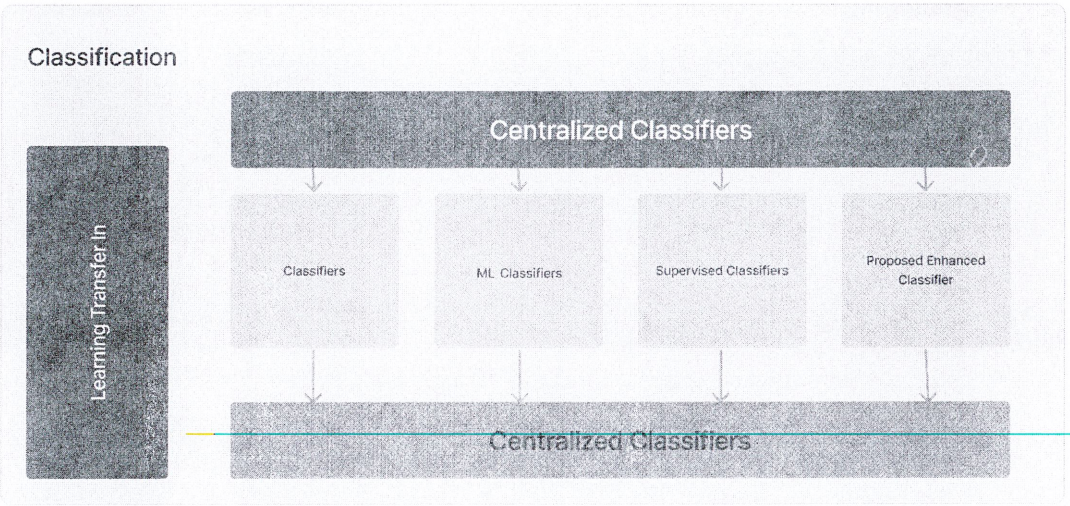


Figure 17: Classification method has a fixed recursive learning transfer data to be used for feature and channel selection actions.

As illustrated in Figure **17**, a new method for the classification of EEG signals is presented in this thesis. Besides all the enhancements implemented in earlier stages of this study, enhancing the final step is the most important part of the work. We present here how we combined recursively three state of art classification methods, the Convolutional Neural Network (CNN), Autoencoder (AE), and Recursive Neural Network (RNN), together.

The rationale for using the approach in the following paragraph is to leverage the strengths of each classification method and overcome their limitations, in Figure **18**. CNNs are good at extracting local features from EEG signals, but they may not capture the global temporal dependencies. AEs are good at

reducing the dimensionality of the input data and learning a compact representation, but they may lose some important information in the process. RNNs are good at modeling sequential data and capturing long-term dependencies, but they may suffer from vanishing or exploding gradients. By combining these methods recursively, we aim to achieve a robust and accurate classification of EEG signals that can handle different types of noise and variability.

In the suggested network, the Autoencoder was used for feature extraction, and logistic regression was used to predict emotional states. First, the features are converted by Short-time Fourier Transform (STFT) and spatiotemporal analysis from time series into 2D images and then encoded and decoded by an Autoencoder (AE) recursively. Then the data with reduced redundancy is used as the input features of the Long short-term memory (LSTM) for the classification task.

Compared to other approaches, the suggested classification model, which takes advantage of CNN's convolutional layers and AE's sparsity, exhibits good classification accuracy and fast convergence.

Both classification and clustering are usually used to evaluate EEG signals and extracted features. We have experimented with and compared different classification methods, such as the Decision tree classifier, the Naive Bayes approach, and the K-NN algorithm. We also used deep learning classifiers such as the CNN algorithm in addition to the proposed enhanced classification method.

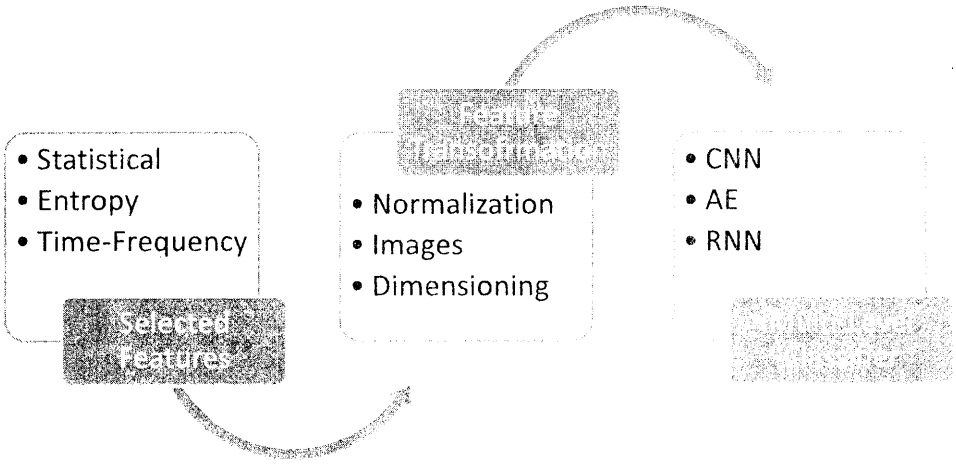


Figure 18: Proposed multi-stage recursive classification model, combines CNN-AE with RNN.

Real-Time Adaptive Decoding Algorithm

We were able to conduct trial-by-trial decoding of the emotion class by assigning a single emotion class to numerous trials within the same group. This was done on the basis of the idea that an emotion class can generate a collection of EEG patterns that are comparable to one another (Mattar & Al-Junaid, 2018).

Because the suggested solution as a whole is intended to address issues that are connected to real-time, the benefits of the comprehensive solution may be highlighted through the process of immediately making decisions and taking action, below figure explain the whole real-time decoding algorithm.

To make a decision as quickly as possible, all procedures should be carried out on the terminal side. However, because of the issue with limited resources, we were forced to go with a methodology that required fewer resources, but this typically resulted in a decision that needed to be more accurate when it was carried out in standalone mode.

In order to solve these problems, we came to the conclusion that we needed to modify the way decisions are made and operate according to both the center-centered and decenter-centered models, depending on the circumstances, see the flow in Figure **19**.

We are going to assume that the solution is already set with a minimum acceptable degree of accuracy or, to put it another way, an accuracy threshold based on the picture below. Following the conclusion of the analysis of the features, the terminal began the classification of emotions by defaulting to the lowest possible degree of accuracy. Yet, it was utterly dependent on the centralized decoding of the signal as well as the adjustment of the model weights.

By utilizing the high-precision centralized model in this way, we are able to keep a certain degree of accuracy while at the same time keeping terminal resources the same and boosting terminal time to live.

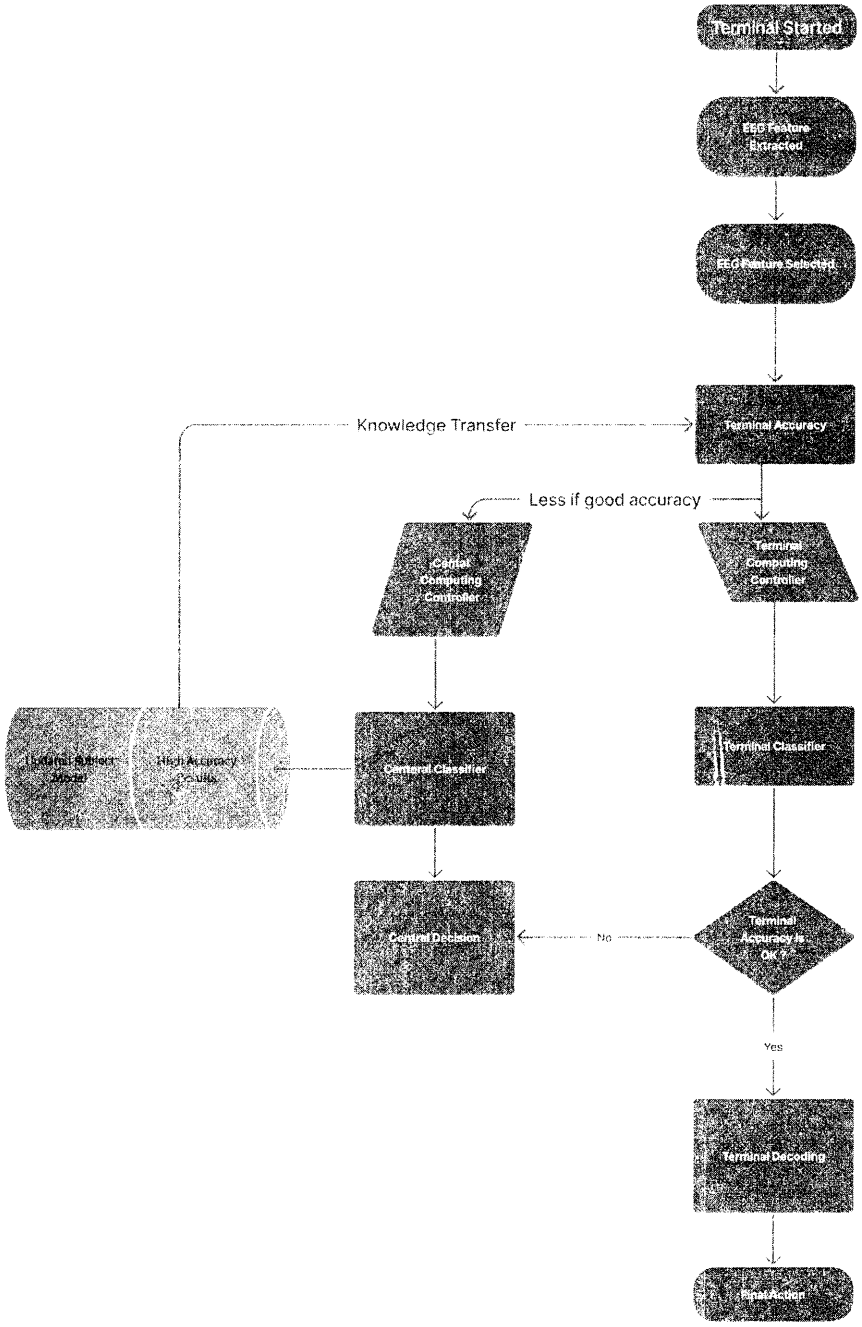


Figure 19: Terminal decode emotion instantly with the help of central classifier, based on the real-time adaptive decoding algorithm that is controlled by the terminal and defined accuracy acceptance level.

3.1.2. Datasets Acquisition

This research work on EEG signals is included in the SJTU Emotion EEG Dataset (SEED), which is accessible to the public. It was conducted by Professor Bao-Liang Lu, who also serves as the team leader and compiles the EEG datasets that are made available by the BCMI laboratory. When compared to other datasets, SEED has a competitive edge in terms of its reputation, the quantity of research and citations conducted, and even the possibility of repeating and improving the experiment (Lan et al., 2019) (Duan et al., 2013).

Finding datasets that are relevant to the issue will be quite difficult for any researcher interested in cross-subject emotion detection using Brain EEG. This is because the intricacy of the recognition process produces a huge cross-subject variance. Therefore, developing a strong framework to pick subjects, studies, and features is a game-changer. [Cause and effect] After doing more in-depth research on similar datasets, we've found that two of the most important datasets cover emotional EEG readings. These are the datasets for emotion analysis using physiological signals (DEAP) and the SJTU emotion EEG dataset (SEED) (X. Li et al., 2018).

SJTU Emotion EEG Dataset (SEED)

Subjects & Environment

The dataset consists of fifteen subjects, each of whom participated in three recording sessions over the course of one month, with a gap of two weeks in between each set of recordings. In order to create the necessary emotional states, each subject saw fifteen movie clips during each session. Five movie clips were assigned to each emotion: positive, neutral, and negative (Lan et al., 2017).

The research group presented the patient with visual stimulation while concurrently recording sixty-two-channeled EEG signals stimuli at a rate of one thousand hertz (Hz) as illustrated in Table 3: The SEED Dataset three categories for emotions triggering .

Measured Emotions

The author chose the movies such that there would be no emotional overlap between the targeted feelings; in this manner, they were able to simplify the label score detection to a single dimension, where even the video stimuli had a single state and were treated as a beginning and ending event.

Table 3: The SEED Dataset three categories for emotions triggering experiments.

Label	Experiment
Positive (1)	1,6,9,10,14
Neutral (0)	2,5,8,11,13
Negative (-1)	3,4,7,12,15

Dataset for Emotion Analysis Using Physiological Signals (DEAP)

Subjects & Environment

DEAP is a multidimensional dataset that was created for the purpose of evaluating the emotional states of humans. The electroencephalogram (EEG) and the physiological signals from the periphery of 32 subjects. During the experiment, each participant saw 41-minute segments of music videos in order to elicit an EEG response from their brain.

When we consider how complicated the experiment was, it is easy to see why the DEAP Dataset has such a significant flaw: it was only created once, in 2012, and has never been revised or updated since then. This is something that is completely understandable when we consider how complicated the experiment was.

Quantitative Emotions

In the DEAP Dataset, the researchers asked the subjects to rate themselves on five different emotions: arousal, dominance, liking, and familiarity. The researchers then compiled these ratings into labels. Labels with values ranging from one to five or from one to nine have been compiled into a collection.

The design of this experiment involves very high dimensions in terms of labels and scales, which increases the difficulty of both the procedure and the computing. In addition, the researchers introduced a great deal of complexity to the problem of emotional detection by enabling individuals to establish their own labels, which can result in bias in various social and private contexts (Koelstra et al., 2012).

SEED vs DEAP

The researchers of the SJTU emotion EEG dataset (SEED), who did three versions of their experiment between 2013 and 2019, simplified the experiment by defining only four opposite labels with zero (not applicable) or one (applicable) score and labelling the stimuli videos at the beginning. This was done seeing as we mentioned that the complexity of the DEAP and how it restricted the ability to improve or repeat the experiment. (W. L. Zheng et al., 2015)(W. L. Zheng & Lu, 2017)(W.-L. Zheng et al., 2019)

Table 4: Comparing DEAP and SEED, two important EEG emotion datasets.

	DEAP	**SEED**
Experiment Dates	2012	2013
Dataset Updates	-	2
Labels Dimension	5	2
Score Dimension	5 – 9	1
Score Defining	Subject	Researcher
Participants	32	15
Experiments	40	45

Based on the previous summary evaluation of the principal EEG emotion dataset (see Table 4), we have chosen to focus our efforts on the SJTU emotion EEG dataset (SEED), which will be the topic of our subsequent discussion in greater depth.

Enhancing Experiments Evaluation and Selection

Proper planning of the experiment design of an EEG Dataset is crucial to have trustworthy that is acceptable from the research community. It should cover and provide details related to the experiment designing, procedure details, environment setup, samples selection, and used stimuli and evoked emotions. For example, a few types of stimuli are usually used to induce emotions in their experiment participants, consisting of music, music videos, video clips, and virtual reality.

From a statistical point of view, a larger sample always leads to more randomness, better accuracy, and fewer errors; that is why bigger datasets will always be a better choice to work on it; however, bigger datasets mean more repetition and time, which may lead to overfitting or unbiased scenarios when the practitioner or the subject get familiar or tired.

We analyzed the execution of the experiments and used two factors to overcome the above problem, i.e., Experiments Count per Month, Experiments Count per day, Rest Period; we think those factors can impact the individual experiment quality. So, we picked the month with average experiments (see Figure 20 and Figure 21), avoided days with high experiment counts, and preferred days with a gap between 4 to 7 days.

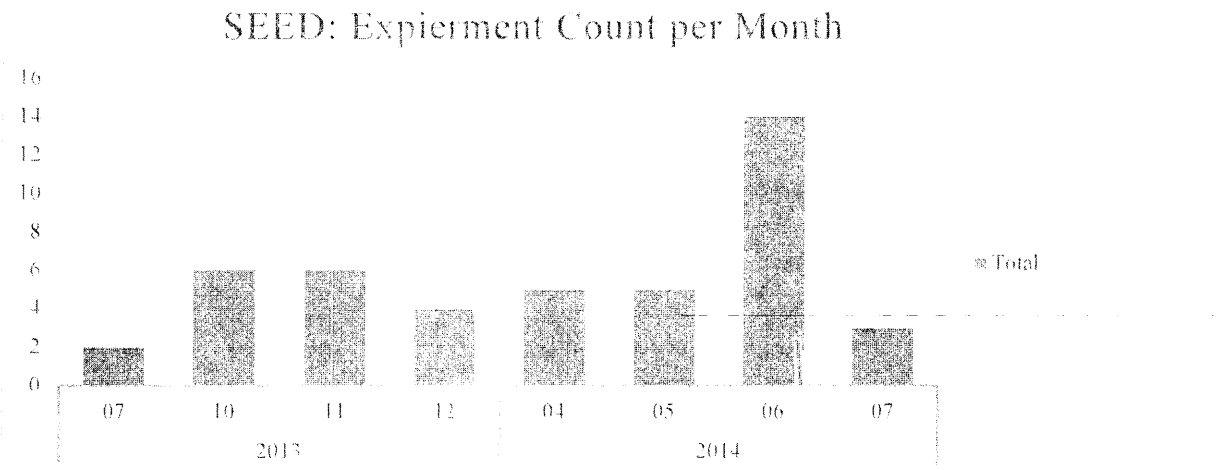


Figure 20: Analyze SEED dataset's experiments count per month, can impact practitioner performance.

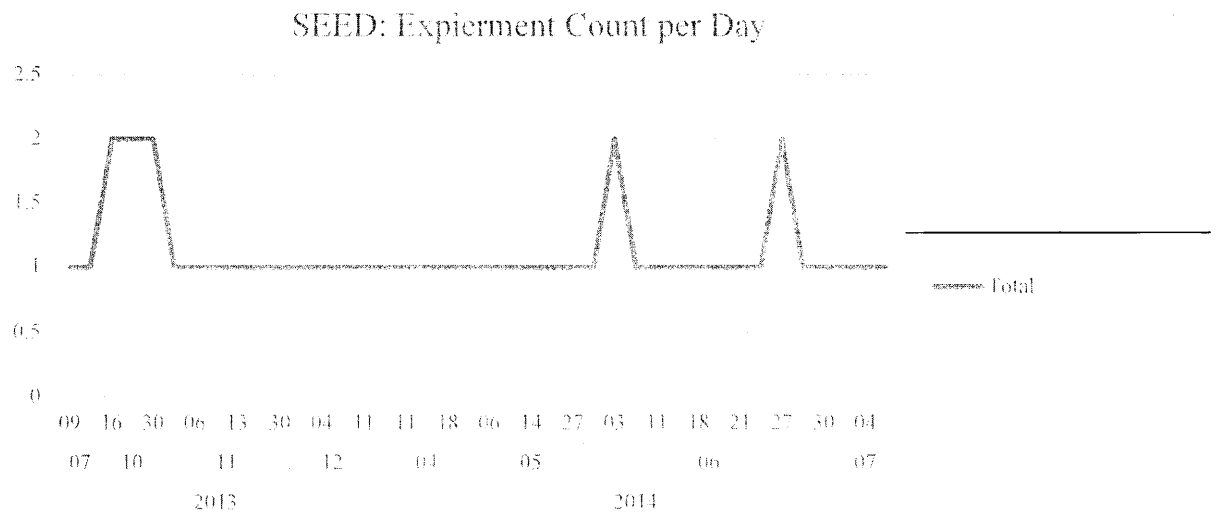


Figure 21: Analyze SEED dataset's experiments count per day give better look on workload.

Table 5: Sample of Dataset Gap Analysis of SEED Dataset, low and high gap can impact expermint performance.

Experiment Date	Subject	Subject Gap	Experiment Gap
7/9/2013	15	-	-
7/12/2013	6	-	3
10/16/2013	6	96	96
10/16/2013	15	99	0
10/27/2013	1	-	11
⋮	⋮	⋮	⋮
12/7/2013	12	6	3
12/11/2013	10	7	4
4/4/2014	2	-	114
4/11/2014	5	-	7
4/13/2014	2	9	2
4/18/2014	5	7	5
6/18/2014	11	-	3
⋮	⋮	⋮	⋮
6/21/2014	4	-	1
6/25/2014	11	7	4
7/2/2014	4	11	2
7/4/2014	9	7	2
7/5/2014	4	3	1

So as a final result general speaking about data set and experiment selecting we need to put in mind many factors, Data Score Calculation equation (1), where factors can be different base on the research topic, which can be represented in by the symbol f_x and each factor with impact weight represented by symbol w_x, n is total number of factors:

$$Data\ Score = \sum_{x=1}^{n} w_x * f_x \qquad (1)$$

We employed a combination of criteria in our example, including dataset updates, label simplicity, experiment workload, idle subject time, and experiment idle time. All of these aspects have a direct impact on data correctness, which indicates signal and data processing as well as overall classification generality.

Transfer Knowledge:

One of the most popular uses of transfer learning is the process of sharing a model for the purpose of assigning categories to problems that are similar. In the real world, it is not very typical to start from scratch when training a model for a dataset recognition job. Instead, researchers and data scientists prefer to start with a model that has already been pre-trained and is familiar with how to categorize different kinds of items. This makes it easier for the model to make accurate classifications.

However, the ultimate classification is only a tiny part of the problem that we are aiming to solve; in Figure 22, we can see the several forms that transfer learning can take depending on the stage complexity that we are attempting to optimize.

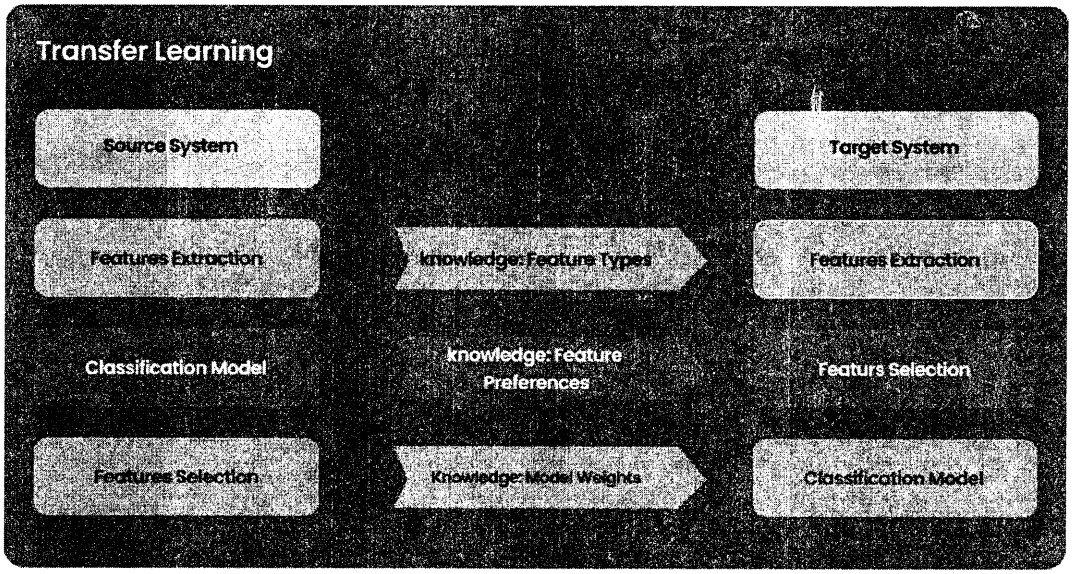


Figure 22: Overview of multi-layer transfer knowledge.

In the realm of signal processing, one of the subjects that is often considered as being very important is the extraction of features from EEG data. Each feature provides its own unique insight into the data, and the behavior of a signal can be described using the features that have been derived from the signal.

It is required, then, to extract features that are capable of accurately characterizing the behavior of signals in order to

increase the learning potential of machine learning models. This is because the behavior of signals is extremely complex.

If the features collected from the signal could be easily split into several categories and if the frontier between the classes was more evident, then the ML model could train more successfully. The basic goal of feature extraction is to discover more relevant information that is disguised inside enormous volumes of data. This may be accomplished by analyzing the data in a variety of ways. The feature extraction method helps significantly reduce the number of resources required for high-dimensional data analysis by reducing the quantity of data that needs to be processed, which in turn helps to reduce the overall amount of data.

One of the very first phases in the process of building a predictive model is feature selection, which refers to the practice of reducing the size of input parameters as much as possible.

It is desirable to restrict the number of variables that are input into the model in order to improve the performance of the model and, in some circumstances, to reduce the amount of computing effort required for modeling. Using feature selection, which entails choosing just the characteristics that are helpful for classification, may be one option for resolving this problem. Feature selection is a method that helps improve the performance of classifiers and minimizes the amount of time that is required for training.

This method works by removing features that are similar to one another. Techniques that are able to explore the whole search space in an efficient manner, such as evolutionary algorithms, are gaining more and more significance in the process of feature selection. This is one reason why these methods are becoming increasingly significant.

$$d_1(A,B) = \sum_{i=1}^{n}\sum_{j=1}^{n}|a_{ij} - b_{ij}| \tag{2}$$

$$d_2(A,B) = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n}(a_{ij} - b_{ij})^2} \tag{3}$$

$$d_{\infty}(A,B) = \max_{1\le i\le n} \max_{1\le j\le n} |a_{ij} - b_{ij}| \quad (4)$$

$$d_m(A,B) = max\{\|(A-B)x\| : x \in \mathbb{R}^n, \|x\| = 1\} \quad (5)$$

Equations (2), (3), (4), (5) shows calculating measure the variations between two domains or subjects, where first domain is A, second domain is B, and a_{ij}, b_{ij} *are the matrices of* A *and* B.

Establishing both the source domain and the target domain is the initial step in the process of developing our strategy for transfer learning. This will require some time and effort on our part. It is important to quantify the degree of similarity between signals and characteristics, as well as across people and between domains or labels.

Because the origin of all existing data sets conceals subject-level details for reasons of privacy, ethics, and the law, the great bulk of the work that has been done in the past has focused on commonalities between domains. We intend to compile a personal emotional profile for each participant. This profile will take into account aspects of the participant's social environment, such as their age, gender, family members, and social status, as well as aspects of the participant's personality analysis, which will primarily be derived from a personality factor questionnaire. The goal of this research is to disseminate information, and in order to do so, we anticipate compiling this profile (16pf) (Cattell & Mead, 2008; Maksimenko et al., 2018).

Therefore, in this research, we analyzed the same EEG data for the same subject with a gap of more than three months in the same experiments, and we observed a significant change in EEG signal between the two reads, as shown in figures Figure **23**, Figure **24** and Figure **25**.

This research was carried out by the same researchers who conducted the previous research. As a result of this, we will utilize the two trials as a comparable inter-subject read, with a personality profile that is generally similar to one another.

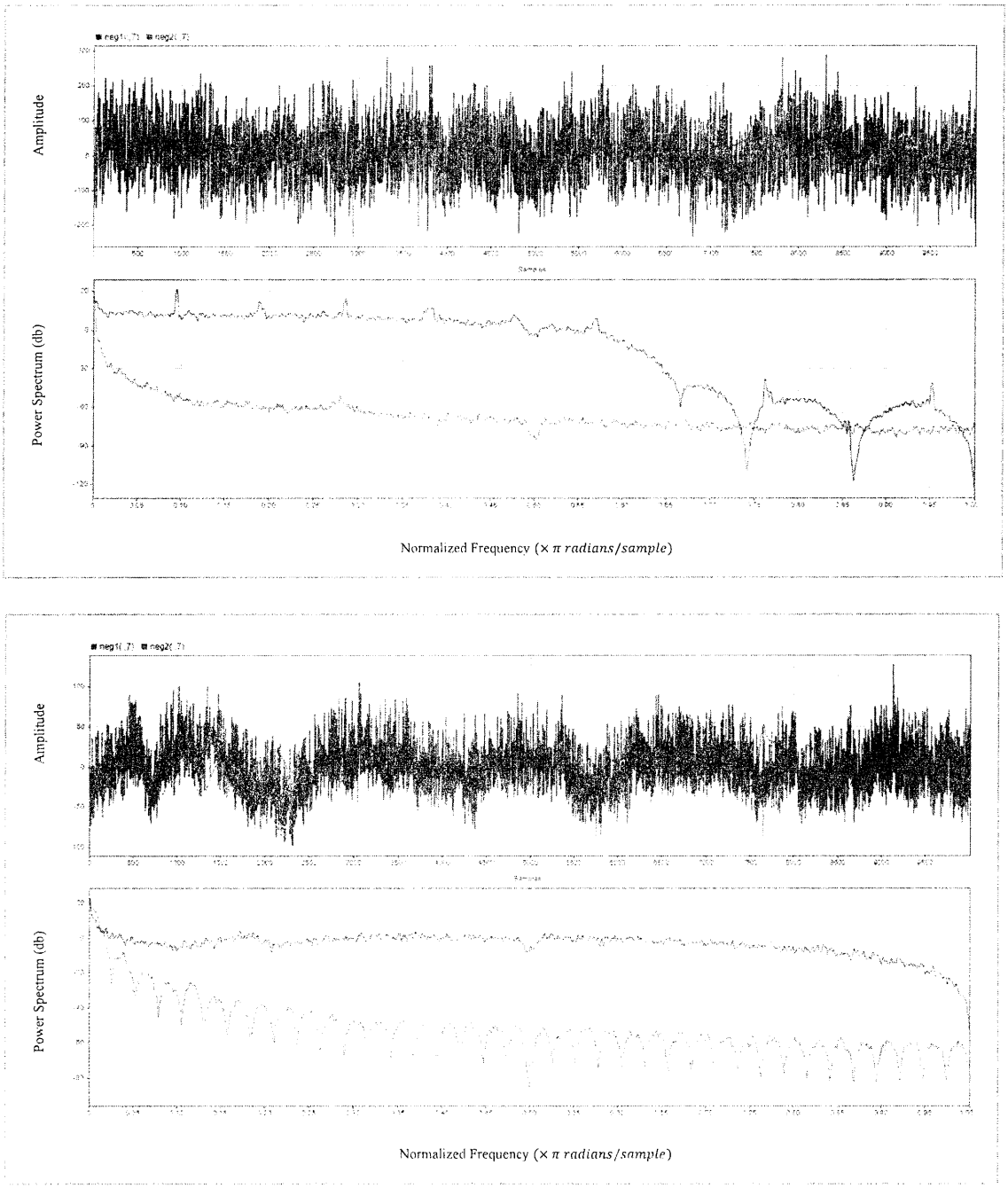
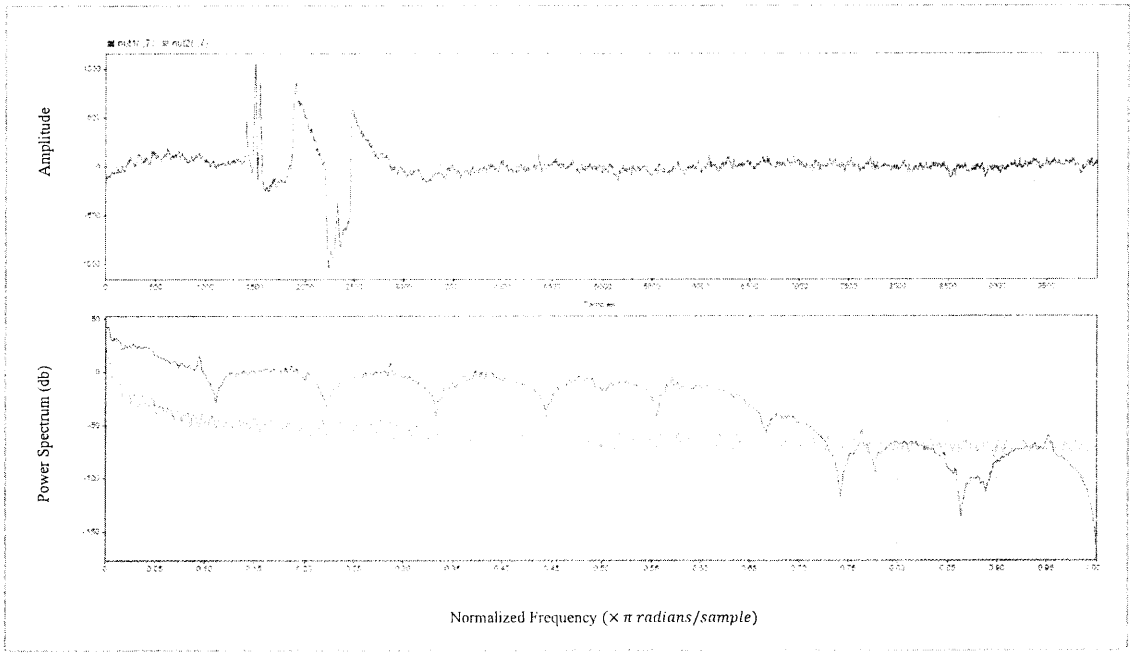


Figure 23: Signal analysis for the same subject where second experiment held after four months (negative emotion). The signal was sampled at 200 Hz.



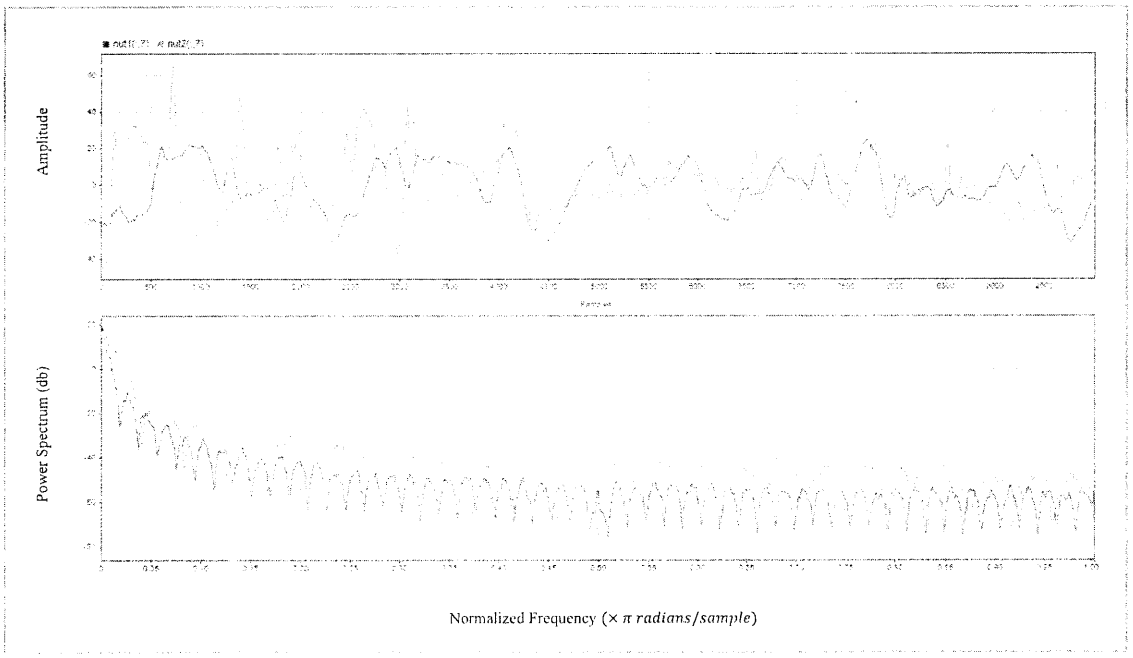
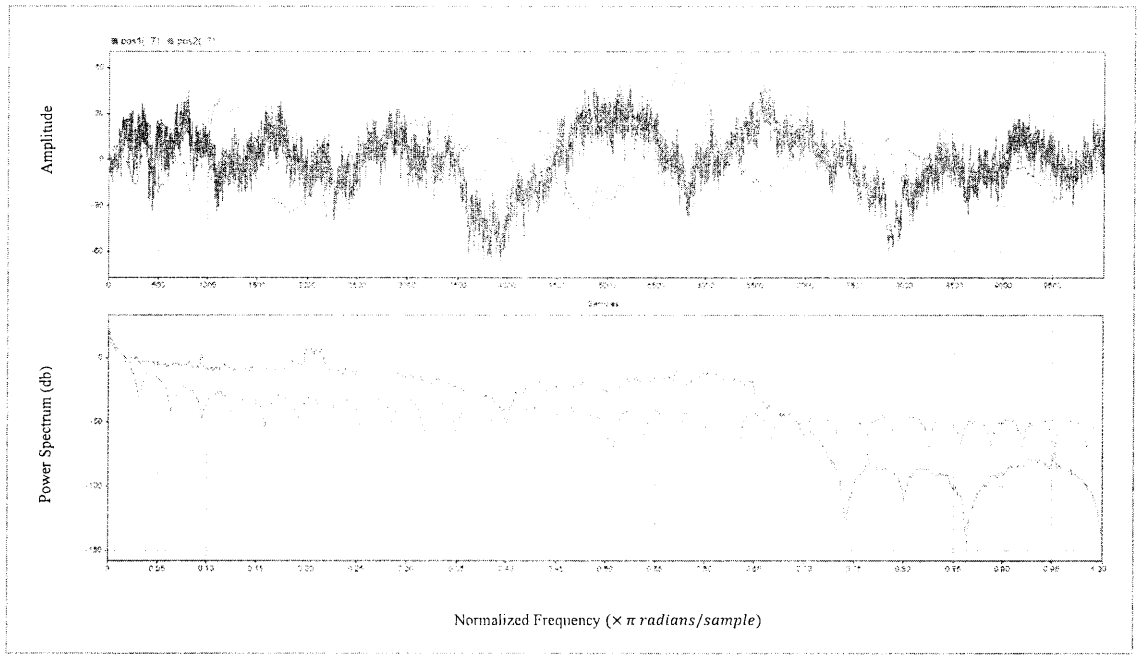


Figure 24: Signal analysis for the same subject where second experiment held after four months (neutral emotion).



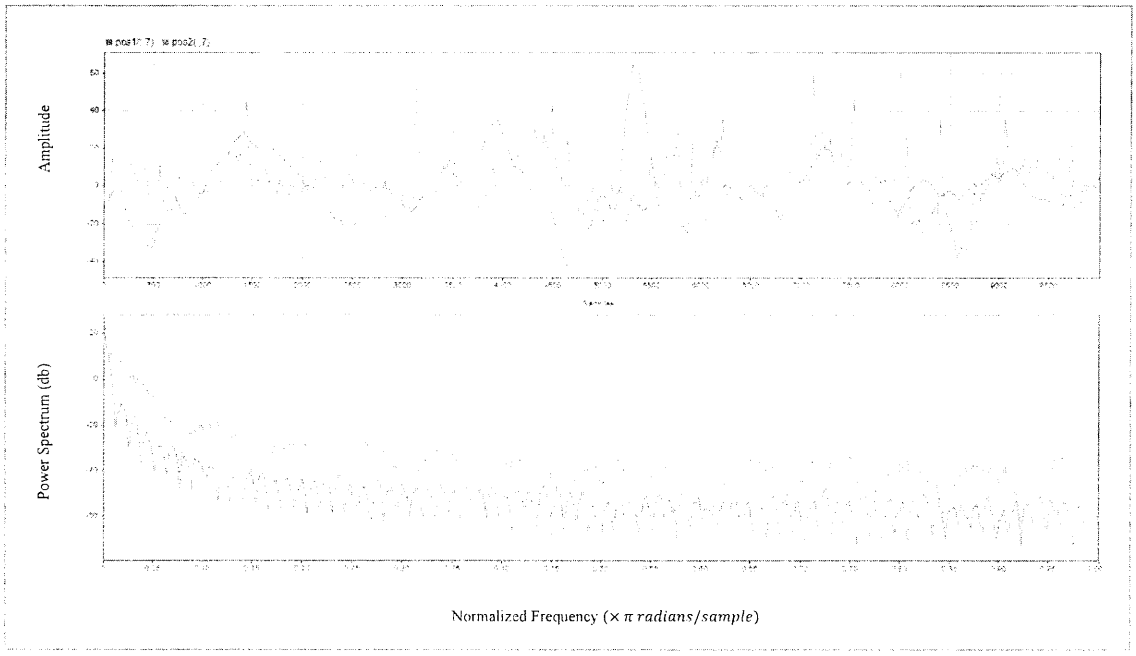


Figure 25: Signal analysis for the same subject where second experiment held after four months (positive emotion).

3.1.3. Signal Preprocessing:

The signals collected from the scalp usually do not accurately represent the original brain signals. Additionally, spatial information may be lost, and the Signal may contain much noise, which can shadow original brain signals.

Also, external events in the skull, such as blinking or other muscle movements, can generate Artifacts that corrupt the EEG signals; for all the above reasons, clarifying Signal is a compulsory process, which is usually named Signal preprocessing.

Generally, we perform preprocessing to transform raw data into a more valuable and usable new format that allows further processing and analysis—commonly in EEG datasets, preprocessing focuses on removing noise purpose to get back the actual brain-generated signals.

Defining the techniques and the level of preprocessing depends on the situation of the raw data on the matter of Sign to noise ratio (SNR), and even on the application nature, such as the resources available for each processing stage and the usage of the Signal is online or offline, and so on. There are several reasons why preprocessing is necessary for EEG data.

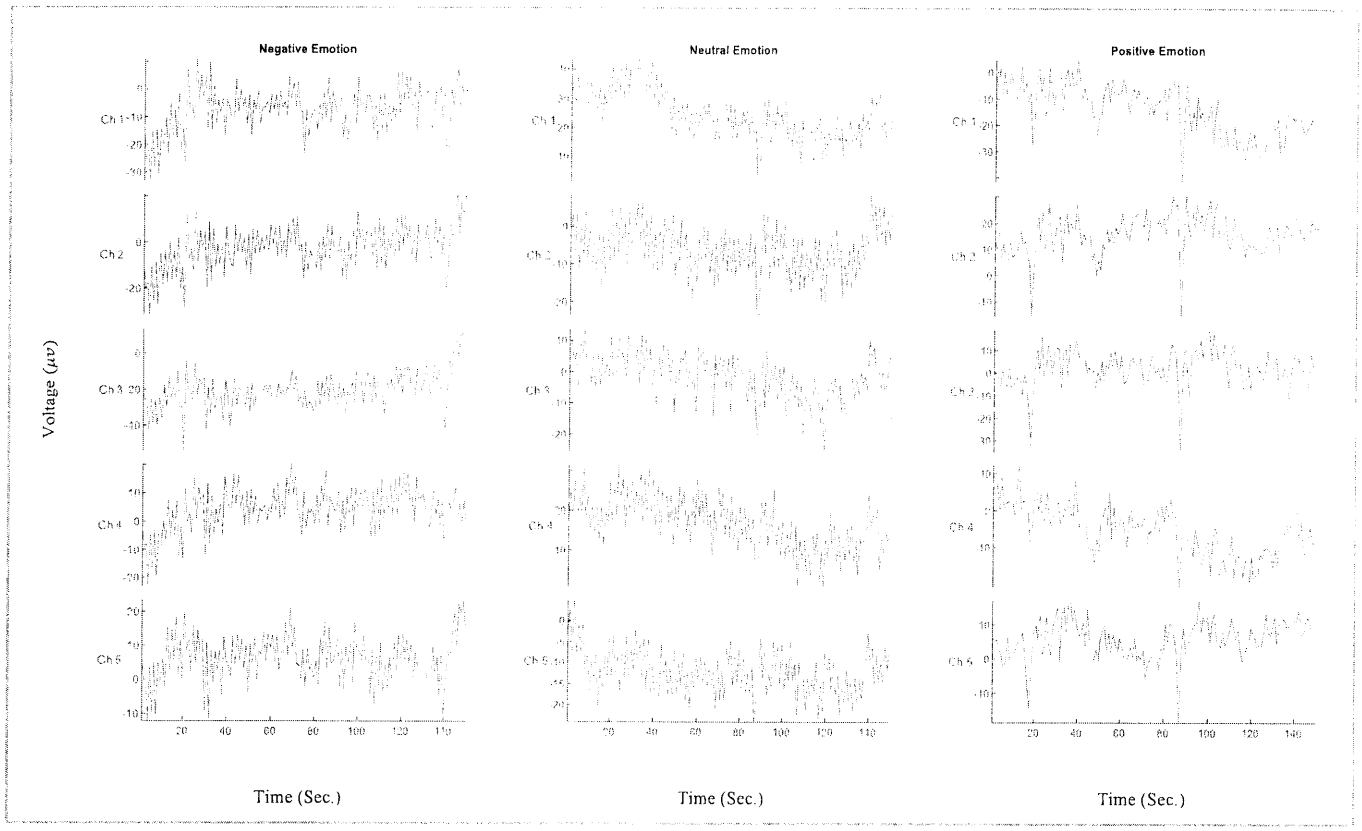


Figure 26: Raw EEG signals in the three different emotion states, used channels are fp1, fpz, fp2, af3, af4.

Figure 26 shows the five channels of each emotion class plotted on the power-time domain. As can be seen, there are little variances in feeling about the power value between neutral, positive, and negative status.

However, when we evaluate their power value, both negative and positive statuses are quite similar, making it difficult to discern which status the subject is in.

We discuss further methods for studying signals in the frequency-time domain using the MATLAB spectrum toolbox, which shows the frequency spectrum of time-domain signals. Figure 27 depicts the results of a better visualization to compare emotion classes. However, we encountered the same resemblance issue while displaying the spectrum of positive and negative classes.

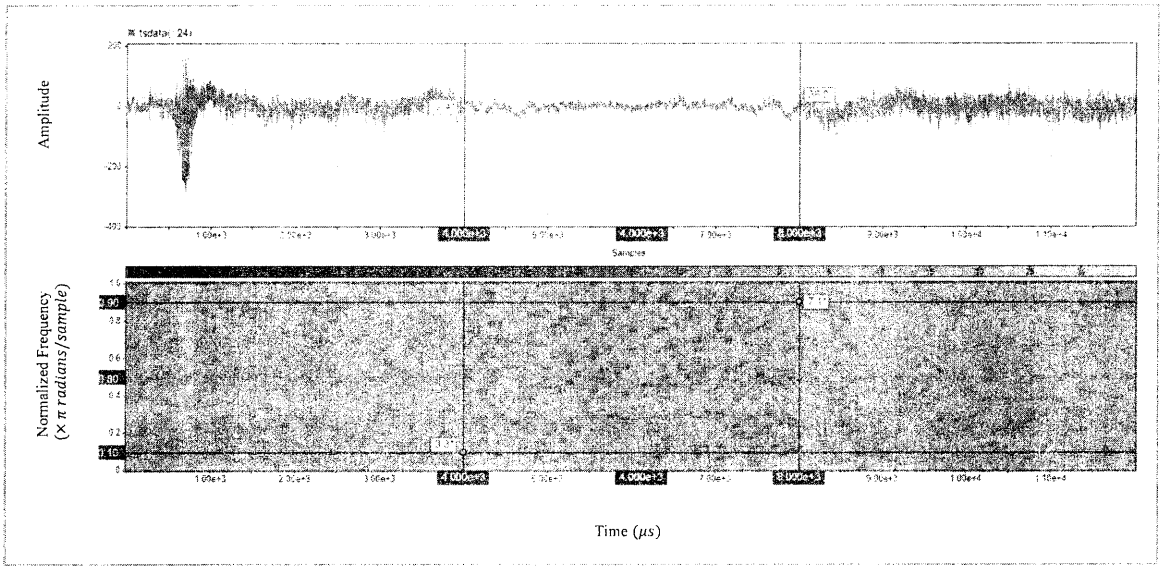


Figure 27: Spectral analysis with spectrogram for time-frequency of the raw EEG signals at the three states, (EEG signal from SEED dataset, with 200 Hz sampling rate).

In our selected dataset - SEED Dataset- the researchers provide the data on two sets, raw signal set and preprocessed signal set, researcher already preprocessed data to be filtered, resampled, and segmented, even though those processes can be applied over raw data and preprocessed one to preprocess it further.

Signal Filtering:

Band-pass Filter

When dealing with EEG signals, signal filtering is a common approach that is utilized. The elimination of frequencies can be accomplished by the use of well-known filtering methods such as low-pass, high-pass, band-pass, and notch filters. However, we deployed the same band-filter with a range from 0.1Hz to 65Hz using the *bandpass()* function in MATLAB, and it generated the desired signal output as shown in Figure 28. The creator of the SEED dataset preprocessed data applied band-filtering of the EEG signal from 0Hz to 75Hz.

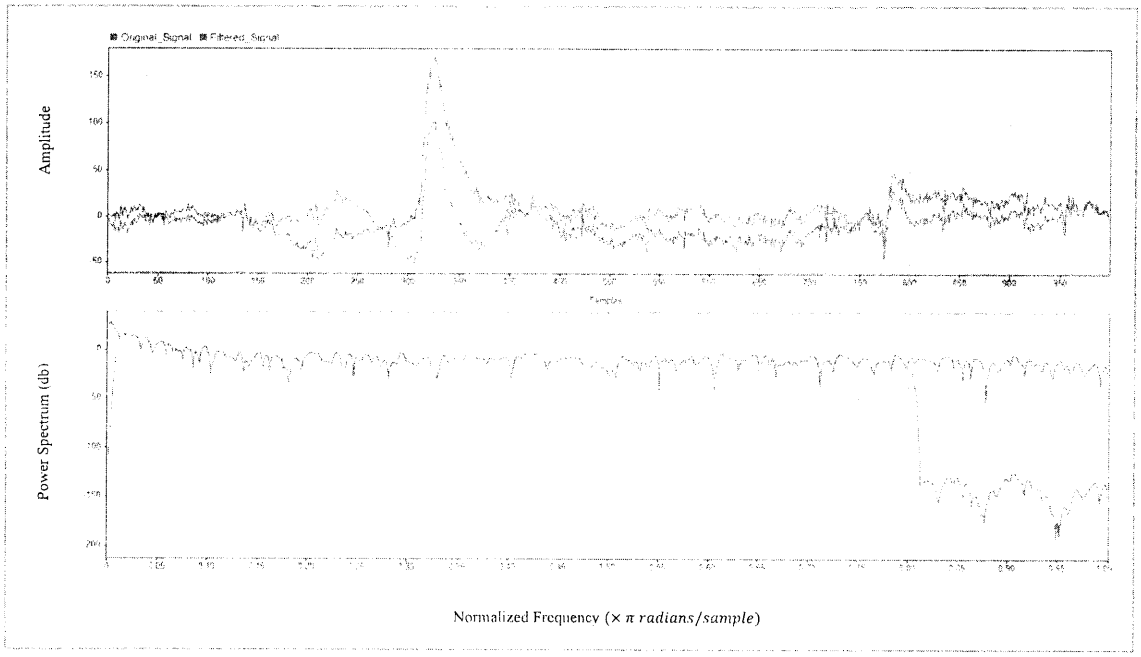


Figure 28: Before and after applying band-pass filter from 0.1 Hz to 65 Hz (EEG signal from SEED dataset, with 200 Hz sampling rate).

Notch Filter

Electrical interference from equipment that operates on voltages ranging from 110 to 230 volts AC is a typical source of environmental noise.

This alternating current (AC) power cycles at either 50 or 60 Hertz, which might show up on an EEG. This is especially true if the electrode is just loosely connected or if it is surrounded by a large number of wires and other electrical equipment.

It is possible to filter out this kind of artefact, which is referred to as "power-line noise," by employing a filter known as a "notch," which removes small frequency bands, such as 50 and 60 Hz, while leaving the majority of the spectrum nearly undistorted (see Figure 29).

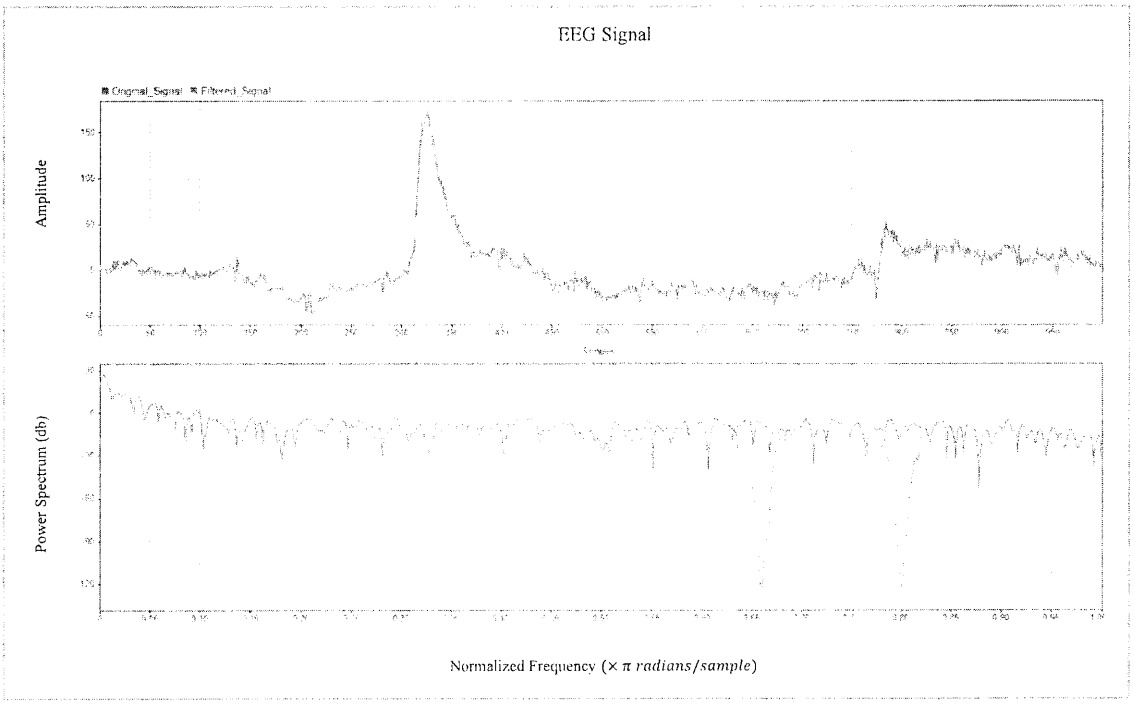


Figure 29: Before and after applying notch filter on bands 50Hz and 60Hz (EEG signal from SEED dataset). The signal was sampled at 200 Hz.

Signal Sampling

A few different names for the method may be used to reduce the number of samples taken from a signal while maintaining the essential information. These methods are referred to as signal sampling, resampling, or down-sampling.

When discussing new applications that are based on edge-computing and cloud-computing, as well as, of course, data streaming, the large amount of data can load transmitting infrastructures such as battery and media throughput and consume processing resources such as memory and processing units, both of which are very expensive for wearable and mobile technologies.

So, we take the case of SEED Dataset as an example to calculate the amount of traffic originally generated, with a 1000 Hz sample rate and 62 channels. If we are representing each sample as a 32-bit float, this is (62 * 1000 * 32), which generates traffic about 2mbps, and 21GB of data on a daily basis.

For this reason, we resampled the EEG to smaller frequencies using MATLAB Signal Processing resample function resample(), Figure 30 shows the data saving in shape of band size before and after applying the resample function.

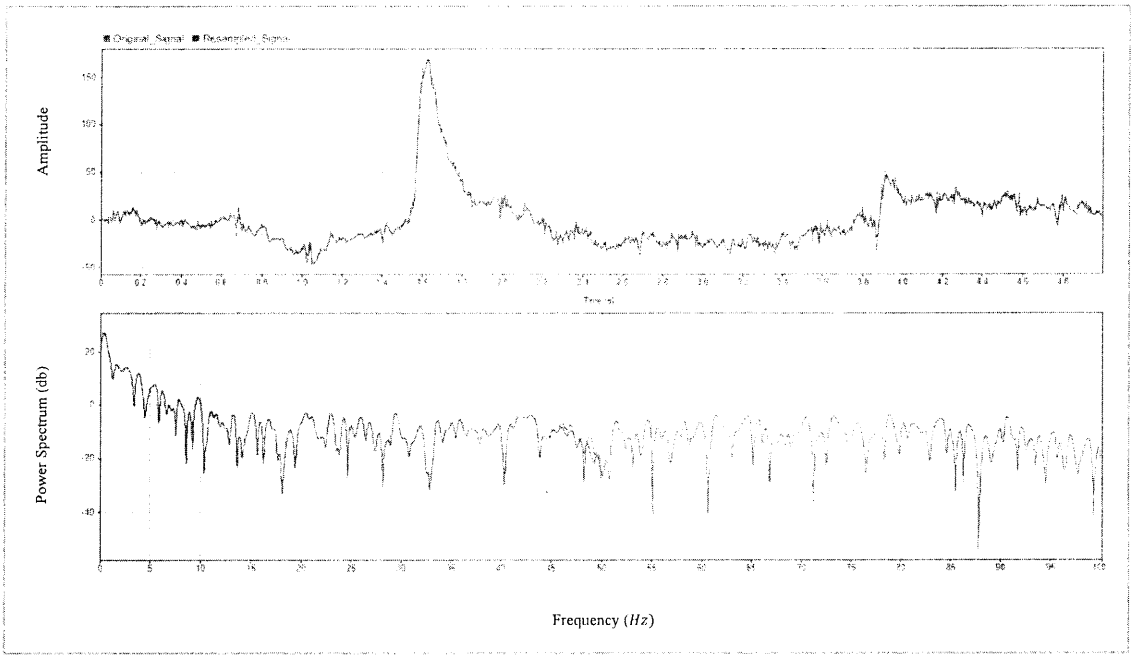


Figure 30: Before and after applying signal down-sampling by half from 200Hz to 100Hz using MATLAB resample() function, note that signal maintain the general characteristics (EEG signal from SEED dataset, with sampling rate 200 Hz).

Signal Segmentation

EEG is known as a non-stationary and nonlinear signal. Suppose we divided EEG signals into segments; in this case, we can consider new segments of the signals as stationary.

Constant Window Segmentation:

The simplest way of doing that is by subdividing the signal into equal segments on the time frame. On the grounds of experimental remarks, the optimum segment durations are 10 to 30 seconds. We tried on our research with segment lengths from approximately 1 second to tens of seconds. The best results were obtained with segment length of 1000 to 5000 bit which is 5 to 20 seconds, according to the type of analyzed EEG signal, see Figure 31.

$$mean([x_m \dots x_n]) = \frac{1}{n-m+1}\sum_{r=m}^{n} x_r \quad (6)$$

In equation (6), x_m to x_n are the numbers in the sequence that we want to find the mean of. The variables n and m are the term numbers of the first and last terms in the sequence, respectively. The variable x_r represents any term in the sequence between x_m and x_n.

For Signal x_1 *to* x_n, find change points J where k that is smallest using equation (7).

$$J = \sum_{i=1}^{m-1}\left(x_i - mean([x_1 \dots x_{m-1}])\right)^2 + \sum_{i=1}^{n}\left(x_i - mean([x_1 \dots x_n])\right)^2 \quad (7)$$

Substituting mean equation (6) to equation (7) give us the final equation form as in equation (8).

$$J = \sum_{i=1}^{m-1}\left(x_i - \frac{1}{m-1}\sum_{r=1}^{m-1} x_r\right)^2 + \sum_{i=1}^{n}\left(x_i - \frac{1}{n}\sum_{r=1}^{n} x_r\right)^2 \quad (8)$$

This technique is already implemented in MATLAB as part of the Signal Processing Toolbox, a function called findchangepts(), in below Figure 32 we can see the verity of window based on signal characteristics.

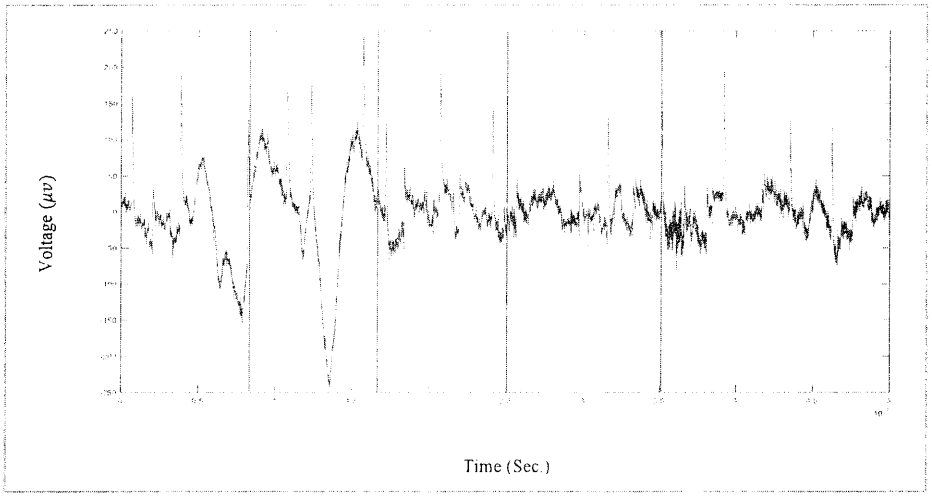


Figure 32: Change point detection segments with variable window sizes, applied on 12000 sample of 60 second duration (EEG signal from SEED dataset, sampled at 200 Hz).

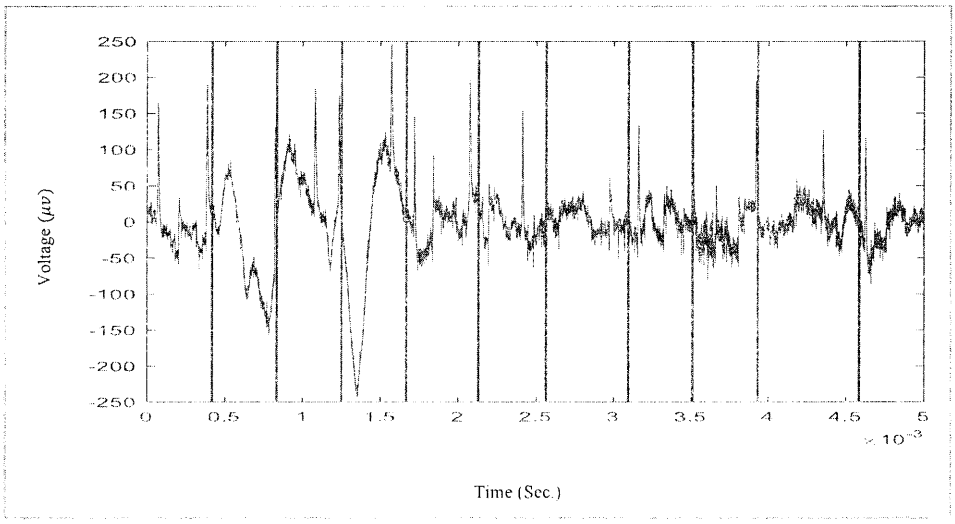


Figure 33: Increasing change point detection sensitivity by reduce minimum window size (1000) and increase allowed number of windows (10) with variable window sizes, applied on 12000 sample of 60 second duration (EEG signal from SEED dataset, sampled at 200 Hz).

Artifact Removal

Artifacts are noisy signals that occur when recording EEG signals because of many physiological factors out of interest. For instance, eye movements, eye blinks, cardiac activity, and other muscular activities on the upper part of the body. Artifacts may cause a faulty reading and lead to a misanalysis of the signals. False handling of those artifacts leads to a major unsuitable use of medications or treatments.

Since noise sources could be widely distinct with complex traits, studies usually eliminate a single type of artifact. Artifact processing techniques range from rejection to eliminating. Rejecting Filter is designed to identify and reject specific artifacts by frequency band, such as Notch filter, or by eliminating impacted segments with poor quality, like the dynamic segmentation technique discussed in the previous section. Although rejection is still an option for segments containing extreme noise, it is often useful to keep as we can.

In offline processing, algorithms such as the average Change Point Detection are typically utilized. In this type of processing, a batch of the signal can be processed together, and the method incorporates a simultaneously measured EEG to channels. The algorithm first carries out local profiling, then subdivides the signal into portions around each change, and finally averages the variable sections to approximate the artifact. For example, The SEED EEG dataset removes artifacts using ICA, spatial filtering and time-domain filtering, which separate signals by statistical, spatial and temporal criteria respectively.

Proposed Channel Selection:

We proposed an adaptive channel selection technique similar to feature selection; the system uses two different channel selection methods, mainly filter and wrapper. In case the system overall or a subject is in an initial state, we use filter methods such as predefined or unsupervised selection methods where no previous trials or feeds are required. We defined a predefined pool of channels based on previous studies, which also permanently exist as default channels; predefined channels will be selected later based on weight feed by transfer learning and from previous trailers.

We propose to use various filter methods to find the top-N ranked channels based on local computation with one or more of the ranking methods, such as maximum variance (Alotaiby et al., 2015) and highest entropy (Q. Liu et al., 2021); the selection method can be chosen based on the live computation limitation.

The other selection type is wrapper selection (Baig et al., 2020). We used wrapper methods to identify the minimum count of channels that met the required accuracy; this mechanism is used to avoid over-fitting due to the use of cross-validation measures of predictive accuracy. We propose to use multiple filter methods to find the top-N ranked channels based on recursive performance ranking through transfer learning; unlike other researchers, the ranking value will be used to rank channels locally however, picking the right number of channels will be done recursively.

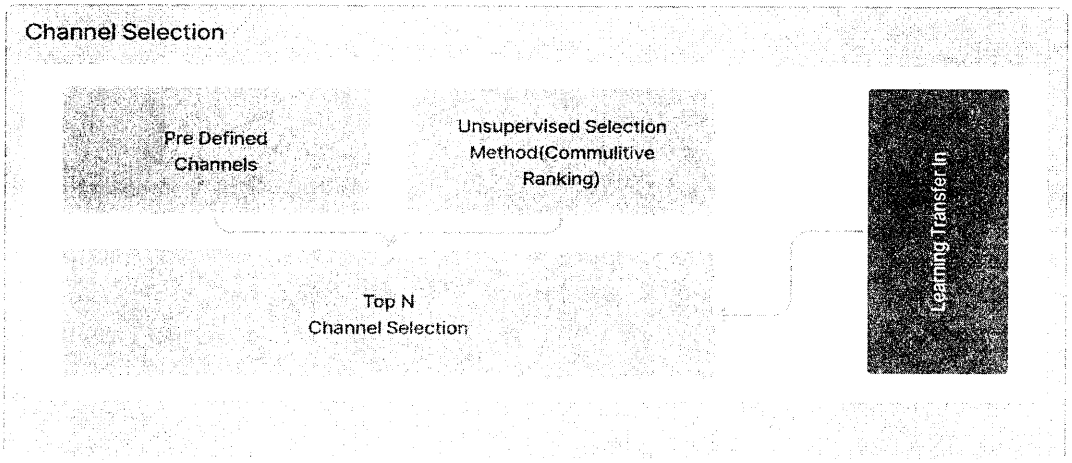


Figure 34: Adaptive channel selection techniques using learning transfer weights.

We added an inter-subject channel selection dimension, in which the selected feature will be different from one subject to another in the same domain; however, we got a new challenge for the new subject comer, and to overcome this issue, the new subject will use the feature set of the closest subject, which will be based on the minimum varieties score calculation that was mentioned earlier in 1-4.

Algorithm 1: EEG channel selection algorithm

Input: Raw EEG Signal to be collected from multiple experiment preictal segments, the global matching profile channel subset

Output: array of reduced channels for each patient that give the same accuracy

Initialization: m = 32 (initial number of channels), selected = all

```
if new subject
    | channels ← matching profile channel
Else
    | Previous Subset ← discriminant by feature selection method
    | for ch ← 1 to 32 do
    |     | Accuracy ← Apply LDA
    | end
    | channels ← Top Accuracy channels
End
return channel subset
```

3.1.4. Features Evaluation and Extraction:

Once signal processing concluded, we came closer to the purpose of this whole analysis journey to classifying or clustering then conclude the state of the processed signal. This method begins with feature extraction, which is a key step to optimize EEG data processing.

Feature Extraction attempts to decrease the number of features in a dataset by producing new features from the current ones and then eliminating the initial features. These new minimized features should then describe most of the information provided in the original features. A summed-up version of the initial features may be constructed by combining the original set in this method.

Another often-used strategy to decrease the number of features in a dataset is Feature Selection. While Feature Extraction focuses on generating new features, Feature selection seeks to rank the importance of the existing characteristics in the dataset and eliminate less important ones.

Feature Extraction

Extracting features are useful in the analysis of an EEG signal, which is regarded as a large and continuous data collection that must be minimized, and redundant data removed from the data set. Each fragment of an EEG signal can be analyzed using either a single generated feature or a combination of generated features. This stage of the EEG processing process is important because it distinguishes between different classes. It has such a direct impact on the overall classification. The subsections that follow provide a brief summary of developed features.

1. Statistical Features

Vital features from the raw EEG data must be obtained for training and testing in the emotion classification process. The EEG data is utilized to create nine statistical features in this work (min, max, mean, mode, median, standard deviation, difference of means, skewness, and kurtosis). The distribution of the amplitude and its moments can be utilized to define an EEG signal. Let the signal recorded from the EEG be indicated by x, and x_N denotes the value of the nth sample of the raw signal, where 200 Hz signal $N = 1000$ correspond to five seconds of EEG recording.

Mean Feature:

Let's say there are n observations in a data collection, and those observations are the numbers like $X(i)$. Calculating the average or mean of the collection of data may be done as follows in (9).

$$\mu_x = \frac{1}{n}\sum_{i=1}^{n} X(i) \qquad (9)$$

Mean of Standard Deviation Feature:

Suppose we express the data set's mean as in (9). As a result, the mean of the data's standard deviation is: According to this definition, the average of the population S of the standard deviation of the population n is the number of observations in each sample, x_i is the *i-th* observation in the *x-th* sample, and μ is

the mean of all observations, and MSD is determined in this manner.

$$MSD = \frac{1}{S}\sum_{x=1}^{n}\sqrt{\frac{\sum(x_i - \mu)^2}{n}} \quad (10)$$

Means of abs. 2nd differences Feature:

In this feature, instead of getting the difference between every two samples, we are leaving a gap sample, so at the end, we will end up with $n-2$ samples, and the mean value will the sum of the absolute values is divided by $n-2$ as in (11), where n is the number of observations in the data set, and X_i is the *i-th* observation. The γ_x is calculated by taking the absolute value of the difference between each observation and the one two steps ahead, and then averaging these values over all possible pairs.

$$\gamma_x = \frac{1}{n-2}\sum_{i=1}^{n-2}|X_{i+2} - X_i| \quad (11)$$

3rd-order and 4th-order moments Features (Skewness and Kurtosis):

To solve the variation in time and power, we are using a high level of moment calculation such as the kurtosis and skewness over a sliding time window because kurtosis is a measure of the 'spikiness' (or outliers) equation represented in (12) while skewness is a measure of the asymmetry of signals, where n is the number of observations in the data set, x_i is the *i-th* observation, $\bar{x}$ is the mean of all observations, and σ_x is the standard deviation of all observations (Xiang et al., 2020).

$$S = \frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^3}{\sigma_x^3} \quad (12)$$

$$k = \frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^4}{\sigma_x^4} \quad (13)$$

2. Entropy-based Features:

German mathematician Rudolf Clausius introduced the concept of entropy to the area of thermodynamics. In 1948, mathematician and electrical engineer Claude E. Shannon released A paper on Mathematical Theory, in which he addressed issues related to the quantification and uncertainty of information.

Entropy is a statistical measure of the "amount of information" in a given state. The amount of uncertainty, which is measured by the number of possible outcomes, also influences the entropy of a variable. Other Entropy methods such as Approximate (ApEn), Sample (SampEn), and Shannon wavelet (SWE) have all been proposed and used to quantify DoA (SE).

Shanoon Entropy Feature:

Shannon entropy and approximate entropy (ApEn) are suitable parameters to characterize the EEGs in this investigation since their value lowers abruptly caused by the extreme synchronous discharge of neurons in the brain during epilepsy activity.

Using the discrete wavelet transform approach. EEG data are divided into sub-bands, these sub-bands are used to construct non-linear characteristics such as ApEn and Shannon entropy, and classification is accomplished with the use of support vector machine methods. Because of this, we decided to employ Shannon Entropy (SE) as a method for feature extraction and compared its effectiveness with that of other widely used approaches (Phung et al., n.d.).

The Shannon entropy of a discrete random variable X is denoted by $h(X)$ and measures the average amount of information or uncertainty that X conveys. It is calculated as the negative sum of the products of the probabilities $P(x_i)$ and the logarithms of the probabilities $P(x_i)$, where x_i are the possible outcomes of X, n is the number of outcomes, and b is the base of the logarithm.

$$h(X) = -\sum_{i=1}^{n} P(x_i) \log_b P(x_i) \qquad (14)$$

Differential Entropy Feature:

Researchers often utilize differential entropy as their go-to method for measuring the complexity of a continuous random variable. Differential entropy distinguishes itself from other entropies of managing continuous random variability. A connection can also be made between differential entropy and minimal description length.

The mathematical formula for its calculations can be represented in (6) and (7), where $h(X)$ is the differential entropy of a continuous random variable X with mean μ and standard deviation σ.

The differential entropy can be calculated by integrating the product of the probability density function of X and the logarithm of its reciprocal over the entire real line. Alternatively, it can be simplified as in (7) where 2π is a constant that represents the circumference of a circle divided by its diameter. (D. W. Chen et al., 2019; W. L. Zheng et al., 2014).

$$h(X) = \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \log\left(\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}\right) dx \quad (15)$$

$$h(X) = \frac{1}{2}\log(2\pi\sigma^2) \quad (16)$$

3. Time-Frequency Features

The spectrum differences for the various emotional states can be seen with the naked eye, and they are representative of all 62 channels. Keeping in mind that we can improve these results by decreasing the number of channels using ICA or PCA dimension reduction techniques, as well as by smoothing the signal, as shown in Figure 36, we are able to see the spectrum differences.

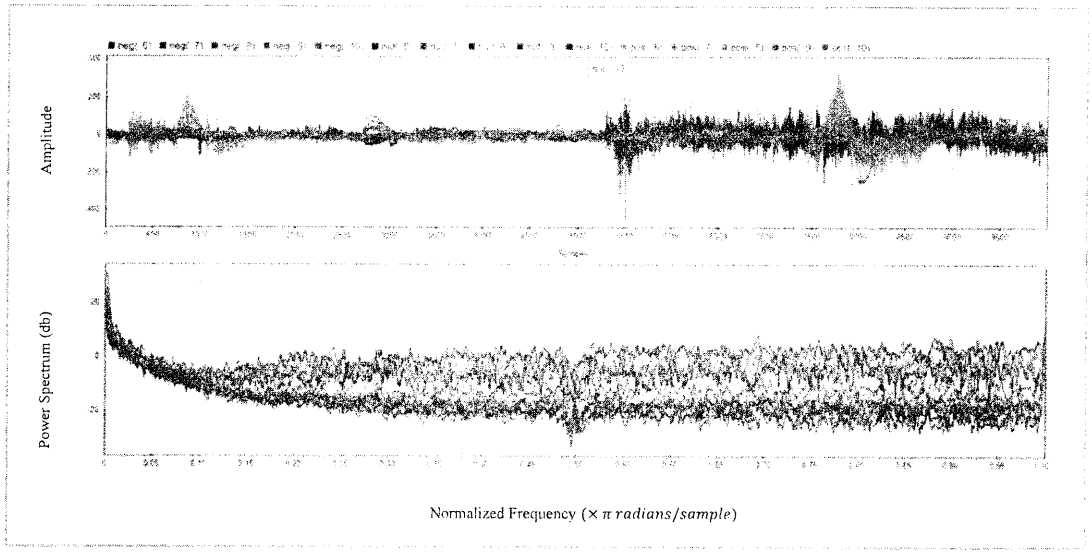


Figure 35: The outcome of doing signal analysis on all 62 channels across the spectrum domain and time domain.

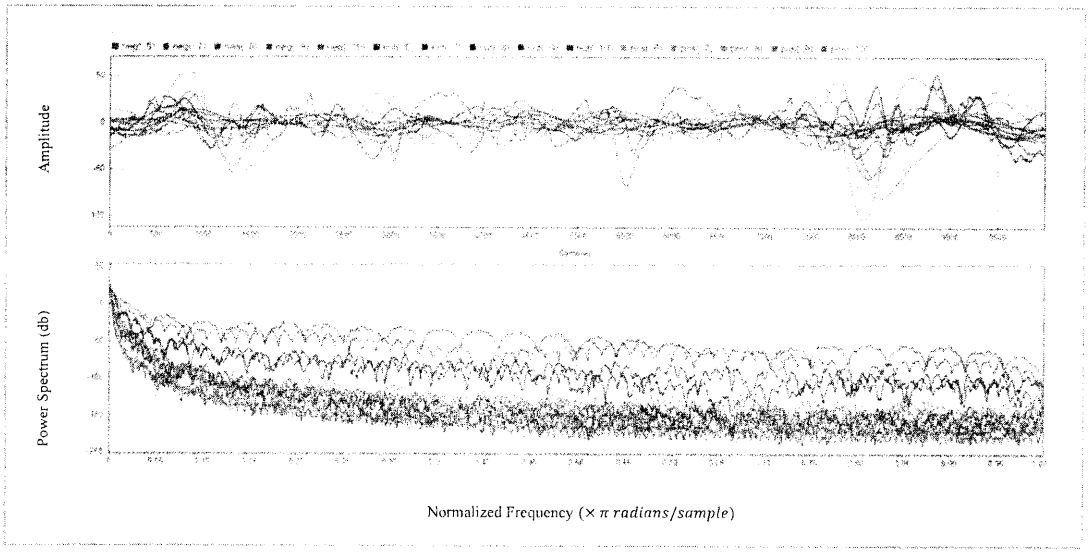


Figure 36: On the same 62 channels, in both the temporal and spectrum domains, we use a function that smooths signals by a factor of three.

Wavelet Transform

EEG signal processing is proposed using the Wavelet family and differential evolution. Because it efficiently operates in this sector, the discrete wavelet transform is commonly utilized in the feature extraction step, as evidenced by prior studies.

The feature selection process reduces dimensionality by removing non-essential features. Differential evolution is used for this step. This paper proposes a model for EEG classification that takes into account feature extraction and selection. During the course of our research, we analyzed seven distinct types of wavelets that are commonly seen.

There are several different forms of Daubechies, including Daubechies, coiflets, fejé-korovkin, discrete meyer, and reverse biorthogonal. To provide a wide range of features, many types of discrete wavelet transforms are applied. After that, we apply differential evolution to select optimal features that will result in the highest signal classification performance. We built classifiers and tested their performance using Bonn databases for the classification step.

The wavelet transform of a function $f(x)$ in (17) can be expressed as a linear combination of scaled and shifted versions of a mother wavelet $\psi(x)$ and a scaling function $\varphi(x)$. The coefficients of this expansion are called the detail coefficients $W_{\psi}(j,k)$ in (18) and the approximation coefficients $W_{\varphi}(j_0,k)$ in (19), respectively.

The variables j and k are integers that control the scale and position of the wavelets and scaling functions, respectively. The variable M is the number of samples in the signal $f(x)$, and N is the length of the wavelet filter. The functions $\varphi_{j_0,k}(x)$ and $\varphi_{j_0,k}(x)$ are obtained by dilating and translating the mother wavelet and scaling function by powers of two and multiples of k , respectively.

$$f(x) = \frac{1}{\sqrt{M}}\sum_{k} W_{\varphi}(j_0,k)\varphi_{j_0,k}(x) + \frac{1}{\sqrt{M}}\sum_{j=j_0}^{\infty}\sum_{k} W_{\psi}(j,k)\psi_{j,k}(x) \quad (17)$$

Approximation coefficients: $W_{\varphi}(j_0, k) = \frac{1}{\sqrt{M}} \sum_k W_{\varphi}(j_0, k) \varphi_{j_0,k}(x)$ **(18)**

Detail coefficients: $W_{\varphi}(j, k) = \frac{1}{\sqrt{M}} \sum_N f(x) x$ **(19)**

Discrete Wavelet Transform (DWT):

It is simply another way of encoding the signal and has no effect on the data held in the signal. The wavelet transform is merely a sampling of the continuous wavelet transform, whereas, in CWT, the information delivered is extremely redundant in terms of signal reconstruction. DWT is more efficient than continuous wavelet transforms at removing redundancy.

It contains enough information to analyze and synthesize the received signals. In comparison to CWT, DWT is simple to implement and reduces computing time and resource requirements. DWT is based on sub-band coding to achieve high computation of wavelet transform. Discrete wavelets are used to generate finite outputs from finite data sets.

In equation (20), we can calculate discrete wavelet transform of a signal t by applying a discrete set of wavelet functions $\psi_{a,b}(t)$ that are derived from a single mother wavelet $\psi(t)$ by scaling and shifting operations. The variables a and b are positive integers that determine the scale and position of the wavelet functions, respectively. The variable a controls the width or frequency of the wavelet, while the variable b controls the location or time of the wavelet. The factor $1/\sqrt{a}$ is a normalization constant that ensures that the wavelet functions have unit energy.

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right) \quad \textbf{(20)}$$

Short Time Fourier Transform (STFT):

Since the 1930s, the electrical wave activities of the brain have been studied using electrical stimulation linked to the scalp. However, for EEG signal feature extraction, several mathematical transform procedures such as Fourier (FT), Fast Fourier (FFT), Short Time Fourier (STFT), and Wavelet (WT) have indeed been presented. However, the Fast Fourier Transform lost information in the time domain and provided only spectrum information in the frequency domain. To address the issues with FFT, STFT was devised, which used the moving window function to represent the signal in both the time and frequency domains, in discrete form could be written as in below formula, where X is the signal, $w(n)$ is the window function, m is the time index, and ω is the frequency.:

$$X(n,w) = \sum_{i=-\infty}^{\infty} x(i)\,\omega(n-i)e^{-jwm} \qquad (21)$$

4. Hybrid Multilevel Features

We present a new multi-level features extraction technique that incorporates various features at various abstraction levels. It takes two or more types of features at various levels of abstraction from executables and combines them into a single hybrid feature set. The benefit of adding more levels of abstraction to the combined feature set has been empirically demonstrated.

Table 6: Some implemented Hybrid Multilevel Features.

#	**Basic Feature**	**Basic Type**	**Extended Feature**	**Extended Type**
1	Mean	Statistical	Shannon Enrtopy	Enrtopy
2	Standard Deviation	Statistical	Shannon Enrtopy	Enrtopy
3	Skewness	Statistical	Shannon Enrtopy	Enrtopy
4	Kurtosis	Statistical	Shannon Enrtopy	Enrtopy
5	Mean	Statistical	Deferential Entropy	Enrtopy
6	Standard Deviation	Statistical	Deferential Entropy	Enrtopy
7	Skewness	Statistical	Deferential Entropy	Enrtopy
8	Kurtosis	Statistical	Deferential Entropy	Enrtopy

Proposed Feature Extraction:

Significant effort has been put into researching different feature extraction strategies. For machine learning purposes, the outcomes of EEG signal segmentation with a distinct subset obtained from the segmentation process are represented by a collection of characteristics (features) that may comprise unnecessary and redundant information. In addition to improving generalization, selecting pertinent features can speed up a classifier's training process.

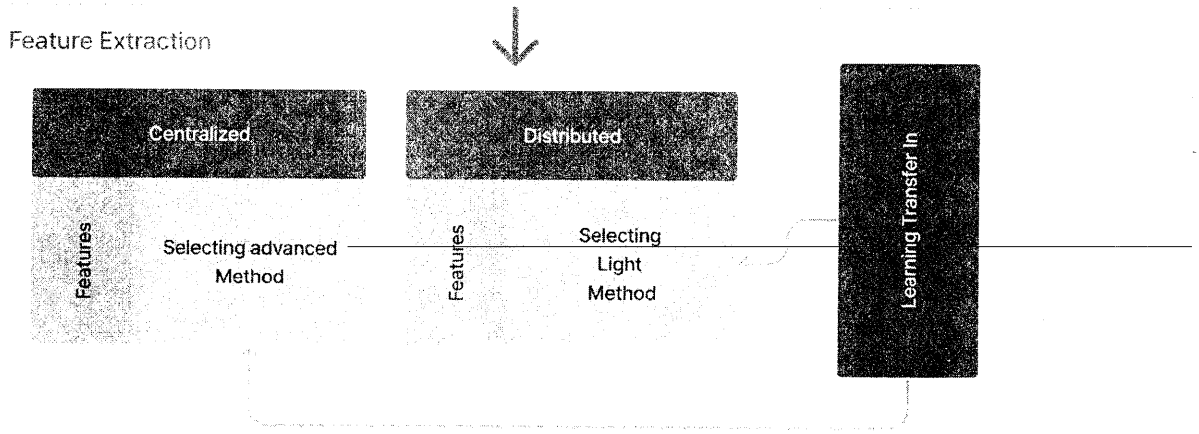


Figure 37: Adaptive feature extraction method using auto switch technique based on learning transfer weights.

Therefore, we are using simple feature extraction methods on the cap, such as statistical and timer-frequency features, and we use heavier methods, such as machine learning, neural network (autoencoder) and deep learning features, on the central unit. We propose to use the wrapper technique to select from those methods, where those methods are ranked based on classification performance results, and those results feed through transfer learning of the same subject or globally over the system.

Feature Selection

Using feature extraction from EEG signals, dozens of features can be generated. A key concern would be whether these features are needed for proper discriminating. By picking a part of the features, the complexity of the feature dimension can be decreased. Feature selection lowers data dimensionality by creating a model with only a subset of measurable features. When searching for a subset of classifiers that can accurately model measured responses, feature selection algorithms take into account limitations such as the features that must be included and eliminated, as well as the size of the subset. The primary benefits of using feature selection are enhanced prediction performance, faster and more cost-effective predictors, and a deeper comprehension of the data-generating process. Even if all relevant features include information on the response variable, the use of an excessive number of features might nevertheless result in a reduction in prediction performance. There are three distinct categories of feature selection algorithms to choose from:

Filter Type: This algorithm determines the value of features based on their qualities, such as feature variance and response relevance. As part of a data preprocessing procedure, choose important features and then use those characteristics to train a model. As a result, the training method has no bearing on the filter type feature selection.



Figure 38: Filter method action flow.

Wrapper Type: This method begins by training with a limited collection of features, and then applies a selection criterion to decide whether to add or delete a feature. The selection criterion makes an indirect comparison between the model's performance before and after the addition or deletion of a feature. The algorithm trains and develops a model until it meets its stopping conditions. Recursive feature removal, forward feature selection, and backward feature elimination are only a few examples of wrapper approaches.

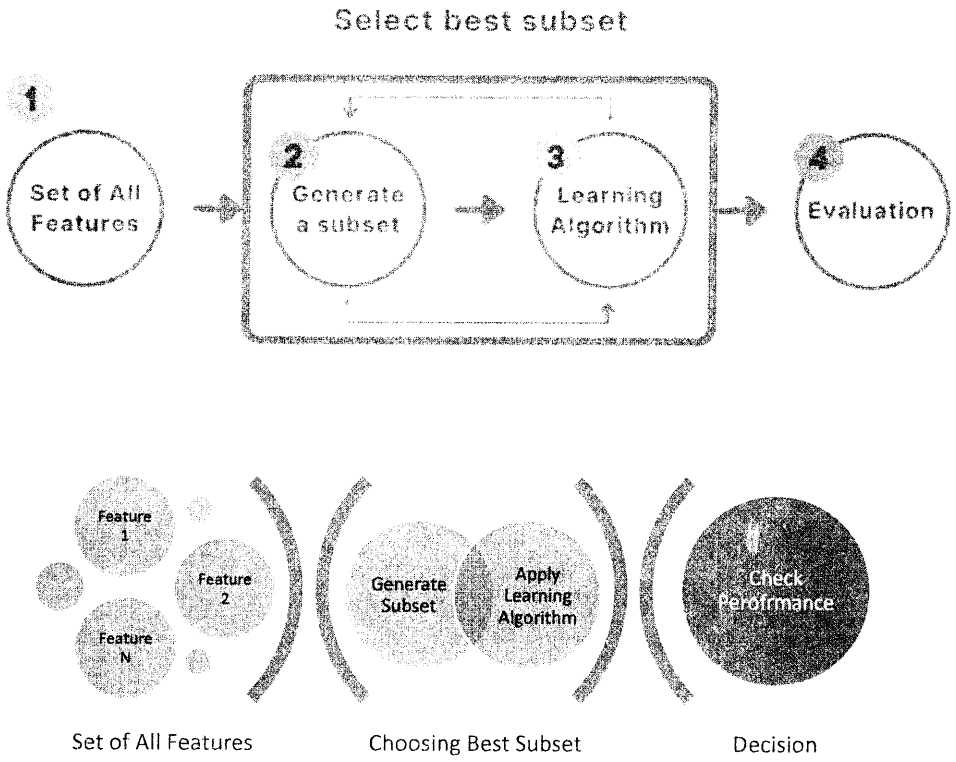


Figure 39: Wrapper method action flow.

Embedded Type: In embedded methods, the filter and wrapper methods are integrated into a single method. Embedded methods are methods that have built-in feature selection techniques that are utilized to implement them.

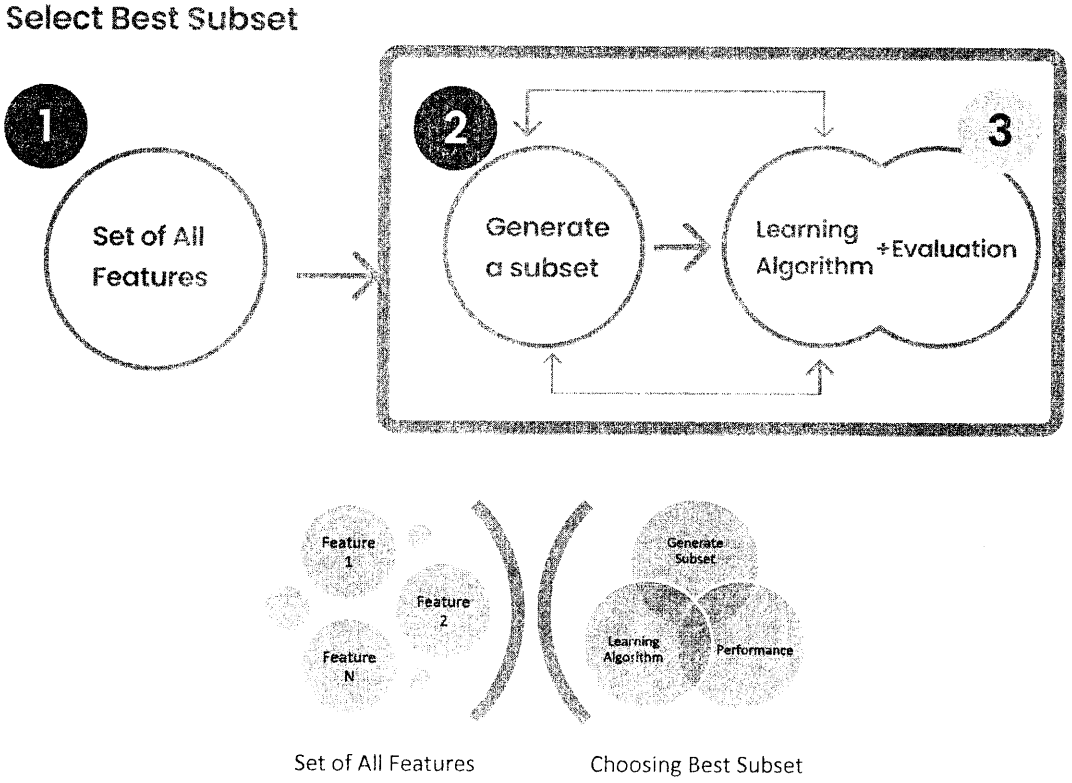


Figure 40: Embedded method action flow.

As part of the process of training the model, this approach discovers the value of the features. When you train a model, you get an understanding of the relative significance of the model's features. This algorithm chooses attributes that are compatible with a specific learning procedure. Some of the most common examples of these algorithms are Univariate feature selection, Recursive feature elimination, LASSO or RIDGE regression which feature inherent penalization mechanisms to reduce overfitting.

In addition, feature selection algorithms can be classified based on whether or not they rank features consecutively. The sequential feature selection approach minimizes redundancy and maximum relevancy.

After selecting features, we can train two models and compare their accuracies. Feature selection is chosen over feature transformation in situations when the initial features and their values are important for modeling, and the purpose of the modeling process is to locate an influential subset. When their categorical features are present, feature selection emerges as the primary strategy for a reduction since numerical transformations are rendered useless.

We used the filtering approach with lightweight computing methods in distributed computation mode, such as maximum variance for feature extraction and PCA for feature reduction. For distributed computation, wrapper and embedded approaches will also be used. We suggest more advanced ways for centralized systems to choose which features to use, like ReliefF minimum redundancy maximum relevance (mRMR) and recursive feature elimination (RFE).

In all modes, we also used PCA to cut down on the number of features. For the embedding approach, we used a simple method for evaluating features, like decision trees or SVM, for distributed computation, and a more complex method, like Ensemble or RNN classifiers, for centralized computation.

Algorithm 2: EEG feature selection algorithm

Input: Previously used Feature Subsets and their performance for both current subject and the top relevant subject.

Output: Reduced subset of features with expected maximum performance

Initialization: N = 32 (initial number of feature subsets), M (Transferred Feature Subsets), selected = one subset

```
for i ← 1 to N do
    Weighted Rank ← N(i) and M(i)
    for j ← 1 to N do
        Resort Subsets base on weighted rank
    end
end
return Weighted Feature Subsets
```

3.2. Classification Methodologies:

The whole total of a person's feelings and actions is what we refer to as their emotion. In addition to this, it frequently reflects the opinions and attitudes of human beings. In addition to this, it always plays an important part in our day-to-day lives, particularly in the interactions between people. Emotion recognition based on a computer is also becoming an essential component of the more advanced brain-computer interface system (Mattar et al., 2019).

Deep learning is commonly recognized as a subfield of machine learning; however, what precisely is machine learning? It is an area of computer science in which computers learn to solve problems that are hard to solve with regular programming codes. We supply an input dataset, and machine learning algorithms learn from it to produce output for previously unseen data.

We shall now categorize machine learning algorithms as parametric or nonparametric. Algorithms are classified into four categories. An algorithm can be supervised or unsupervised, parametric or nonparametric in nature.

Talking about parametric and nonparametric algorithms, we frequently make assumptions about the function structure that builds our model in parametric approaches to simplify the task. Nonetheless, these assumptions can significantly ease the learning process while limiting what can be learned. Logistic Regression, Linear Discriminant Analysis, Naive Bayes, and Simple Neural Networks are examples of parametric machine learning algorithms. There are various advantages of using metric algorithms. For example, they can easily understand and analyze results, learn from data faster, and operate successfully even if the data is not perfect. However, because parametric algorithms are severely restricted to the stated form, they are better suited to simpler problems.

On the other side, nonparametric techniques are a category of algorithms that do not make any underlying assumptions about the shape of the function that is to be estimated. This contrasts with parametric approaches, which do make such assumptions. If they refrain from forming assumptions about the training data, they will be able to learn any functional form. Nonparametric methods include things like decision trees, support vector machines, and k-nearest neighbors, to name a few examples.

Nonparametric algorithms provide advantages to parametric algorithms, such as fitting a wide variety of functional forms while making less assumptions about the underlying function. Furthermore, they can lead to higher-performing prediction models. However, they necessitate a lot more training data to estimate the model, resulting in a slower training process with a higher danger of overfitting the training data, and it is harder to explain why specific predictions are made.

In addition to parametric and nonparametric which are statistics methods based categorized, we have three more categories based on data shape which are supervised, unsupervised and reinforcement methods (Sakkalis et al., 2008)(Isaksson et al., 1981).

There are a lot of algorithms well known with no specific categories, in Figure 41 we visualize the algorithms in tree view, trying to give an overall view of classifications and where they fit the machine learning methods.

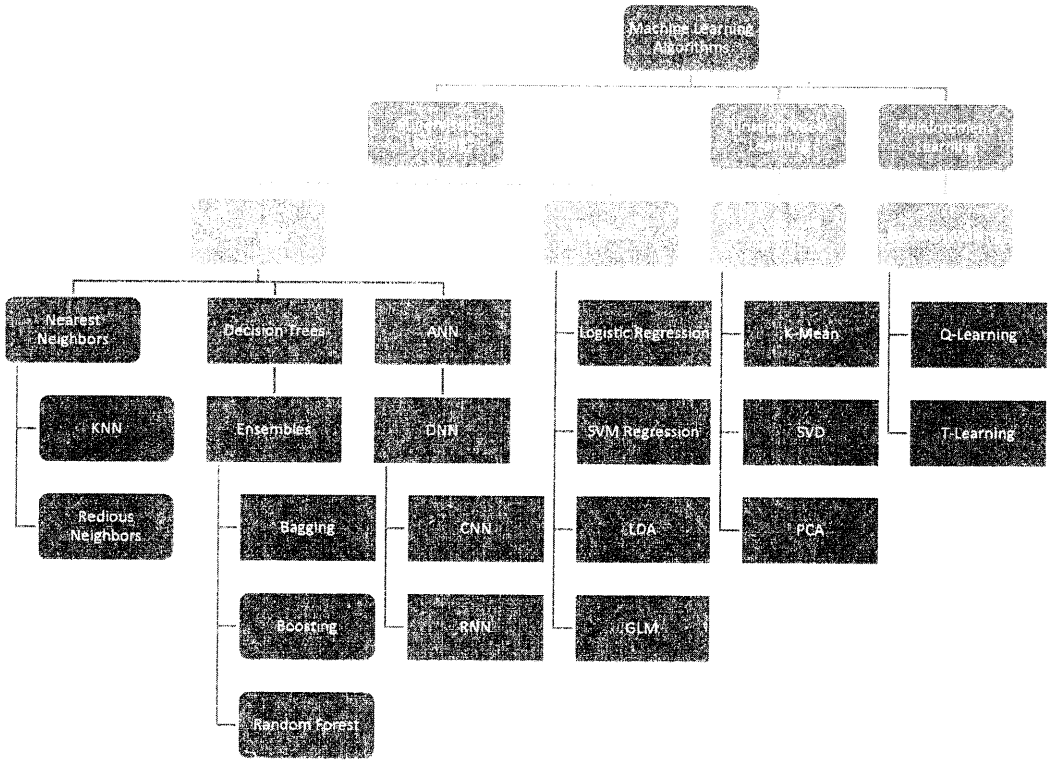


Figure 41: Overview of machine learning.

Given this context, this study examines advanced adaptive classification algorithms utilized in EEG-based BCIs, such as genetic algorithms (GAs), particle swarm optimization (PSO), and Group of Adaptive Models Evolution inductive models. The adaptive classifiers, as the name implies, are deep learning classifiers whose parameters are incrementally re-evaluated and changed over time when new EEG signals are sent to the classifier. Furthermore, the research frame was chosen to establish a comprehensive overview of this specific category of classifier, because dynamic classifiers have been shown to outperform static classifiers, even with low supervision or unsupervised adaptation (Lotte et al., 2018).

3.2.1. Supervised Classifiers

The categorization of electroencephalography (EEG) is a crucial component of the Brain-Computer Interface (BCI), which serves as an interface from the human mind to a computer. This paves the way for subjects with physical limitations to communicate even with the help of a computer. Supervised learning is a technique for developing a function that implies an expectation from labeled training records and then uses correlations to determine which label should be applied to unlabeled fresh records. A supervised learning technique examines training data and creates an implied function that may be used to map new samples.

Classification and regression are the two primary categories of supervised procedures. Whereas classification recognizes the category to which the data belongs, regression forecasts a numerical value based on previously observed data; price prediction and body weight prediction are two instances of regression.

Decision Trees Classifier

Decision tree learning is a Supervised Machine Learning Algorithm and one of the most extensively used and useful inductive inference methods that use a set of rules to make judgments for both classification and Regression; in our case of EEG signals, these criteria are comparable to those used by clinicians when interpreting EEG scans. Learned trees can be expressed as conditional rules, which may increase human readability by continuously splitting data based on particular criteria. Furthermore, the outcomes of decision tree classifiers can be likened to the guidelines employed by clinicians when analyzing EEG. The decision tree comprises core nodes that reflect the attributes of a dataset, branches that represent the rule base, and leaf nodes that indicate the conclusion (Coppersmith et al., 1999). We tested the Fit binary decision tree, which employs these processes to split nodes, computes node weighted impurity, and calculates the chance of observation in node t using the formula below.

$$P(T) = \sum_{j \in T} w_{j^\circ} \qquad (22)$$

Where w_{j° denotes the weight of observation j° and T is the set of all observation indices in node t If neither Prior nor Weights are specified, $w_{j^\circ} = 1/n$, where n is the sample size.

Nearest Neighbor Classifier (KNN)

The k-nearest neighbors' method (KNN) is a non-parametric classification approach developed by Evelyn Fix and his colleague Joseph Hodges in 1951 and later modified by Thomas Cover. It is used for classification and regression, and both the distance metric and the number of nearest neighbors may be adjusted. The input in both situations consists of the nearest k training sample in a data collection.

When used for classification, the KNN produces a class choice based on a majority vote of its neighbors, with the item being selected to the most popular class among its k closest neighbors. If $k = 2$, for example, the object is allocated to the class of the two closest neighbors.

When used in regression, the KNN produces a property value that is the average of the values of the k nearest neighbors.

$$d(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \qquad (23)$$

The typical train-test split would likely be 70:30, which would not be effective for unexpected future data, and the concept of cross-validation would assist us in firmly asserting that we can attain a specific level of accuracy for freshly unlabeled data.

So instead of separating the data into two portions, we used cross-validation to separate it into three. Data for training, cross-validation, and evaluation. We use training data to detect nearest neighbors, cross-validation data to determine the optimal value of K, and completely unknown test data to assess our technique. This

test data corresponds to previously unknown data sets in the future.

Support Vector Machine (SVM)

Support vector machine (SVM) is a supervised learning method for classifying, predicting, and detecting outliers.

The following are some of the benefits of using a support vector machine: It uses a subset of the training set in the decision unknown as support vectors which is also computationally efficient in high-dimensions and in circumstances when the number of dimensions is greater than the number of samples.

Support vector machines have some drawbacks, such as not directly providing estimates; those are also calculated using costly five-fold cross-validation. Overfitting problem if the number of features is much larger than the total of samples.

Naive Bayes Classifier

The Bayes' Theorem is put to use in the process of developing a group of probabilistic classification strategies that are referred to as Naive Bayesian classifiers. It is not a single algorithm but rather a series of approaches with a similar premise, which is that it makes strong feature-independent assumptions.

The Bayes' Theorem is a mathematical formula that determines the chance of an event happening based on the probability of another occurrence. They are among the most straightforward Bayesian network models, but when paired with kernel density estimation, they are capable of producing results with higher degrees of precision.

The following equation (24) expresses Bayes' Theorem mathematically, where the probability of a hypothesis A given some evidence B. The variable $P(A|B)$ is the posterior probability of A given B, which is the probability of A after observing B. The variable $P(B|A)$ is the likelihood of B given A, which is the probability of observing B if A is true. The variable $P(A)$ is the prior probability of A. The variable $P(B)$ is the marginal probability of B, which is the probability of observing B regardless of A:

$$P(A|B) = \frac{P(B|A)\,P(A)}{P(B)} \qquad (24)$$

3.2.2. Deep Learning Classification Models:

Deep neural networks, abbreviated as DNN, have emerged as one of the most powerful and widely used methodologies across a variety of study subfields during the past several years. Deep neural networks have demonstrated remarkable effectiveness in a variety of EEG-related applications. Following feature extraction, the output data were then split into two parts: 80% was used for training the DNN, and 20% was used for testing. (Tsihrintzis & Jain, 2020)

We chose a few of the most well-known models to test with MATLAB's machine learning tools and python's deep learning libraries like TensorFlow and Keras in order to compare the efficacy of deep neural networks with that of other traditional machine learning models for EEG emotion classification. Our goal was to determine how well deep neural networks perform in comparison to these other models.

In recent years, deep neural networks (DNN) have been developed into one of the most effective and popular methods in many research fields. Deep neural networks have shown exceptional efficiency in several EEG applications. The output data after feature extraction were divided into 80% for training and 20% for testing in the DNN. Sample of chosen models are listed below:

- Simple Neural Network
- Bagged (Ensemble)
- Discriminant Analysis
- KNN

Deep Learning Models:

Bagged (Ensemble)

A technique for producing many copies of a predictor and integrating them into a single cumulative model is known as bagging predictors. When estimating a numerical result, the aggregate utilizes the mean of the versions, and when estimating a class, it employs the majority vote. Numerous variants are

produced by creating bootstrap copies of the learning set and employing these as new learning sets. Bagging enhances the accuracy of tests on actual and simulated data sets using classification and regression trees and subset choosing in linear regression by a substantial amount. Critically unstable is the prediction approach. If perturbing the learning set produces big changes in the resulting predictor, bagging can improve accuracy. (Breiman, 1996)

Discriminant Analysis

The discriminant analysis method is yet another strategy for dimensionality reduction. The distance that a sample is from the middle of the class is calculated using discriminant analysis (DA). It does this by constructing a new set of axes to arrange individuals of the same group to be as near to one another as possible while simultaneously shifting the groupings as far away as possible. The original variables are combined in a linear fashion to form the discriminant axes, which are sometimes referred to as canonical variates (CVs). DA is an example of supervised pattern recognition since its operation depends on the user's information. In particular, DA calls for a complete cognition of the various groupings to which each sample belongs. Both principal component analysis (PCA) and discriminant analysis (DA) are routinely used on the same sample types; nevertheless, PCA is typically used since it reduces the total number of samples. (Mendlein et al., 2013)

Convolutional Neural Network

Convolutional neural networks (CNNs) have been used to do automated feature extraction and classification in computing visual and speech recognition, have been effectively used to EEG-based BCIs. Convolutional neural networks use the original multidimensional signals as input and use the supervised learning method to transform the hidden layers into suitable feature extractors, eliminating the need for a complicated artificial feature extraction process. Convolutional neural networks are well suited to data that fluctuate dramatically over time, such as EEG.

CNN was utilized to classify objects into K distinct categories, each of which was determined by a collection of attributes. In

order to properly classify each item, we calculated its category using the sum of the quadratic of the distance that separated it from the most relevant cluster. CNN is an algorithm for deep learning used to evaluate images' different representations. The Fourier transform's shared-weight architecture, which scans the hidden units and translational affine features, is what gives these ANNs their name, shift neutral spatially ANN. They are referred to as so because of their nature. The following are some examples of applications: image and video detection, advanced analytics, image classification, segmentation approaches, computer vision, text analysis, links with the central nervous system, and economic time series (Guo et al., 2021).

A 3D CNN technique was introduced by Salama et al. to identify emotions based on multichannel EEG inputs. In order to increase the overall performance of their 3D CNN model, they designed a data augmentation phase. They used the DEAP dataset to reach an accuracy of 87.44% for the valence dimension and 88.49% for the arousal dimension (Salama et al., 2018). Tripathi et al. described a CNN-based emotion identification system using the DEAP dataset.

The technology was used to analyze EEG signals. They investigated a basic deep neural network and a convolutional neural network as potential brain representation. The latter exhibited a 4.96% increase in performance compared to the most recent and cutting-edge approaches (Tripathi et al., 2017). Researchers Moon et al. used CNN based on EEG data for emotion identification. In order to explain the simultaneous activation of several distinct brain areas, they took advantage of brain connection features that had not been considered in earlier research. Therefore, their approach successfully catches asymmetry patterns of brain activity, which play a key part in the perception of emotions (Moon et al., 2018).

Autoencoder Neural Network

A neural network known as an Autoencoder is one that is made up of numerous layers of Autoencoders. In this type of neural network, the outputs of each layer are connected to the inputs of the consecutive levels. The autoencoder makes an effort to learn a function that is approximately equivalent to the identity function, as seen here: The unsupervised pretraining strategy is utilized in

conjunction with greedy layer-wise training in the Deep Learning Network. Autoencoder can formulated as in equations (25) to (27), where:

- $\hat{x}$ is the reconstructed output of the autoencoder, which is an approximation of the input x.
- h is the latent representation of the input x, which is a lower dimensional vector.
- W and b are the weight matrix and bias vector of the encoder or decoder neural network, respectively.
- f and g are the activation functions of the encoder or decoder neural network, respectively. They can be nonlinear functions such as sigmoid, tanh, or ReLU.
- k is a scaling factor that controls the sparsity of the latent representation. A larger k means a sparser representation.

$$\hat{x} = h_{W,b}(x) \approx x \tag{25}$$

$$h = encoder(x) = f(W_k \times x + b_x) \tag{26}$$

$$h = decoder(x) = g(W_k \times x + b_x) \tag{27}$$

Unsupervised pretraining is carried out by the algorithm on a layer-by-layer basis, beginning with the input layer and progressing all the way up to the output layer. In order to learn the principal features h of the inputs, the first Autoencoder, which is located in the first hidden layer, is trained on the raw data. In order to keep the costs to a minimum, during the pretraining phase, all of the weight and bias factors were studied and memorized.

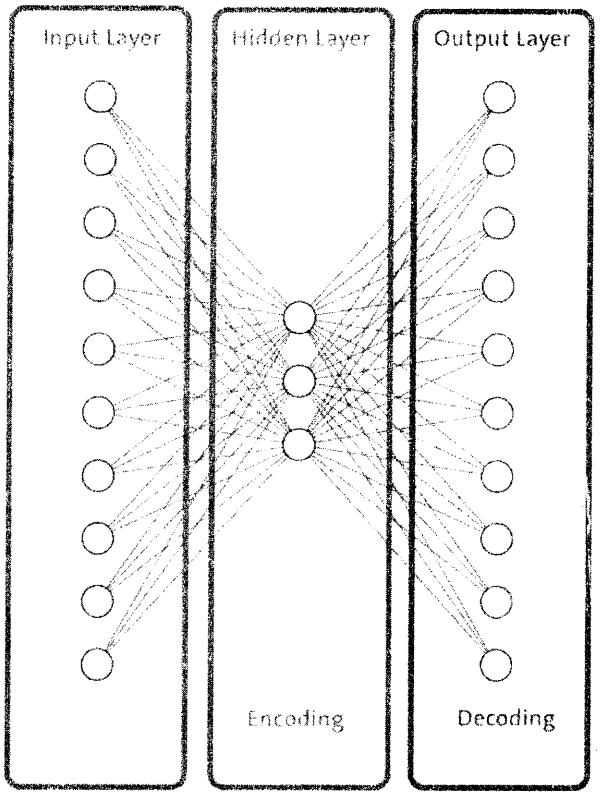


Figure 42: General autoencoder structure.

After that, the method uses the untrained Autoencoder's raw inputs to propagate forward into the trained version of the autoencoder in order to retrieve the principal feature activations. For the purpose of pretraining in the subsequent hidden layer, the algorithm computes its features using the exact same process that it did for the features that it had learned in the layers that came before it.

Recurrent Neural Network

A directed or undirected graph can be generated by the connections between the nodes in a recurrent neural network throughout the course of time. Because of this, it is able to behave in a way that is temporally dynamic. RNNs, which are built from feedforward neural networks, have the capability to handle input sequences of varying lengths by exploiting the internal state of the network (memory). That is why RNNs have short-term memory problems caused by the vanishing gradient problem, which is common in neural networks that use gradient-based learning

methods and backpropagation. Two types of specialized RNNs were developed to overcome this challenge: LSTM (Long Short-Term Memory) and GRU (Gated Recurrent Unit).

The complex RNN (sequences network) known as the Long Short-Term Memory Network (LSTMN) is responsible for storing the needed information. RNNs cannot remember long-term dependencies since the gradient they employ is constantly decreasing. LSTMs are designed to steer clear of any potential long-term dependence issues consciously. The GRU is comparable to an LSTM that has a forget gate, but it differs in that it does not have an output gate. As a result, it has fewer parameters. On tasks involving modeling polyphonic music, modeling voice signals, and natural language processing, GRU performed in a manner analogous to that of LSTM. On some smaller and less frequent datasets, GRUs have fared better.

DNN Classification Method:

In the model the Autoencoder was utilized in order to extract features, and logistic regression was utilized in order to predict emotional states. First, the features are transformed from time series into 2D pictures using a Short-time Fourier Transform (STFT) and a spatiotemporal analysis, and then an Autoencoder (AE) is used to encode and decode the images in a recursive fashion. After that, the data with the decreased amount of redundancy is utilized as the input features of the Long short-term memory (LSTM) for the classification job.

The recommended classification model, which makes use of CNN's convolutional layers and AE's sparsity, demonstrates high classification accuracy and rapid convergence in comparison to alternative methods of attack.

In practice, the evaluation of EEG signals and derived features typically involves using classification in addition to clustering. We have conducted research on and made comparisons among a variety of classification strategies, including the Decision tree classifier, the Naive Bayes technique, and the K-NN algorithm.

In addition to the improved classification strategy that was proposed, we also applied deep learning classifiers like the CNN algorithm.

Convolutional neural networks, often known as CNNs, play an important part in our design because of their widespread application in the analysis of visual data. The concept of matrix multiplications typically comes to mind when we think about neural networks; however, this is not the case with CNN.

Convolution is the technique that is used by this strategy. The term "convolution" refers to a mathematical operation that is carried out on two functions to produce a third function that indicates how the form of one function is altered by the shape of the second function.

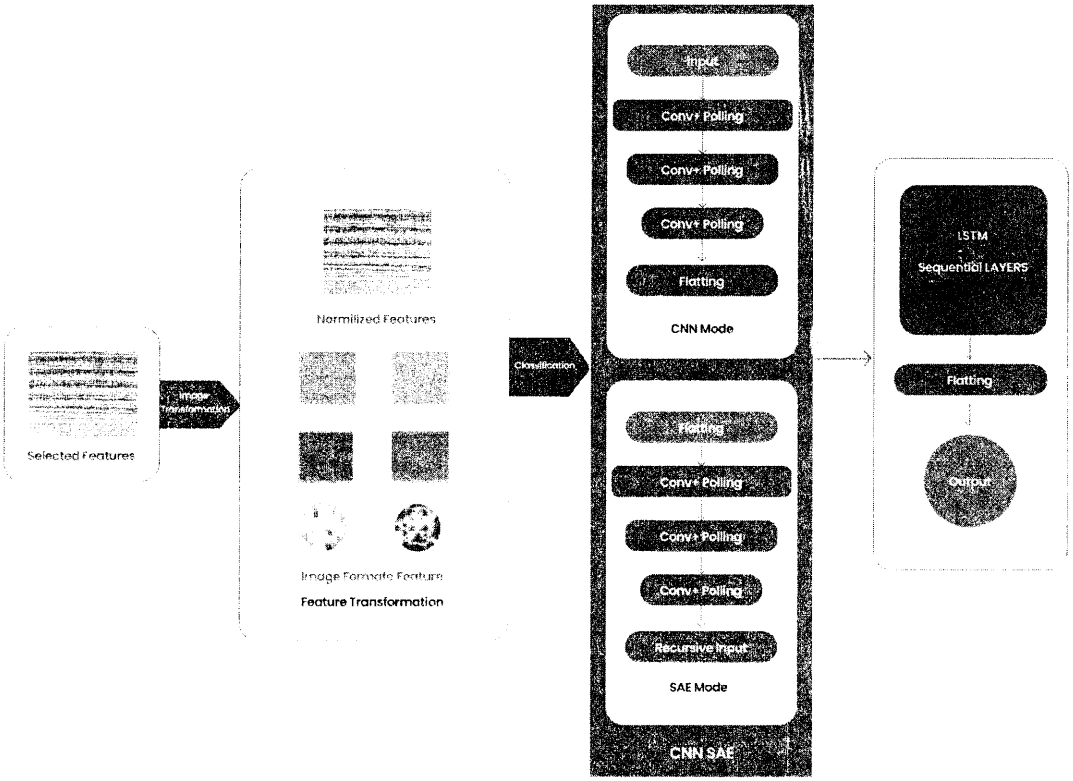


Figure 43: Proposed multi-stage recursive classification model, combines CNN, AE and RNN.

CNN Architecture:

A convolutional neural network's primary objective is to identify an image's properties and then compress those features into a smaller space without altering their essential qualities. Layers of a convolutional neural network (CNN) can be divided into one of three categories:

- input layers, which collect raw data from the source.
- hidden layers, which collect input from one layer and transmit output to another.
- output layers, which provide predictions.

Through the use of more straightforward mathematical operations, the Activation Function decides whether or not neurons should be fired and whether or not the neuron's input to the network should be considered significant in the prediction model.

The convolutional layer's mathematical expression is as follows:

$$y_{mn} = f\left(\sum_{j=0}^{j-1}\sum_{i=0}^{I-1} x_{m+i,n+j} w_{ij} + b\right) \quad (28)$$

Hereby x is the input 2D data; y is the output of M × N; $0 \leq m \leq M$, $0 \leq n \leq N$; w is the convolutional kernel with J × I; b is the bias; and f is the activation function.

The primary goal of the Activation Function is to transform the cumulative weighted input of the node into an output value that may be passed on to the next hidden layer or output. The hidden layers of a CNN typically include convolutional layers, pooling layers, fully connected layers, and normalizing layers, among other types of layers.

It essentially signifies the utilization of convolution and pooling functions as activation functions as opposed to the conventional activation functions that are used in traditional neural networks.

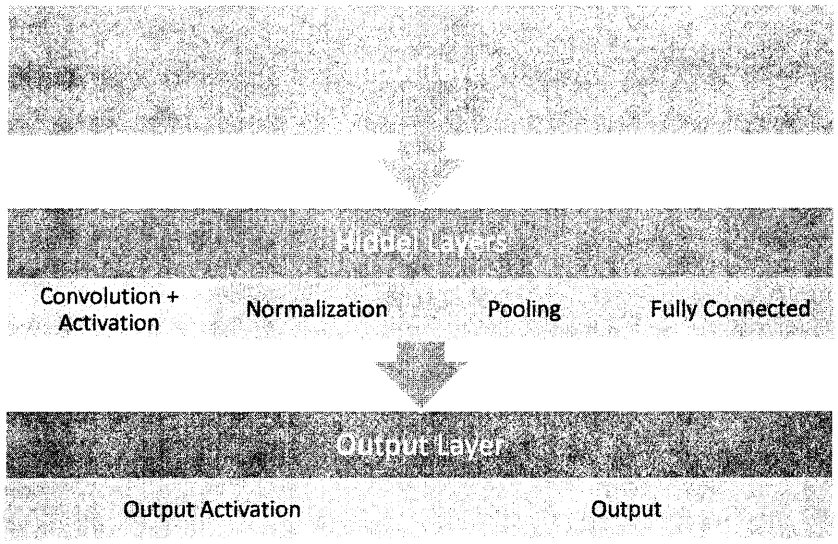


Figure 44: High level CNN architecture.

A convolutional layer is the primary component that goes into the construction of a CNN. It is comprised of a group of filters (also known as kernels), all of which need to be taught throughout the course of the training process.

In this layer, the activation function is ReLU changes all negative activation values to zero as in equation (29), where x The function $max(x,0)$ returns the maximum of x and 0, and the layer executes a dot product of the convolution kernel and the layer's input matrix.

$$f(x) = \max(x, 0) \tag{29}$$

A feature map is produced by the convolution technique as the convolution kernel slides along the input matrix for the layer. This matrix then contributes to the input of the layer that comes after it. Additional layers, such as the pooling layer, the fully connected layer, and the normalizing layer.

Pooling is a technique that reduces the size and complexity of the feature maps in convolutional neural networks. There are different ways to pool, such as taking the mean or the maximum value of a region of the feature map.

The equation for pooling is found in equation (30), and for the mean and maximum pooling layer is given in Equations (31) and (32), where:

The variables in the equations are:

- $[v_\ell^f]_n$ is the pooled feature map of the n-th input sample for the f-th filter.
- $\rho\ell$ is the pooling function of the l-th pooling layer.
- $[u_\ell^f]_{n\ell}$ is the feature map of the n-th input sample for the f-th filter before pooling at the l-th layer.
- f is the number of elements in the feature map.
- $|n_\ell|$ is the size of the pooling region for the n-th input sample at the l-th layer.

$$[v_\ell^f]_n = \rho\ell\left([u_\ell^f]_{n\ell}\right) \tag{30}$$

$$\rho\ell\left([u_\ell^f]_{n\ell}\right) = 1^T[u_\ell^f]_{n\ell}/|n_\ell| \tag{31}$$

$$\rho\ell\left([u_\ell^f]_{n\ell}\right) = max\ [u_\ell^f]_{n\ell} \tag{32}$$

Convolutional neural networks have several benefits over other types of neural networks when it comes to the extraction of features; nevertheless, they are unable to remember past time-series patterns.

LSTM neural networks have a higher rate of success in the processing of temporal data when compared to convolutional neural networks.

The mathematical expression of the LSTM unit is defined as follows:

$$f(t) = \sigma\left(W_f h^{t-1} + U_f x_t + b_f\right) \tag{33}$$

$$i(t) = \sigma(W_i h^{t-1} + U_i x_t + b_i) \tag{34}$$

$$a(t) = \tanh(W_a h^{t-1} + U_a x_t + b_a) \quad (35)$$

$$o(t) = \sigma(W_o h^{t-1} + U_o x_t + b_o) \quad (36)$$

Where Wf, Wi, Wo and Wa a are the weight coefficients of $ht{-}1$ in the processes of forget gate, input gate, output gate, and feature extraction, respectively; Uf, Ui, Uo and Ua are the weight coefficients of xt in the processes of forget gate, input gate, output gate, and feature extraction, respectively. $h(t)$ represents the input of moment t. $ht{-}1$ represents the hidden layer state value of moment t.

The offset values for the forget gate, the input gate, the output gate, and the feature extraction procedure are represented by the symbols $bfbf$, bi, bo and ba, respectively. The tangent hyperbolic function is represented by tanh, while the sigmoid activation function is represented by.

The cell state $c(t)$ at time t is created by the outcomes of the forget gate and input gate calculations acting on $c(t{-}1)$ and is represented by the following formula:

$$c(t) = c(t-1) \odot f(t) + i(t) \odot a(t) \quad (37)$$

where the Hadamard product is present. The output o(t) and the current cell state $c(t)$ are then used to determine the hidden layer state $h(t)$ at time t:

$$h(t) = o(t) \odot \tanh(c(t)) \quad (38)$$

A discretization method that makes use of samples is known as the pooling layer. The objective is to decrease the dimensionality of an input representation (such as an image or the output matrix of a hidden layer) while simultaneously making assumptions about the properties that are included in the binned subregions. If

we do not employ pooling or max pooling, it will be a costly undertaking from a computing standpoint.

A technique for standardizing the data before it is processed is known as the normalization layer. Having many data sources that fall within the same range, to put it another way. If the data is not normalized before training, it might cause problems in our network, which makes training much more difficult and slows down its learning rate. Instead of being conducted on the raw data, the normalizing technique known as Batch Norm is carried out between the layers of a neural network. Instead, then processing the complete data collection at once, it is done in smaller chunks. It quickens the learning process by employing quicker learning rates and speeds up training, making education easier to grasp.

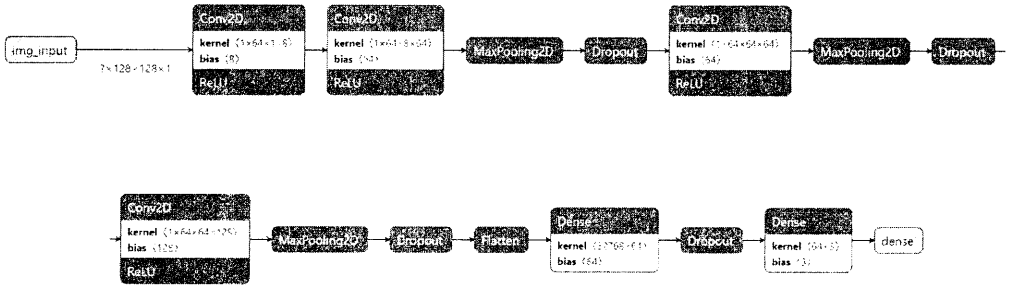


Figure 45: Sample CNN model, saved in h5 Keras model format and visualized by netron app (Ganesh, n.d.; Roeder, n.d.).

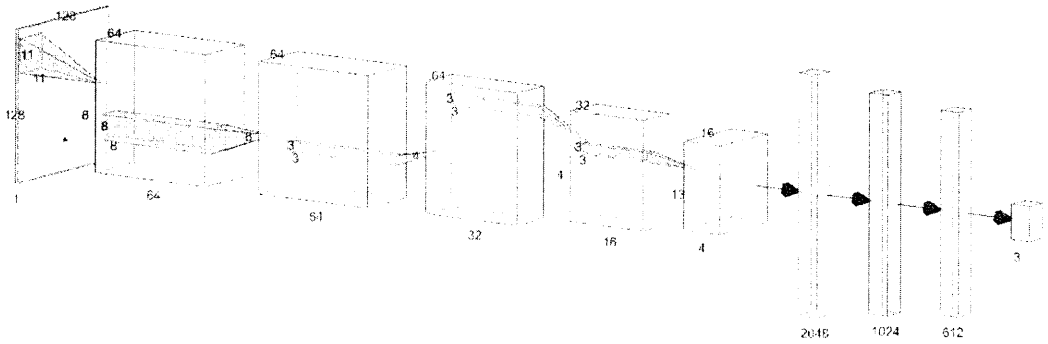


Figure 46: Sample CNN model designed by NN-SVG (LeNail, 2019).

Signal Transformation:

The continuous activity of EEG signals that were captured over a period of time is what is contained in a raw EEG file. EEG is a common tool we employ to investigate how the nervous system reacts to events from the outside world. For instance, we could be interested in researching how the brain reacts when presented with a series of pictures or noises. For this reason, rather than examining a continuous time course of activity, we want to focus on areas of the EEG signals that correlate to the period when external events happened. This will allow us to better understand how the brain responds to different stimuli.

In a typical experiment, we will indicate the time when significant events occur by including triggers in an EEG recording so that they may later be imported as events. This is done while the experiment is being conducted. An event will, in essence, provide information on when, in time, an external stimulus was provided and the nature of this stimulus.

After we have collected all of this data, we will be able to transform our raw EEG signal into epochs. Each Epoch will extract chunks of EEG data centered on a particular window. These chunks will be timestamped with the moment at which each external event took place.

Data from an EEG provides a wealth of information across many dimensions, including time and location (in the form of electrodes scattered around the head). It is essential to effectively represent this information in order to gain an understanding of the types of reactions and activity that are contained within our EEG data; a variety of visualization options will be taught using an example dataset.

When visualizing any EEG dataset, the initial stage often involves a visual evaluation of single-trial data. This may be done in a variety of ways. With the help of this examination, we will be able to examine the recorded epochs in order to determine the integrity of our dataset.

Because neural activity displays both non-linear dynamics and non-stationarity across temporal scales, it is impossible to study neuronal activity in an accurate manner using traditional methods. There is a need for tools that can uncover the complex spatiotemporal hierarchical patterns that are concealed within these signals. To gain information from the dynamics of EEG

signals in the subject, algorithmic complexity metrics of EEG spectrograms need to be applied; such metrics may be beneficial per se for categorization or scoring. On the other hand, the ways in which the algorithms immediately discover the required characteristics would be great for us to apply.

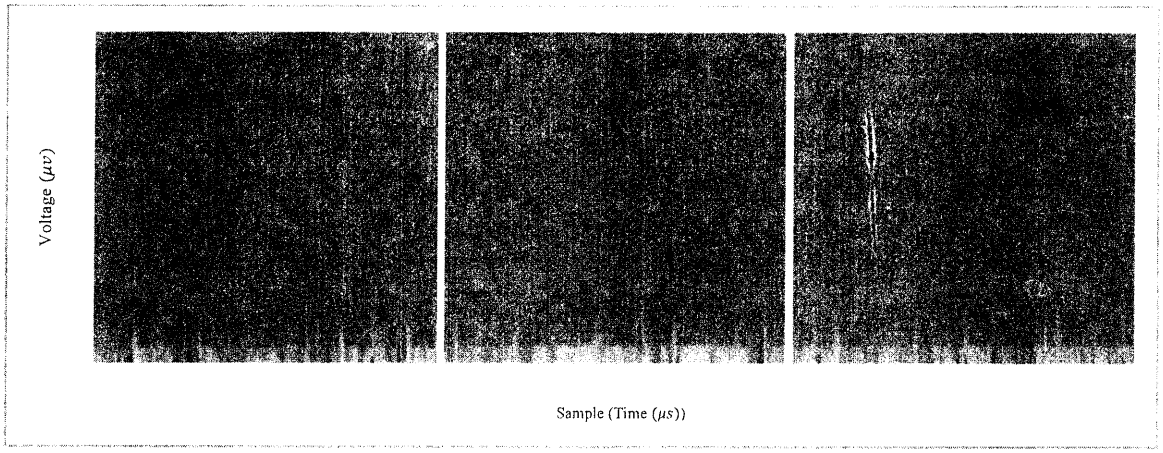


Figure 47: Three spectrogram samples of EEG signal representation, with 200 Hz sampling rate.

In order to construct spectrograms, we used EEG data from each channel, and after detrending the blocks, we analyzed the data using Fourier analysis (FFT). Utilizing a sliding window as shown in Figure 47, we were able to get artifact-free epochs of the selected channel for each of our subjects. Therefore, the data frames generated as a result are multidimensional arrays with the form size of the various channels and Epochs. We decided on a minimum number of frames to collect from each participant in order to ensure that each individual contributed the same amount. As part of the training process, datasets were made more equitable for subjects by randomly duplicating subjects in the group that had fewer subjects.

As a result, both the training and test sets were well-balanced in terms of the number of subjects and frames allocated to each class. In the end, the data were standardized to unit variance and centered before being applied to each frequency and channel.

Voltage topographies are an additional method that may be used to display EEG data in addition to time courses of activity. The EEG voltage levels at each channel on the scalp are represented

graphically by the topographies. They operate in a manner comparable to that of a heat map, in which varying voltage ranges are shown using a variety of colors, see sample of those heatmaps in Figure 48.

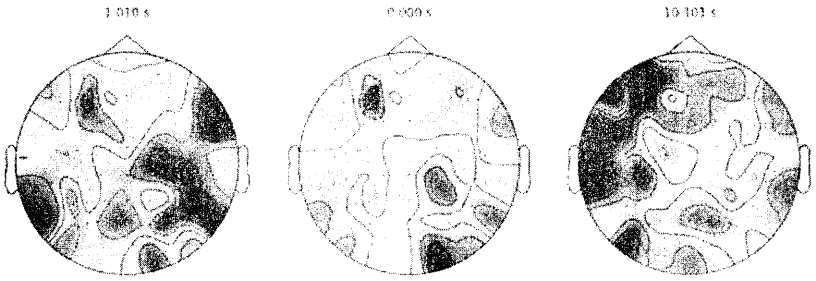


Figure 48: Some topographies for different EEG time windows.

We are given the ability to examine all at once the distribution of voltage levels over the scalp thanks to topographies. Because of this phenomenon, they can be informative of all electrodes simultaneously rather than only looking at individual ERP components. This is a particularly useful property of the phenomenon.

Models:

Here, we discuss three proposed deep neural networks based on CNN, AE, and RNN models. The new DNN architecture for EEG-based signals is designed to be:

- Simple convolutions may be trained with a small number of samples.
- share similar data formats for input and output.
- can transfer knowledge easily with a minimal number of parameters and weights.

Figure 49 to Figure 51 shows how our models look and give a full description of them for EEG trials with 62 channels, 150 samples, and a sampling rate of 128 Hz. We put the model together using the Adam optimizer, which is a stochastic gradient descent method that minimizes the categorical cross-entropy loss function. The optimization method is based on the adaptive estimation of

low-order moments and uses a stochastic gradient. (Kingma & Ba, 2014)

We run 200 training iterations and perform early stopping as a regularization technique to combat the overfitting challenge. During the training process, Early Stopping checks the performance of the model for each epoch on a set of validation data that has been held out. The training is terminated contingent on the validation performance, which we specified to be no change in the accuracy of the validation after waiting for 10 epochs with no improvement before the training is terminated.

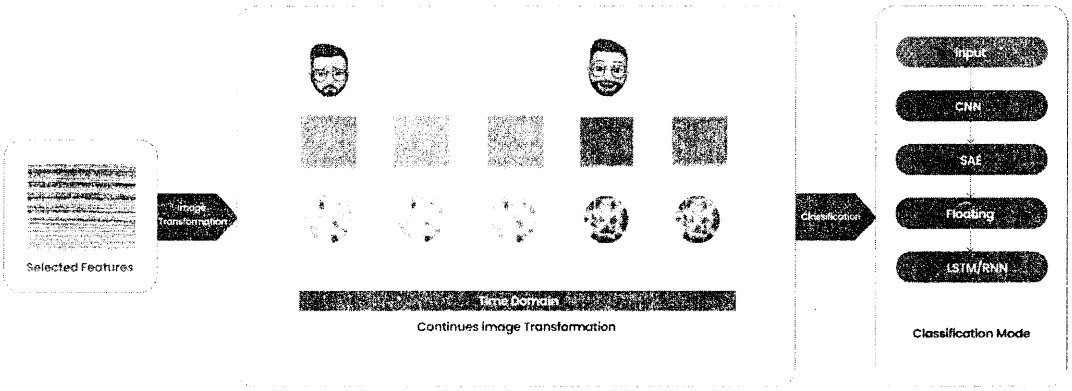


Figure 49: Overall used deep learning model, including data formatting , preparation and overall classification methods structure.

All of the models were trained without the use of a GPU, and the processing was done on a commercial Intel Core i7 CPU. The code was created by making use of well-known libraries like TensorFlow and the Keras API. It is necessary to generate a bias vector, which is then appended to the outputs of each convolutional layer. According to the standard procedure for implementing CNN, we make use of convolution functions that are two-dimensional.

We conduct two 2D convolutional stages in sequence for CNN architecture. Whereas in the first, we used 2D convolutional filters with a length of 20 and a kernel size of (1, 10) and kernel limiting with max normalization to constrain the weights incident to each hidden unit to have a norm less than or equal to 2. We then use average pooling to down-sample the input along its spatial dimensions (height and width) by calculating the average value

for each channel of the input throughout an input window. Along each dimension, the window is moved by (1, 1).

We carried out the second 2D convolutional stages in the same manner as the first ones; however, we used the batch normalizing function to apply normalization to the output. The function utilizes the mean and standard deviation of the most recent batch of inputs to normalize the output it generates. The layer produces a normalized result with a momentum of value 0.9 for the moving average, and this result is returned for each channel that is being normalized.

The batch normalization process involves the application of a transformation that keeps the output means near to the value 0 and the output standard deviation close to the value 1. Batch normalization applies a transformation after it has been called, keeping these values relatively near one another. This occurs when it is called after the layer.

The RELU activation function was utilized in our multilayer CNN to create a link between the upper node's output and the lower node's input. This allowed us to create a more accurate prediction of the data. In point of fact, we put each of the various activation functions—primarily RELU, Sigmoid, Tanh, SELU, and ELU—through their paces. However, RELU demonstrated greater outcomes in comparison to others, with no wander; according to Alexander Craik et al., approximately 70% of the studies on convolutional architecture utilized a ReLU activation (Craik et al., 2019).

The dropout, flatten, and dense layers are applied one after the other throughout the process, and the final step is classification. The purpose of dropout is to lessen the amount of information that is taken in by eliminating an excessive amount of association between features. This is accomplished by dropping the weights at a predetermined probability, which in this instance is 0.5. Dropout does not affect the weights that are associated with it.

The next step that we took was to apply the flattened layer, which helps to reshape the multidimensional output into a size that can be passed on to the Dense layer. As was indicated, the next layer is the dense layer, which is a basic layer of neurons. In this layer, each neuron receives input from all of the neurons in the layer below it, which is why it is referred to as the dense layer.

The dense layer is used to categorize images based on the output from the convolutional network. It is comprised of the hidden layer with neurons that compute the weighted average of their input and pass through a non-linear activation function. The classification will come after our dense layer, which contains a number of inputs equal to three, which is the count of our classes. Last but not least, we made use of the SoftMax activation function, and the primary objective here is to convert the values of the inputs into a probability distribution with an output vector that ranges between 0 and 1 and adds up to 1.

The SoftMax function is typically applied in the role of activation for the very last layer of a classification network. This is due to the fact that the output may be interpreted as a probability distribution.

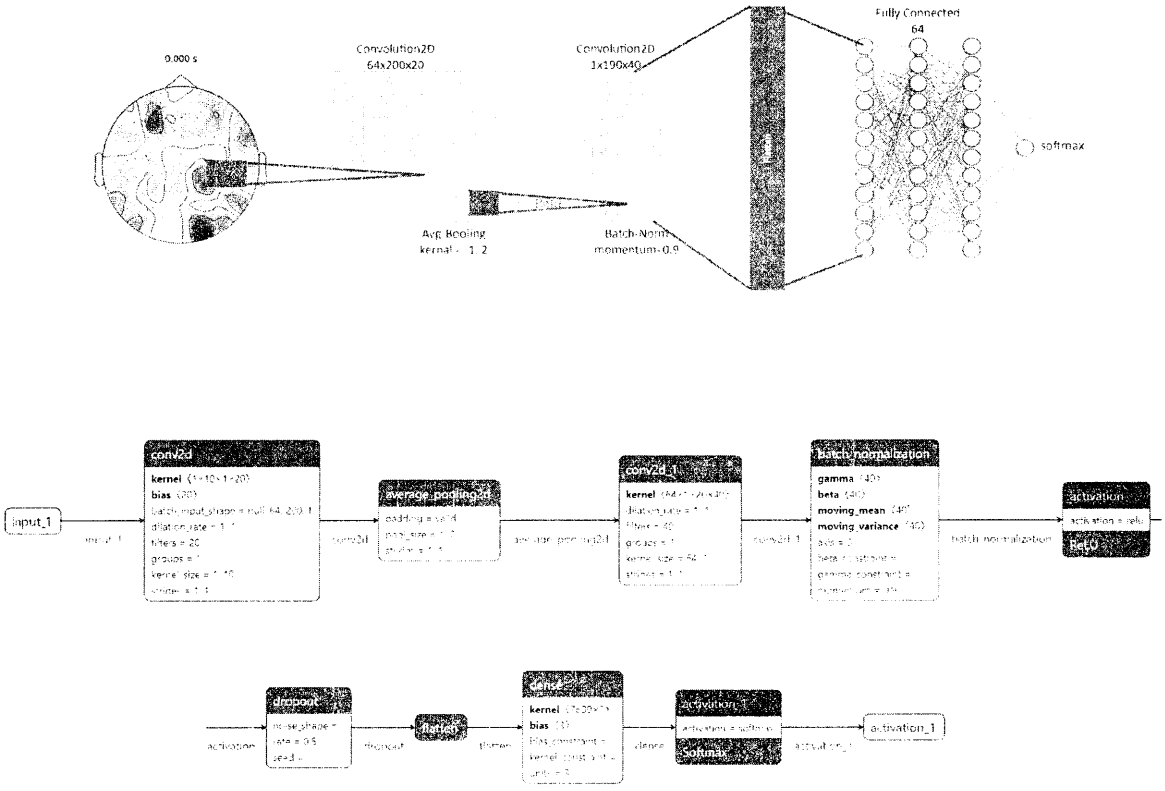


Figure 50: Proposed CCN model.

One advantage of employing an autoencoder is that the encoder learns the compact representation of the input data, and the Decoder decompresses it to recreate the input data.

The encoder of the Convolutional autoencoder is made up of convolutional layers and pooling layers, and it down-samples the input picture while the decoder up-samples it. We previously used down-sampling (in python pooling) on our CNN in the form of average pooling to reduce feature size; now, all that remains is to do up-sampling (known as nun-pooling), which is one of the ways to up-sample the compressed image is by un-pooling (the reverse of pooling) using Nearest-Neighbor, which is the simplest approach to up-sampling. The pixel values are multiplied successively by the kernel weights to generate the up-sampled image.

The kernel weights in up-sampling are taught in the same way that they are in convolutional operations, which is why it is also known as learnable up-sampling. It duplicates a pixel value (response) from the input feature map to all pixels in the associated output sub-region.

The other method of up-sampling is to utilize transpose convolution, which is more difficult to regulate the dimension of features, so we used the nearest neighbor approach in two sequence layers, with a pool size of (1, 2) and (1, 10) in sequence as can be seen in the figure. The convolution operation normally results in down-sampling. The transpose convolution is the reverse of the convolution operation. Here, the kernel is placed over the input image pixels.

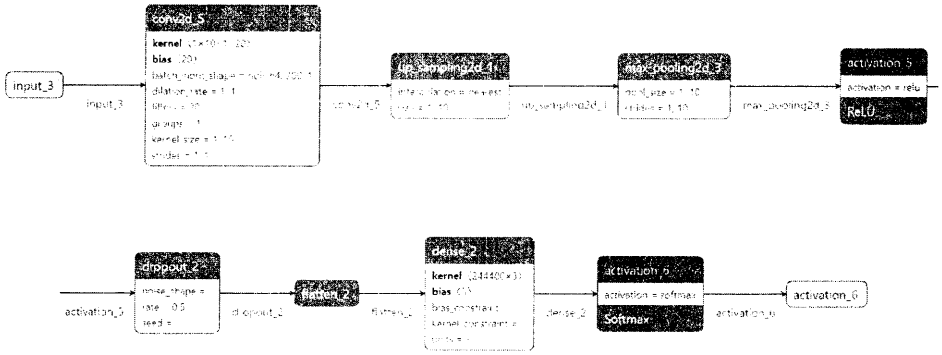


Figure 51: Proposed CCN AE model.

We propose a recurrently time-distributed convolutional neural network with long short-term memory (CNN-LSTM) algorithm for classification of EEG-based data in modern applications. Before, we talked about how we turned EEG signals into 2D before sending them to CNN to extract key features and down-sample and up-sample.

In this stage, we will use LSTM to learn the temporal features and correlations that depend on the time that is needed to find brain activity. The model, as seen in Figure 52, consists of two types of deep learning structures. It combines CNN's huge capacity to process data with a 2D grid-like topology and RNN's ability to handle sequential data, which can be thought of as an extra dimension, to make data in a 3D shape.

Mining intercorrelation and extracting features from two-dimensional frames are the responsibilities of the CNN unit. The Long Short-term Memory (LSTM) unit, which is a redesigned version of the RNN structure, is responsible for simulating the context information for long-term sequences of arbitrary length. When dealing with data that is sequential in two dimensions, this hybrid paradigm works quite well.

The suggested model's average layer (decision layer) is utilized to capture decision information at each time step, which serves as the basis for the trial's final conclusion. Specifically, like with CNN, the final decision is produced by averaging the values in the SoftMax nodes of the decision layer at each time step.

The trial is then classified based on the node with the highest average probability. This technique conforms to our notion that participants' emotional ratings are based on their complete trial experience. Because the weights of the time-distributed CNN are linked across time, this time-series model can also be thought of as having a single CNN. The size of the convolutional filter was chosen to look for correlations across channels and scales.

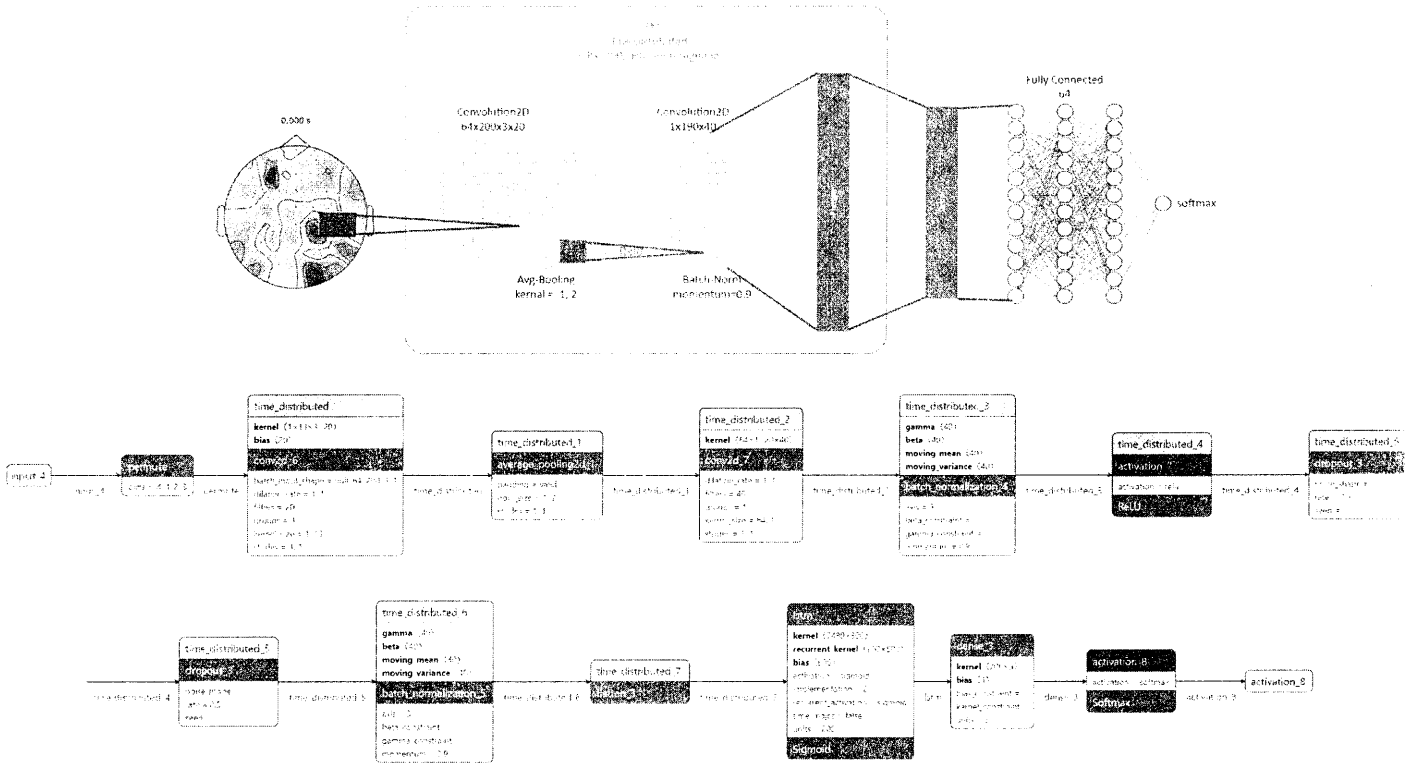


Figure 52: Proposed CCN LSTM model.

3.2.3. Classifiers Learning Transfer

The human emotion is the term we use for the person's overall feelings and actions.. In addition to this, it frequently reflects the opinions and attitudes of human beings. In addition to this, it always plays an important part in our day-to-day lives, particularly in the interactions between people. When it comes to human-machine contact, emotion recognition based on a computer system has also developed into an essential component of the most modern brain-machine interface system (Mattar et al., 2019).

Storing and Loading Knowledge

In the few chapters before this one, we went over the evaluation of each area of enhancement separately. However, when we were talking about online and real-time applications, we saw that modern deep-learning neural networks had the upper hand over other methods. This is because DNNs gather the ability to be generalized, simplicity of knowledge transfer, and reduction of additional manual work such as feature selection and extraction.

There is a widespread perception that one of the primary ways in which humans may rapidly acquire new, complicated ideas with relatively little exposure to training sets is through the process of knowledge transfer. In this section, we will demonstrate the knowledge transfer process using deep neural networks, which we consider particularly effective.

Knowledge Transfer Impact

Our first findings indicate that the components of a trained convolutional and recurrent neural network have the ability to constructively transmit information between various persons who share the same or distinct backgrounds or personalities. In addition, one of the most important aspects to consider when determining the effectiveness of the knowledge transfer process is the capacity to solve real-time classification problems.

This task can be accomplished by reducing the amount of time spent in training, increasing the amount of time spent waiting before the system is ready to use, and taking advantage of the globalization factor. The studies that we ran entailed initially training our CNN and RNN models with a huge training set that included up to five minutes of the signal stream. After that, we evaluate the established model by fitting a modest quantity of data, usually between 30- and 60-seconds worth of the signal

stream. When presented with limited training sets, our findings indicate that relying on previously acquired information to learn new tasks can be quite beneficial, provided that the new activities are sufficiently analogous to the previously mastered ones. However, as the quantity of training sets increases, this benefit becomes less significant.

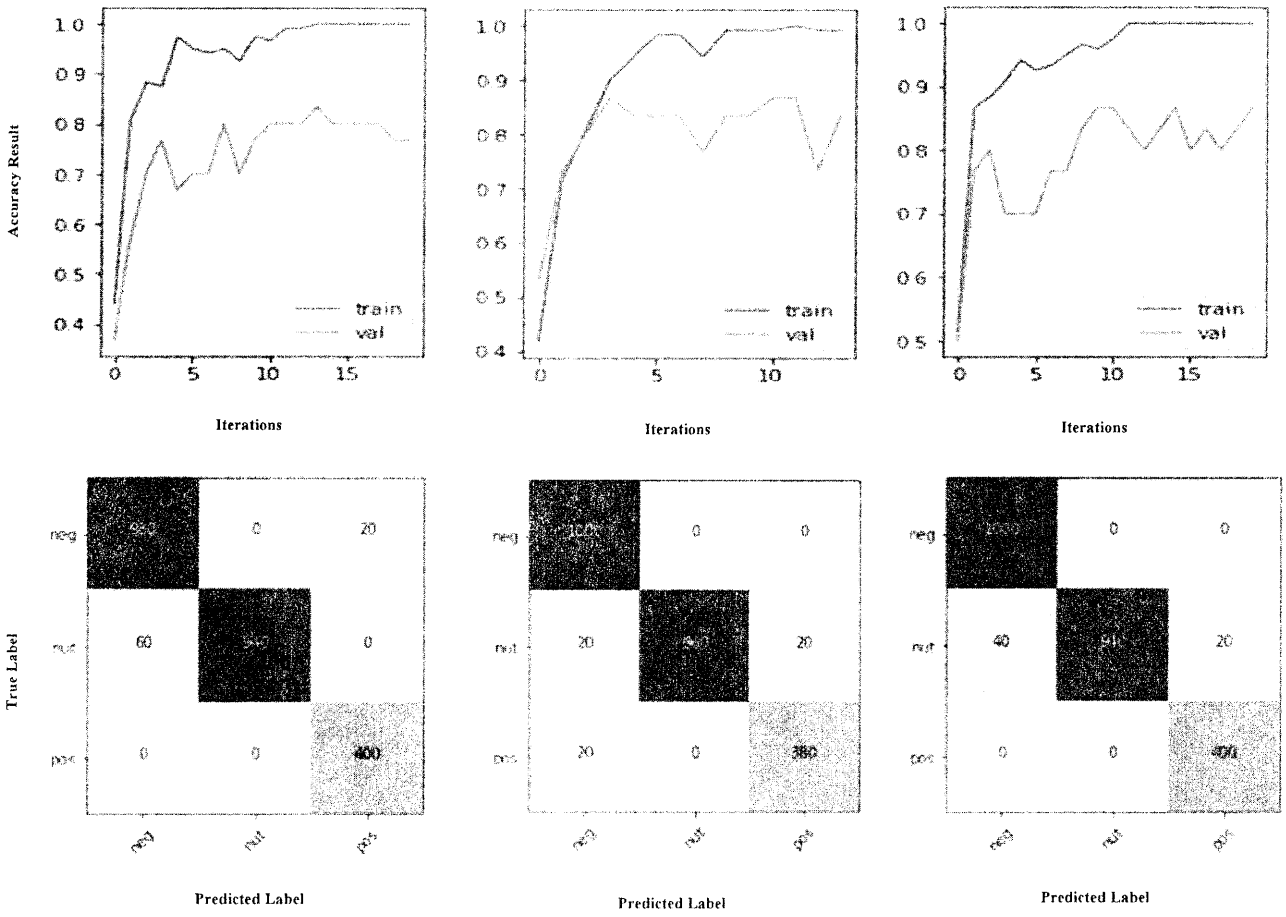


Figure 53: CNN (IMAGE) classification accuracy in real-time micro training (1-2 minutes). in the left without knowledge transfer, in the middle with applying for knowledge transfer from different experiments, and finally, on the right with applying for knowledge transfer between inter-subjects.

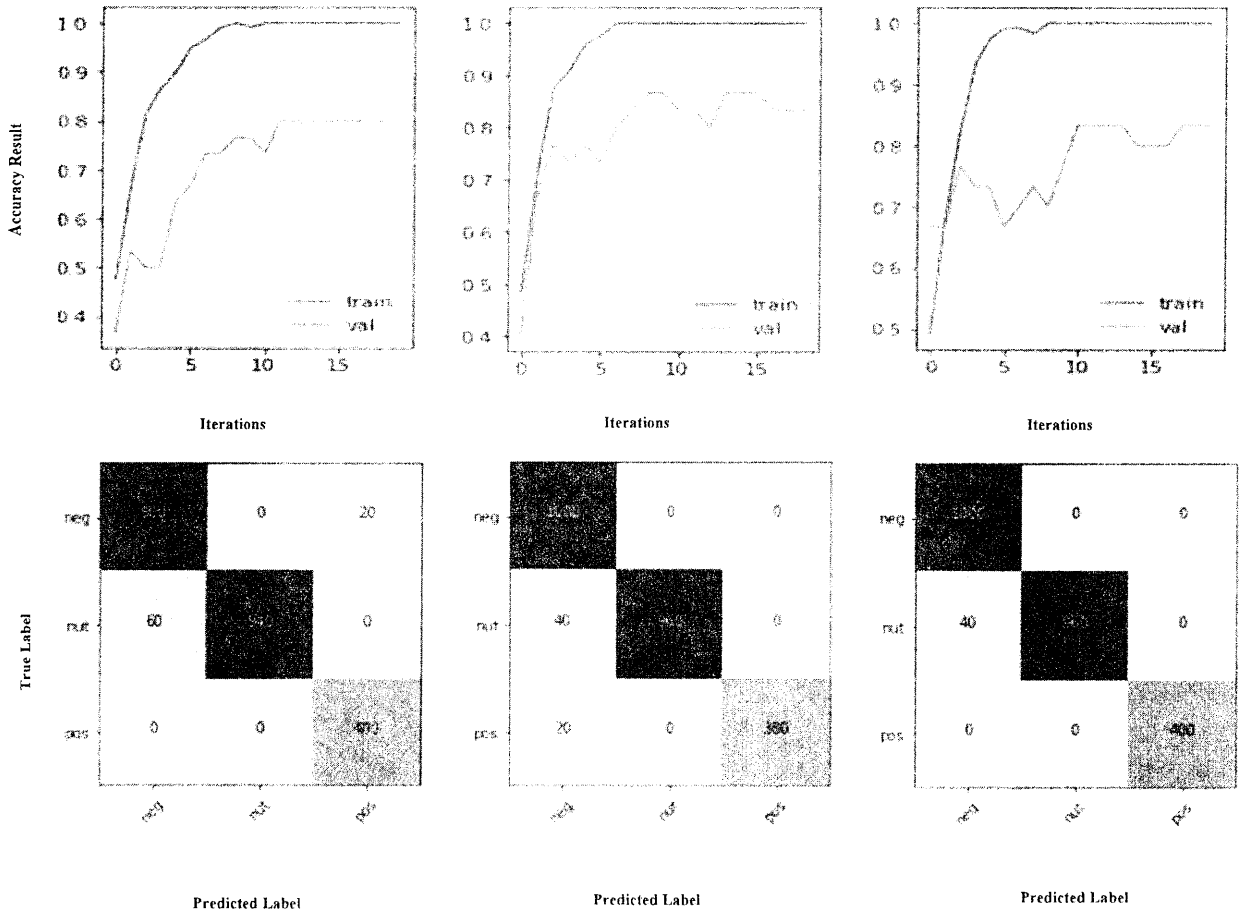


Figure 54: RNN (IMAGE) classification accuracy in real-time micro training (1-2 minutes). in the left without knowledge transfer, in the middle with applying for knowledge transfer from different experiments, and finally, on the right with applying for knowledge transfer between inter-subjects.

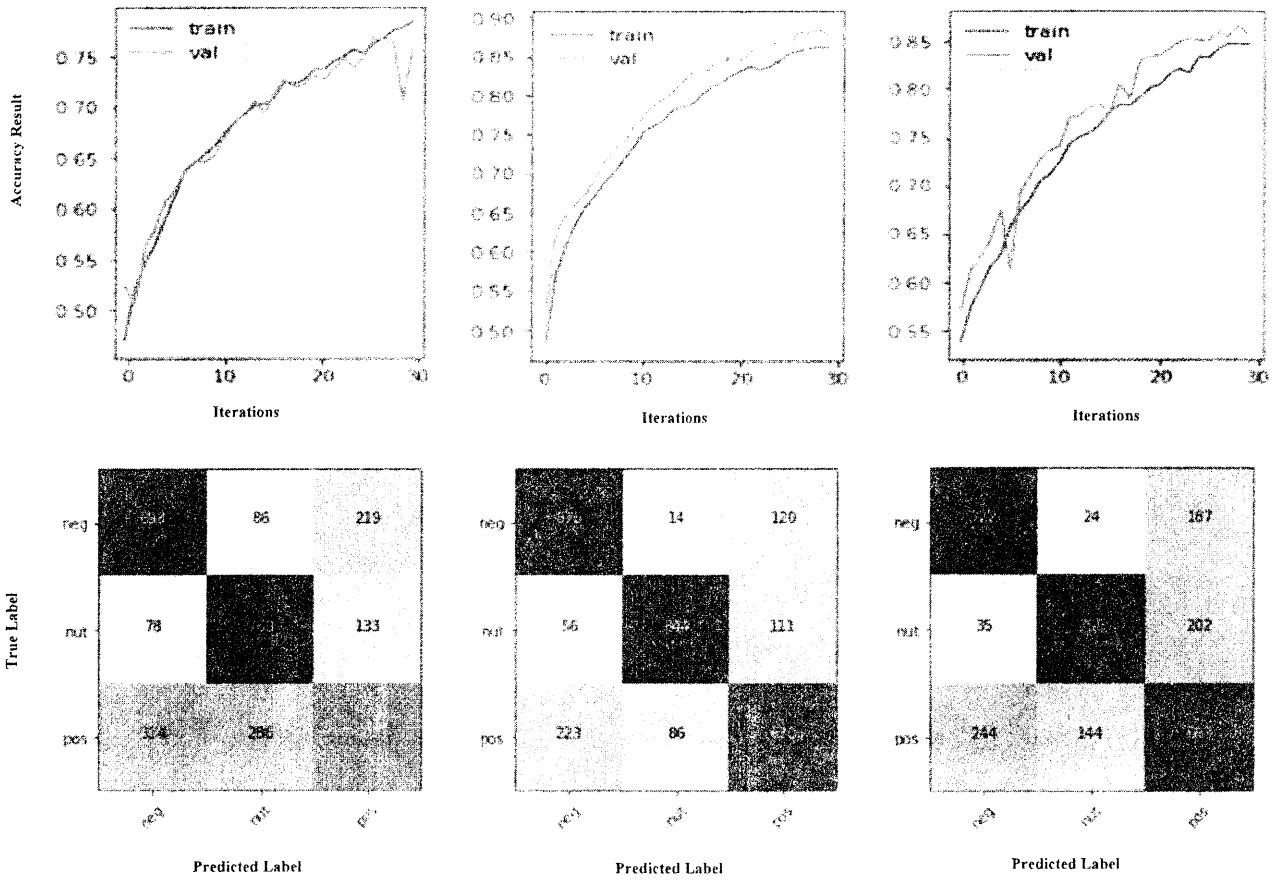


Figure 55: CNN (Signal Flow) classification accuracy in real-time micro training (1-2 minutes). in the left without knowledge transfer, in the middle with applying for knowledge transfer from different experiments, and finally, on the right with applying for knowledge transfer between inter-subjects.

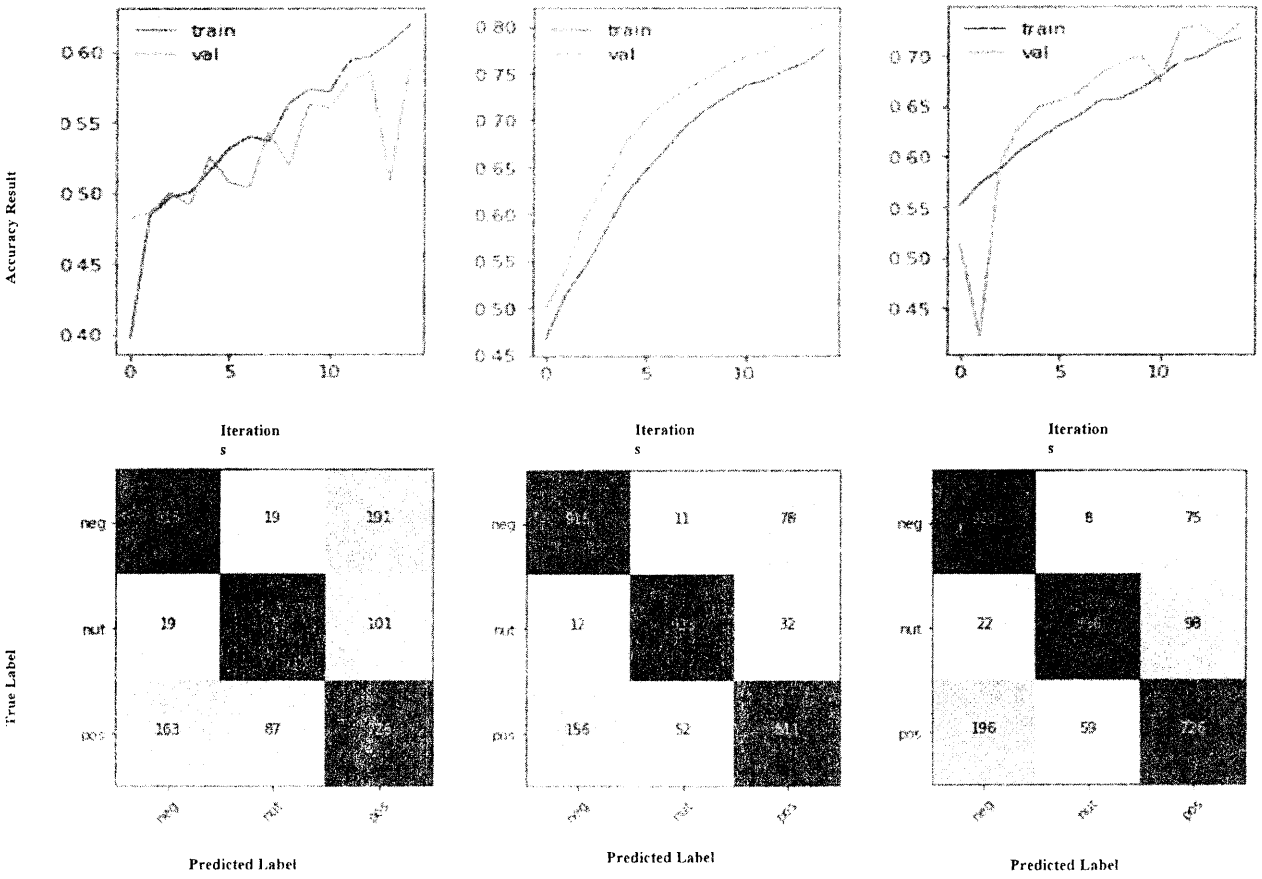


Figure 56: RNN (Signal Flow) classification accuracy in real-time micro training (1-2 minutes). in the left without knowledge transfer, in the middle with applying for knowledge transfer from different experiments, and finally, on the right with applying for knowledge transfer between inter-subjects.

The real-time nature application calls for nearly instantaneous categorization and encoding operations, which limits the amount of time and data that can be taken into consideration for the purpose of computing system efficiency effectively. In this location, we keep the training to a duration of less than one minute. Figures 1, 2, 3, and 4 make it abundantly evident that the accuracy of the classifier is improved for micro-training by employing the accumulated knowledge of a topic that is analogous to the one being trained on, as well as by bringing together subjects from throughout the system.

Chapter Four:
Results and Discussion

Emotion is a state that includes all a person's feelings, thoughts, and actions. It is an integral part of how people interact with one another. Emotion classification aims to automatically distinguish between different emotional states using physiological and non-physiological signals collected from humans in order to facilitate efficient communication and interaction between humans and machines. One of the most popular and practical approaches to emotion estimation is Brainwaves-Based Emotion Estimation. Technology is vital in many research areas, including the treatment of emotional disorders, the brain-computer interface for people with disabilities, entertainment, and many others. This thesis presents improved methods, schemes, and solid frameworks for estimating human emotions based on Electroencephalogram (EEG).

4.1. Results

4.1.1. Signal Processing

When dealing with a real-time application, making data flow instantly available plays a major role in reaching the goals of the application; therefore, minimizing processing delay and optimizing transformed data from the terminal unit (EEG cap) to the centralized processing unit not only minimizes event latency but also extends battery life before needing to recharge.

To be able to demonstrate the impact of channel counts on feature extraction processing time, an experimental application has been coded using MATLAB to record the time needed to extract features for different groups of EEG channels (1, 5, 20, and all channels), in addition, to loading different amount of batched data loads (that could have happened in case of terminal lost internet connection for a while).

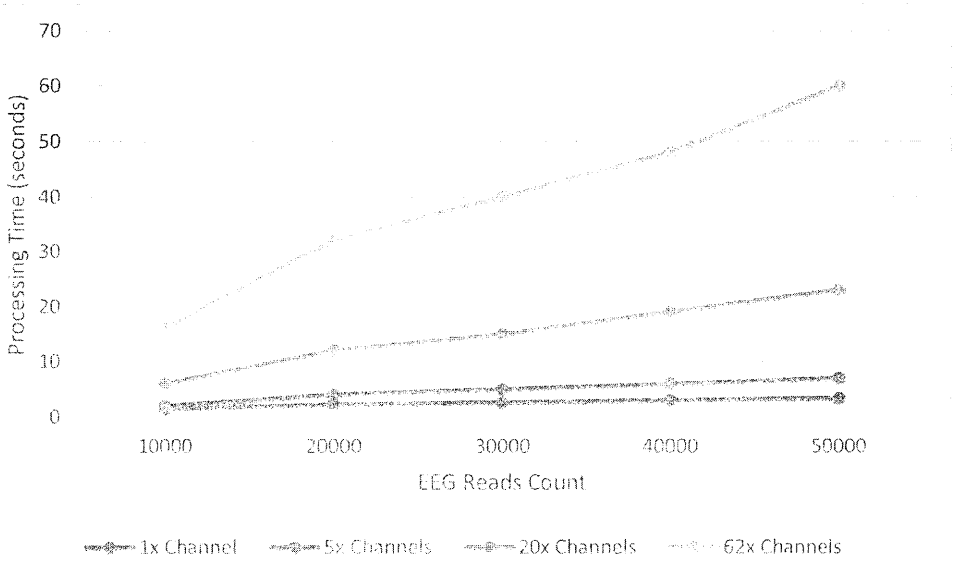


Figure 57: Feature extraction time based on number of channels counts.

As shown in the Figure 57 above, the feature extraction process for simple statistical features dropped significantly with reduced channel counts, which impacted battery consumption caused by processing, storing, and transmitting data.

When looking at the final system accuracy using simple classification methods like the decision tree and the support vector

machine (SVM) algorithm, we can see in the figure that the accuracy impact is minimal in the case of the decision tree.

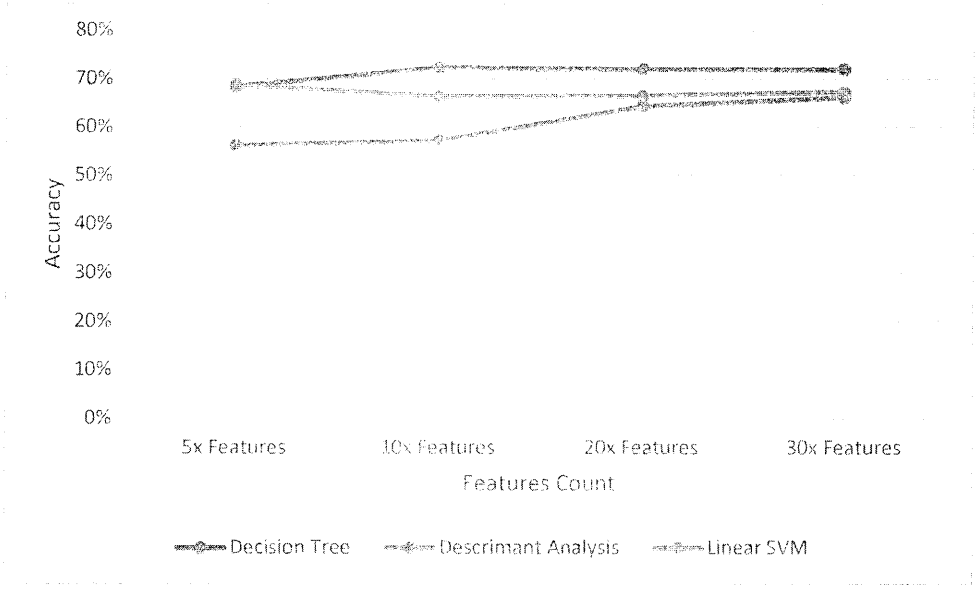


Figure 58: Accuracy of sample classification methods based on number of channels.

We found a surprising result in Figure 58. The support vector machine, which is a machine learning model that can sort data into different groups, performed worse with more channels. Channels are the features that the model uses to make a decision. For example, if the data is EEG signals, channels are the sensors that record the brain activity. More channels usually mean more information and better performance. But here, the opposite was true. The support vector machine became less correct with more channels. Correctness is how often the model predicts correctly. The correctness decreased by two percentage points, which means it was two percent less right with more channels than with fewer channels.

4.1.2. Feature Processing

Consequently, we express samples of the selected and extracted features, such as standard deviations, different degrees of moments, and other statistical features in (Figure 59, Figure 60, Figure 61, Figure 62), on the power-time domain, with the results averaged by the fixed window.

We enabled feature evaluation utilizing unsupervised learning methods after each attempt of feature extraction to raise variations between the feature results of each class.

After each try of feature extraction, we enabled feature evaluation using unsupervised learning methods to increase variations between the feature outputs of each class.

As the plots show, we may perform a visual examination of them, where greater statistical computed features with higher windows result in higher variances and represent seniority to select over other features.

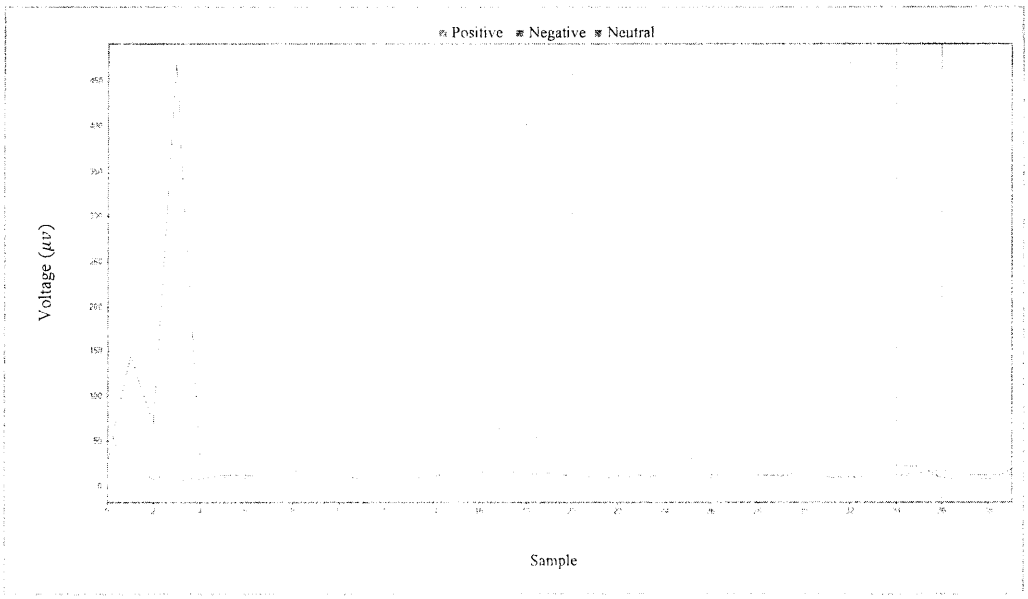


Figure 59: Example plot of averaged standard deviation feature of 1,000 reads on 200 Hz.

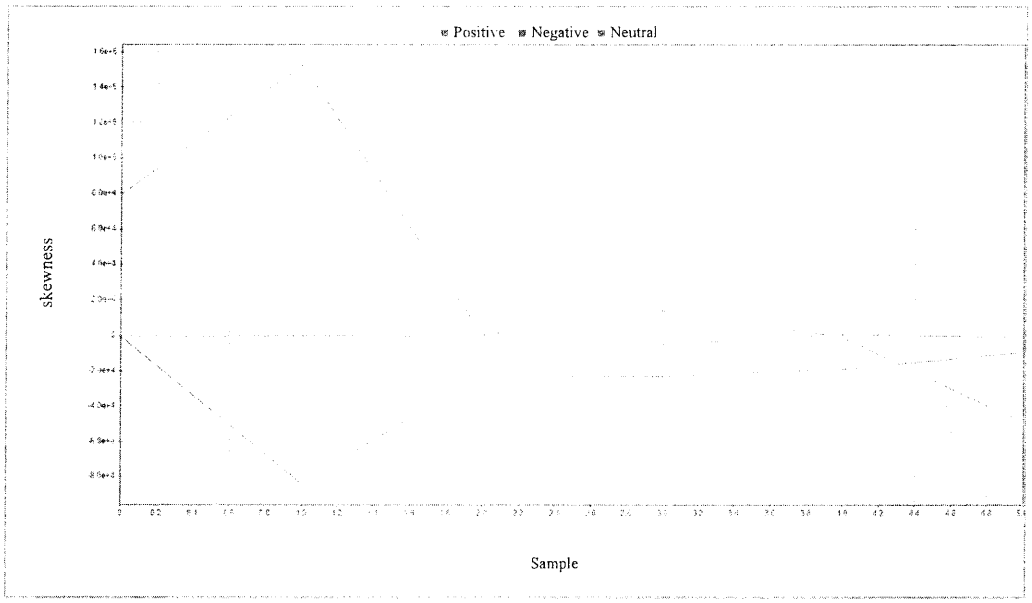


Figure 60: Example plot of averaged third order moments (skewness) feature of 5000 reads on 200 Hz.

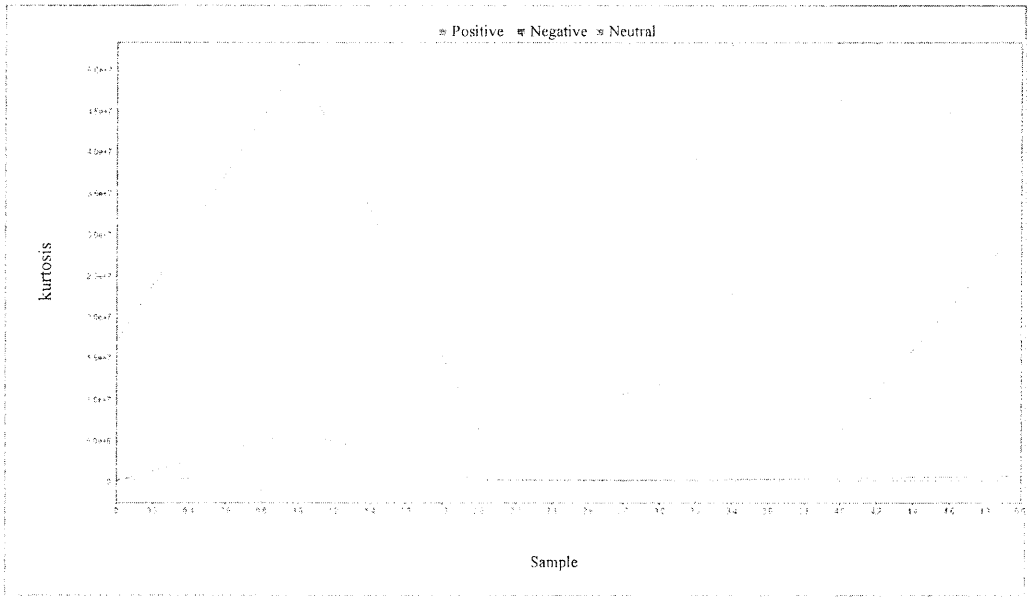


Figure 61: Example plot of averaged fourth order moments (kurtosis) feature of 5000 reads on 200 Hz.

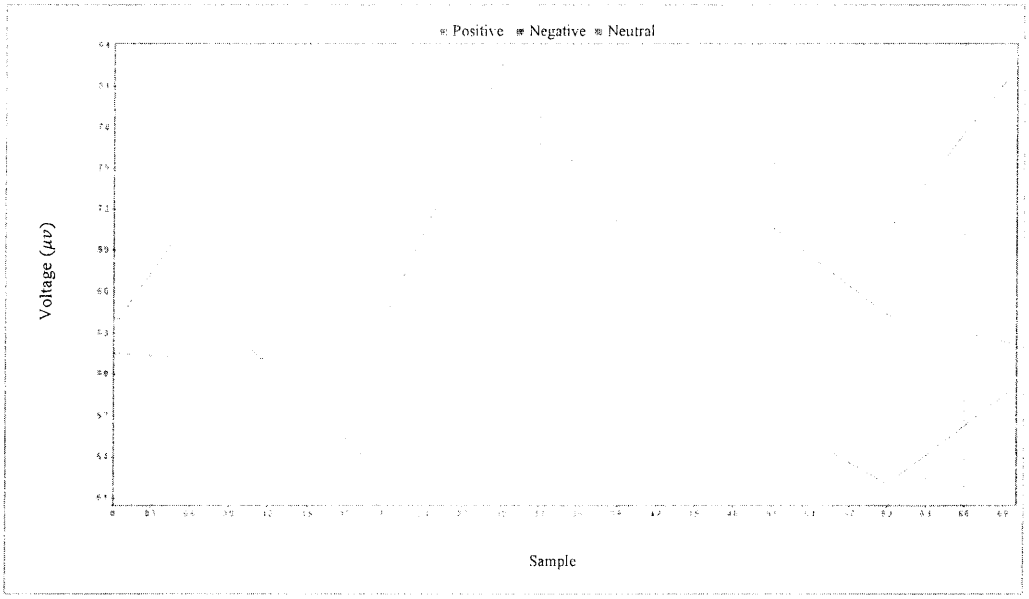


Figure 62: Example plot of absolute second difference feature of 5000 reads on 200 Hz.

Feature selection activities are usually done in the early stages, which is proposed to be done in two stages; the first one is in the terminal end, where simpler features can be extracted and selected, which requires fewer computation resources in purpose to evaluate MATLAB application in equivalent systems for both terminal and server devices.

To complete this simulation, we set CPU as our system's standard, ARM CPU, and Intel CPU for distributed mobile devices and centralized devices, respectively. We used it for server Intel Core i7 and terminal ARM Cortex-A53; as can be seen in Figure **63** to Figure **65**, the processing time increased exponentially, and the time also increased aggressively with increasing the number of channels or data size.

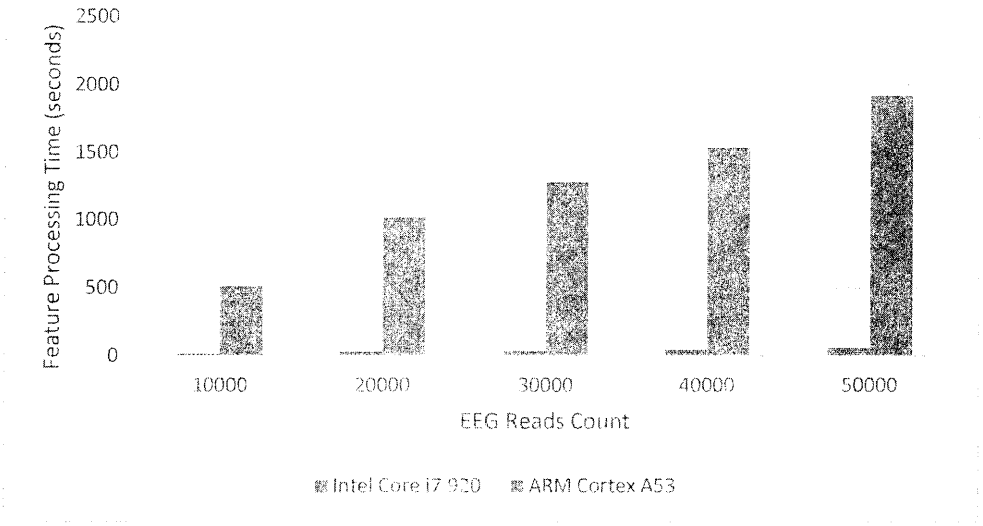


Figure 63: Computation time of entropy feature in distributed mode (Intel Core i7 920) and centralized mode (ARM Cortex A53).

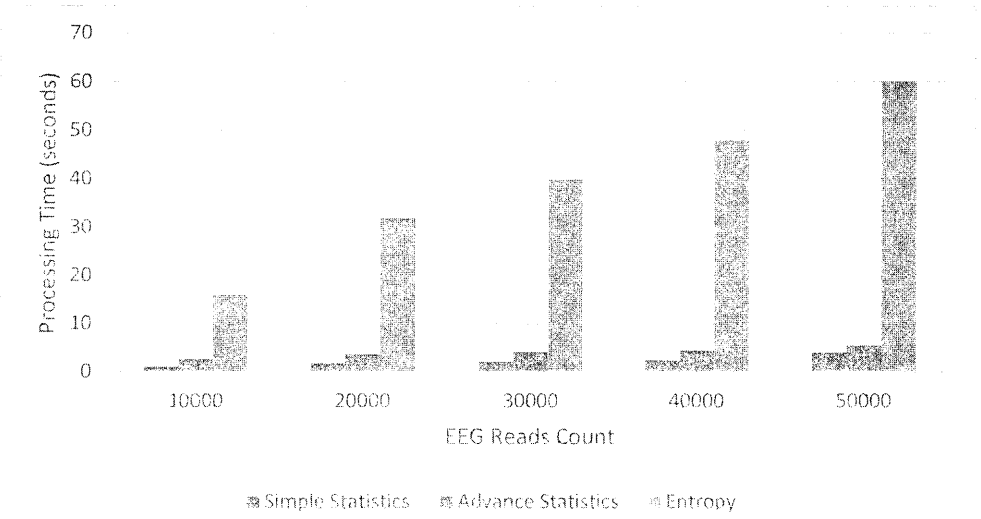


Figure 64: Computation time in centralized mode (ARM Cortex A53) for different type of features.

Figure **65** demonstrates that the amount of time spent calculating on the terminal for more complex feature extractions, such as entropy, is significantly more than that required for more straightforward statistical feature extraction.

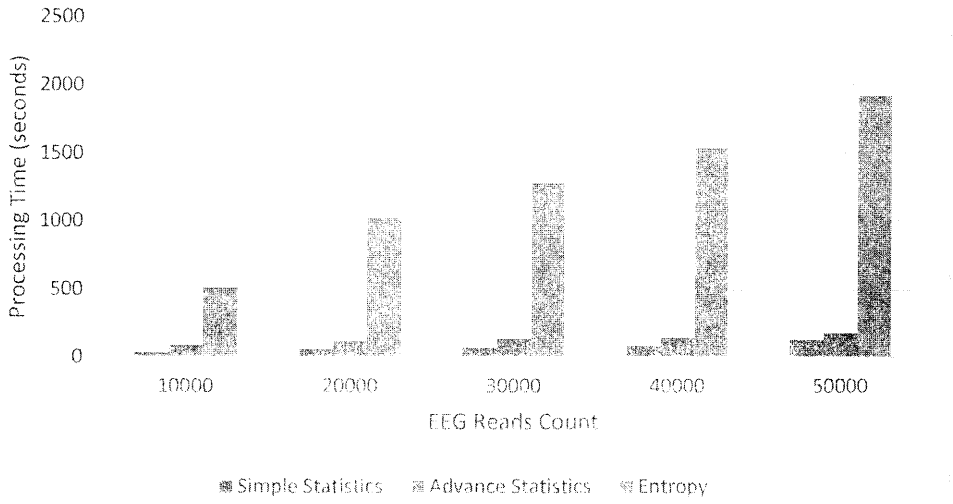


Figure 65: Computation time decentralized mode (ARM Cortex A53) for different type of features.

However, on the other hand, as seen in Figure **66** and Figure **67**, processing feature extraction at the edge saves transmission and saves battery consumption for transmission; however, we can observe in all of those figures that with the proposed adaptive technique, the delay associated with feature extraction is very close to that of the centralized technique, and bandwidth usage is very close to that of the distributed mode. As a result, it makes use of both modes' strengths while avoiding the drawbacks of each.

The simulation results show that processing feature extraction at the edge can save transmission and battery consumption, while the proposed adaptive technique can effectively balance the delay and bandwidth usage between centralized and distributed modes, making use of their strengths and avoiding their drawbacks. This approach can potentially improve the overall performance of the system in terms of efficiency and reliability.

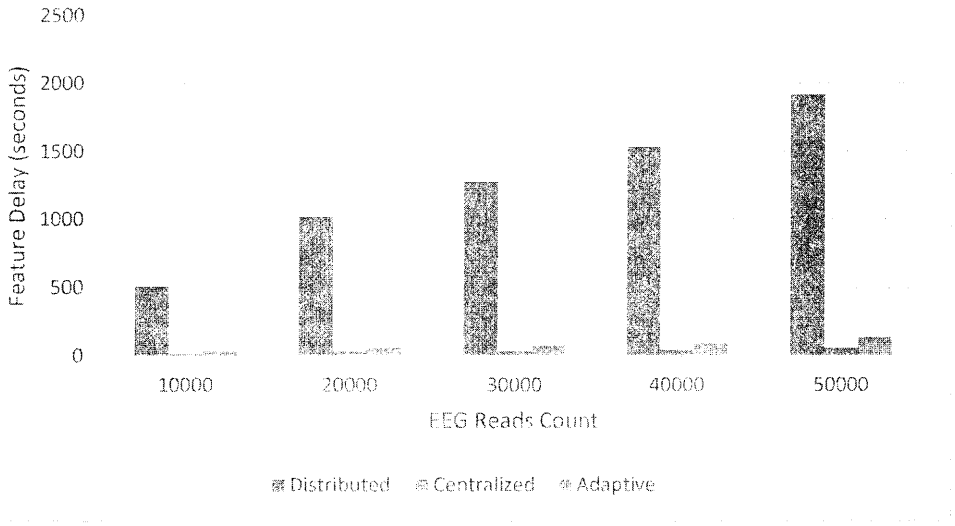


Figure 66: Estimated delay of feature extraction and getting ready for classification in different processing modes.

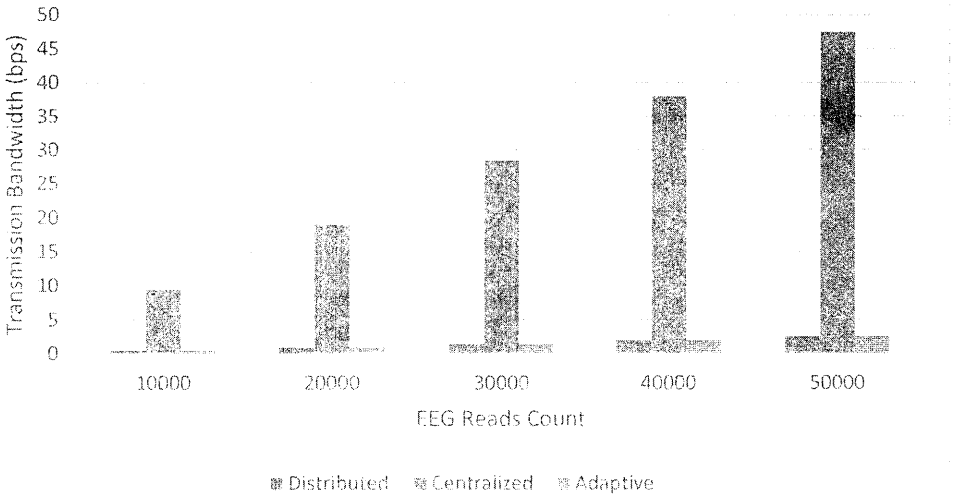


Figure 67: Estimated bandwidth usage of features data transformation to the system for classification in different processing modes.

4.1.3. Classifications Methods

Emotional detection is one of the more contemporary EEG classification tasks. It lacks the same regions of concentration and motor-based detection.

Despite the fact that it is advantageous to analyze the emotional condition of the subjects, especially given the roughly 25% error probability of motor-based features, motor-based features are relatively comparable in the majority of features. (Kevric & Subasi, 2017)

To prevent overfitting issues, we tested and compared the classification effectiveness of multiple machine learning models with differing numbers and types of information.

The model's classification performance is measured by accuracy, sensitivity, and specificity. Accuracy is the fraction of correctly classified samples as in equation (39). Sensitivity is the fraction of correctly classified positive samples in equation (40). Specificity is the fraction of correctly classified negative samples see equation (41). *TP*, *TN*, *FN*, and *FP* are true positive, true negative, false negative, and false positive respectively.

$$Accuracy = \frac{TP + TN}{Total\ Samples} \tag{39}$$

$$Sensitivity = \frac{TP}{TP + FN} \tag{40}$$

$$Specificity = \frac{TN}{TN + FP} \tag{41}$$

We evaluated the classification methods with different channel quantities, i.e., 5, 10, 20, 30, 60 channels where we pick out of the extracted features directly, because one of the main goals of this research is to improve the EEG classification performance, especially for new applications that added more restrictions to the speed and density of the data stream of the signal reads. We also tested the same classification algorithms with similar characteristics, but these features were reduced using PCA to define channel amounts of 5, 10, 20, and 30.

Two things leap out when we look at classical statistical measures like standard deviations and absolute second differences in Figure 68 to Figure 73. First, with a greater number of features, most classes achieve slightly better results, with a maximum accuracy of less than 80%. Second, PCA-based features perform better with fewer extracted features, making them an effective technique for lowering the number of features on network edges.

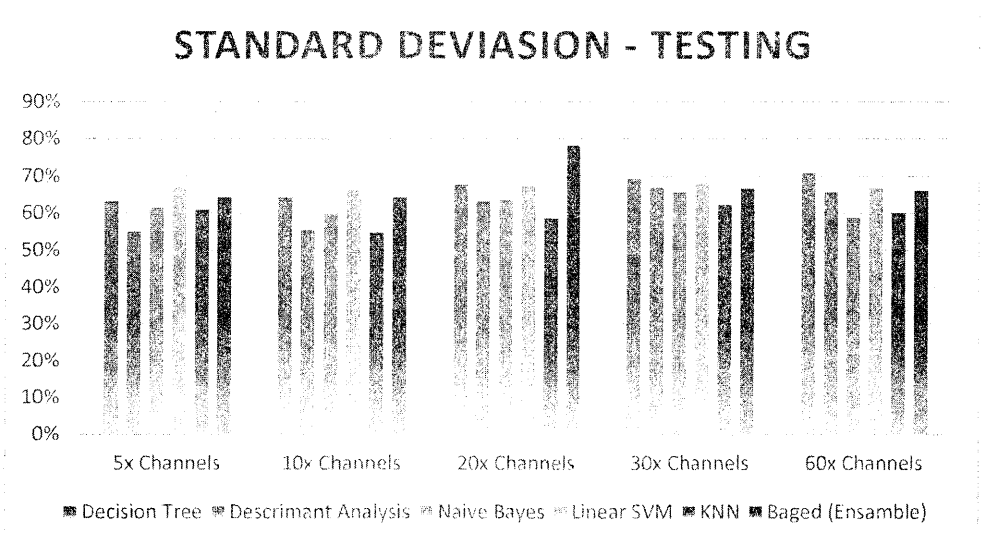


Figure 68: The categorization algorithms were tested on different counts of standard deviation features.

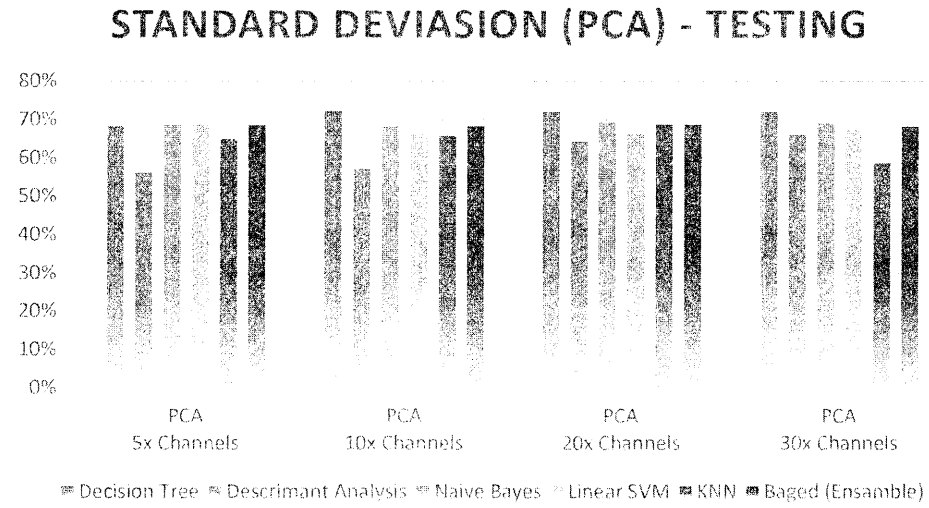


Figure 69: The categorization algorithms were tested on different counts of PCA generated features those are based on standard deviation features.

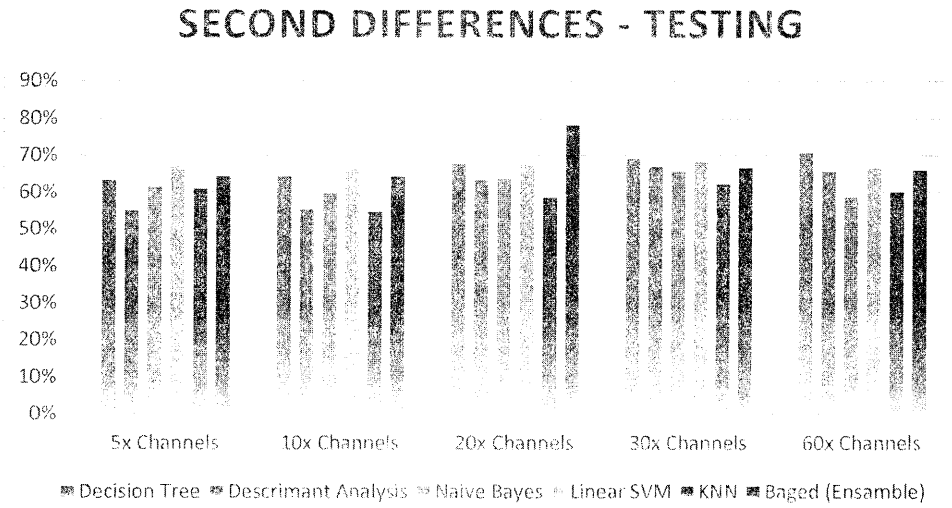


Figure 70: The categorization algorithms were tested on different counts of 2nd differences features.

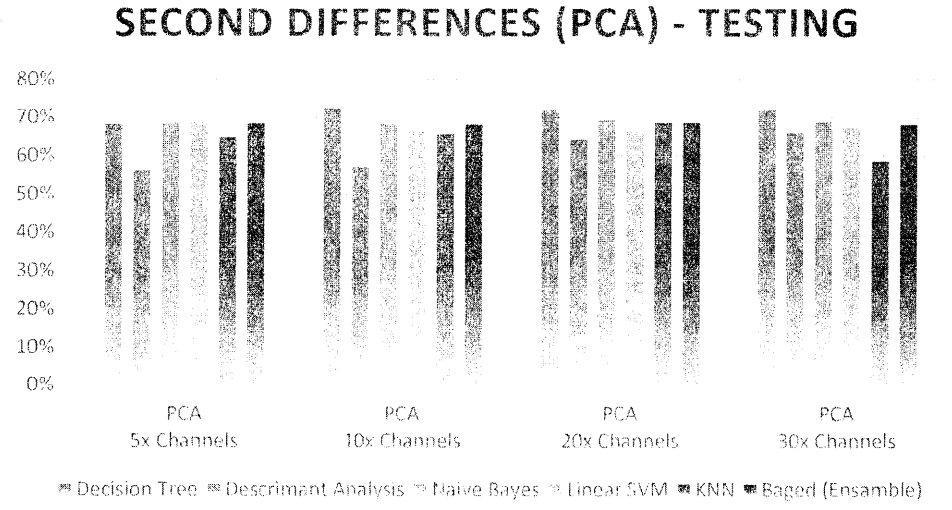


Figure 71: The categorization algorithms were tested on different counts of PCA generated features those are based on 2nd differences features.

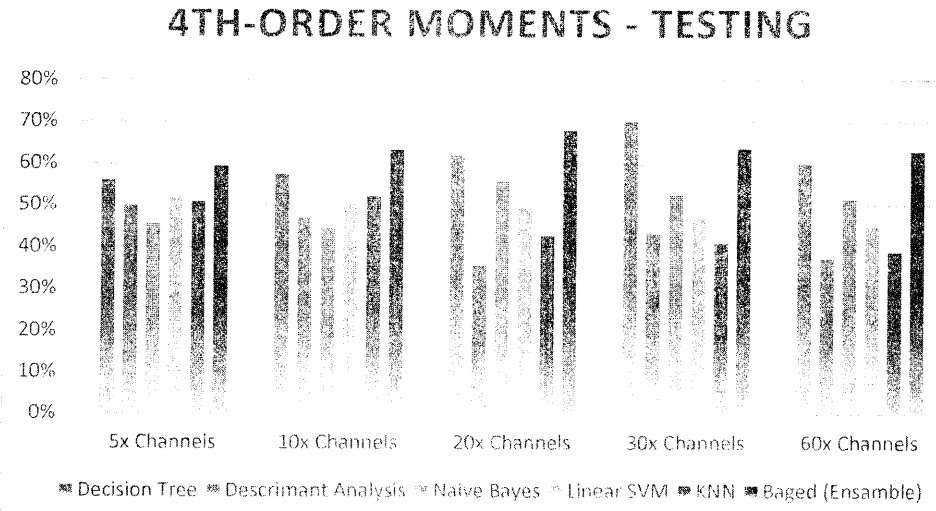


Figure 72: The categorization algorithms were tested on different counts of 4th-order moments features.

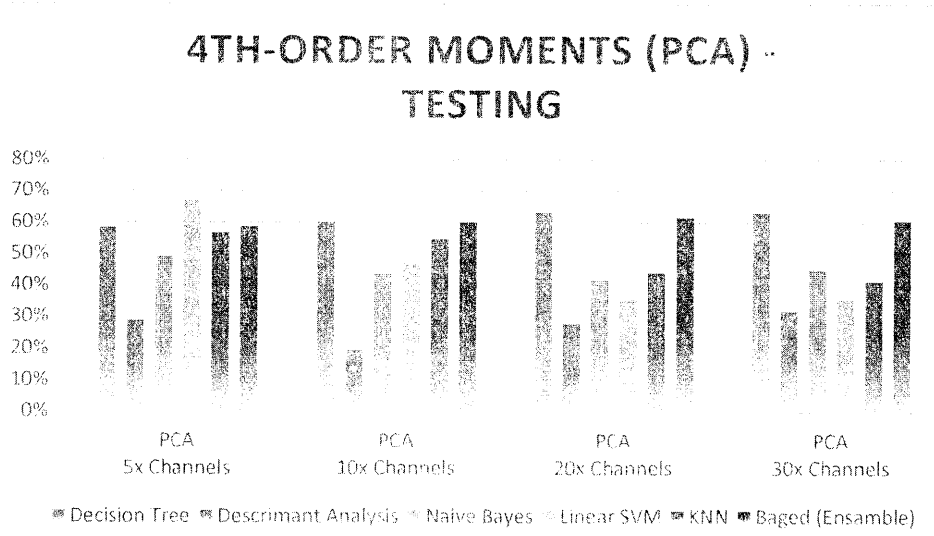


Figure 73: The categorization algorithms were tested on different counts of PCA generated features those are based on 4th-order moments features.

Let us look at higher statistical computed features such as Shannon entropy and differential entropy in Figure 74 to Figure 77, where we can notice greater results than the previous group, with accuracy above 80%. However, similar to the last group of features, we see that employing a higher number of features

improves results. Similarly, PCA-based features perform better when dealing with fewer features.

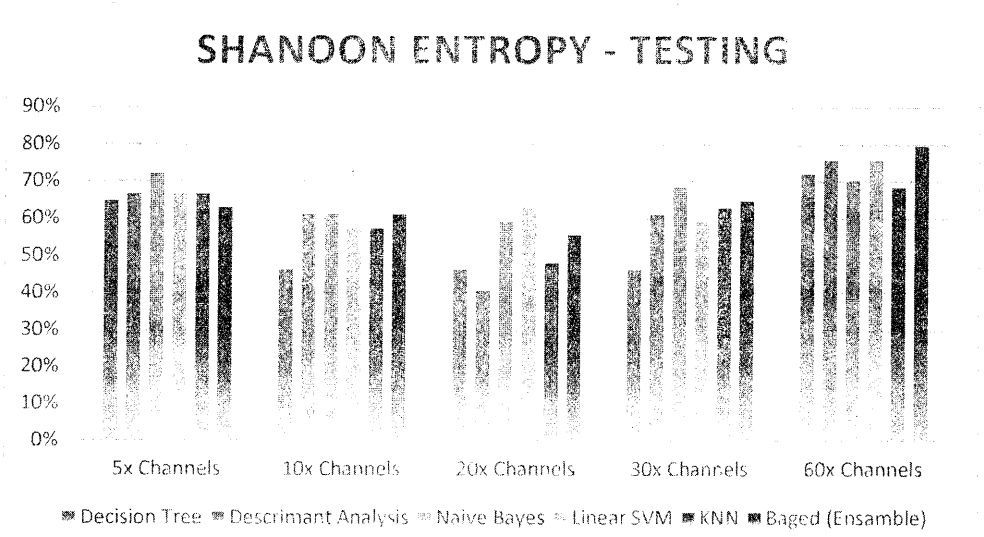


Figure 74: The categorization algorithms were tested on different counts of Shanoon entropy features.

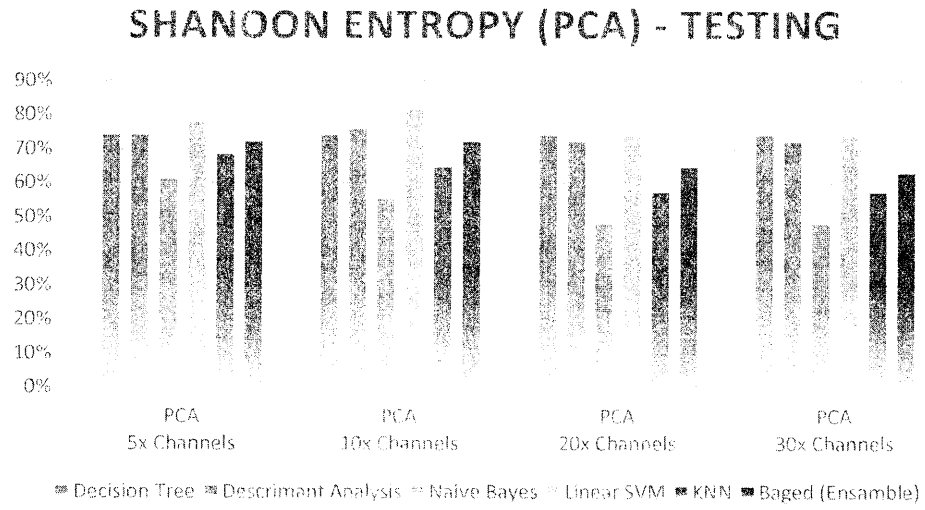


Figure 75: The categorization algorithms were tested on different counts of PCA generated features those are based on Shanoon entropy features.

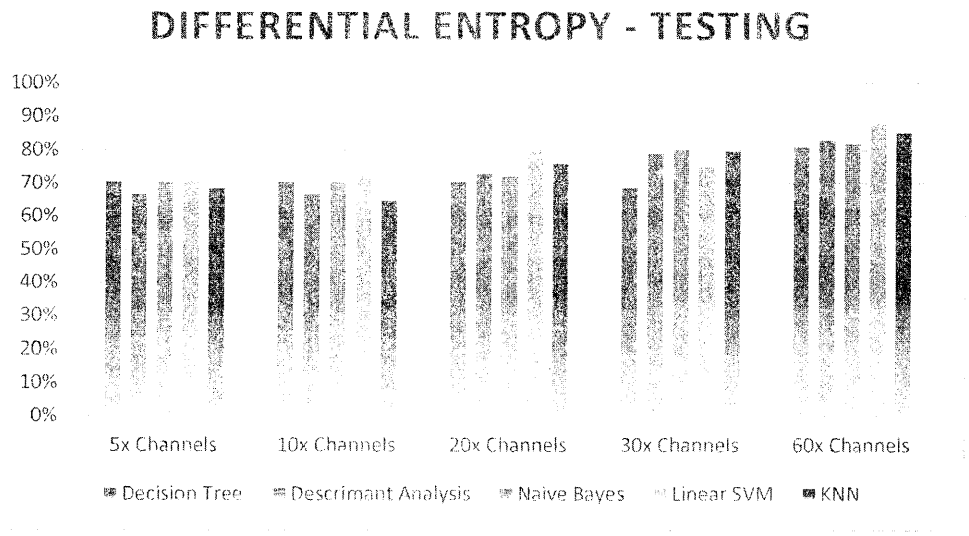


Figure 76: The categorization algorithms were tested on different counts of differential entropy features.

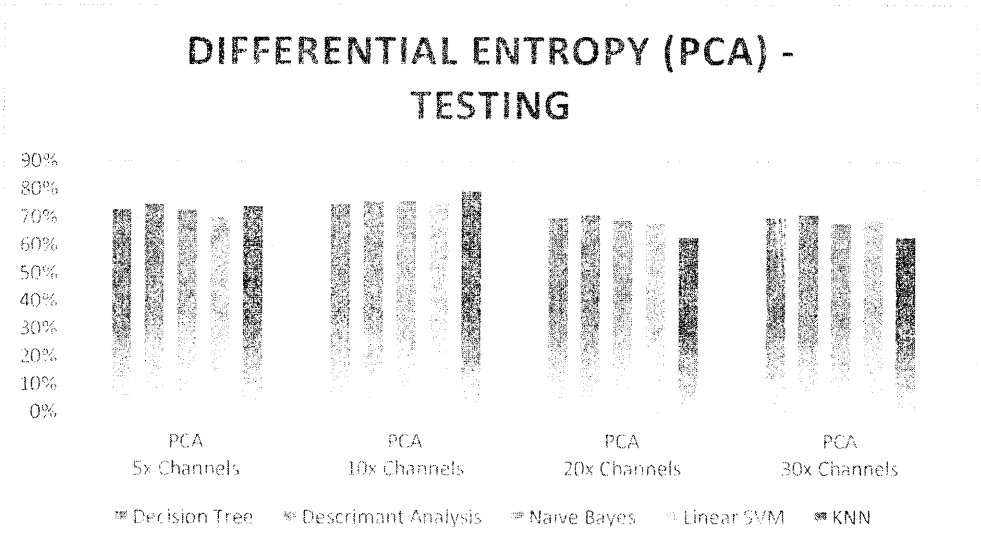


Figure 77: The categorization algorithms were tested on different counts of PCA generated features those are based on differential entropy features.

4.1.4. Convolutional Neural Networks

We started the preparation of the CNN model by training the compiled classification model with training and validation data at a rate of 80/20 in sequence. as we can see in the accuracy plot, the model started with low training and validation accuracy, and it kept enhancing while repeating the training during selected epochs, the accuracy reaches to level 0.8 before the training early stopped at 18 epochs.

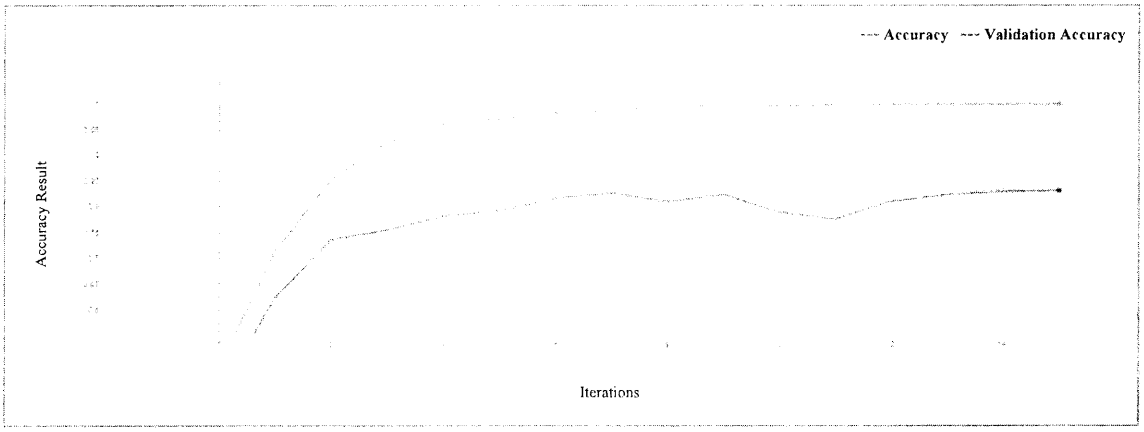


Figure 78: Proposed CNN accuracy result; the x-axis displays the accuracy per iteration, the orange line represents the training results, and the blue line represents the validation accuracy.

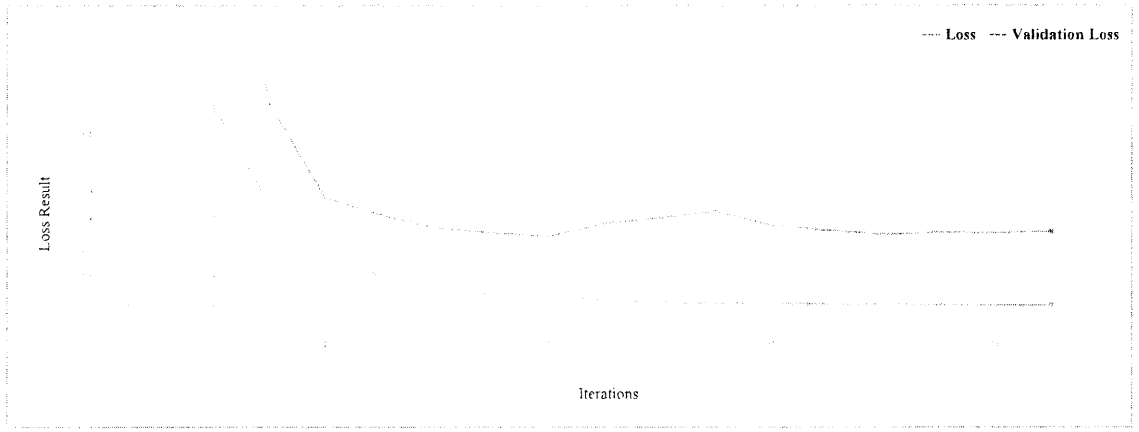


Figure 79: Proposed CNN loss result; the x-axis displays the loss per iteration, the orange line represents the training results, and the blue line represents the validation accuracy.

The result of evaluating the trained classification model with new data is that each row of the confusion matrix represents the target class, and each column represents the predicted class that a classifier outputs. This is because each row of the confusion matrix corresponds to a different column in the confusion matrix.

The percentage of samples from class i that were determined to belong to class j is denoted by the element (i, j). According to the findings from using the proposed CNN classifier, it is relatively easy to identify positive and negative emotions, however neutral emotions are the ones that are the most challenging to identify. They are unable to separate positive and negative emotions from neutral ones, and they do not understand how to place negative feelings in a positive category.

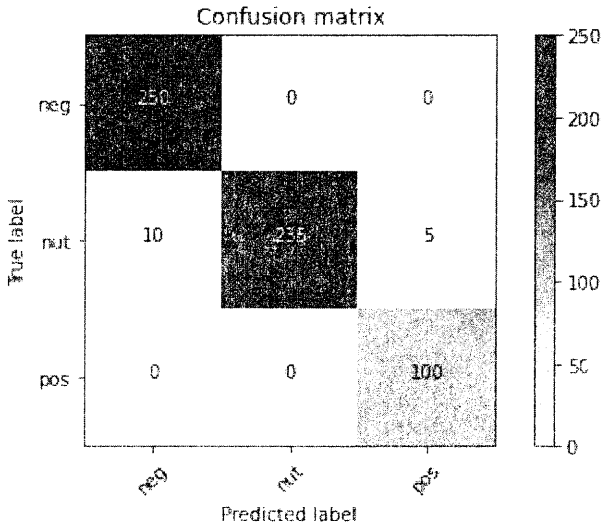


Figure 80: Trained classification model is tested with new data, each row of the confusion matrix represents the target class, and each column represents the predicted class that a classifier outputs.

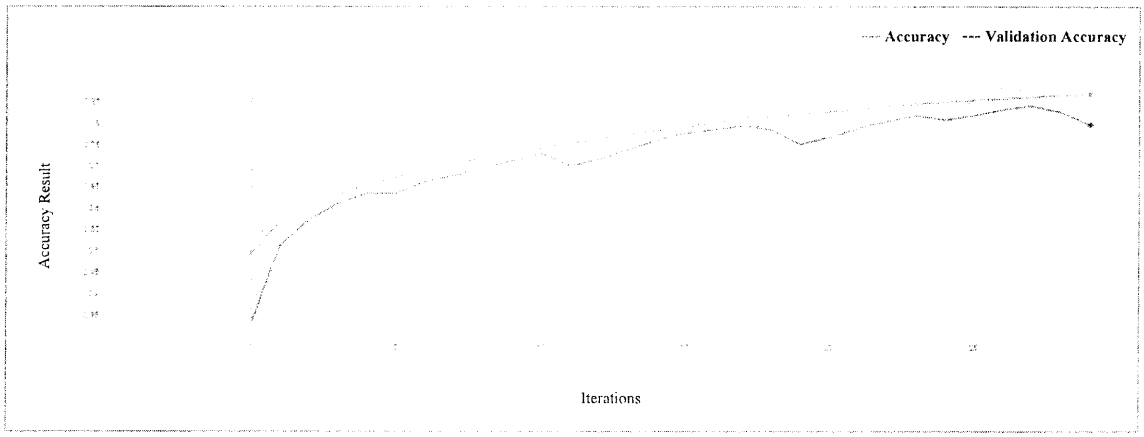


Figure 81: Proposed CNN accuracy result; the x-axis displays the accuracy per iteration, here we used continuous raw features as an input to the model instead of images, the orange line represents the training results, and the blue line represents the validation accuracy.

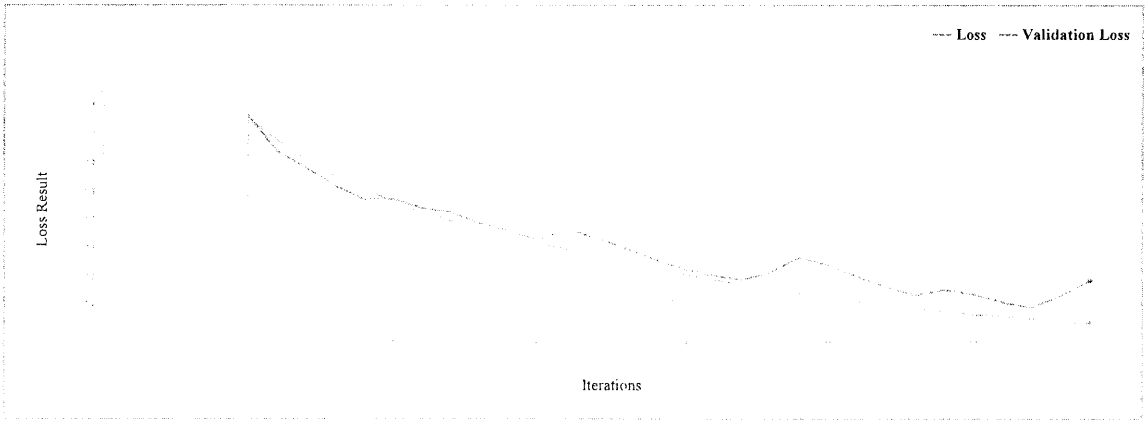


Figure 82: Proposed CNN loss result; the x-axis displays the loss per iteration, here we used continuous raw features as an input to the model instead of images, the orange line represents the training results, and the blue line represents the validation accuracy.

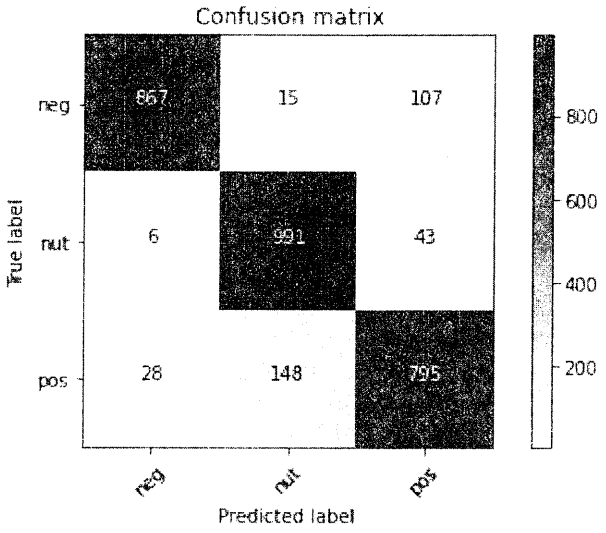


Figure 83: Trained classification model is tested with new data, here we used continuous raw features as an input to the model, each row of the confusion matrix represents the target class, and each column represents the predicted class that a classifier outputs.

We began the preparation by training the collected classification model with training and validation data at the same pace, just as the CNN model does. When we look at the plot of the model's accuracy, we can see that it initially had a low training and validation accuracy level. On the other hand, in order to attain higher classification scores requires a smaller amount of data.

Similar to CNN, it continued to improve as it repeated the training at specific epochs. However, unlike CNN, it achieved an accuracy that was greater than 0.8 even when the training was stopped early.

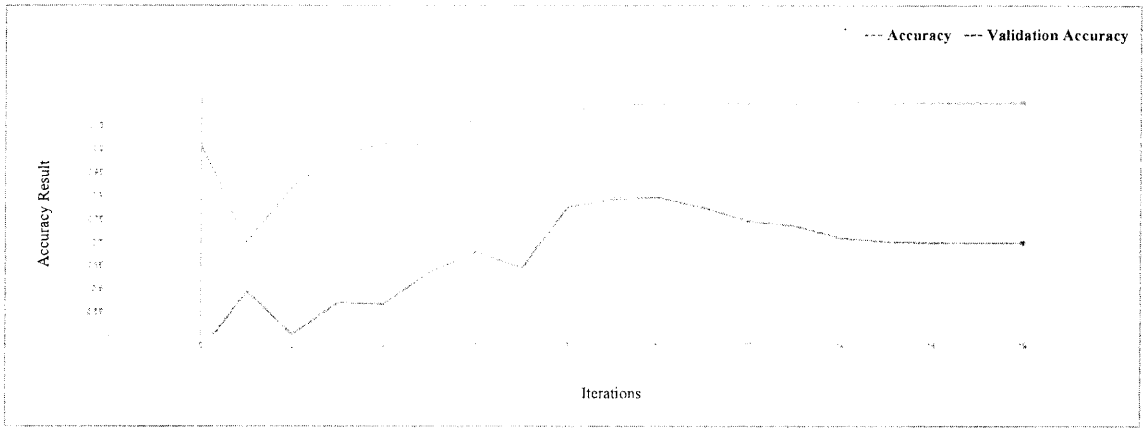


Figure 84: Proposed CNN-AE accuracy result; the x-axis displays the accuracy per iteration, the orange line represents the training results, and the blue line represents the validation accuracy.

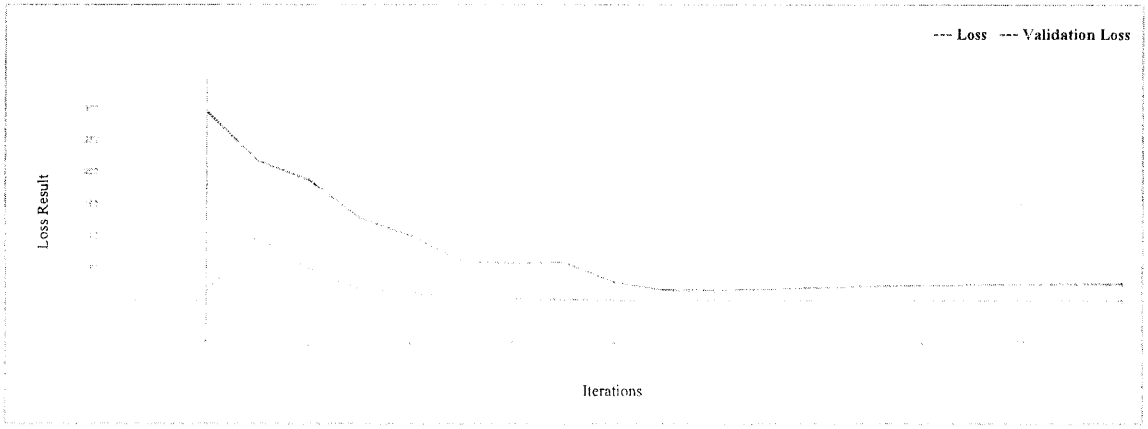


Figure 85: Proposed CNN-AE loss result; the x-axis displays the loss per iteration, the orange line represents the training results, and the blue line represents the validation accuracy.

According to the findings from using the proposed CNN classifier, even though the wrong prediction rate is very close to CNN, but in this model, it is relatively easy to identify positive emotions, however negative and neutral emotions are the ones that are the most challenging to identify. They cannot separate positive and negative emotions from neutral ones and do not understand how to place negative feelings in a positive category.

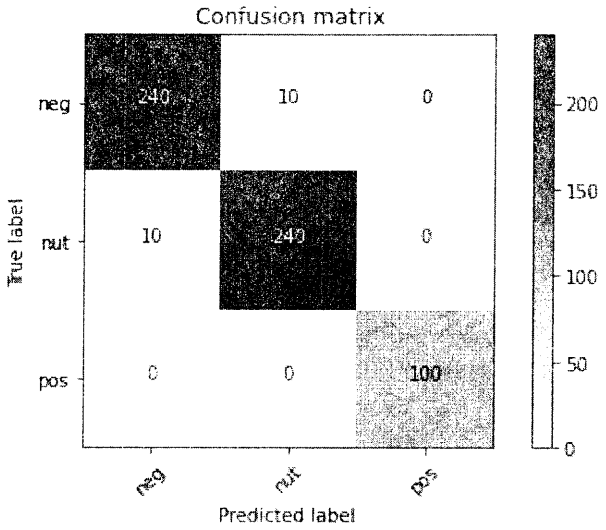


Figure 86: Trained CNN classification model is tested with new data, each row of the confusion matrix represents the target class, and each column represents the predicted class that a classifier outputs.

Table 7: Comparing newly proposed deep learning classification performance with other classification methods.

Classification Method	*Accuracy*
Decision Tree	*81%*
Discriminant Analysis	*86%*
Naive Bayes	*70%*
Linear SVM	*73%*
KNN	*61%*
Boosted (Ensemble)	*82%*
Bagged (Ensemble)	*80%*
Proposed Distributed CNN	*76%*
Proposed Distributed CNN-LSTM	*85%*
Proposed Centralized CNN	*97%*
Proposed Centralized CNN-LSTM	*98%*

Additionally, using RNN and LSTM speeds up the enhancement of accuracy in case of a limited amount of data, which can be good for classifying emotional stimuli using EEG-based signals, which could be very sensitive to different types of noises.

In general, deep learning networks show a high accuracy rate with a low complexity structure, which leads to a faster processing time for real-time applications. In addition, using RNN and LSTM speeds up the enhancement of the accuracy in case of a limited.

4.1.5. Recurrent Neural Networks

However, this time the input data should be reshaped to three dimensions before it can be processed in the system. We started the preparation of the CNN-LSTM by training the collected classification model using training and validation data at the same rate.

When we looked at the plot of the model's accuracy, we saw some interesting results: adding the LSTM decreased the data size required to achieve good accuracy, where it started with accuracies higher than 0.8 at early stages, and it kept improving with more iterations until it reached 0.9 accuracies, which was higher than earlier models.

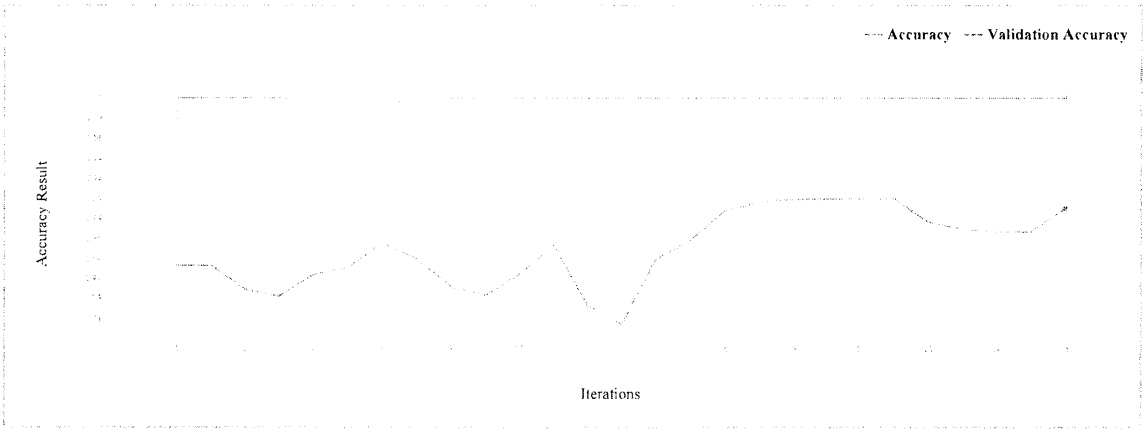


Figure 87: Proposed CNN-LSTM accuracy result; the x-axis displays the accuracy per iteration, the orange line represents the training results, and the blue line represents the validation accuracy.

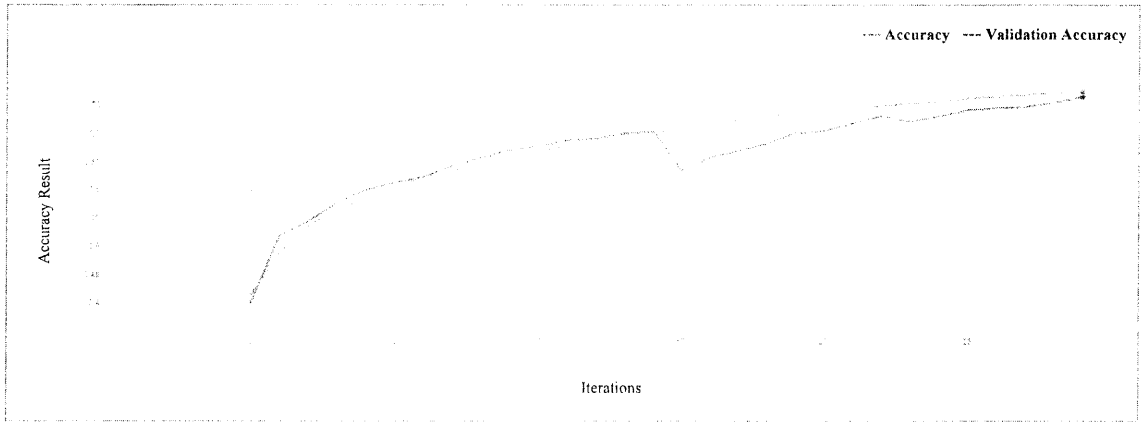


Figure 88: Proposed CNN-LSTM accuracy result; the x-axis displays the accuracy per iteration, here we used continuous raw features as an input to the model instead of images, the orange line represents the training results, and the blue line represents the validation accuracy.

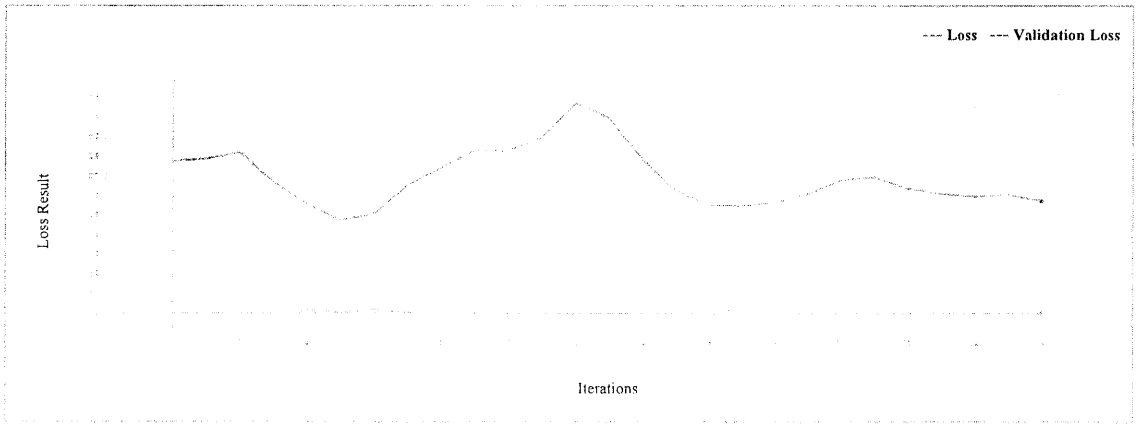


Figure 89: Proposed CNN-LSTM loss result; the x-axis displays the loss per iteration, the orange line represents the training results, and the blue line represents the validation loss.

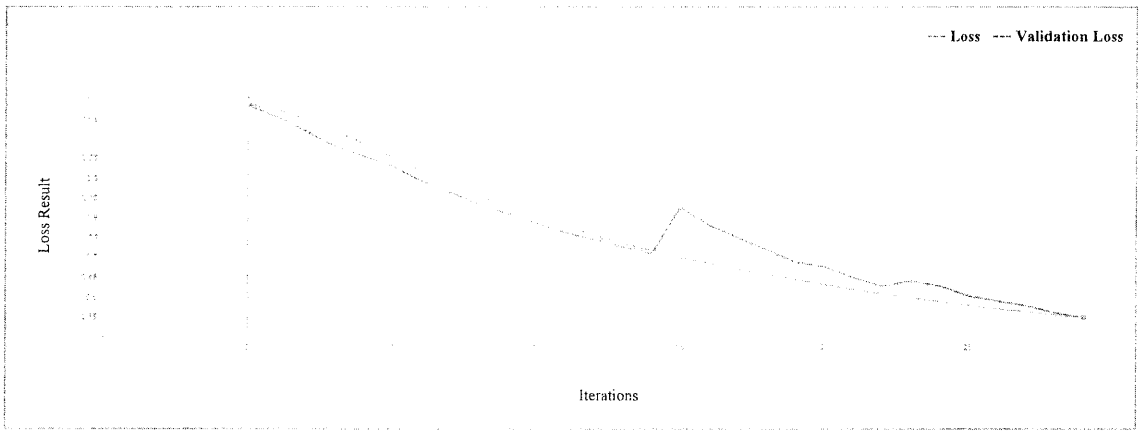


Figure 90: Proposed CNN-LSTM loss result; the x-axis displays the loss per iteration, here we used continuous raw features as an input to the model instead of images, the orange line represents the training results, and the blue line represents the validation loss.

Similarly, to the earlier model, CCN-LSTM shows superiority on the prediction test, where only 10 mistakes happen, but in this model, it is relatively easy to identify positive and negative emotions; however neutral emotions are the ones that are the most challenging to identify.

They cannot separate positive and negative emotions from neutral ones and do not understand how to place negative feelings in a positive category.

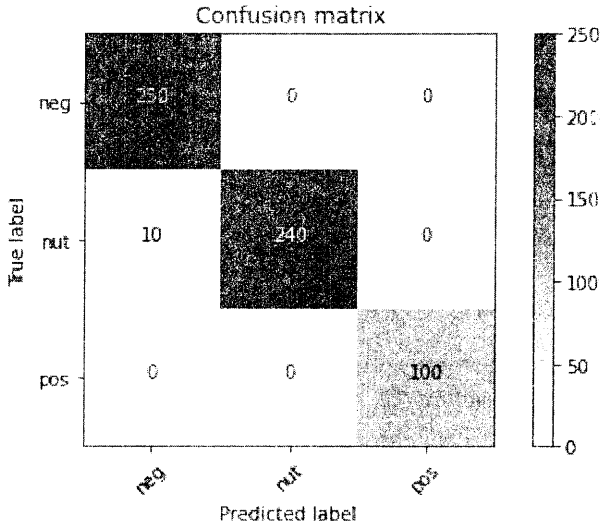


Figure 91: Trained CNN-LSTM classification model is tested with new data, each row of the confusion matrix represents the target class, and each column represents the predicted class that a classifier outputs.

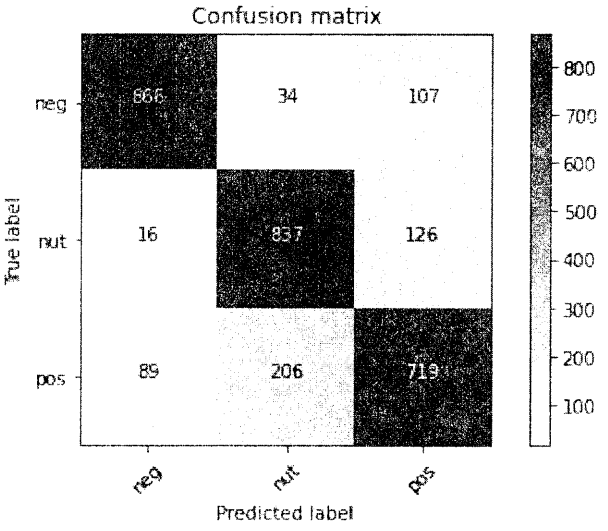


Figure 92: Trained CNN-LSTM classification model is tested with new data, here we used continuous raw features as an input to the model instead of images, each row of the confusion matrix represents the target class, and each column represents the predicted class that a classifier outputs.

4.2. Discussions

After eliminating any EEG aberrations that could be present, Li et al. (2020) (X. Li et al., 2020) published the most recent benchmark for the categorization of emotional states.

They used a z-score approach to perform normalization on the channel data of each participant, and they achieved an accuracy of 85.81% as a result. Our greatest accuracy for binary classification is 97.5%, which is higher than any other approach that has been discovered.

Our proposed method has been surpassed by the methods proposed by Zhang et al. (2015) (W. L. Zheng et al., 2015), whose model was based on a deep belief network (DBN) and achieved an accuracy of 86.08%, and by Li et al. (2018) (X. Li et al., 2018), who achieved an accuracy of 83.33% by using adaptive subspace feature matching (ASFM) and Support Vector Machine.

The articles that have been listed can be found here (SVM). Nevertheless, in contrast to our method, Wang et al. (2019) (W. Zhang et al., 2019) and Zhang et al. (2020) (Song et al., 2020) employed the validation set throughout the training phase, which can boost the cross-subject accuracy. Our technique does not use the validation set.

Table 8: Classification performance based on processing and signal format.

Classification Goal	*Three Classes*				*Depression*			
Feature Processing	Centralized		Distributed		Centralized		Distributed	
Used signals	Topographic Map		Raw Signal		Topographic Map		Raw Signal	
Classification Method	CNN	LSTM	CNN	LSTM	CNN	LSTM	CNN	LSTM
Subject Dependent Accuracy	97.50%	97.50%	79.66%	88.63%	98.30%	97.50%	86.23%	91.43%
Subject Independent Accuracy	97.48%	98.33%	76.47%	84.73%	98.33%	98.33%	83.67%	89.70%

Table 9: lists a performance comparison between state-of-the-art models and our proposed method for EEG emotion categories.

Method	*Proposed*	*Proposed*	*Zheng and Lu (2015)*	*Li et al. (2018)*	*Zhang et al. (2020)*	*Wang et al. (2019)*	*Li et al. (2020)*
Classification Goal	Three Classes						
Classification Method	CNN	LSTM	DBN	SVM	DGCNN	CNN	AE
Dependent Accuracy	97.50%	97.50%	86.08%	83.33%	90.4%	90.59%	85.81%

Chapter Five:
Conclusions and Future Works

Emotion is a state that includes all a person's feelings, thoughts, and actions. It is an integral part of how people interact with one another. Emotion classification aims to automatically distinguish between different emotional states using physiological and non-physiological signals collected from humans in order to facilitate efficient communication and interaction between humans and machines. One of the most popular and practical approaches to emotion estimation is Brainwaves-Based Emotion Estimation. Technology is vital in many research areas, including the treatment of emotional disorders, the brain-computer interface for people with disabilities, entertainment, and many others. This thesis presents improved methods, schemes, and solid frameworks for estimating human emotions based on Electroencephalogram (EEG).

Machine learning is gradually becoming an integral component of people's day-to-day activities in a variety of new domains, including education, training, and entertainment, among others. These shifts are mostly caused by the rapid advancement in the technology of data transformation, such as 5G networks, data delivery technologies, such as Content Delivery Networks, and big data streaming and distributed computing technologies, such as Hadoop ecosystems.

5.1. Research Contributions and Conclusions

The main aim of the work is to propose enhanced approaches to building real-time EEG-based emotion recognition solutions and investigate various machine learning and deep learning techniques. We proposed both theoretical and practical implementations to build our real-time inter-subject emotion recognition architecture.

In Chapter Three, we took an in-depth look at the current needs and trends of the industry, as well as the difficulties that arise as a result of the nature of the new usages and applications. Following that, we translated those inputs into the form of requirements and limitations, which our subsequent research would address and circumvent. In the rest of the chapter, we highlighted a high-level overview of the proposed solution. Our work is unlike the typical

work in this area, and it focuses on reducing the gap of taking the EEG base research into a new applicable level that is in-line with recent changes of challenges and requirements of the actual life situation; more specifically, enhance classification and decoding of the EEG signal on a more extensive scale and take advantage of the availability of limitless resources and data size to enhance overall solution classification accuracy and decoding effectiveness. Our work is one of the first of its type to build realistic collaborative solutions that are equipped with the capacity for knowledge transfer through the internet.

In Chapter Four, we had a look at some of the most well-known EEG datasets for emotions, which are extremely important to the reliability of our research. In contrast to any other study that had been done, we investigated the specifics of the experiment and assessed the influence of the randomization of the data. We suggested a technique for the dataset and experiment selection based on the research-stated problem, and we especially analyzed the most well-known datasets, which are known as SEED and DEAP. Even after deciding to employ SEED, we chose the experiment that had a high likelihood of randomization in order to guarantee that the final results would be objective.

In Chapter Five, we covered the next step after acquiring the dataset, signal processing, so we developed and validated a new approach for optimizing the EEG signal processing. This work devotes a large portion of its time and effort to the development and validation of a novel strategy for improving EEG data processing under unknown settings. In addition, given that the density and complexity of EEG signals is one of their primary characteristics, those processes designed to reduce signal dimension enable the live extraction of emotional information from EEG data collected using mobile devices.

This is possible because one of the primary characteristics of EEG signals is their density and complexity. Here, we offer an adaptive channel selection methodology that employs several filter techniques to identify the top-N ranked channels utilizing recursive performance ranking via transfer learning. The ranking value will be used to rank channels locally; however, the selection of the appropriate number of channels will be done in a recursive manner. In this chapter, we will assess the suggested channel selection and the influence it has on both the accuracy and performance of the solution. We will take into consideration the

amount of time required for processing based on the number of channels. The function of information transfer in this situation is to narrow down the channels by considering the whole solution. As a consequence of this, the accuracy with which the system minimizes chosen tracks remains unaffected.

This adaptive methodology shifts depending on the processing location and available resources in order to be tuned to real-time situations; however, the topic that we will be focusing on in this chapter is resource consumption. In Chapter 6, after reviewing the most common features, we proposed a new feature extraction and selection processing methodology in the field. In this instance, knowledge transfer is employed as a supporting component to assist in the resolution of issues with overfitting that is brought on by data that is localized. We tested the performance of feature extraction and selection on real hardware, such as Intel Core i7 and ARM A53 processors, which respectively represent centralized and decentralized processing units. This was done because the amount of time that a battery can last is one of the most important limitations. The findings of this experiment indicate how to become acclimated to the benefits of each feature by automatically deploying it, and the experiment itself exhibits close-to-real-life Internet of Things systems.

We find that entropy features give outcomes that are more accurate than those produced by other basic features. Unfortunately, the processing of it needed either additional resources or additional time. As a result, we divided characteristics into centralized and non-centralized categories. The one that was centralized demanded more resources but delivered a greater level of accuracy. We provided examples of the time required to extract an entropy feature, as well as more complex statistical features and advanced statistical features. Our solution demonstrated a 10x reduction in the number of resources required for data computation and transformation. We compared the outcome of the resource consumption to the results of existing centralized and decentralized methodologies, both of which are often utilized in academic research.

In Chapter Seven, we begin to move to the classification layer, which is where the majority of the heavy lifting is done. In this chapter, we tested and evaluated popular supervised classification methods such as Decision Tree Classifier, Nearest Neighbor Classifier, Support Vector Machine, and Naive Bayes Classifier,

as well as unsupervised classification methods such as Hierarchical Clustering and K-mean Clustering. Since one of the main goals of this research is to improve the EEG classification performance, particularly for new applications that add more restrictions to the speed and density of the data stream of the signal reads, we focused on supervised classifiers in this paper. We applied the selected supervised classification methods with different channel quantities between 5 and 60 to test performance vs. accuracy.

We perform the test again, each time with a different approach to feature reduction; this time, we use the PCA-based feature reduction method in addition to the random feature removal method. The findings of the tests showed that the accuracy of the system was not significantly affected by the addition of additional features. In addition, given that we wish to maximize the efficiency with which the total number of features will be processed, we have discovered that the PCA-based feature selection approach yields superior outcomes than the random feature removal method. For instance, when comparing the accuracy of each classification approach, PCA-based features increased accuracy by 5–10%. In addition, we discovered that the K-Nearest Neighbors Algorithm and the linear Support Vector Machine (SVM) obtained higher accuracy than the majority of the other approaches, independent of the feature selection method or the number of features.

In Chapter Eight, we reviewed different types of deep learning classification methods, including artificial and deep neural networks. The convolutional neural network (CNN) layers are utilized in our suggested classification model, which possesses both high classification accuracy and rapid convergence. We tried to design our model to match our higher goal of improving a real-time classification solution, so our model was designed to have simple convolutions that give good results with a small number of samples at the initiation, and the model will have unified formats for input and output to be in line with real-time knowledge transfer and weight forwarding. The foundation of our model for deep learning is made up of convolutional neural networks. The first thing that has to be done is to prepare the characteristics that were retrieved. We made the decision to evaluate the 1-D and 2-D CNN algorithms, where 1-D symbolizes the array data format that is the simplest to understand.

Nevertheless, two dimensions denote a tabular data format, and given that we are working in the biomedical industry, we have chosen to work with a format that a physician can comprehend in a short amount of time. In order to do this, we made use of the scalp spectrogram representation, which is a standard visual signal format. In addition to this, we made use of an EEG-specific format known as topography. Both representations have the same level of processing power and accuracy regardless of which one is used. Both centralized and decentralized deep learning classification approaches outperform other classification methods by up to 10%.

Our proposed When we looked at the plot of the model's accuracy, we saw some interesting results: adding the LSTM decreased the data size required to achieve good accuracy, where it started with accuracies higher than 0.8 at early stages, and it kept improving with more iterations until it reached 0.9 accuracies, which was higher than earlier models.

In the previous chapter, we returned to the functionality of knowledge transfer that we had provided for the overall solution structure. In this chapter, we discussed how transferring knowledge plays a part at each stage of the emotion classification process. On the other hand, in contrast to channel selection and feature processing, where the primary purpose of knowledge transfer is to conserve resources, our goal in this chapter is to concentrate more on the accuracy impact of knowledge transfer. Here, we examine deep learning networks since they automatically cover all classification layers, including feature extraction and feature extraction.

We put our suggested CNN and RNN models to the test on tiny data sources that reflected real-time scenarios or what is known as "micro-batching" in the industry, which is processing that is very near to real-time. We then compared the results with and without knowledge transfer.

Our experiment was meant to assess three alternative setups: first, 1-D input against 2-D input; second, knowledge transfer versus no information transfer; and third, inter-subject knowledge transfer versus intra-subject knowledge transfer.

The phrase "ecosystem" refers to a group of people who work in the construction industry. We discovered that the accuracy of the same model when using 2-D is 10-15% greater than that of the 1-

D classifier after evaluating the classification of both 1-D and 2-D signals. The second discovery is one that was rather predictable: when we compared the accuracy of findings obtained with and without the utilization of knowledge transfer, we discovered that the utilization of knowledge transfer led to an improvement in accuracy of between 10 and 20 percent. The last discovery relates to the subject dependencies of knowledge transmission. We discovered that the accuracy of utilizing inter-subject information is quite effective and delivers a result that is comparable to using knowledge of the same topic. This paves the way for a significant chance to use more centrally managed global knowledge storage. Lastly, we evaluate our whole effort in comparison to previous studies and find that our method produces superior outcomes by 7%–12%.

The relevance of the study that is being presented is the development of an EEG-based solution for emotion identification that is tailored to contemporary applications that run in real-time. Our solution makes it possible to construct a large-scale online solution that is able to dynamically apply a variety of processing techniques that strike a balance between accuracy, resource consumption, and the urgency of decoding. In spite of the fact that our solution makes available a platform for solution-wide knowledge transfer, which may make use of data centered on subscribers or collaborators in order to improve the accuracy of decoding answers and increase their speed.

5.2. Future Work

Although significant headway has been achieved in the field of emotional brain-computer interactions over the course of the past several decades, there is still a great deal of unanswered concern.

Prepare first for the design of an experiment that can have various types of emotional stimuli that are targeted in emotion research. These can include emotional film clips with both scenes and audio that are reliable and effective for evoking subjects' emotions in the laboratory. Additionally, the experiment should have greater access to personal characteristics that may be included in the input features of our classifier.

The majority of the already-existing experiments are divorced from any sort of social collaboration, where each experiment is isolated from any real environment. The individuals do not actively engage with their surroundings and are only passively exposed to stimuli. One way to make the experiments better is to incorporate social interactions into them. This may be done, for instance, through group experiments in educational or professional settings or by the use of new virtual reality technologies that make emotion tests more realistic.

We would propose an offline emotional profile assessment protocol for future improvements related to knowledge transfer. This protocol would be based on Cattell's sixteen factors of personality (16PF), and it would be used to generate subject profiles that would filter the knowledge transfer process and introduce a matching mechanism between different subjects.

It's possible that in the not too distant future, we'll have a better knowledge of how emotional processing is connected to a wide variety of mental diseases, including anxiety and depression. On the other hand, there is a dearth of knowledge concerning the fundamental neurological mechanism underlying the processing of emotions in the brain. Research on subjects connected to emotions, such as the analysis of brain and behavioral data and the search for treatments for mood disorders, should interact more closely.

Bibliography:

Ackermann, P., Kohlschein, C., Bitsch, J. A., Wehrle, K., & Jeschke, S. (2016). EEG-based automatic emotion recognition: Feature extraction, selection and classification methods. *2016 IEEE 18th International Conference on E-Health Networking, Applications and Services (Healthcom)*, 1–6. https://doi.org/10.1109/HealthCom.2016.7749447

Alarcao, S. M., & Fonseca, M. J. (2019). Emotions Recognition Using EEG Signals: A Survey. *IEEE Transactions on Affective Computing*, *10*(3), 374–393. https://doi.org/10.1109/TAFFC.2017.2714671

Alotaiby, T., El-Samie, F. E. A., Alshebeili, S. A., & Ahmad, I. (2015). A review of channel selection algorithms for EEG signal processing. *EURASIP Journal on Advances in Signal Processing*, *2015*(1), 66. https://doi.org/10.1186/s13634-015-0251-9

Al-Shargie, F., Tariq, U., Alex, M., Mir, H., & Al-Nashash, H. (2019). Emotion Recognition Based on Fusion of Local Cortical Activations and Dynamic Functional Networks Connectivity: An EEG Study. *IEEE Access*, *7*, 143550–143562. https://doi.org/10.1109/ACCESS.2019.2944008

Arendt-Nielsen, L., & Chen, A. C. N. (2003). Lasers and other thermal stimulators for activation of skin nociceptors in humans. *Neurophysiologie Clinique/Clinical Neurophysiology*, *33*(6), 259–268. https://doi.org/10.1016/J.NEUCLI.2003.10.005

Arnin, J., Kahani, D., Lakany, H., & Conway, B. A. (2018). Evaluation of Different Signal Processing Methods in Time and Frequency Domain for Brain-Computer Interface Applications. *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 235–238. https://doi.org/10.1109/EMBC.2018.8512193

Arunkumar, N., Ram Kumar, K., & Venkataraman, V. (2018). Entropy features for focal EEG and non focal EEG. *Journal of Computational Science*, *27*, 440–444. https://doi.org/10.1016/j.jocs.2018.02.002

Baig, M. Z., Aslam, N., & Shum, H. P. H. (2020). Filtering techniques for channel selection in motor imagery EEG applications: a survey. *Artificial Intelligence Review*, *53*(2), 1207–1232. https://doi.org/10.1007/s10462-019-09694-8

Bao, F. S., Liu, X., & Zhang, C. (2011). PyEEG: An Open Source Python Module for EEG/MEG Feature Extraction. *Computational*

Intelligence and Neuroscience, *2011*, 1–7. https://doi.org/10.1155/2011/406391

Baveye, Y., Dellandrea, E., Chamaret, C., & Liming Chen. (2015). LIRIS-ACCEDE: A Video Database for Affective Content Analysis. *IEEE Transactions on Affective Computing*, *6*(1), 43–55. https://doi.org/10.1109/TAFFC.2015.2396531

Black, M. H., Chen, N. T. M., Iyer, K. K., Lipp, O. V., Bölte, S., Falkmer, M., Tan, T., & Girdler, S. (2017). Mechanisms of facial emotion recognition in autism spectrum disorders: Insights from eye tracking and electroencephalography. *Neuroscience & Biobehavioral Reviews*, *80*, 488–515. https://doi.org/10.1016/j.neubiorev.2017.06.016

Bradley, M. M., & Lang, P. J. (1994). Measuring emotion: The self-assessment manikin and the semantic differential. *Journal of Behavior Therapy and Experimental Psychiatry*, *25*(1), 49–59. https://doi.org/10.1016/0005-7916(94)90063-9

Bradley, M. M., & Lang, P. J. (2017). International Affective Picture System. In *Encyclopedia of Personality and Individual Differences* (pp. 1–4). Springer International Publishing. https://doi.org/10.1007/978-3-319-28099-8_42-1

Breiman, L. (1996). Bagging predictors. *Machine Learning 1996 24:2*, *24*(2), 123–140. https://doi.org/10.1007/BF00058655

Brenner, C. A., Rumak, S. P., Burns, A. M. N., & Kieffaber, P. D. (2014). The role of encoding and attention in facial emotion memory: An EEG investigation. *International Journal of Psychophysiology*, *93*(3), 398–410. https://doi.org/10.1016/j.ijpsycho.2014.06.006

Cabañero-Gomez, L., Hervas, R., Gonzalez, I., & Rodriguez-Benitez, L. (2021). eeglib: A Python module for EEG feature extraction. *SoftwareX*, *15*, 100745. https://doi.org/10.1016/J.SOFTX.2021.100745

Carvalho, S., Leite, J., Galdo-Álvarez, S., & Gonçalves, Ó. F. (2012). The Emotional Movie Database (EMDB): a self-report and psychophysiological study. *Applied Psychophysiology and Biofeedback*, *37*(4), 279–294. https://doi.org/10.1007/S10484-012-9201-6

Cattell, H. E. P., & Mead, A. D. (2008). The Sixteen Personality Factor Questionnaire (16PF). In *The SAGE Handbook of Personality Theory and Assessment: Volume 2 — Personality Measurement and Testing*

(pp. 135–159). SAGE Publications Ltd. https://doi.org/10.4135/9781849200479.n7

Chang, W.-D., Lim, J.-H., Im, C.-H., Aarabi A, K. K. G. R. M. H. A. and W. F., Barbati G, P. C. Z. F. R. P. M. and T. F., Breuer L, D. J. R. T. P. L. and S. N. J., Chang W-D, C. H.-S. K. K. and I. C.-H., C-H, C. W.-D. and I., J, C. R. J. and B. R., Durka P, K. H. B. K. S. W. and N. S., Fukuda K, S. J. A. B. T. B. and R. M. B., Galley N, S. R. and G. L. ed G. A., N, G. G. and G. S., Gratton G, C. M. G. and D. E., E, H. D. and N., M, H. S. and F., Jiang J-A, C. C.-F. C. M.-J. L. R.-G. T. C.-L. and L. R., Jung T-P, M. S. W. M. T. J. C. E. and S. T. J., J, K. S. and M., … Yong X, W. R. K. B. G. E. and S. N. S. (2016). An unsupervised eye blink artifact detection method for real-time electroencephalogram processing. *Physiological Measurement*, *37*(3), 401–417. https://doi.org/10.1088/0967-3334/37/3/401

Chaudhary, R., & Jaswal, R. A. (2021). A Review of Emotion Recognition Based on EEG using DEAP Dataset. *International Journal of Scientific Research in Science, Engineering and Technology*, 298–303. https://doi.org/10.32628/IJSRSET218352

Chen, D. W., Miao, R., Yang, W. Q., Liang, Y., Chen, H. H., Huang, L., Deng, C. J., & Han, N. (2019). A feature extraction method based on differential entropy and linear discriminant analysis for emotion recognition. *Sensors (Switzerland)*, *19*(7). https://doi.org/10.3390/s19071631

Chen, Z., Lu, G., Xie, Z., & Shang, W. (2020). A Unified Framework and Method for EEG-Based Early Epileptic Seizure Detection and Epilepsy Diagnosis. *IEEE Access*, *8*, 20080–20092. https://doi.org/10.1109/ACCESS.2020.2969055

Cimtay, Y., & Ekmekcioglu, E. (2020). Investigating the Use of Pretrained Convolutional Neural Network on Cross-Subject and Cross-Dataset EEG Emotion Recognition. *Sensors*, *20*(7), 2034. https://doi.org/10.3390/s20072034

Clerc, M., Bougrain, L., & Lotte, F. (2016). *Brain–Computer Interfaces 1* (M. Clerc, L. Bougrain, & F. Lotte, Eds.). Wiley. https://doi.org/10.1002/9781119144977

Coppersmith, D., Hong, S. E. J., & Hosking, J. R. M. (1999). Partitioning Nominal Attributes in Decision Trees. *Data Mining and Knowledge Discovery 1999 3:2*, *3*(2), 197–217. https://doi.org/10.1023/A:1009869804967

Craik, A., He, Y., & Contreras-Vidal, J. L. (2019). Deep learning for electroencephalogram (EEG) classification tasks: a review. *Journal of Neural Engineering*, *16*(3), 031001. https://doi.org/10.1088/1741-2552/ab0ab5

Cudlenco, N., Popescu, N., & Leordeanu, M. (2020). Reading into the mind's eye: Boosting automatic visual recognition with EEG signals. *Neurocomputing*, *386*, 281–292. https://doi.org/10.1016/j.neucom.2019.12.076

Daoud, H., & Bayoumi, M. A. (2019). Efficient Epileptic Seizure Prediction Based on Deep Learning. *IEEE Transactions on Biomedical Circuits and Systems*, *13*(5), 804–813. https://doi.org/10.1109/TBCAS.2019.2929053

Delorme, A., & Makeig, S. (2004). EEGLAB: an open source toolbox for analysis of single-trial EEG dynamics including independent component analysis. *Journal of Neuroscience Methods*, *134*(1), 9–21. https://doi.org/10.1016/J.JNEUMETH.2003.10.009

Deweese, M. M., Bradley, M. M., Lang, P. J., Andersen, S. K., Müller, M. M., & Keil, A. (2014). Snake fearfulness is associated with sustained competitive biases to visual snake features: Hypervigilance without avoidance. *Psychiatry Research*, *219*(2), 329–335. https://doi.org/10.1016/j.psychres.2014.05.042

Douglas-Cowie, E., Cox, C., Martin, J.-C., Devillers, L., Cowie, R., Sneddon, I., McRorie, M., Pelachaud, C., Peters, C., Lowry, O., Batliner, A., & Hönig, F. (2011). *The HUMAINE Database* (pp. 243–284). https://doi.org/10.1007/978-3-642-15184-2_14

Dow, J. (2011). Aristotle's Theory of the Emotions: Emotions as Pleasures and Pains. In *Moral Psychology and Human Action in Aristotle* (pp. 47–74). Oxford University Press. https://doi.org/10.1093/acprof:oso/9780199546541.003.0003

Duan, R.-N. N., Zhu, J.-Y. Y., & Lu, B.-L. L. (2013). Differential entropy feature for EEG-based emotion classification. *2013 6th International IEEE/EMBS Conference on Neural Engineering (NER)*, 81–84. https://doi.org/10.1109/NER.2013.6695876

Ekman, P. (1992). Are there basic emotions? *Psychological Review*, *99*(3), 550–553. https://doi.org/10.1037/0033-295X.99.3.550

Ganesh, S. K. (n.d.). *Awesome Drawing tools for Neural Net Architecture | Data Science and Machine Learning | Kaggle*. Retrieved July 31, 2022, from https://www.kaggle.com/getting-started/253300

Gannouni, S., Aledaily, A., Belwafi, K., & Aboalsamh, H. (2021). Emotion detection using electroencephalography signals and a zero-time windowing-based epoch estimation and relevant electrode identification. *Scientific Reports*, *11*(1), 7071. https://doi.org/10.1038/s41598-021-86345-5

Gatti, E., Calzolari, E., Maggioni, E., & Obrist, M. (2018). Emotional ratings and skin conductance response to visual, auditory and haptic stimuli. *Scientific Data*, *5*(1), 180120. https://doi.org/10.1038/sdata.2018.120

Gerla, V., Djordjevic, V., Lhotska, L., & Krajca, V. (2010). PSGLab Matlab toolbox for polysomnographic data processing: Development and practical application. *Proceedings of the 10th IEEE International Conference on Information Technology and Applications in Biomedicine*, 1–4. https://doi.org/10.1109/ITAB.2010.5687737

Gunes, H., Schuller, B., Pantic, M., & Cowie, R. (2011). Emotion representation, analysis and synthesis in continuous space: A survey. *Face and Gesture 2011*, 827–834. https://doi.org/10.1109/FG.2011.5771357

Guo, Z., Shen, Y., Wan, S., Shang, W., & Yu, K. (2021). Hybrid Intelligence-Driven Medical Image Recognition for Remote Patient Diagnosis in Internet of Medical Things. *IEEE Journal of Biomedical and Health Informatics*, 1–1. https://doi.org/10.1109/JBHI.2021.3139541

Haamer, R. E., Rusadze, E., Lüsi, I., Ahmed, T., Escalera, S., & Anbarjafari, G. (2018). Review on Emotion Recognition Databases. In *Human-Robot Interaction - Theory and Application*. InTech. https://doi.org/10.5772/intechopen.72748

Hajare, R., & Kadam, S. (2021). Comparative Study Analysis of Practical EEG sensors in Medical Diagnoses. *Global Transitions Proceedings*. https://doi.org/10.1016/J.GLTP.2021.08.009

Hammal, Z., Couvreur, L., Caplier, A., & Rombaut, M. (2007). Facial expression classification: An approach based on the fusion of facial deformations using the transferable belief model. *International Journal of Approximate Reasoning*, *46*(3), 542–567. https://doi.org/10.1016/j.ijar.2007.02.003

Hossain, I., Khosravi, A., & Nahavandhi, S. (2016). Active transfer learning and selective instance transfer with active learning for motor imagery based BCI. *2016 International Joint Conference on Neural*

Networks (IJCNN), 4048–4055. https://doi.org/10.1109/IJCNN.2016.7727726

Huishi Zhang, C., Sohrabpour, A., Lu, Y., & He, B. (2016). Spectral and spatial changes of brain rhythmic activity in response to the sustained thermal pain stimulation. *Human Brain Mapping*, *37*(8), 2976–2991. https://doi.org/10.1002/hbm.23220

Isaksson, A., Wennberg, A., & Zetterberg, L. H. (1981). Computer Analysis of EEG Signals with Parametric Models. *Proceedings of the IEEE*, *69*(4), 451–461. https://doi.org/10.1109/PROC.1981.11988

Jayaram, V., Alamgir, M., Altun, Y., Scholkopf, B., & Grosse-Wentrup, M. (2016). Transfer Learning in Brain-Computer Interfaces. *IEEE Computational Intelligence Magazine*, *11*(1), 20–31. https://doi.org/10.1109/MCI.2015.2501545

Jenke, R., Peer, A., & Buss, M. (2014). Feature Extraction and Selection for Emotion Recognition from EEG. *IEEE Transactions on Affective Computing*, *5*(3), 327–339. https://doi.org/10.1109/TAFFC.2014.2339834

Jiang, L., Stocco, A., Losey, D. M., Abernethy, J. A., Prat, C. S., & Rao, R. P. N. (2019). BrainNet: A Multi-Person Brain-to-Brain Interface for Direct Collaboration Between Brains. *Scientific Reports*, *9*(1), 6115. https://doi.org/10.1038/s41598-019-41895-7

Kevric, J., & Subasi, A. (2017). Comparison of signal decomposition methods in classification of EEG signals for motor-imagery BCI system. *Biomedical Signal Processing and Control*, *31*, 398–406. https://doi.org/10.1016/J.BSPC.2016.09.007

Khalaf, A., & Akcakaya, M. (2020). A probabilistic approach for calibration time reduction in hybrid EEG–fTCD brain–computer interfaces. *BioMedical Engineering OnLine*, *19*(1), 23. https://doi.org/10.1186/s12938-020-00765-4

Kingma, D. P., & Ba, J. (2014). *Adam: A Method for Stochastic Optimization*. https://doi.org/10.48550/arXiv.1412.6980

Koelstra, S., Mühl, C., Soleymani, M., Lee, J. S., Yazdani, A., Ebrahimi, T., Pun, T., Nijholt, A., Patras, I., Muhl, C., Soleymani, M., Jong-Seok Lee, Yazdani, A., Ebrahimi, T., Pun, T., Nijholt, A., & Patras, I. (2012). DEAP: A Database for Emotion Analysis ;Using Physiological Signals. *IEEE Transactions on Affective Computing*, *3*(1), 18–31. https://doi.org/10.1109/T-AFFC.2011.15

Krishnan, P. T., & Balasubramanian, P. (2016). Automated EEG seizure detection based on S-transform. *2016 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*, 1–5. https://doi.org/10.1109/ICCIC.2016.7919558

Lan, Z., Sourina, O., Wang, L., Scherer, R., & Muller-Putz, G. (2017). Unsupervised Feature Learning for EEG-based Emotion Recognition. *2017 International Conference on Cyberworlds (CW)*, 182–185. https://doi.org/10.1109/CW.2017.19

Lan, Z., Sourina, O., Wang, L., Scherer, R., & Muller-Putz, G. R. (2019). Domain Adaptation Techniques for EEG-Based Emotion Recognition: A Comparative Study on Two Public Datasets. *IEEE Transactions on Cognitive and Developmental Systems*, *11*(1), 85–94. https://doi.org/10.1109/TCDS.2018.2826840

LeNail, A. (2019). NN-SVG: Publication-Ready Neural Network Architecture Schematics. *Journal of Open Source Software*, *4*(33), 747. https://doi.org/10.21105/joss.00747

Li, G., & Chung, W.-Y. (2015). A Context-Aware EEG Headset System for Early Detection of Driver Drowsiness. *Sensors*, *15*(8), 20873–20893. https://doi.org/10.3390/s150820873

Li, H., Mao, X., & Chen, L. (2020). An emotion classification method from electroencephalogram based on 1/f fluctuation theory. *Measurement and Control*, *53*(5–6), 824–832. https://doi.org/10.1177/0020294020913893

Li, J., Cheng, K., Wang, S., Morstatter, F., Trevino, R. P., Tang, J., & Liu, H. (2018). Feature Selection. *ACM Computing Surveys*, *50*(6), 1–45. https://doi.org/10.1145/3136625

Li, X., Song, D., Zhang, P., Zhang, Y., Hou, Y., & Hu, B. (2018). Exploring EEG Features in Cross-Subject Emotion Recognition. *Frontiers in Neuroscience*, *12*(MAR), 162. https://doi.org/10.3389/fnins.2018.00162

Li, X., Zhao, Z., Song, D., Zhang, Y., Pan, J., Wu, L., Huo, J., Niu, C., & Wang, D. (2020). Latent Factor Decoding of Multi-Channel EEG for Emotion Recognition Through Autoencoder-Like Neural Networks. *Frontiers in Neuroscience*, *14*. https://doi.org/10.3389/fnins.2020.00087

Li, Y., Zheng, W., Cui, Z., Zong, Y., & Ge, S. (2019). EEG Emotion Recognition Based on Graph Regularized Sparse Linear Regression. *Neural Processing Letters*, *49*(2), 555–571. https://doi.org/10.1007/s11063-018-9829-1

Lin, Y.-P., & Jung, T.-P. (2014). Exploring day-to-day variability in EEG-based emotion classification. *2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2226–2229. https://doi.org/10.1109/SMC.2014.6974255

Liu, J., Wu, G., Luo, Y., Qiu, S., Yang, S., Li, W., & Bi, Y. (2020). EEG-Based Emotion Classification Using a Deep Neural Network and Sparse Autoencoder. *Frontiers in Systems Neuroscience*, *14*. https://doi.org/10.3389/fnsys.2020.00043

Liu, Q., Liu, Y., Chen, K., Wang, L., Li, Z., Ai, Q., & Ma, L. (2021). Research on Channel Selection and Multi-Feature Fusion of EEG Signals for Mental Fatigue Detection. *Entropy*, *23*(4), 457. https://doi.org/10.3390/e23040457

Lopes da Silva, F. H., Gonçalves, S. I., & De Munck, J. C. (2010). Electroencephalography (EEG). In *Encyclopedia of Neuroscience* (pp. 849–855). John Wiley & Sons, Inc. https://doi.org/10.1016/B978-008045046-9.00304-1

Lopez-Bernal, D., Balderas, D., Ponce, P., & Molina, A. (2022). A State-of-the-Art Review of EEG-Based Imagined Speech Decoding. *Frontiers in Human Neuroscience*, *16*. https://doi.org/10.3389/fnhum.2022.867281

Lotte, F., Bougrain, L., Cichocki, A., Clerc, M., Congedo, M., Rakotomamonjy, A., & Yger, F. (2018). A review of classification algorithms for EEG-based brain–computer interfaces: a 10 year update. *Journal of Neural Engineering*, *15*(3), 031005. https://doi.org/10.1088/1741-2552/aab2f2

Macnab, R. M., & Koshland, D. E. (1972). The Gradient-Sensing Mechanism in Bacterial Chemotaxis. *Proceedings of the National Academy of Sciences*, *69*(9), 2509–2512. https://doi.org/10.1073/pnas.69.9.2509

Maksimenko, V. A., Runnova, A. E., Zhuravlev, M. O., Protasov, P., Kulanin, R., Khramova, M. V., Pisarchik, A. N., & Hramov, A. E. (2018). Human personality reflects spatio-temporal and time-frequency EEG structure. *PLOS ONE*, *13*(9), e0197642. https://doi.org/10.1371/journal.pone.0197642

Markiewicz, C. J., Gorgolewski, K. J., Feingold, F., Blair, R., Halchenko, Y. O., Miller, E., Hardcastle, N., Wexler, J., Esteban, O., Goncalves, M., Jwa, A., & Poldrack, R. A. (2021). OpenNeuro: An open resource for sharing of neuroimaging data. *BioRxiv*, 2021.06.28.450168. https://doi.org/10.1101/2021.06.28.450168

Matiko, J. W., Beeby, S. P., & Tudor, J. (2014). Fuzzy logic based emotion classification. *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 4389–4393. https://doi.org/10.1109/ICASSP.2014.6854431

Mattar, E. A., & Al-Junaid, H. J. (2018). Machine learning for electroencephalography decoding and robotics dextrous hands movement. *2018 International Conference on Power, Signals, Control and Computation (EPSCICON)*, 1–6. https://doi.org/10.1109/EPSCICON.2018.8379585

Mattar, E. A., Al-Junaid, H. J., & Al-Mutib, K. N. (2019, September 1). Electroencephalography features extraction and deep patterns analysis for robotics learning and control through brain-computer interface. *2019 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies, 3ICT 2019*. https://doi.org/10.1109/3ICT.2019.8910277

Mattar, E. A., Al-Junaid, H. J., & Al-Seddiqi, H. H. (2018). Biomimetic Based EEG Learning for Robotics Complex Grasping and Dexterous Manipulation. *Biomimetic Prosthetics*. https://doi.org/10.5772/INTECHOPEN.72455

Mehdizadehfar, V., Ghassemi, F., Fallah, A., & Pouretemad, H. (2020). EEG study of facial emotion recognition in the fathers of autistic children. *Biomedical Signal Processing and Control*, *56*, 101721. https://doi.org/10.1016/j.bspc.2019.101721

Mehrabian, A., & Russell, J. (1980). *An approach to environmental psychology*. The MIT Press. https://psycnet.apa.org/record/1974-22049-000

Mendlein, A., Szkudlarek, C., & Goodpaster, J. V. (2013). Chemometrics. *Encyclopedia of Forensic Sciences: Second Edition*, 646–651. https://doi.org/10.1016/B978-0-12-382165-2.00259-2

Moon, S.-E., Jang, S., & Lee, J.-S. (2018). Convolutional Neural Network Approach for EEG-Based Emotion Recognition Using Brain Connectivity and its Spatial Information. *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2556–2560. https://doi.org/10.1109/ICASSP.2018.8461315

Moshirian Farahi, S. M., Asghari Ebrahimabad, M. J., Gorji, A., Bigdeli, I., & Moshirian Farahi, S. M. M. (2019). Neuroticism and Frontal EEG Asymmetry Correlated With Dynamic Facial Emotional

Processing in Adolescents. *Frontiers in Psychology, 10.* https://doi.org/10.3389/fpsyg.2019.00175

Motamedi-Fakhr, S., Moshrefi-Torbati, M., Hill, M., Hill, C. M., & White, P. R. (2014). Signal processing techniques applied to human sleep EEG signals—A review. *Biomedical Signal Processing and Control, 10*, 21–33. https://doi.org/10.1016/j.bspc.2013.12.003

Moussa, M. M., Tariq, U., Al-Shargie, F., & Al-Nashash, H. (2022). Discriminating Fake and Real Smiles Using Electroencephalogram Signals With Convolutional Neural Networks. *IEEE Access, 10*, 81020–81030. https://doi.org/10.1109/ACCESS.2022.3195028

Mühl, C., Allison, B., Nijholt, A., & Chanel, G. (2014). A survey of affective brain computer interfaces: principles, state-of-the-art, and challenges. *Brain-Computer Interfaces, 1*(2), 66–84. https://doi.org/10.1080/2326263X.2014.912881

Mullen, T. R., Kothe, C. A. E., Chi, Y. M., Ojeda, A., Kerth, T., Makeig, S., Jung, T.-P., & Cauwenberghs, G. (2015). Real-time neuroimaging and cognitive monitoring using wearable dry EEG. *IEEE Transactions on Biomedical Engineering, 62*(11), 2553–2567. https://doi.org/10.1109/TBME.2015.2481482

Nahum, A. (2015). *System and method for a compact eeg headset* (Patent No. US20150112153A1). https://patents.google.com/patent/US20150112153A1/en

Nakisa, B., Rastgoo, M. N., Tjondronegoro, D., & Chandran, V. (2018). Evolutionary computation algorithms for feature selection of EEG-based emotion recognition using mobile sensors. *Expert Systems with Applications, 93*, 143–155. https://doi.org/10.1016/J.ESWA.2017.09.062

Nielek, R., Ciastek, M., & Kopeć, W. (2017). Emotions make cities live. *Proceedings of the International Conference on Web Intelligence*, 1076–1079. https://doi.org/10.1145/3106426.3109041

Nir, R.-R., Sinai, A., Raz, E., Sprecher, E., & Yarnitsky, D. (2010). Pain assessment by continuous EEG: Association between subjective perception of tonic pain and peak frequency of alpha oscillations during stimulation and at rest. *Brain Research, 1344*, 77–86. https://doi.org/10.1016/j.brainres.2010.05.004

Osgood, C., Suci, G., & Tannenbaum, P. (1957). *The measurement of meaning*. Univer. Illinois Press. https://books.google.com/books?hl=en&lr=&id=Qj8GeUrKZdAC&o

i=fnd&pg=PA1&ots=RJL2ZXLKb7&sig=VfVR6GHOKW8kZhA1y
f8_GIO9hTQ

Pan, S. J., Tsang, I. W., Kwok, J. T., & Yang, Q. (2011). Domain Adaptation via Transfer Component Analysis. *IEEE Transactions on Neural Networks*, *22*(2), 199–210. https://doi.org/10.1109/TNN.2010.2091281

Petridis, S., Martinez, B., & Pantic, M. (2013). The MAHNOB Laughter database. *Image and Vision Computing*, *31*(2), 186–202. https://doi.org/10.1016/j.imavis.2012.08.014

Phung, D., Tran, D., Ma, W., Nguyen, P., & Pham, T. (n.d.). *Using Shannon Entropy as EEG Signal Feature for Fast Person Identification*. Retrieved October 31, 2020, from http://www.i6doc.com/fr/livre/?GCOI=28001100432440.

Plutchik, R. (1991). *The emotions*. https://books.google.com/books?hl=en&lr=&id=JaQauznPoiEC&oi=fnd&pg=PR3&ots=EFBb0A8sGf&sig=ShZGgMp_nvMv0f7ZuQEoUQMa5pM

Ponz, A., Montant, M., Liegeois-Chauvel, C., Silva, C., Braun, M., Jacobs, A. M., & Ziegler, J. C. (2014). Emotion processing in words: a test of the neural re-use hypothesis using surface and intracranial EEG. *Social Cognitive and Affective Neuroscience*, *9*(5), 619–627. https://doi.org/10.1093/scan/nst034

Qiang Ji, Lan, P., & Looney, C. (2006). A probabilistic framework for modeling and real-time monitoring human fatigue. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, *36*(5), 862–875. https://doi.org/10.1109/TSMCA.2005.855922

Quesada-Tabares, R., Molina-Cantero, A. J., Gómez-González, I., Merino-Monge, M., Castro-García, J. A., & Cabrera-Cabrera, R. (2017). Emotions Detection based on a Single-electrode EEG Device. *Proceedings of the 4th International Conference on Physiological Computing Systems*, 89–95. https://doi.org/10.5220/0006476300890095

Raichle, M. E., MacLeod, A. M., Snyder, A. Z., Powers, W. J., Gusnard, D. A., & Shulman, G. L. (2001). A default mode of brain function. *Proceedings of the National Academy of Sciences*, *98*(2), 676–682. https://doi.org/10.1073/pnas.98.2.676

Rodrigues, P. L. C., Jutten, C., & Congedo, M. (2019). Riemannian Procrustes Analysis: Transfer Learning for Brain–Computer

Interfaces. *IEEE Transactions on Biomedical Engineering*, *66*(8), 2390–2401. https://doi.org/10.1109/TBME.2018.2889705

Roeder, L. (n.d.). *Netron: Visualizer for neural network, deep learning, and machine learning models*. Retrieved July 31, 2022, from https://github.com/lutzroeder/netron

Rojas, G. M., Alvarez, C., Montoya, C. E., de la Iglesia-Vayá, M., Cisternas, J. E., & Gálvez, M. (2018). Study of resting-state functional connectivity networks using EEG electrodes position as seed. *Frontiers in Neuroscience*, *12*(APR). https://doi.org/10.3389/FNINS.2018.00235/FULL

Russell, J. A. (1980). A circumplex model of affect. *Journal of Personality and Social Psychology*, *39*(6), 1161–1178. https://doi.org/10.1037/H0077714

Sakkalis, V., Cassar, T., Zervakis, M., Camilleri, K. P., Fabri, S. G., Bigan, C., Karakonstantaki, E., & Micheloyannis, S. (2008). *Parametric and nonparametric EEG analysis for the evaluation of EEG activity in young children with controlled epilepsy. 2008*. https://doi.org/10.1155/2008/462593

Salama, E. S., A.El-Khoribi, R., E.Shoman, M., & A.Wahby, M. (2018). EEG-Based Emotion Recognition using 3D Convolutional Neural Networks. *International Journal of Advanced Computer Science and Applications*, *9*(8). https://doi.org/10.14569/IJACSA.2018.090843

Sallout, M., & Mattar, E. (2022). *EEG-based emotion classification using deep neural network*. 325–332. https://doi.org/10.1049/ICP.2022.0364

Sallout, M., & Mattar, E. A. (2021). EEG-based classification of visually evoked emotions using multiscale entropy features. *3rd Smart Cities Symposium (SCS 2020)*, *2020*(CP777), 136–142. https://doi.org/10.1049/icp.2021.0967

Schaefer, A., Nils, F., Sanchez, X., & Philippot, P. (2010). Assessing the effectiveness of a large database of emotion-eliciting films: A new tool for emotion researchers. *Cognition & Emotion*, *24*(7), 1153–1172. https://doi.org/10.1080/02699930903274322

Schlogl, A., & Brunner, C. (2008). BioSig: A Free and Open Source Software Library for BCI Research. *Computer*, *41*(10), 44–50. https://doi.org/10.1109/MC.2008.407

Schölkopf, B., Smola, A., & Müller, K.-R. (1998). Nonlinear Component Analysis as a Kernel Eigenvalue Problem. *Neural Computation*, *10*(5), 1299–1319. https://doi.org/10.1162/089976698300017467

Schulz, E., May, E. S., Postorino, M., Tiemann, L., Nickel, M. M., Witkovsky, V., Schmidt, P., Gross, J., & Ploner, M. (2015). Prefrontal Gamma Oscillations Encode Tonic Pain in Humans. *Cerebral Cortex*, *25*(11), 4407–4414. https://doi.org/10.1093/cercor/bhv043

Shin, H.-C., Roth, H. R., Gao, M., Lu, L., Xu, Z., Nogues, I., Yao, J., Mollura, D., & Summers, R. M. (2016). Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics and Transfer Learning. *IEEE Transactions on Medical Imaging*, *35*(5), 1285–1298. https://doi.org/10.1109/TMI.2016.2528162

Shojaei, G., & Razzazi, F. (2019). Semi-supervised domain adaptation for pedestrian detection in video surveillance based on maximum independence assumption. *International Journal of Multimedia Information Retrieval*, *8*(4), 241–252. https://doi.org/10.1007/s13735-019-00180-z

Song, T., Zheng, W., Song, P., & Cui, Z. (2020). EEG Emotion Recognition Using Dynamical Graph Convolutional Neural Networks. *IEEE Transactions on Affective Computing*, *11*(3), 532–541. https://doi.org/10.1109/TAFFC.2018.2817622

Stone, J. L., & Hughes, J. R. (2013). Early History of Electroencephalography and Establishment of the American Clinical Neurophysiology Society. *Journal of Clinical Neurophysiology*, *30*(1), 28–44. https://doi.org/10.1097/WNP.0b013e31827edb2d

Swartz Center for Computational Neuroscience. (2017). *EEGLAB is an interactive Matlab toolbox*. Swartz Center for Computational Neuroscience. https://sccn.ucsd.edu/eeglab/

Tangermann, M., Müller, K.-R., Aertsen, A., Birbaumer, N., Braun, C., Brunner, C., Leeb, R., Mehring, C., Miller, K. J., Müller-Putz, G. R., Nolte, G., Pfurtscheller, G., Preissl, H., Schalk, G., Schlögl, A., Vidaurre, C., Waldert, S., & Blankertz, B. (2012). Review of the BCI Competition IV. *Frontiers in Neuroscience*, *6*, 55. https://doi.org/10.3389/fnins.2012.00055

Thomas, K. P., & Vinod, A. P. (2018). EEG-Based Biometric Authentication Using Gamma Band Power During Rest State.

Circuits, Systems, and Signal Processing, *37*(1), 277–289. https://doi.org/10.1007/s00034-017-0551-4

Topic, A., & Russo, M. (2021). Emotion recognition based on EEG feature maps through deep learning network. *Engineering Science and Technology, an International Journal*, *24*(6), 1442–1454. https://doi.org/10.1016/j.jestch.2021.03.012

Topic, A., Russo, M., Stella, M., & Saric, M. (2022). Emotion Recognition Using a Reduced Set of EEG Channels Based on Holographic Feature Maps. *Sensors*, *22*(9), 3248. https://doi.org/10.3390/s22093248

Tripathi, S., Acharya, S., Sharma, R., Mittal, S., & Bhattacharya, S. (2017). Using Deep and Convolutional Neural Networks for Accurate Emotion Classification on DEAP Data. *Proceedings of the AAAI Conference on Artificial Intelligence*, *31*(2), 4746–4752. https://doi.org/10.1609/aaai.v31i2.19105

Tripathy, B., & Anuradha, J. (2020). *Internet of Things (IoT) : Technologies, Applications, Challenges and Solutions*.

Tsihrintzis, G. A., & Jain, L. C. (2020). *Machine Learning Paradigms* (G. A. Tsihrintzis & L. C. Jain, Eds.; Vol. 18). Springer International Publishing. https://doi.org/10.1007/978-3-030-49724-8

Urigüen, J. A., & Garcia-Zapirain, B. (2015). EEG artifact removal-state-of-the-art and guidelines. *Journal of Neural Engineering*, *12*(3), 031001. https://doi.org/10.1088/1741-2560/12/3/031001

VanderPlas, J. (2016). Python Data Science Handbook: Essential Tools for Working with Data. *O'Reilly*, 541. http://shop.oreilly.com/product/0636920034919.do%0Ahttps://jakevdp.github.io/PythonDataScienceHandbook/05.01-what-is-machine-learning.html

Venkatesh, B., & Anuradha, J. (2019). A Review of Feature Selection and Its Methods. *Cybernetics and Information Technologies*, *19*(1), 3–26. https://doi.org/10.2478/cait-2019-0001

Vidaurre, C., Sander, T. H., & Schlögl, A. (2011). BioSig: The Free and Open Source Software Library for Biomedical Signal Processing. *Computational Intelligence and Neuroscience*, *2011*, 1–12. https://doi.org/10.1155/2011/935364

Vijayakumar, V., Case, M., Shirinpour, S., & He, B. (2017). Quantifying and Characterizing Tonic Thermal Pain Across Subjects From EEG Data Using Random Forest Models. *IEEE Transactions on*

Biomedical Engineering, *64*(12), 2988–2996. https://doi.org/10.1109/TBME.2017.2756870

Vikram Shenoy, H., & Vinod, A. P. (2014). An iterative optimization technique for robust channel selection in motor imagery based Brain Computer Interface. *2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 1858–1863. https://doi.org/10.1109/SMC.2014.6974191

Wang, J., & Wang, M. (2021). Review of the emotional feature extraction and classification using EEG signals. *Cognitive Robotics*, *1*, 29–40. https://doi.org/10.1016/J.COGR.2021.04.001

Whissell, C. (2009). Using the Revised Dictionary of Affect in Language to Quantify the Emotional Undertones of Samples of Natural Language. *Psychological Reports*, *105*(2), 509–521. https://doi.org/10.2466/PR0.105.2.509-521

Wright, J. A. (2016). An update of systemic analgesics in children. *Anaesthesia & Intensive Care Medicine*, *17*(6), 280–285. https://doi.org/10.1016/j.mpaic.2016.03.001

Wundt, W. (1894). *Lectures on human and animal psychology (trans. from German 2nd ed.)*. Swan Sonnenschein & Co. https://doi.org/10.1037/12937-000

Xiang, J., Maue, E., Fan, Y., Qi, L., Mangano, F. T., Greiner, H., & Tenney, J. (2020). Kurtosis and skewness of high-frequency brain signals are altered in paediatric epilepsy. *Brain Communications*, *2*(1). https://doi.org/10.1093/braincomms/fcaa036

Yan, K., Kou, L., & Zhang, D. (2018a). Learning Domain-Invariant Subspace Using Domain Features and Independence Maximization. *IEEE Transactions on Cybernetics*, *48*(1), 288–299. https://doi.org/10.1109/TCYB.2016.2633306

Yan, K., Kou, L., & Zhang, D. (2018b). Learning Domain-Invariant Subspace Using Domain Features and Independence Maximization. *IEEE Transactions on Cybernetics*, *48*(1), 288–299. https://doi.org/10.1109/TCYB.2016.2633306

Yang, S., & Deravi, F. (2017). On the Usability of Electroencephalographic Signals for Biometric Recognition: A Survey. *IEEE Transactions on Human-Machine Systems*, *47*(6), 958–969. https://doi.org/10.1109/THMS.2017.2682115

Zangeneh Soroush, M., Maghooli, K., Setarehdan, S. K., & Motie Nasrabadi, A. (2017). A Review on EEG Signals Based Emotion

Recognition. *International Clinical Neuroscience Journal*, *4*(4), 118–129. https://doi.org/10.15171/icnj.2017.01

Zhang, J., Chen, M., Zhao, S., Hu, S., Shi, Z., & Cao, Y. (2016). ReliefF-Based EEG Sensor Selection Methods for Emotion Recognition. *Sensors*, *16*(10), 1558. https://doi.org/10.3390/s16101558

Zhang, W., Wang, F., Jiang, Y., Xu, Z., Wu, S., & Zhang, Y. (2019). *Cross-Subject EEG-Based Emotion Recognition with Deep Domain Confusion* (pp. 558–570). https://doi.org/10.1007/978-3-030-27526-6_49

Zhang, Y., Zhao, L., & Wang, Q. (2023). MiDA: Membership inference attacks against domain adaptation. *ISA Transactions*. https://doi.org/10.1016/j.isatra.2023.01.021

Zheng, W. L., & Lu, B. L. (2017). A multimodal approach to estimating vigilance using EEG and forehead EOG. *Journal of Neural Engineering*, *14*(2). https://doi.org/10.1088/1741-2552/aa5a98

Zheng, W. L., Lu, B. L., Wei-Long Zheng, & Bao-Liang Lu. (2015). Investigating Critical Frequency Bands and Channels for EEG-Based Emotion Recognition with Deep Neural Networks. *IEEE Transactions on Autonomous Mental Development*, *7*(3), 162–175. https://doi.org/10.1109/TAMD.2015.2431497

Zheng, W. L., Zhu, J. Y., Peng, Y., & Lu, B. L. (2014). EEG-based emotion classification using deep belief networks. *Proceedings - IEEE International Conference on Multimedia and Expo*, *2014-September*(Septmber). https://doi.org/10.1109/ICME.2014.6890166

Zheng, W.-L., Liu, W., Lu, Y., Lu, B.-L., & Cichocki, A. (2019). EmotionMeter: A Multimodal Framework for Recognizing Human Emotions. *IEEE Transactions on Cybernetics*, *49*(3), 1110–1122. https://doi.org/10.1109/TCYB.2018.2797176

Zheng, W.-L., & Lu, B.-L. (2016). Personalizing EEG-Based Affective Models with Transfer Learning. *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, 2732–2738. https://doi.org/10.5555/3060832.3061003

Zheng, W.-L., Zhang, Y.-Q., Zhu, J.-Y., & Lu, B.-L. (2015). Transfer components between subjects for EEG-based emotion recognition. *2015 International Conference on Affective Computing and Intelligent Interaction (ACII)*, 917–922. https://doi.org/10.1109/ACII.2015.7344684

Appendix A:
Used ML/AI Python Language Libraries

Python is a high-level, general-purpose programming language. Python is a high-level, general-purpose programming language designed for ease of use by human beings accomplishing all sorts of tasks. Python was created by Guido van Rossum and first released in the early 1990s. Python is a mature language developed by hundreds of collaborators around the world.

MATLAB is widely known as a high-quality environment for any work that involves arrays, matrices, or linear algebra. Python is newer to this arena but is becoming increasingly popular for similar tasks. As you'll see in this article, Python has all of the computational power of MATLAB for science tasks and makes it fast and easy to develop robust applications.

Moreover, since Python is available at no cost, a much broader audience can use the code you develop. As you'll see a little later on in the article, Python has an awesome community that can help you get started with the language and advance your knowledge. For all of these reasons, and many more, a lot of developers replace MATLAB with as their programming language of choice.

Machine Learning using SciKit-Learn

Scikit-learn is a free software machine learning library for the Python programming language. scikit-learn is a Python module for machine learning built on top of SciPy. It provides a selection of efficient tools for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction via a consistency interface in Python.

This library, which is largely written in Python, is built upon NumPy, SciPy and Matplotlib. It features various classification, regression and clustering algorithms including support-vector machines.

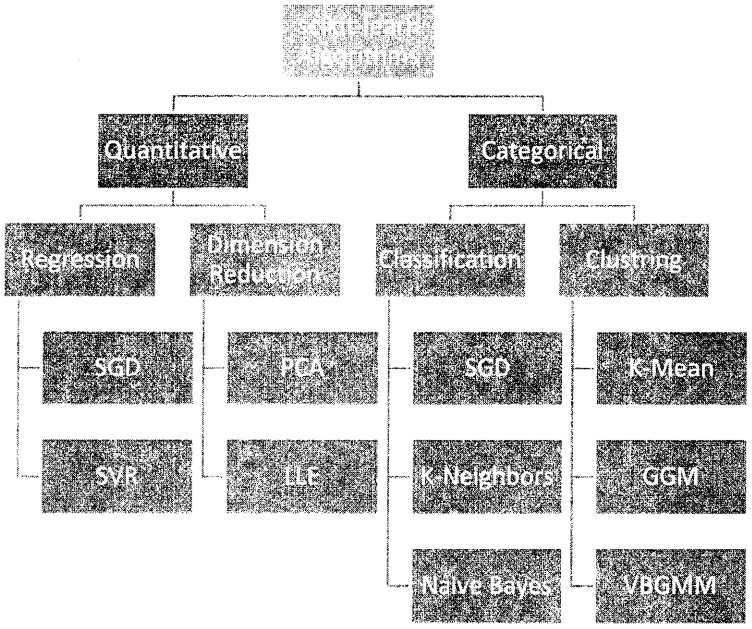


Figure 93: Summary of algorithms enabled in the python machine learning library (scikit-learn).

Deep Learning Models using TenserFlow and Keras

TensorFlow is an end-to-end open-source platform for machine learning. It has a comprehensive, flexible ecosystem of tools, libraries, and community resources that lets researchers push the state-of-the-art in ML and developers easily build and deploy ML-powered applications. TensorFlow was originally developed by researchers and engineers working on the Google Brain team within Google's Machine Intelligence Research organization to conduct machine learning and deep neural networks research. The system is general enough to be applicable in a wide variety of other domains, as well.

One of the most powerful and easy-to-use Python libraries for developing and evaluating deep learning models is Keras; It wraps the efficient numerical computation libraries Theano and TensorFlow. The advantage of this is mainly that you can get started with neural networks in an easy and fun way. Keras is an open-source software library that provides a Python interface for artificial neural networks. Keras is a powerful and easy-to-use free open-source Python library for developing and evaluating deep learning models.

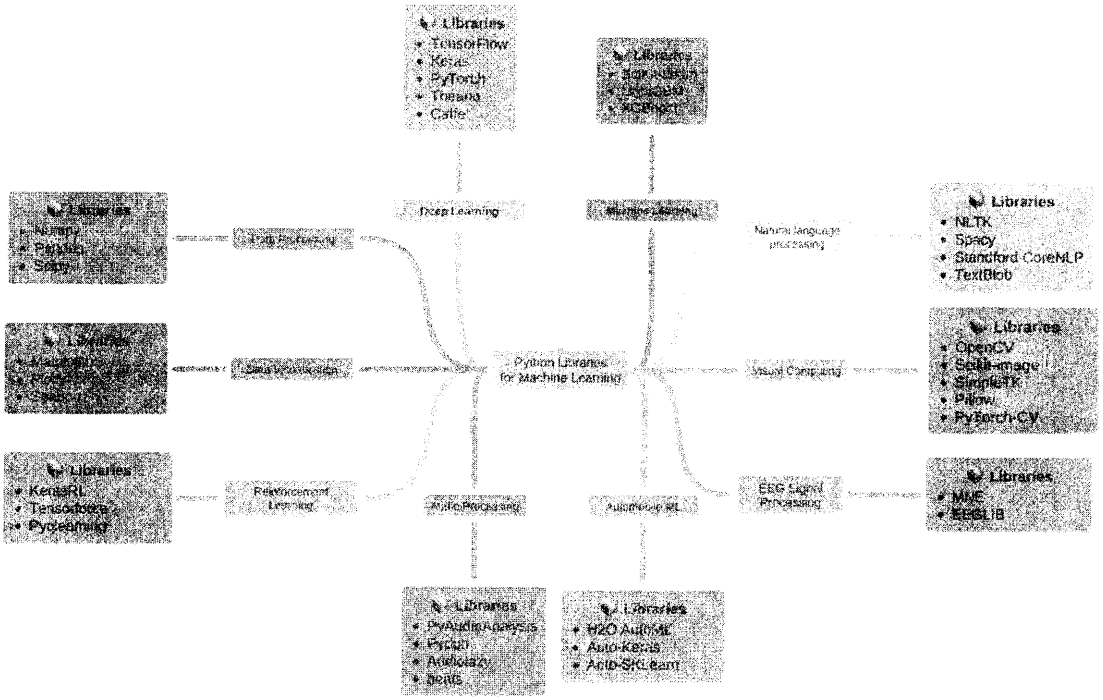


Figure 94: Mind map of most popular python library applying machine learning algorithms.

Data Migration From MATLAB using SciPy:

SciPy is a free and open-source Python library used for scientific computing and technical computing. SciPy contains modules for optimization, linear algebra, integration, interpolation, special functions, FFT, signal and image processing, ODE solvers and other tasks common in science and engineering.

```
import matplotlib.pyplot as plt
plt.subplot(2,1,1)
plt.plot(t,x1,t,x2,t,x3)
plt.subplot(2,1,2)
plt.plot(t,x)
plt.xlabel('Time (s)')
plt.ylabel('Amplitude');
```

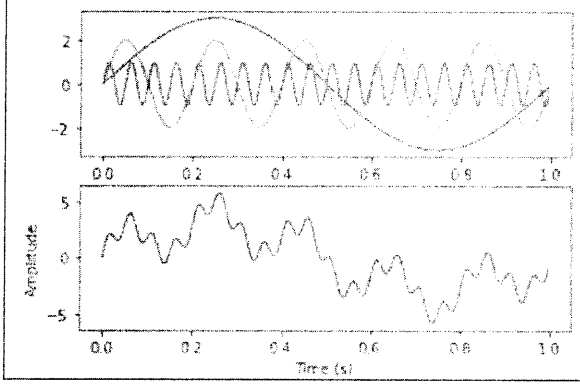


Figure 95: Plotting different signal representation using matplotlib python library.

Feature Extraction using eeglib:

An open-source Python library which is a powerful feature extraction tool oriented towards EEG signals and based on sliding windows. The design of eeglib is oriented towards compatibility with the most used machine learning and data analysis libraries for Python, so its output can be an input for them, which simplifies the workflow (Cabañero-Gomez et al., 2021).

Table 10: Summary of the features that can be extracted through eeglib by default.

Single channel	Transformations	Fast Fourier Transform Discrete Wavelet Transform Power Spectral Density
	Frequency domain	Band Power
	Time domain	Hjorth Parameters
	Complexity	Detrended Fluctuation Analysis Sample Entropy Lempel–Ziv Complexity Petrosian Fractal Dimension Higuchi Fractal Dimension
Dual channel		Pearson Cross-Correlation Coefficient Dynamic Time Warping Synchronization Likelihood

PyEEGLTE (Python Library for EEG Load, Transform and Extract)

PyEEGLTE is a high-level open-source library that allows researchers to handle classical datasets from closed or old sources in the python environment, then design and test deep learning models such as CNN, LSTM, and RNN with minimal programming knowledge.

This project was built on top of the latest python opensource libraries such as MNE (visualization), MatPlotLib (Plotting), sklearn (Artificial intelligence), TensorFlow (Deep Neural Networks), NumPy (Mathematical computing), pandas (Data Handler), Scipy (Scientific computing) and other libraries.

These libraries enable the project to perform complex data analysis and machine learning tasks with ease. The use of Python also allows for easy collaboration and sharing of code among researchers.

Since we aim to come up with practical and modern enhancement of the old classification and decoding methodology of EEG, where the cloud, on-edge application market is growing exponentially, those modern applications came with great opportunities and challenges as well, where on-edge resources such as power, processor, and memory are limited, and the application should optimize sent data to a minimum to balance between network utilization and speed.

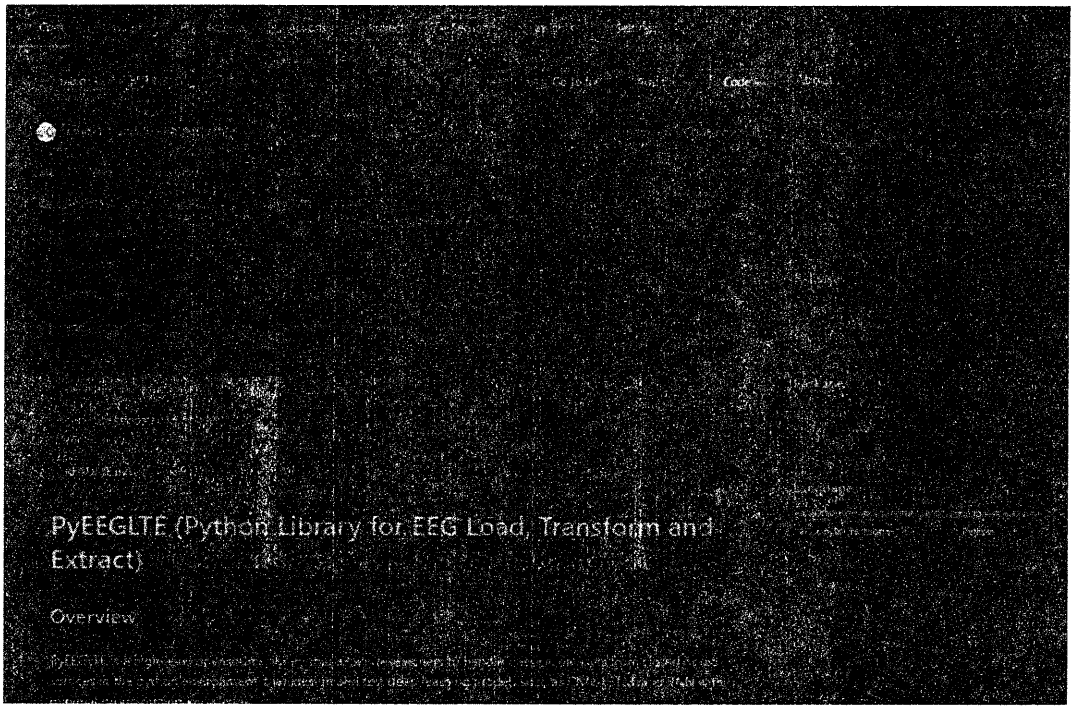


Figure 96: PyEEGLTE (Python Library for EEG Load, Transform and Extract), project page in GitHub.

Considering all these constraints, we target to optimize accuracy with minimal training data, and we programmed PyEEGLTE to give us the ability to add additional optimization and performance indicator to evaluate classifier and to be able to tune data source and channel size. So, we added two keys to performance measurement: the number of electrodes (signal channels) and the training samples.

Appendix B:
Data Preparation

Preprocessing EEG signal is only the first step for processing the data and transforming it for the final results, so while all previous processes are related to preparing signals, regardless of the tool which data will be further, in this step, we will prepare data to fit with the final processing tool which in our case the MATLAB and it is specialized too EEGLAB. Furthermore, since the available MATLAB for us is the most public one, a stand-alone license running on a single server or computer, we ended up with limited processing resources that should be used carefully, and we should prepare data with this constraint in mind.

Data Representation

EEGLAB is an open-source GUI MATLAB package for analyzing continuous and event-related electroencephalogram (EEG), magnetoencephalogram (MEG), and other electrophysiology data. The tool provides ICA, artifact removal, and numerous helpful presenting data. EEGLAB is running on top of MATLAB. It provides features for uploading EEG data, details related to channels and events, visual analytics, and multi-trial ERP-image plotting). The functionalities of EEGLAB are divided into three categories. Experienced MATLAB users can create batch analysis scripts utilizing EEGLAB data structures and stand-alone signal processing tools. There is much help and lesson information supplied. EEGLAB enables adding additional plug-ins to collaborate with the EEG researching community. For more information about EEGLAB, refer to the authors Arnaud Delorme and Scot Makeig (Delorme & Makeig, 2004).

The main problem with EEGLAB that it didn't accept data on it is row format, to be able to process EEG data it should be restructured and formatted to EEGLAB project format, this action can be done by mapping the SEED experiment to an accepted format such EDF/EDF+,BDF,GDF,CNT or VHR, unfortunately none of those format exists for SEED, which require us to structure the EEGLAB dataset manually.

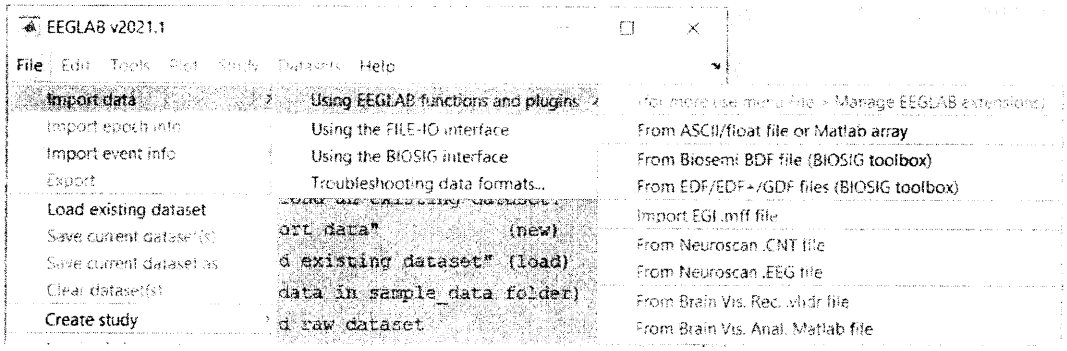


Figure 97: Available EEGLAB import formats.

As observed in the EEGLAB dataset dialogue in Figure **98**, several adjustments must be done ahead of importing the raw data. These options pertain to the nature of the experiment, the signal format, and specific data pertaining to event representations.

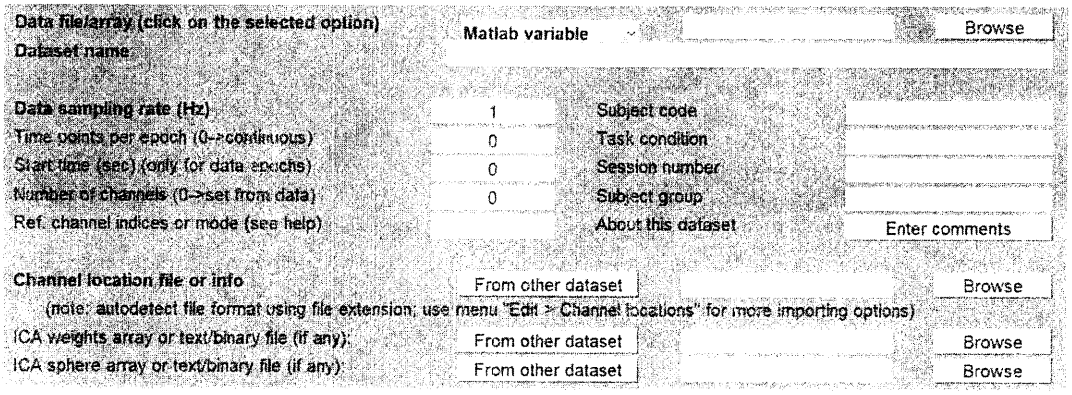


Figure 98: EEGLAB import raw data dialogue.

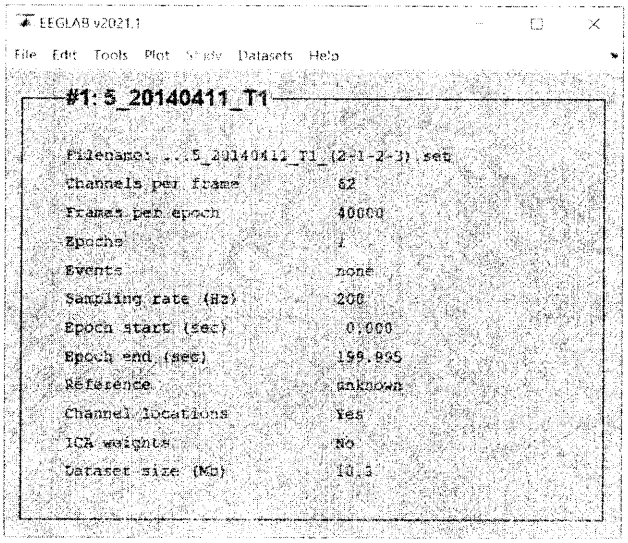


Figure 99: Our SEED EEG dataset settings.

Thus, the initial duty is to construct the channel location file information in CED format; this file should contain all channel details such as channel IDs, labels, radius, locations, and radius. According to the SEED description, the tests were designed utilizing the International (10–20) system which presented in Figure **100**, which is a widely accepted approach for standardizing EEG electrode placement and assuring uniform interelectrode spacing.

Electrode placement is proportional to the size and shape of the skull, with spacing between close electrodes equivalent to 10 percent or 20 percent of the skull's overall front-back or right-left distance. The second purpose is to generate events on the signal that reflect the generated emotions; this includes connecting several experiments and aligning events and labels in a consistent order.

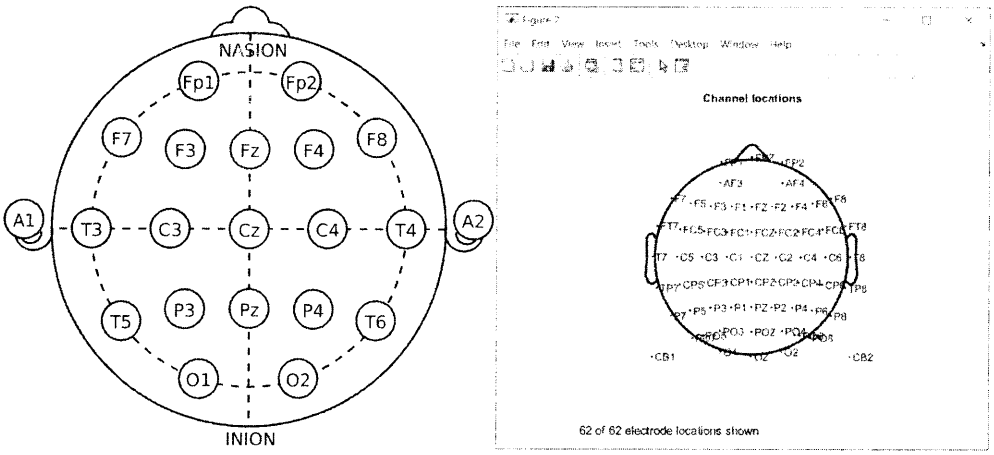


Figure 100: The international 10-20 system's electrode placements for EEG recording (Rojas et al., 2018) (right), EEGLAB channel location displayed for SEED dataset (right).

We developed an EEGLAB dataset consisting of four events, each having ten thousand recordings and fifty seconds latency see [Table **11**: EEGLAB dataset event configuration.]; this dataset may be used to analyze the SEED utilizing a diverse range of EEGLAB features such as signal visualization in Figure **102** and Artifact detection and removal tools see Figure **105**. Although we did not need to apply these tools to our EEG recordings, since the

SEED dataset had already undergone artifact removal, we could use them to further enhance the quality of our EEG signals. For example, we could use the AAR plugin to automatically detect and correct any remaining ocular, muscle, or channel artifacts with default settings. We could also use ICA to separate the EEG signals into 64 components and check them using the EEGLAB GUI. We could discard the components with obvious artifact characteristics, such as high amplitudes, irregular shapes, or non-scalp distributions. By doing so, we could obtain a more refined EEG signal for further analysis.

The power of EEGLAB enables a simple interactive graphical interface where the specialized individual with no deep technology knowledge can still explore and edit the data set by eye and can diagnose signals from different perspectives, such as power, time, event, artifacts channels overlapping, and others.

Table 11: EEGLAB dataset event configuration.

Type	Position	Latency
Neutral	1	50
Positive	10001	50
Neutral	20001	50
Negative	30001	50

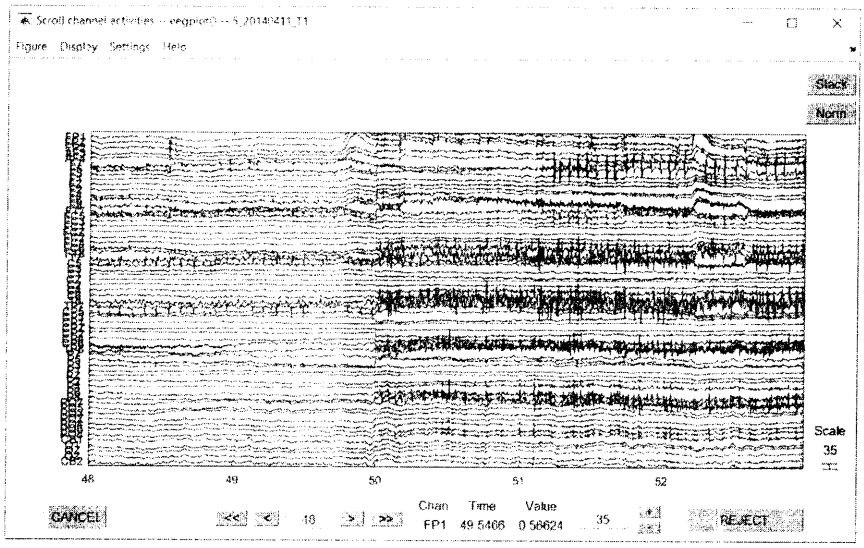


Figure 101: Channel activities for the first event.

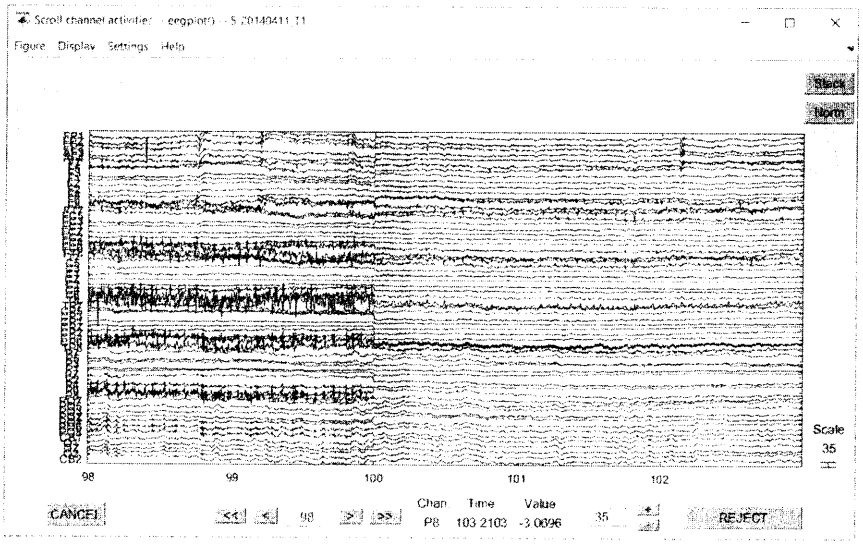


Figure 102: Channel activities for the second event.

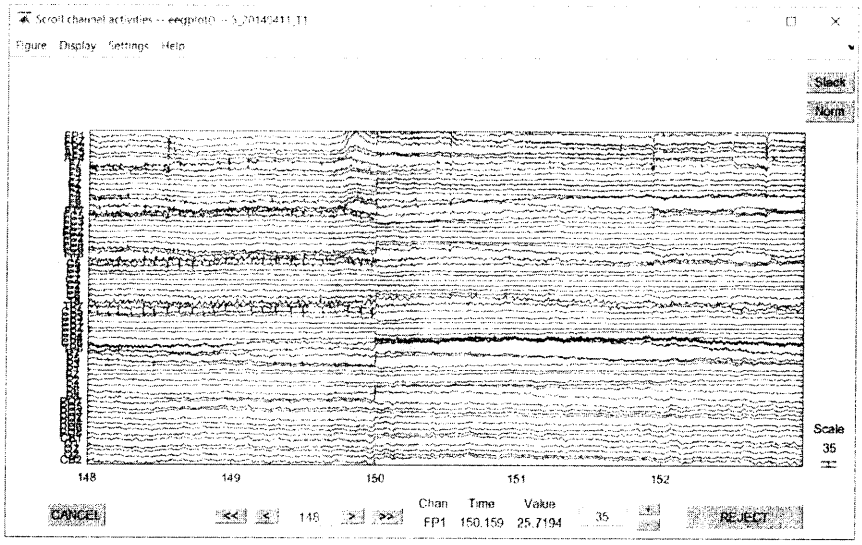


Figure 103: Channel activities for the third event.

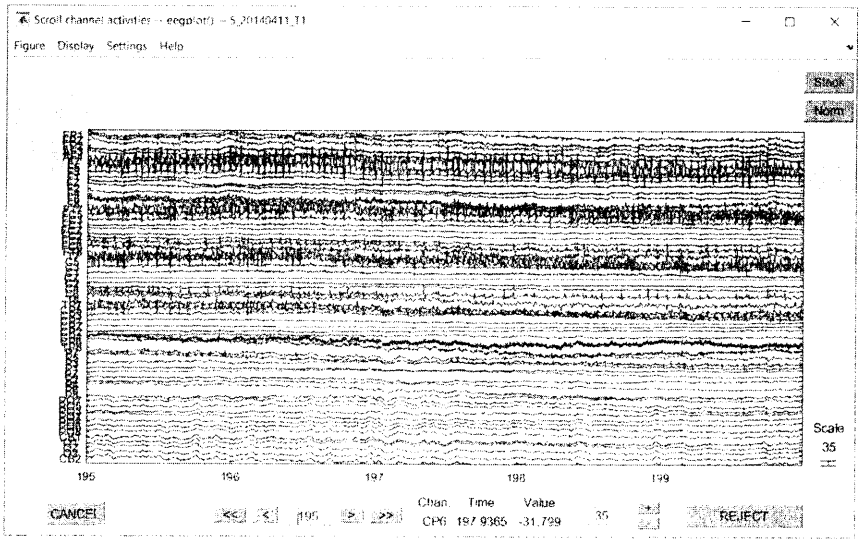


Figure 104: Channel activities for the last event.

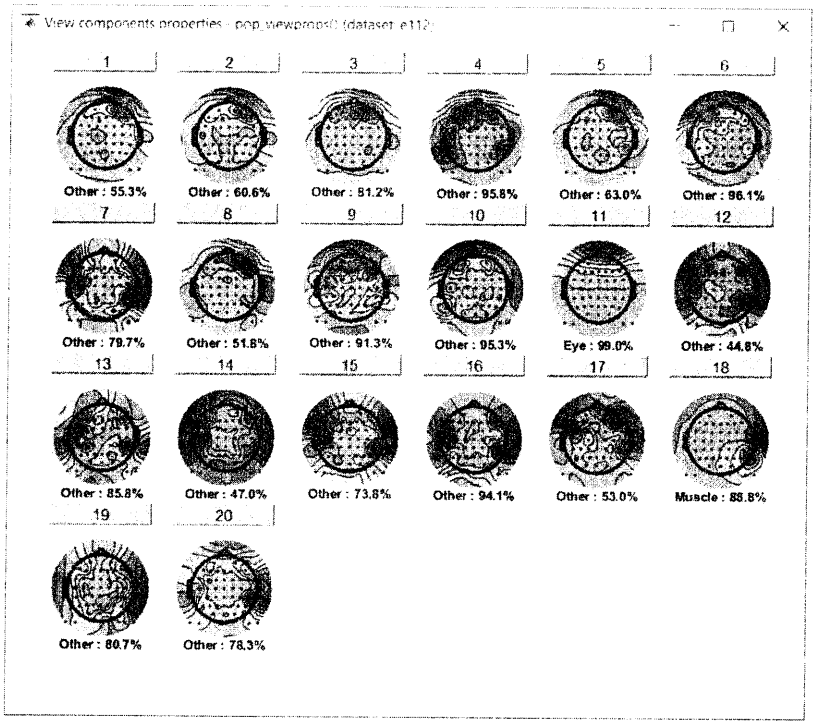


Figure 105: Sample of EEGLAB artifact detection tool.

Appendix C:
Unsupervised Clustering

The system uses uncategorized data and learns without any supervision in Unsupervised Learning. In the unlabeled data and answer, the machine detects a pattern. Unsupervised learning is categorized into two kinds: clustering detection algorithms, which are especially helpful in detecting potential fraud or defective products, and association mining, which uses algorithms to find associations between data points to identify what products are frequently purchased together, for example.

Even though clustering algorithms are classified in a variety of ways in the literature, one easy classification divides them into two categories:

Clustering based on parameters. In general, parametric approaches reduce the cost of each instance cluster assignment by minimizing a cost function or an optimality criterion. This algorithm seeks to solve an optimization problem to meet the model's optimality requirement. This frequently entails lowering the cost function. This approach frequently makes some assumptions about the data structure underneath. The K-means algorithm is one of these strategies that can be used.

Clustering is not parametric. Agglomerative and divisive algorithms, often known as hierarchical algorithms, are two notable instances of the non-parametric approach to clustering. In each cycle, both techniques measure the dissimilarities among the current cluster set. Some clusters are merged using agglomerative methods, while others are separated using divisive algorithms.

Running simple random function of the sklearn dataset in below, which is part of the training attached to the Python Data Science Handbook by Jake VanderPlas.(VanderPlas, 2016)

```
from sklearn.datasets import make_blobs
X, y_true = make_blobs(n_samples=300, centers=4,
                       cluster_std=0.60, random_state=0)
plt.scatter(X[:, 0], X[:, 1], s=50);
```

https://colab.research.google.com/github/jakevdp/PythonDataScienceHandbook/blob/master/notebooks/05.11-K-

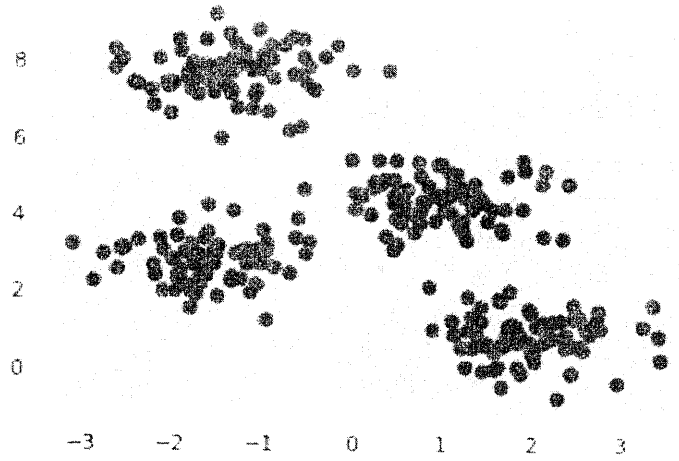


Figure 106: Visualizing of a two-dimensional dataset containing four distinct blobs before applying K-Means.

Hierarchical Clustering:

Hierarchical clustering, also known as hierarchical cluster analysis, is a non-parametric clustering technique that attempts to construct a cluster hierarchy. There are several hierarchical clustering techniques available, each with its distance measure. We tested them through their paces using the MATLAB toolbox.

Clustering with K-means:

Based on the distance metric, the k-means method can be used to partition data points into K groups. K-means clustering can be used in the context of EEG for similar clustering segments in the EEG recording.

The algorithm's input parameter is the number of clusters K. If this number is uncertain, we can use the k-means approach to find the best one by running it multiple times with different numbers of clusters. The ability to use k-means for really large datasets is an undeniable benefit.

```
from sklearn.cluster import KMeans
kmeans = KMeans(n_clusters=4)
kmeans.fit(X)
y_kmeans = kmeans.predict(X)
plt.scatter(X[:, 0], X[:, 1], c=y_kmeans, s=50, cmap='viridis')
```

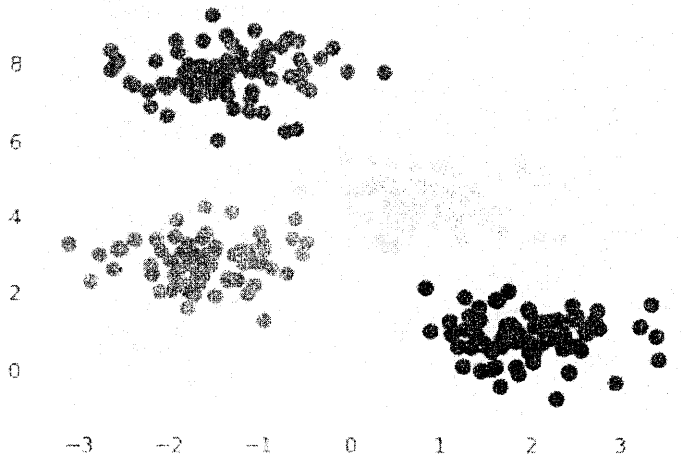


Figure 107: Visualizing four distinct blobs clustering after applying K-Means.

ملخص

تستكشف هذه الرسالة العلمية العديد من أساليب التعلم الآلي والتعلم العميق لإنشاء أنظمة التعرف على المشاعر في الوقت الحقيقي بناءً على إشارات الدماغ الكهربائية المعروفة ب (EEG). نهدف في هذه الرسالة إلى تعزيز تصنيف وفك تشفير إشارات الدماغ الكهربائية على نطاق واسع وتقليل الفجوة بين التطبيقات الحقيقية والأبحاث العلمية الصرفة في هذا المجال. استخدمنا بعض من أكثر مجموعات البيانات شيوعًا للمشاعر واستكشفنا كيف تؤثر عشوائية البيانات على النتائج. كما قدمنا طريقة اختيار ديناميكية لقنوات التقاط إشارات الدماغ الكهربائية باستخدام أساليب ترشيح مختلفة مبنية على التقييم المستمر للأداء مع نقل المعرفة للعثور على أفضل قنوات الالتقاط. كما قدمنا نهجًا جديدًا لاستخراج واختيار سمات البيانات باستخدام نقل المعرفة لتجاوز مشكلات الانحراف الزائد الناجمة عن الافراط في استخدام البيانات المحلية. كما قسمنا سمات البيانات إلى مجموعات مركزية وغير مركزية، حيث تكون المركزية أكثر دقة ولكنها تقوم باستهلاك المزيد من الموارد. وقمنا بمقارنة العديد من طرق التصنيف الموجه (supervised) وغير الموجه (unsupervised) ، مثل Decision Tree, Nearest Neighbor, Support Vector Machine, Naive Bayes, Hierarchical Clustering, K-mean, Clustering. كما قيمنا العديد من أساليب التعلم العميق، مثل الشبكات العصبية الاصطناعية والعميقة. صممت الرسالة نموذجًا بتحويلات بسيطة (convolutions) تعمل بشكل جيد مع العينات الصغيرة، وتحتوي على تنسيقات مدخلات ومخرجات متسقة لنقل المعرفة والوزن في الوقت الحقيقي. تستغل الرسالة فعالية شبكات التعلم العميق العصبية التكرارية (Convolutional Neural Networks) وشبكات العصب الطويل المتسلسل (Recurrent Neural Networks) لإجراء الاستخراج والاختيار التلقائي للميزات في النماذج المركزية وغير المركزية. يمكن لهذه النماذج التعامل مع تنسيقات مختلفة للبيانات المدخلة اعتمادًا على الموارد المتاحة والدقة والكفاءة المطلوبة، مما يؤدي إلى مستويات مختلفة من الدقة والأداء. بشكل عام، تعد هذه الرسالة العلمية مساهمة قوية في مجال تعرف المشاعر عن طريق الإشارات الدماغية الكهربائية، وتساعد في تعزيز التطبيقات الحقيقية للأبحاث العلمية البحتة في هذا المجال. كما تفتح أبوابًا جديدة للبحث والتطوير في هذا المجال المتطور والمتنامي، وتشير إلى الاستخدامات الواسعة التي يمكن تطبيقها في صناعات مختلفة.

الكلمات المفتاحية: الذكاء الاصطناعي، اكتشاف المشاعر، هندسة طبية، تخطيط الدماغ الكهربائي، تصنيف إشارات الدماغ الكهربائية ، واجهات الدماغ-الحاسوب، تطبيقات BCI، تصنيف المشاعر، معالجة الإشارات، استخراج ميزات البيانات، التصنيف.

جامعة البحرين

كلية الهندسة



قسم الهندسة الكهربائية والالكترونية

تحسين دقة التعرف على المشاعر وفك تشفيرها عبر اشارات الدماغ الكهربائية باستخدام تعلم الآلة

أطروحة مقدمة كجزء من متطلبات الحصول على درجة دكتوراة
الفلسفة في الهندسة الكهربائية

اعداد

محمد نبيل السلوت

20150117

اشراف

إبراهيم عبدالله مطر

أستاذ دكتور

جامعة البحرين

مملكة البحرين

مايو 2023