

An investigation into decoding imagined and inner speech processes from commercial EEG devices

Vladislav Shishkin

Master of Science in Computer Science
The University of Bath
2021-2022

This dissertation may be made available for consultation within the University Library and may be photocopied or lent to other libraries for the purposes of consultation.

An investigation into decoding imagined and inner speech processes from commercial EEG devices

Submitted by: Vladislav Shishkin

Copyright

Attention is drawn to the fact that copyright of this dissertation rests with its author. The Intellectual Property Rights of the products produced as part of the project belong to the author unless otherwise specified below, in accordance with the University of Bath's policy on intellectual property (see https://www.bath.ac.uk/publications/university-ordinances/attachments/Ordinances_1_October_2020.pdf).

This copy of the dissertation has been supplied on condition that anyone who consults it is understood to recognise that its copyright rests with its author and that no quotation from the dissertation and no information derived from it may be published without the prior written consent of the author.

Declaration

This dissertation is submitted to the University of Bath in accordance with the requirements of the degree of Master of Science in the Department of Computer Science. No portion of the work in this dissertation has been submitted in support of an application for any other degree or qualification of this or any other university or institution of learning. Except where specifically acknowledged, it is the work of the author.

Code Availability

All code written as part of this project is available at: https://github.com/vshis/emotiv_epoc_eeg_decoding_imagined_and_inner_speech.

Abstract

Verbal communication is a key process allowing individuals to effectively exchange information. We investigate the use of a mobile, affordable, 14-channel Electroencephalography (EEG) headset in translating neural activity into speech, to be utilised by individuals with speech impairments. Defined interpretation and characterisation of the imagined and inner speech phenomena are derived, contributing to the brain-computer interface research field with a novel open-access set of low-fidelity EEG data covering 16 English phones. We study whether either of the speech modalities results in higher classification performance; discovering it is highly dependent on the individual performing the task. Furthermore, an investigation of various data decomposition techniques is performed. We achieve a reliable above-chance classification accuracy of 8% on 16-classes and 90% on binary classification using minimally-processed EEG signals, suggesting its superiority over preprocessing and statistical feature extraction alternatives, which include linear features, non-linear features, and Mel Frequency Cepstral Coefficients.

Contents

1	Introduction	1
2	Literature and Technology Survey	2
2.1	Electroencephalography	2
2.1.1	Comparison with Other Neuroimaging Techniques	2
2.1.2	The Electroencephalography Acquisition Principles	3
2.1.3	The Emotiv EPOC+ Headset	4
2.1.4	Neuronal Information Transfer Mechanisms	5
2.2	Preprocessing	6
2.3	Feature Extraction and Classification	7
2.3.1	Feature Extraction Methodologies	7
2.3.2	Classification with Machine Learning	8
2.4	Experimental Paradigm Design	8
2.4.1	Prompt Selection	9
2.4.2	Imagined Speech and Inner Speech	10
2.5	Hypotheses	11
2.6	Summary	12
3	Methodology	13
3.1	Experimental Paradigm	13
3.1.1	Paradigm Description	13
3.1.2	Paradigm Limitations	14
3.2	Data Acquisition	15
3.2.1	Software Packages	15
3.2.2	Device Drift Correction	16
3.3	The Dataset	16
3.4	Dataset Preprocessing	17
3.4.1	Raw Signal Analysis	17
3.4.2	Digital Filtering	19
3.4.3	Individual Component Analysis	19
3.4.4	Data Epoching	21
3.5	Feature Extraction	21
3.6	Classification	22
3.6.1	K-Nearest Neighbours	22
3.6.2	Gaussian Naïve Bayes	22
3.6.3	Linear Discriminant Analysis	23
3.6.4	Support Vector Machine	23

3.6.5	Decision Trees with Adaptive Boosting	24
3.6.6	EEGNet	24
3.6.7	Hyperparameter Sweeps	25
3.7	Summary	26
4	Results	27
5	Discussion	34
5.1	Training and Testing Data Correlation	34
5.2	Participant 00 Dataset	34
5.3	Conventional Classifiers	35
5.3.1	K-Nearest Neighbours	35
5.4	EEGNet	36
5.5	Data Decomposition Types	36
5.6	Imagined Speech and Inner Speech	38
5.7	Summary	39
6	Conclusions and Future Work	40
Bibliography		41
A	Participant Information Sheet	47
B	Participant Consent Form	53
C	Dataset File Tree	56
D	EEGNet Architecture	57
E	EEGNet Hyperparameters	58
F	Full Results	59
F.1	Testing Accuracy results	59
F.2	Training Accuracy results	63

List of Figures

2.1	(a) Diagram of the electric dipole model, currents and the differential amplifier. [22] (b) Circuit diagram representation of the electric dipole.	4
2.2	Emotiv EPOC+ placed on the researcher's head with labelled AF3 and AF4 channels. (a) Front. (b) Back.	5
2.3	(a) Map of electrode positioning according to the international 10-20 system [30]. (b) Map of electrode positioning of the Emotiv EPOC+ headset.	5
2.4	Recorded potentials of neural oscillations, their frequency ranges and attributions to various brain activities. Adapted [34].	6
2.5	Diagram depicting a typical pipeline of procedures incorporated in the processing of EEG signal and the direct approach of learning from raw data.	9
2.6	Diagram summarising speech modalities discussed and how they are produced. Highlighting exaggerates on the modalities studied in this work.	11
3.1	Diagram depicting the state flow during the experiment, where the circles represent the states and the arrows represent the transitions between them.	14
3.2	Illustration of this study's experimental paradigm implemented within OpenVibe [67] Designer software package, where the modules represent the various components of the experimental paradigm, acquisition client integration, input via the timeline generator, and output via visual cues, auditory cues and CSV files, whilst the lines represent the output/input links between the modules.	16
3.3	Time domain graphs, both demonstrate a sample of the first 10 seconds of data at identical scale, where the x-axis is time in seconds and the y-axis is the scaled signal amplitude in volts. (a) P00 raw EEG signal collected with the Emotiv EPOC headset. (b) P01 raw EEG signal collected with the Emotiv EPOC+ headset.	17
3.4	Power spectral density graphs, where x-axis is the frequency in Hz and y-axis is the power density in $\mu V^2/Hz$ (dB) (a) P00 raw EEG signal collected with the Emotiv EPOC headset. (b) P04 raw EEG signal collected with the Emotiv EPOC+ headset. (c) P04 1 Hz high-pass filtered signal, where the vertical dotted line represents the filter cut-off point.	18
3.5	Time domain graphs of P03 imagined speech raw EEG signal between 65 and 75 seconds at identical scale, where the x-axis is time in seconds and the y-axis is the scaled signal amplitude in volts. (a) Bad channel T8 before interpolation. (b) Bad channel T8 after interpolation.	19
3.6	Time domain P04 inner speech raw EEG signal, where the x-axis is time in seconds and the y-axis is the scaled signal amplitude in volts. Vertical green line highlights an EOG artefact. (a) Pre-ICA raw signal. (b) Post-ICA raw signal.	20
3.7	P04 imagined speech topographic map plot of the individual components.	21

3.8	Binary classification with two features using KNN algorithm, where $k = 3$. (a) New query point, marked with symbol ?, where the dotted lines indicate the three nearest data points. (b) Query point assigned to Class 2 as the majority.	23
3.9	Binary classification with two features using SVM algorithm, where w is the margin distance. (a) Potential classification hyperplanes constructed. (b) Optimal classification hyperplane found.	24
3.10	EEGNet architecture, where the layer dimensions correspond to a 3-second, at 256 Hz, across 14 channels, EEG signal input.	25
3.11	Examples of validation accuracy graph generated during a hyperparameter sweep. (a) Poor accuracy graph. (b) Desired accuracy graph.	26
4.1	AB classification accuracy using different datasets.	32
4.2	Gaussian NB classification accuracy using different datasets.	32
4.3	LDA classification accuracy using different datasets.	33
4.4	EEGNet classification accuracy using different datasets.	33
5.1	Graphs of KNN accuracy against the value of k across datasets. (a) Graph representing the accuracy trends in datasets for P01-P04 and FEIS. (b) Graph representing the accuracy trends in binary classification.	36
5.2	Mean accuracy across datasets on each data type using different classifiers. (a) AB. (b) Gaussian NB. (c) LDA. (d) EEGNet.	37
5.3	Mean accuracy of different classifiers on imagined speech and inner speech between different participants. (a) AB. (b) Gaussian NB. (c) LDA. (d) EEGNet.	38
C.1	A portion of the study's dataset [64] file hierarchy, depicting how the raw and the labelled EEG recording files are structured with the relevant sequence and Python files.	56

List of Tables

2.1	Consonant phoneme types studied.	10
2.2	Vowel phoneme types studied.	10
4.1	P00 testing accuracy classification results on conventional algorithms, where e stands for the number of estimators in AB, lr stands for the learning rate of AB and k stands for the number of nearest neighbours in KNN.	28
4.2	P01 testing accuracy results on conventional algorithms, where cyan and red colours highlight the best and the worst mean classifier accuracy respectively, e stands for the number of estimators in AB, lr stands for the learning rate of AB and k stands for the number of nearest neighbours in KNN.	29
4.3	Binary and FEIS dataset classification testing accuracy results on conventional algorithms, where cyan and red colours highlight the best and the worst mean classifier accuracy respectively, e stands for the number of estimators in AB, lr stands for the learning rate of AB and k stands for the number of nearest neighbours in KNN.	30
4.4	EEGNet classification results, where cyan and red colours highlight the best and the worst mean classifier accuracy respectively.	31
D.1	EEGNet architecture, where F_1 is the number of temporal filters, D is the number of spatial filters, F_2 is the number of pointwise filters, S is the EEG device sampling rate, C is the number of EEG channels and p is the probability of dropout.	57
E.1	EEGNet hyperparameter configurations.	58
F.1	P02 testing accuracy classification results on conventional algorithms, where cyan and red colours highlight the best and the worst mean classifier accuracy respectively, e stands for the number of estimators in AB, lr stands for the learning rate of AB and k stands for the number of nearest neighbours in KNN.	60
F.2	P03 testing accuracy classification results on conventional algorithms, where cyan and red colours highlight the best and the worst mean classifier accuracy respectively, e stands for the number of estimators in AB, lr stands for the learning rate of AB and k stands for the number of nearest neighbours in KNN.	61
F.3	P04 testing accuracy results on conventional algorithms, where cyan and red colours highlight the best and the worst mean classifier accuracy respectively, e stands for the number of estimators in AB, lr stands for the learning rate of AB and k stands for the number of nearest neighbours in KNN.	62

List of Abbreviations

- AB Decision Trees with Adaptive Boosting classifier
AdaBoost Adaptive Boosting
ANN Artificial Neural Network
BCI Brain Computer Interface
BSS Blind Source Separation
CCV Channel Cross-Covariance
CE Cross-entropy
CNN Convolutional Neural Network
DNN Deep Neural Network
DT Decision Tree
ECOG Electrocorticography
EEG Electroencephalography
EOG Electrooculography
fMRI functional Magnetic Resonance Imaging
fNIRS functional Near-Infrared Spectroscopy
GNAA Gradient Norm Adversarial Augmentation
ICA Independent Component Analysis
IPA International Phonetic Alphabet
KNN K-Nearest Neighbours
LDA Linear Discriminant Analysis
LF Linear Features
MEG Magnetoencephalography
MFCC Mel Frequency Cepstral Coefficients
ML Machine Learning
NB Naïve Bayes

- NLF Non-Linear Features
NLL Negative Log Likelihood
PET Positron Emission Tomography
PSD Power Spectral Density
RBF Radial Basis Function
RF-RFE Random Forest and Recursive Feature Elimination
RNN Recurrent Neural Network
SC Stratum Corneum
SEEG Stereoelectroencephalography
SGD Stochastic Gradient Descent
SNR Signal-to-Noise Ratio
SSI Silent Speech Interface
STFT Short Time Fourier Transform
SVM Support Vector Machine
VOCA Voice Output Communication Aids
WandB Weights & Biases

Chapter 1

Introduction

Daily communication relies upon a complex process resulting in the ability to speak. Conditions such as motor neurone disease, amyotrophic lateral sclerosis and brainstem stroke compromise this process, resulting in speech impairment. Individuals with speech impairments resort to communication aids including signing, communication boards, or eye-tracking technologies in more severe scenarios. Multiple methods have been proposed to restore individuals' ability to verbally communicate through decoding speech from neural activity [1]. Brain-computer interfaces (BCI) restore speech by decoding cortical brain activity [2, 3, 4]. Silent speech interfaces (SSI) enable communication without creating acoustic signals by acquiring sensor data from articulatory gestures, neural pathways, or the brain [5]. Other methods gather biosignals, providing the user with articulatory feedback of their own voice production, assisting articulatory awareness when producing speech [6, 7].

This project will address BCIs as a promising technology for decoding and transforming neural activities into control commands for wheelchairs, prostheses, voice output communication aids (VOCA) and other virtual interface devices [8], focusing on VOCA development. These systems use a range of modalities to capture the electromagnetic activity of the brain, including electrocorticography (ECoG), electroencephalography (EEG), magnetoencephalography (MEG), stereoelectroencephalography (SEEG), positron emission tomography (PET), functional magnetic resonance imaging (fMRI) and functional near-infrared spectroscopy (fNIRS). A commercially available, non-invasive solution for speech synthesis with the Emotiv EPOC+ [9] EEG headset will be used in this study.

Chapter 2

Literature and Technology Survey

This chapter outlines the problem task and description of the proposed brain activity measurement methodology in Section 2.1. Section 2.2 describes the methodologies employed to remove noise from raw EEG signals. Common preparation methods of resulting signal data for feeding into machine learning (ML) algorithms are analysed in Section 2.3, with the identification and discussion of which ML algorithms have gained success and popularity in decoding imagined speech. Section 2.4 discusses considerations when designing an experimental paradigm for imagined speech research using EEG. Throughout the chapter, gaps in current research are identified, with Section 2.5 explicitly stating the derived hypotheses.

2.1 Electroencephalography

The brain is the least understood organ in the human body due to the complexity and uniqueness of its structure. Currently, technologies exist allowing researchers to observe brain functionality in real-time, extracting useful findings and data. Some of this data can be used (with limitations, which will be explored in this project), to translate thought processes into vocalised speech, with techniques such as EEG: the non-invasive technique of measuring varying electric fields of the brain with electrodes placed on the scalp [10]. EEG is relatively cheap and provides comparatively good temporal resolution. However, it has a low signal-to-noise ratio (SNR) and is frequently corrupted by artefacts, such as ambient noise, salt bridges and electrode movement. It also offers limited spatial resolution [2].

2.1.1 Comparison with Other Neuroimaging Techniques

Several alternative neural activity measuring methods to EEG exist, such as fMRI, PET and MEG. fMRI detects changing concentrations of oxygenated and deoxygenated haemoglobin, which carries different magnetic properties depending on oxygenation levels [11]. This method has high spatial resolution over the brain but a low temporal resolution compared to EEG, due to the low hemodynamic response speed, leading to consistent accuracy drops in acquired data [12]. PET creates brain images by employing radioactive tracers: areas with high concentrations of tracer indicate high blood concentration. Brain functionality information is obtained by comparing radioactivity distribution images [11]. This method outperforms invasive brain mapping techniques; however, due to the use of radioactive tracers and lower temporal and spatial resolutions, PET is sub-optimal to EEG and fMRI [13]. Both MEG and EEG measure

brain electromagnetic activity: MEG utilises magnetic fields generated by neural currents whilst EEG uses electric potentials. MEG provides less spatially distorted signals and is less sensitive to motion-related disturbances [13, 14]. Despite the advantages over EEG; MEG, fMRI and PET are unsuitable as practical, commercially available tools of communication interfaces due to prolonged setup times, lack of portability and price [2].

Another set of alternative methods includes ECoG and SEEG. The former places electrodes directly onto a small area of the cerebral cortex. The latter inserts thin cylindrical electrodes through the scalp into the brain. SEEG electrodes have multiple recording contacts along their length, allowing data to be acquired from the deeper brain and the shallow cortex layer, overall offering a better understanding of brain activity [15]. Both techniques provide superior results to EEG: the acquired data contains fewer artefacts, and due to less spatial summation and phase cancellation, more data can be collected in the gamma frequency range [16]. These methods are invasive and are not suitable options as commercially available devices.

The last alternative to EEG considered in this work is fNIRS. This method is analogous to fMRI and it equally suffers from low temporal resolution due to slow hemodynamic changes, and offers a lower spatial resolution than fMRI [17]. fNIRS has multiple advantages over fMRI and EEG: it can be used around other electrical devices, having fewer electromagnetic-related and motion artefacts. It is non-invasive and is comparable to EEG in terms of portability, at a higher price [18]. Previous research has suggested and studied the combined benefits of using fNIRS and EEG simultaneously, resulting in improved performance [17, 19]. Currently, this approach is not commercially feasible due to its high cost and complex setup.

2.1.2 The Electroencephalography Acquisition Principles

EEG records brain electric potential oscillations using electrodes located on the scalp [20] and can be summarised as a measure of synchronised activity occurring from around 60 million neurons firing simultaneously. Figure 2.1a shows electrical potential signals in the brain cortex modelled as dipoles. The poles act as the source and sink of the created ionic currents. The three types of current inflow into a cortical patch are radial, oblique (shown in Figure 2.1a) and tangential. The latter current flow does not propagate to the scalp surface and cannot be captured by EEG devices [21], resulting in signals lost with a tangential net current.

EEG electrodes acting as voltage meters are attached to chosen locations. To ensure the electrodes are on the ionic current line they are placed on the scalp at the top of the Stratum Corneum (SC) on the outer epidermis layer. An electrolytic gel is applied to electrodes forming a flexible lattice for charge conductivity. The gold plating on electrodes prevents additional signal noise and reduces signal distortion in EEG frequency ranges. The half-cell voltage source (V_{hc}) along with the gel impedance ($R_g//C_g$) added with SC impedance ($V_{SC} + R_{SC}//C_{SC}$) characterises the electrode-skin impedance. The highest impedance is found in SC, whereas the impedance in electrodes, copper leads and electrolytic gel is comparatively lower. Figure 2.1b represents the described electric setup, where the dipole is modelled as a Norton equivalent circuit, with $V_{ts}/(R_b//C_b)$ representing the current in the dipole, R_e representing the electrode impedance and $R_{in}/2$ representing the amplifier impedance. Incremental nodes A and B split the electric circuit into stages and represent the positive and the negative polarities of the circuit respectively. [22]

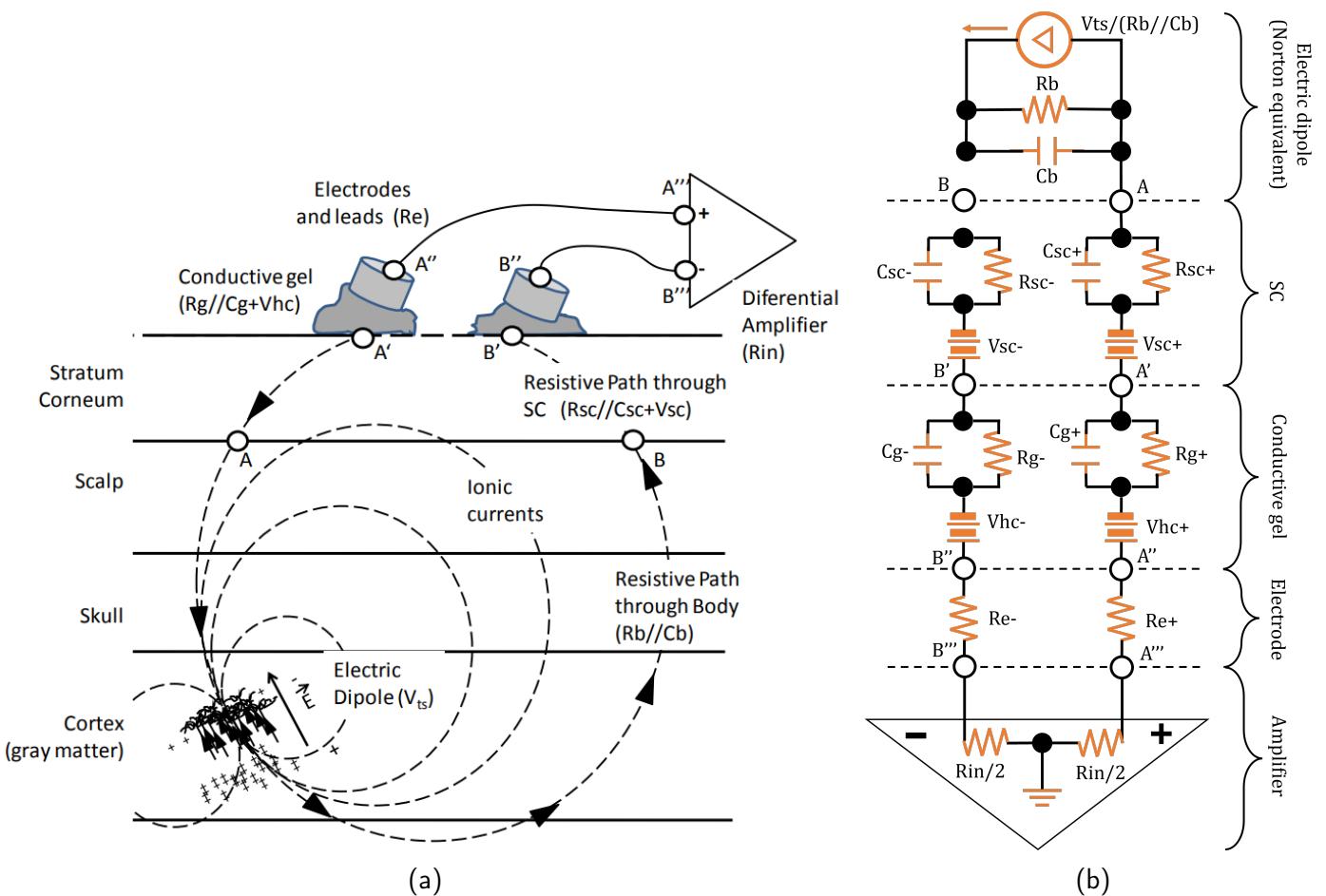


Figure 2.1: (a) Diagram of the electric dipole model, currents and the differential amplifier. [22] (b) Circuit diagram representation of the electric dipole.

2.1.3 The Emotiv EPOC+ Headset

A range of options for EEG systems exists; the EEG system used in this study is the Emotiv EPOC+ [9] mobile headset, shown in Figure 2.2. It provides a universal fitting and demonstrates competitive performance in literature [23, 24, 25], whilst remaining commercially available.

Two common research-grade EEG configurations exist: with 64 [26, 27] and 128 [28, 29] active electrodes and sampling rates of up to 1 kHz. The devices follow the standard 10-20 electrode mounting system, depicted in Figure 2.3a. However, Emotiv EPOC+ uses 14 active electrodes, with sampling rates of 128 Hz or 256 Hz. Their positioning within the standard 10-20 mounting system is located at AF3, AF4, F3, F4, F7, F8, FC5, FC6, T7, T8, P7, P8, O1, O2, as shown in Figure 2.3b. Two additional ground electrodes assist in filtering out ambient electrical noise and are located at P3 and P4 [30]. Each electrode is held in place with a semi-flexible plastic arm, assisting in maintaining similar relative positioning on the head, allowing for easy placement and removal during regular use. However, maintaining consistent electrode placement between participants is unfeasible due to this characteristic [24].



Figure 2.2: Emotiv EPOC+ placed on the researcher’s head with labelled AF3 and AF4 channels. (a) Front. (b) Back.

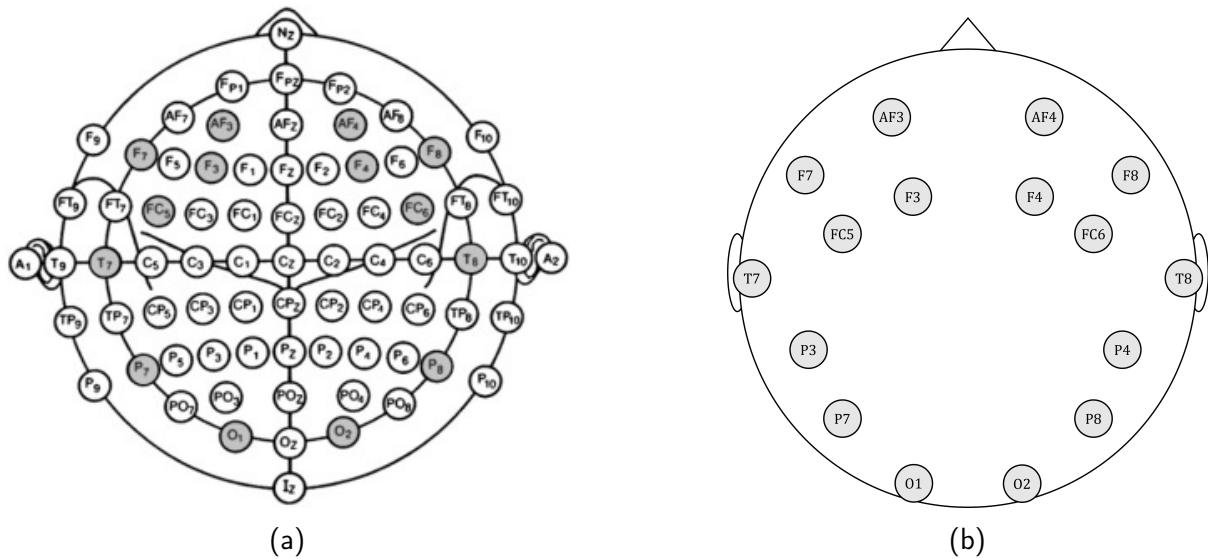


Figure 2.3: (a) Map of electrode positioning according to the international 10-20 system [30]. (b) Map of electrode positioning of the Emotiv EPOC+ headset.

2.1.4 Neuronal Information Transfer Mechanisms

Neurons are vital for information transfer. Action potentials, the rapid changes in voltage differences across plasma membranes, induce information transfer. Impulses propagate along and between nerve cells, allowing for the functioning of the nervous system. Electric signalling in neurons is brought about by membrane potentials across plasma membranes. These arise from unequal distributions of charge between intra- and extra-cellular environments, themselves arising from positive and negative ion concentration differences. Neurons can manipulate ion location, through the use of ion channels, to pass the information along.

When no impulses are conducted, the membrane is at “resting potential”, measured to be in the range of -40 to -90 mV. A local response will provide enough stimulus to pass the threshold point of approximately -55 mV, instigating an influx of sodium (Na^+) ions into the cell, resulting in depolarisation and an increase in membrane potential up to +35 mV.

The repolarisation phase follows, indicated by decreasing Na^+ concentrations, along with an increasing potassium ion (K^+) efflux, leading to membrane polarisation reduction, measured below the threshold level. This is the hyperpolarisation phase. The state gradually returns to the threshold level, which it will pass if stimulated again. The functionality of this process depends on ion movement across the plasma membrane through ion channels.

Channel proteins can select between Na^+ and K^+ , allowing for the increasing and decreasing concentrations of these ions. The flow of ions across the plasma membrane through channel proteins gives rise to transmembrane ion currents. The sum of various currents flowing at a given point determines the overall membrane potential of the neuron. Clusters of neurons undergoing this process result in the electric potential recorded by EEG. [31]

The recorded signals are classified based on their frequencies and respective brain activities, as shown in Figure 2.4. All bands carry useful information, and they are all analysed throughout this investigation. [32, 33]

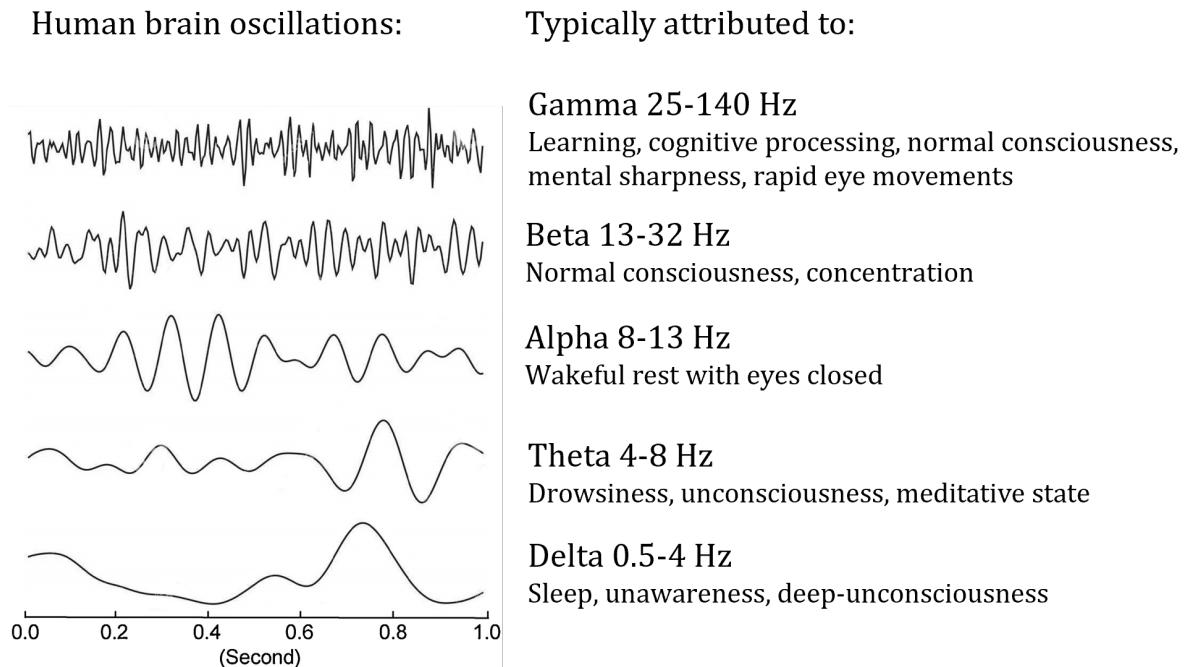


Figure 2.4: Recorded potentials of neural oscillations, their frequency ranges and attributions to various brain activities. Adapted [34].

2.2 Preprocessing

Raw EEG recordings do not provide direct access to information about brain activities due to various artefacts in the data. Signal artefacts are typically removed through extensive preprocessing. One of these artefacts is the EEG electrode drift, where the recorded potential gradually grows. This drift can be treated as a low-frequency signal and is suppressed with a digital band-pass filter with a low-frequency cut-off, also known as a high-pass filter. Notch filtering can be applied to mitigate the second artefact, sourcing from the residual mains interference (50 Hz in the UK) [35]. These processes are referred to as temporal filtering. Another artefact type present in neural activity data is electrooculography (EOG). EOG signals are caused by eyeball and eyelid motion, resulting in sudden changes in the potential field.

These artefacts cannot be simply filtered and are instead removed as follows: first, two EEG channels are allocated to collecting EOG signals. In the Emotiv EPOC+ setup in Figure 2.3b, these are the front-most channels, AF3 and AF4, labelled in Figure 2.2a. Afterwards, independent component analysis (ICA) is applied to the data, combined with a 'sanity-check' of both the topographic map and component waveform plots [36].

Additionally, some researchers [2, 37] resample EEG data to a lower sampling rate, typically from 1 kHz to 256 Hz, prior to any other preprocessing. This downsampling aims to reduce computational complexity without major classification performance losses.

At this stage, the preprocessed signal data is considered 'clean' of major artefacts. Any distorted data is omitted from further investigation. The clean data can be windowed and analysed, extracting useful statistical features before the final stage, or passed directly to the final stage, this being classification.

2.3 Feature Extraction and Classification

The most common inputs for ML classifiers are extracted statistical features, accounting for 41% of employed methodologies [38]. Furthermore, the raw EEG data is often first translated to a frequency domain, due to closer association with behavioural patterns. The subsequent method, accounting for 39%, directly feeds raw signal strength from the time domain into deep neural network (DNN) classifiers, as the latter promises to automatically learn complex features from large data amounts [38]. Downsides arise from the scarcity of signal data for individuals, as data from different people varies significantly and transfer learning between individuals proves challenging and less successful [39, 40]. The remaining 20% of methodologies rely on exploiting convolutional neural networks' (CNN) ability to learn from image data, using spectrograms - 2D and 3D grids generated from EEG data as the input into the classifier [38].

2.3.1 Feature Extraction Methodologies

Across the literature, feature extraction is performed on data windows; small segments incorporating fractions of the dataset. Sliding window methodologies employ either 125 ms, 250 ms or 500 ms intrachannel window sizes with 50% overlap between consecutive windows [23, 24, 25, 26]. However, widely known BCI researchers Nguyen *et al.* [37] and Saha and Fels [41], report successful results with simultaneous feature extraction from multiple channels using Channel Cross-Covariance (CCV) matrices capturing interchannel statistical EEG relationships.

Extracting intrachannel statistical features from EEG signal data leaves researchers with large numbers of such feature sets across EEG channels. Typical extracted linear features in the time domain include *mean*, *root-mean-square*, *standard deviation*, *sum*, *median*, *variance*, *maximum*, *minimum*, *minimum + maximum*, *minimum - maximum*, *kurtosis*, *skewness*, *common spatial pattern*, *dynamic energy*, the first and second derivatives of each. The non-linear features in the frequency domain typically are *fractal dimension*, *Hurst exponent*, *spectral power*, *spectral entropy*, *magnitude* and *phase*. [23, 24, 25, 26, 38, 42]

Mel Frequency Cepstral Coefficients (MFCC) have proved to be efficient information extraction methods from audio signals in speech recognition tasks by closely approximating human auditory system responses [43]. Cooney *et al.*, well-established researchers in the literature, extracted thirteen MFCCs from the Kara One dataset [26], observing a significant increase in

classification performance over linear and non-linear features [42].

Reducing statistical features to feasible amounts to use as ML classifier inputs can be done with multiple methods. An example is Pearson Correlation; features are selected by ranking the correlation between baseline and test conditions [26, 44]. An alternative method proposed by Shen *et al.* [45] utilises the random forest and recursive feature elimination scheme (RF-RFE) in combination, reporting an increase in feature reduction performance. Ali *et al.* [46] brought a recent advancement in feature selection, improving the established Short-time Fourier transform (STFT) methodology by utilising multiple sliding windows of varying sizes for transformation, namely anchored-STFT. The methodology was further improved by proposing a Gradient Norm Adversarial Augmentation (GNAA) method, aimed at enhancing the robustness and classification accuracy of the ML classifiers [46].

We aim to further investigate statistical feature extraction methods, determining whether a universal feature set outperforming others across various ML classifiers exists.

2.3.2 Classification with Machine Learning

The final stage of converting thought into speech is classification. Reviewed literature indicates classification is performed by machine learning algorithms trained by supervised, unsupervised, or mixed methodologies [26, 33, 46].

Due to limited available training data for deep learning classifiers, conventional machine learning techniques, such as linear discriminant analysis (LDA), decision tree (DT), support vector machine (SVM), k-nearest neighbours (KNN) and naïve Bayes (NB) classifiers show higher performance than deep learning techniques [2, 23, 24, 25, 38, 46]. However, compact CNN-based architectures show successful results in learning from small data amounts in the speech decoding domain [41, 47, 48] and the computer vision domain [49]. In conjunction with augmented data, a shallow CNN with skip connection, Skip-Net, recently outperformed the current state-of-the-art classifiers on motor-imagery EEG classification tasks [46]. Furthermore, Vernon *et al.* proposed a CNN-based architecture with depthwise and separable convolutions, potentially capable of generalising over a range of BCI paradigms [50]. However, no researched literature proposed an artificial neural network (ANN) approach able to significantly outperform conventional methods across the whole range of imagined speech decoding paradigms. Due to the low amount of training data for a single individual available, own data will be collected for this study with a novel public dataset released. Various machine learning approaches will be tested on this data, including previously successful conventional approaches, expanding towards CNNs.

The key procedures discussed in Sections 2.2 and 2.3 are summarised and chronologically arranged in Figure 2.5.

2.4 Experimental Paradigm Design

Over the last decade, the research field has been populated with numerous imagined speech experimental paradigms and publicly available datasets of brain activity data [23, 26, 51, 52].

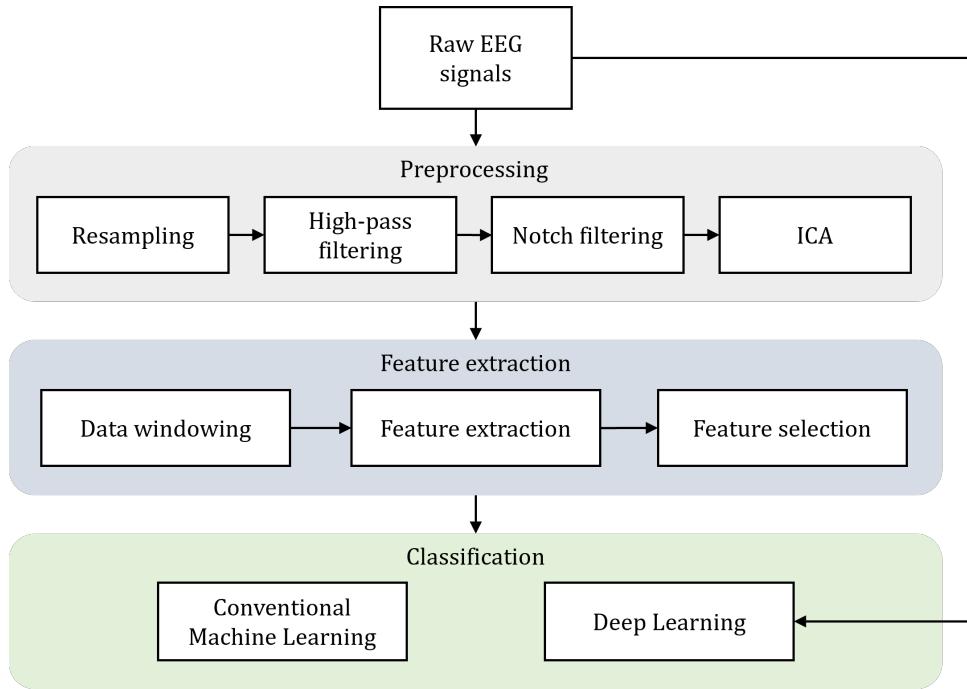


Figure 2.5: Diagram depicting a typical pipeline of procedures incorporated in the processing of EEG signal and the direct approach of learning from raw data.

2.4.1 Prompt Selection

The first step in designing an experimental paradigm for imagined speech is the choice of prompts. This choice is commonly concerned with the following two aspects:

1. Relevance in completing everyday tasks, such as controlling a computer or a wheelchair with prompts such as "up", "down", "left", "right", "forward" and "backwards" [52, 53, 54, 55], and responding to binary questions with prompts such as "yes" and "no" [56, 57].
2. Phonetic dissimilarity, which creates distinct neural activities, greatly assisting in prompt classification [58]. Prompt examples include /a/, /e/, /i/, /o/ and /u/ [52, 55]; /f/, /z/, /n/ and /k/ [23]; "yes" and "no"; "pat" and "pot" [26]; "in" and "cooperate" [37].

Prompt delivery and imagination have been done in various languages to suit regional needs, including English [56], Spanish [52, 55], Hindi [57] and Japanese [54].

The prompt selection in this work was based on the FEIS dataset [23], which, in turn, was based on the Kara One dataset [26]. Both works became the basis for various novel research papers [24, 25, 42]. This work aims to grow upon and expand beyond the limitations of existing datasets, allowing researchers to experiment with novel feature extraction and classification ideas.

The prompt choice covers a wide and balanced spread of binary phonological features across the International Phonetic Alphabet (IPA). As demonstrated in Table 2.1, the twelve chosen consonants cover plosives, fricatives, nasals, vocals, labials, alveolar, postalveolar and velar phonemes. The chosen vowels, as seen in Table 2.2, cover the four extremities of the peripheral vowel space: high front, high back, low front and low back.

Table 2.1: Consonant phoneme types studied.

	Labial	Alveolar	Postalveolar / Velar
Plosive (-voice)	/p/	/t/	/k/
Fricative (-voice)	/f/	/s/	/ʃ/
Fricative (+voice)	/v/	/z/	/ʒ/
Nasal (+voice)	/m/	/n/	/ŋ/

Table 2.2: Vowel phoneme types studied.

	Front	Back
Close	/i/	/u/
Open	/æ/	/ɔ/

2.4.2 Imagined Speech and Inner Speech

The second step of paradigm design is the choice of speech modalities. Speech production is a complex brain process performed by humans since it requires simultaneous interaction of various brain regions. The majority of language theories and models agree the production of speech consists of the following defined steps [59, 60, 61]:

1. Auditory processing.
2. Semantic and syntactic processing.
3. Motor planning.
4. Articulation.

This study explores imagined and inner speech modalities as ways to solve limitations experienced by individuals following this speech production process. These modalities are often inconsistently referred to in literature, although clear definitions exist [32, 62]. Using silent speech as a baseline, where articulation is produced as in normal speaking, without emitting sound, defined characteristics of imagined and inner speeches are presented for this work, as follows:

- Imagined speech: similar to silent speech, but produced without any physical articulatory movement. Instead, a person focuses their mind on the different articulatory gestures used when producing speech. EEG has been extensively used to research imagined speech modality [23, 26, 27], focusing on prompts with distinct articulations, such as plosive, fricative and nasal consonants, as well as open and close vowels.
- Inner speech: as opposed to the other modalities, is a fully internal process, where a person thinks in pure meaning and is often associated with "inner voice". No phonetic properties are retained during inner speech [62]. Fewer works have studied inner speech using EEG. Those using EEG did not provide evidence of the generation of more or less distinguishable brain processes than imagined speech [32, 63].

The summary of the discussion above is depicted in Figure 2.6, concisely defining each modality and highlighting the areas researched in this work.

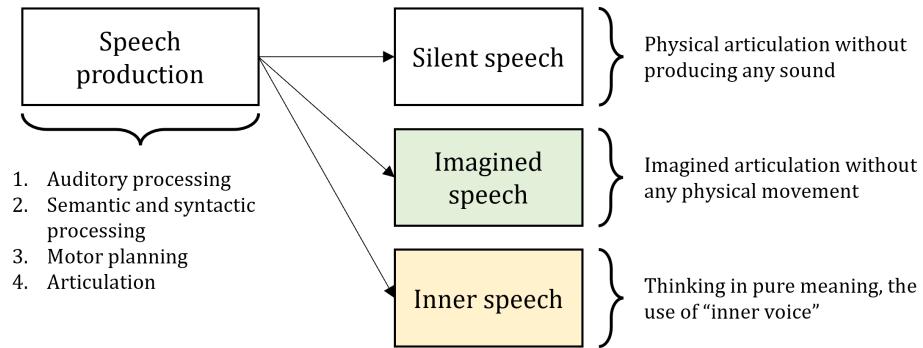


Figure 2.6: Diagram summarising speech modalities discussed and how they are produced. Highlighting exaggerates on the modalities studied in this work.

This work aims to populate BCI research with a novel EEG dataset of imagined and inner speech modalities, using the prompts defined in Section 2.4.1. Inner speech controls BCI more intuitively as individuals do not focus on articulatory gestures. A target of this study will be to provide quantitative evidence to support the use of either inner or imagined speech for BCI systems.

2.5 Hypotheses

Throughout the literature review, gaps in the current research were identified with the novel hypotheses derived. As noted in Section 2.3.1, Cooney *et al.* [42] report that extracting MFCCs from windowed EEG signals yields better classification results with conventional methods than extracting linear or non-linear features. Ali *et al.* [46] and Lawhern *et al.* [50] report high accuracy results when supplying minimally processed EEG signals to DNN classification pipelines with the latter model promising to adapt to a wide range of BCI paradigms.

These studies perform experiments on research-grade EEG systems. However, no investigation has been made for comparing the full spectre of data decomposition techniques with a low-fidelity commercial-grade EEG device (14 channels at 256 Hz). We anticipate that MFCC will outperform other decomposition methods. This is the first hypothesis (H1).

Nieto *et al.* [32] highlight that BCI researchers inconsistently refer to imagined and inner speech modalities. Both imagined speech [26] and inner speech [63] have been individually studied, reporting high classification accuracy. The only study involving both modalities was performed by Nieto *et al.* [32]. However, the applied device is research-grade and no evidence of better classification performance of either is provided. We predict that since phonological properties in the motor system are not retained during inner speech [1], as opposed to imagined speech, inner is ought to produce signals harder to capture with a commercial grade EEG headset, resulting in higher classification accuracy of imagined speech. This is the second hypothesis (H2).

2.6 Summary

This chapter draws information from the literature to introduce imagined speech research fundamentals using EEG. An introduction of the technology, its principles of work and the comparison to other brain imagining technologies is followed by a discussion of EEG data processing before it is used in ML classifiers. Statistical feature extraction and ML methodologies were introduced, which have found success in the literature. A discussion of considerations made in experimental paradigm design follows with novel hypotheses.

Chapter 3

Methodology

The chapter introduces the methodology, derived from supporting literature, to conduct the study in Section 3.1. Section 3.2 introduces the data acquisition software and techniques used to capture EEG data from the headset. The structure of the collected dataset is discussed in Section 3.3, with consecutive discussions on raw data preprocessing in Section 3.4. Section 3.5 details the statistical features extracted from the preprocessed signal. Finally, Section 3.6 discusses the adopted classification algorithms for this study.

3.1 Experimental Paradigm

This section describes the experimental paradigm developed for the performance of this study and the testing of the hypotheses. The EEG device used is Emotiv EPOC+ [9] and its principles of work are described in detail in Section 2.1. It has found success in performing similar research [23, 24].

3.1.1 Paradigm Description

Section 2.3 identifies the possible reason for the poor performance of deep learning methodologies to classify imagined speech is the scarcity of training data for an individual. We ensure a sufficient data amount is recorded per participant, per speech modality, by repeating every phoneme class described in Section 2.4.1 one hundred times. The imagined and inner speech both have 100 data instances for every class. Overall, individuals are tasked with performing: $16 \text{ classes} \times 100 \text{ repetitions} \times 2 \text{ speech modalities} = 3200 \text{ instances of imagination}$.

The experiment runs in two parts: the former concerning imagined speech modality, the latter with inner speech. Participants complete the experiment sitting in a comfortable chair alone, before a laptop monitor. The following steps are carried out:

1. Three-second stimuli state: a phone prompt appears and the respective audio is rhythmically played five times.
2. One-second preparation state: the participant prepares to imagine articulator movement or inner voicing.
3. Three-second imagination state: the participant repeats the imagination of the phone five times, whilst a visual prompt moves across the screen to set the rhythm.

4. Four-second rest state: giving the participant time to clear the mind before the next iteration.

The flow between states of the experiment is demonstrated in Figure 3.1.

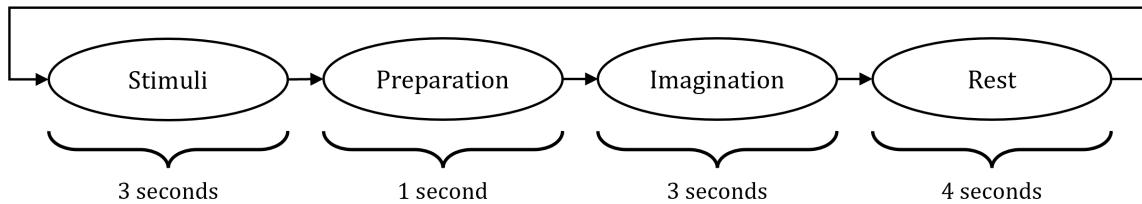


Figure 3.1: Diagram depicting the state flow during the experiment, where the circles represent the states and the arrows represent the transitions between them.

Many experimental paradigms in literature use auditory clicks for setting imagination rhythm, however, this study adopts visual prompts thereby introducing less noise in the data [2].

Each iteration amounts to eleven seconds. Rapid experiment iteration with randomisation of phone order is introduced to keep participants engaged. Furthermore, every 20 minutes participants are given breaks to refresh their minds. As such, 3×20 minutes recording sessions are conducted for each speech modality and a total recording time of approximately two hours. The BCI device stays on the participant's head at the same location for the whole duration.

Prior to commencing the experiment, participants are briefed on the experimental structure verbally and in writing. A copy of the Participant Information Sheet can be found in Appendix A, where participants are provided with simplified definitions of the two speech modalities, as follows:

- Imagined speech: a person imagining themselves saying something; mentally carrying out the process of how they would move their articulators (tongue, lips, vocal cords, *et cetera*) in the production of speech, without moving those organs.
- Inner speech: a person imagining hearing their voice (often referred to as the 'inner voice'), such as when 'silent reading' at the same speed they would speak those words.

The information regarding participants' English speaking proficiency and whether they are left or right-handed is recorded and anonymised. A copy of a blank Participant Consent Form can be found in Appendix B.

3.1.2 Paradigm Limitations

Despite thorough work performance ensuring reliable and consistent data is collected, certain limitations exist in the experimental paradigm and resulting dataset. The qualitative limitations must be considered when comparing quantitative results between participants. Although efforts were made to place the EEG headset identically on each participant, physiological differences in head shape resulted in inconsistent electrode placement with respect to the brain regions between subjects. It is also uncertain whether participants executed correct mental activities, despite the clear instructions provided. Furthermore, individual interpretations of imagined and inner speech may vary. All participants were naive users of BCI technologies; it is unlikely they were able to discriminate between various speech components.

3.2 Data Acquisition

Acquiring EEG data from the Emotiv EPOC+ headset required a combination of various software packages working together. This section will summarise the required packages and their limitations. The practical implementation and dataset can be found in this project's GitHub repository [64].

3.2.1 Software Packages

Each software package corresponds to a step in the process of acquiring EEG data from the Emotiv EPOC+ headset. The first package is the Emotive Launcher [65]. There are two modes the headset can connect to a computer. The former is through a USB cable, allowing the user to edit device configuration settings in the Launcher, such as the EEG sampling rate. The latter is through a Bluetooth connection with the help of a USB dongle. This option blocks the user from altering the settings but allows using the device for EEG data collection. The primary use of the Emotiv Launcher in this work was to assist the researcher with ensuring the EEG signal and the conductivity between the electrodes were of adequate quality. The Launcher provides a clear graphical user interface; assisting with the headset set-up.

The second package in the pipeline is CyKit [66]; a small tool used to set up a local Python server, which delivers EEG signals to other clients via a TCP stream.

Receiving the TCP stream is OpenVibe [67], the final software package in the pipeline of capturing the neural EEG data. One of the key purposes of OpenVibe is to align the recording computer with the EEG headset to a sub-millisecond accuracy. OpenVibe Acquisition Server communicates with the CyKit server, receiving the TCP data stream and forwarding it onto OpenVibe Designer. The Designer is a flexible graphical environment for creating custom BCI experiments. Figure 3.2 illustrates the OpenVibe Designer implementation of the experimental paradigm described in Section 3.1.

The paradigm implementation in the Designer consists of the following building modules:

- Timeline generator: a module working in conjunction with a Lua script, which defines the sequence of events, such as the image to be displayed or a sound to be played. This module sends signals to other modules via the connector arrows to stimulate them.
- Acquisition client: here, it is the Acquisition Server, which forwards the data from the headset to other modules.
- Player Controller: a small module, whose sole purpose is to terminate the recording when the corresponding signal is received.
- Sound Players: play the corresponding auditory cue when receiving a signal from the timeline generator. OpenVibe integrates an additional software package OpenAL [68] to play audio cues.
- Display cue image: similarly to the sound players, displays the corresponding image upon receiving a signal from the timeline generator.
- Stimulation based epoching and CSV File Writers: helper modules, which separate the incoming EEG signals based on the stimulation received from the timeline generator, recording the separated raw signals into the respective CSV files.

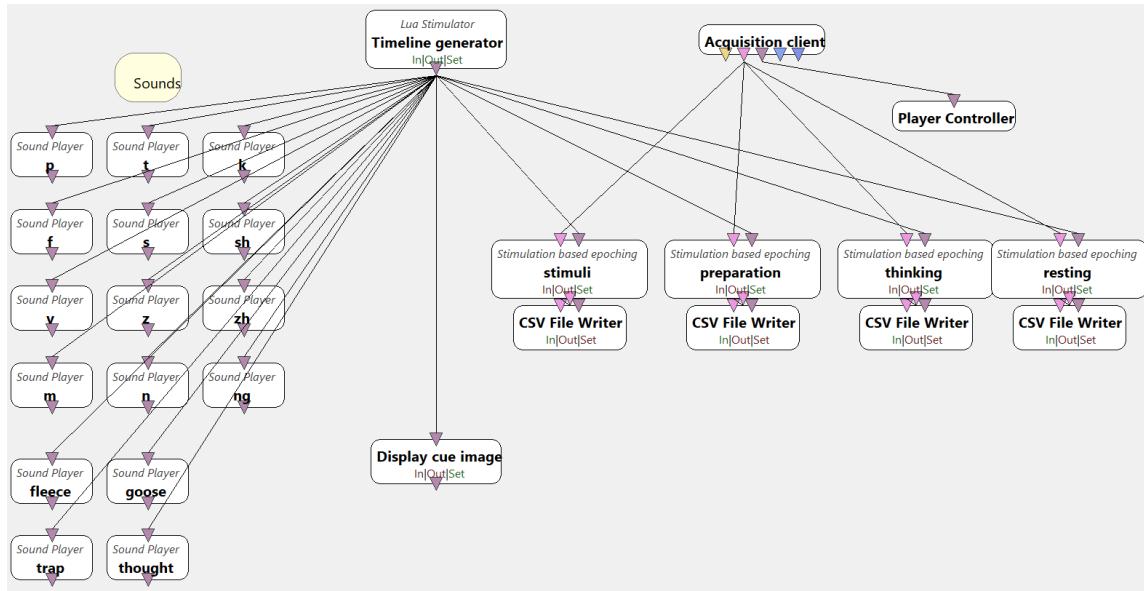


Figure 3.2: Illustration of this study's experimental paradigm implemented within OpenVibe [67] Designer software package, where the modules represent the various components of the experimental paradigm, acquisition client integration, input via the timeline generator, and output via visual cues, auditory cues and CSV files, whilst the lines represent the output/input links between the modules.

3.2.2 Device Drift Correction

A software limitation discovered during experimentation was device drift. The clocks on the headset and the computer running the experiment were asynchronous, the former lagging, thereby increasing drift time. A single 20-minute-long experiment resulted in a device drift of 5 minutes. Consequently, data collected drifts behind the experiment. For instance, when the device drift builds to 11 seconds, the participant may be imagining a phone trial N whilst the device labels the data collected as trial $N - 1$. A setting in the OpenVibe Acquisition Server mitigates this, forcing device drift correction. The study uses a 2ms threshold, hence drift correction occurs approximately every second. A 256 Hz sampling rate indicates 1 in 256 samples are drift corrected. These samples are averaged with the remaining data: indicating they do not represent the true signal received from the device. Although a distributed fraction of the data is corrupted, it is done to prevent the accumulating device drift.

3.3 The Dataset

The collected data consists of EEG recordings of 5 right-handed fluent English speakers. All data files can be found in the project's GitHub repository [64]. Raw data output from the device is contained within the `raw_eeg_recordings` folder, the directory organisation following the experimental procedure. The raw data is unlabelled, as such, a Python script was written to automatically label the recordings according to the sequence of stimuli. The labelled recordings are merged and combined for a full continuous recording per participant per speech modality. Refer to Appendix C for a depiction of the dataset's folder structure.

3.4 Dataset Preprocessing

Raw EEG signals are polluted with artefacts, hence, we perform preprocessing to remove these artefacts. The software built into the Emotiv EPOC+ performs automatic notch filtering at 50 Hz and 60 Hz, removing the powerline noise. MNE-Python package [69] is subsequently used to interpolate bad channels, apply high pass filter, ICA and baseline correction.

3.4.1 Raw Signal Analysis

With MNE, raw EEG signals in full_labelled.csv files are converted to RawArray; an MNE datatype compatible with package functionality. Before preprocessing, signals are visualised in time and frequency domains, using professional expertise to locate major artefacts in the signal. This technique was used to detect anomalous signal behaviour and discard data from the first participant (P00) from further preprocessing.

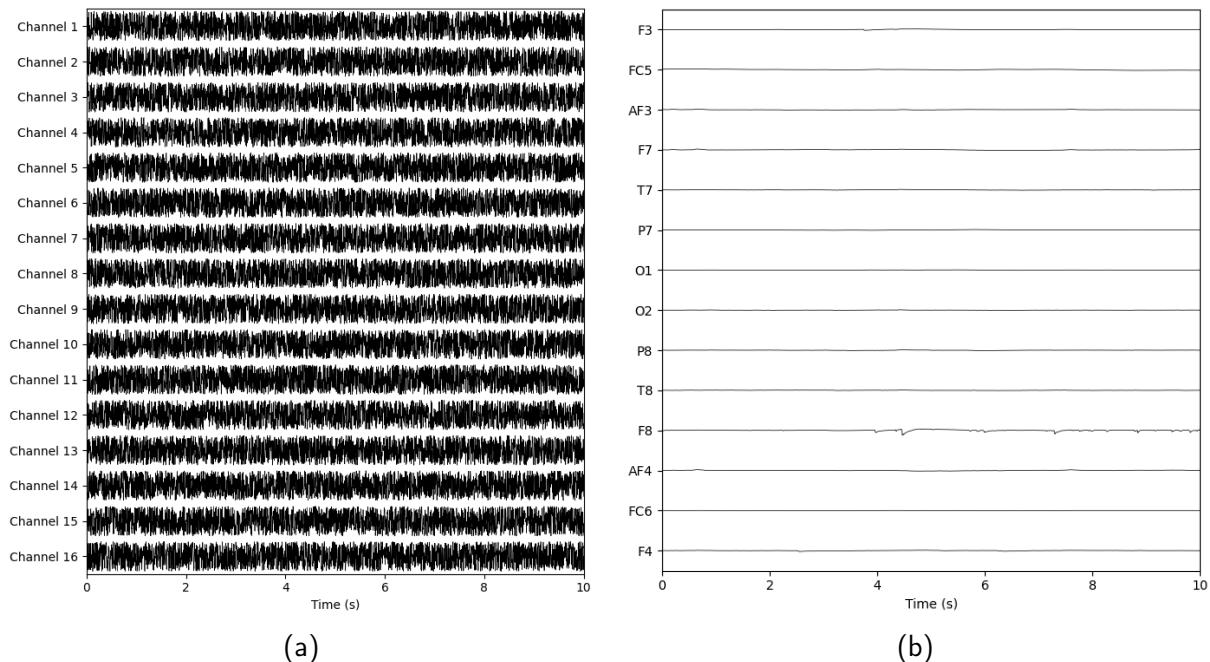


Figure 3.3: Time domain graphs, both demonstrate a sample of the first 10 seconds of data at identical scale, where the x-axis is time in seconds and the y-axis is the scaled signal amplitude in volts. (a) P00 raw EEG signal collected with the Emotiv EPOC headset. (b) P01 raw EEG signal collected with the Emotiv EPOC+ headset.

P00 data was collected using a predecessor of the Emotiv EPOC+ device, Emotive EPOC. The devices have minimal differences: EPOC+ is capable of 128 Hz and 256 Hz sampling rates, whereas EPOC is limited to 128 Hz. EPOC has been used for BCI studies [70], although many literature sources studying commercially available BCI devices opt for the upgraded EPOC+ [23, 30]. Despite the differences in the devices, the EPOC headset was discovered to be malfunctioning, as can be observed in Figure 3.3a: the EEG signal rapidly oscillates at a significantly higher amplitude than the typical EEG signal shown in Figure 3.3b, where both graphs are plotted at the same scale. An incorrect number of 16 channels was used to collect P00 data. The additional channels, 15 and 16, cannot be accessed from the headset, thus CyKit creates two fictitious channels and inserts interpolated data from the real channels.

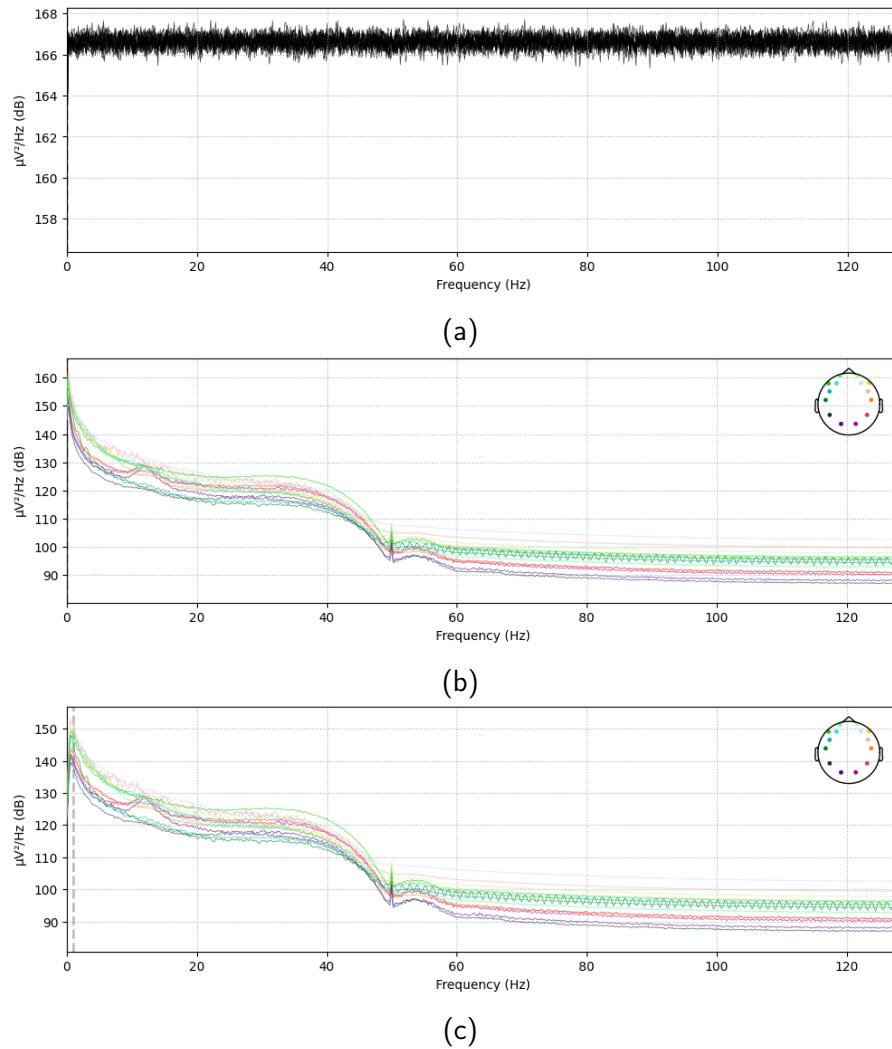


Figure 3.4: Power spectral density graphs, where x-axis is the frequency in Hz and y-axis is the power density in $\mu V^2/Hz$ (dB) (a) P00 raw EEG signal collected with the Emotiv EPOC headset. (b) P04 raw EEG signal collected with the Emotiv EPOC+ headset. (c) P04 1 Hz high-pass filtered signal, where the vertical dotted line represents the filter cut-off point.

Power spectral density (PSD) of each signal was also visually analysed. As observed in Figure 3.4a, the raw EEG signal from the EPOC headset produces an anomalously high power density across the whole frequency range. In contrast, Figure 3.4b demonstrates a typical PSD plot of raw EEG signal.

The EPOC device was diagnosed for possible issues. The correct number of channels was set in the Acquisition Server but this did not resolve the issue. The headset recorded identical signals when worn on the head and rested on a flat table surface. As such, a diagnosis for strong ambient noise was performed by constructing a Faraday cage, ensuring efficiency by placing the USB dongle outside the cage and the headset inside the cage, whilst observing the signal quality between the receiving computer and the headset. The device produced a similar noisy output, indicating that extreme noise artefacts are inherent to the headset. As a result of this diagnosis, a decision was made to carry out the remaining data collection experiments with the Emotiv EPOC+ headset, which reproduces typical EEG signals.

Lastly, visual inspection is used to locate bad channels, such as channel T8 in the P03 dataset,

as indicated in Figure 3.5a. The source noise in a single channel may be due to poor electrode placement. An MNE in-built feature to mark and interpolate bad channels is used to handle these artefacts, resulting in the signal seen in Figure 3.5b.

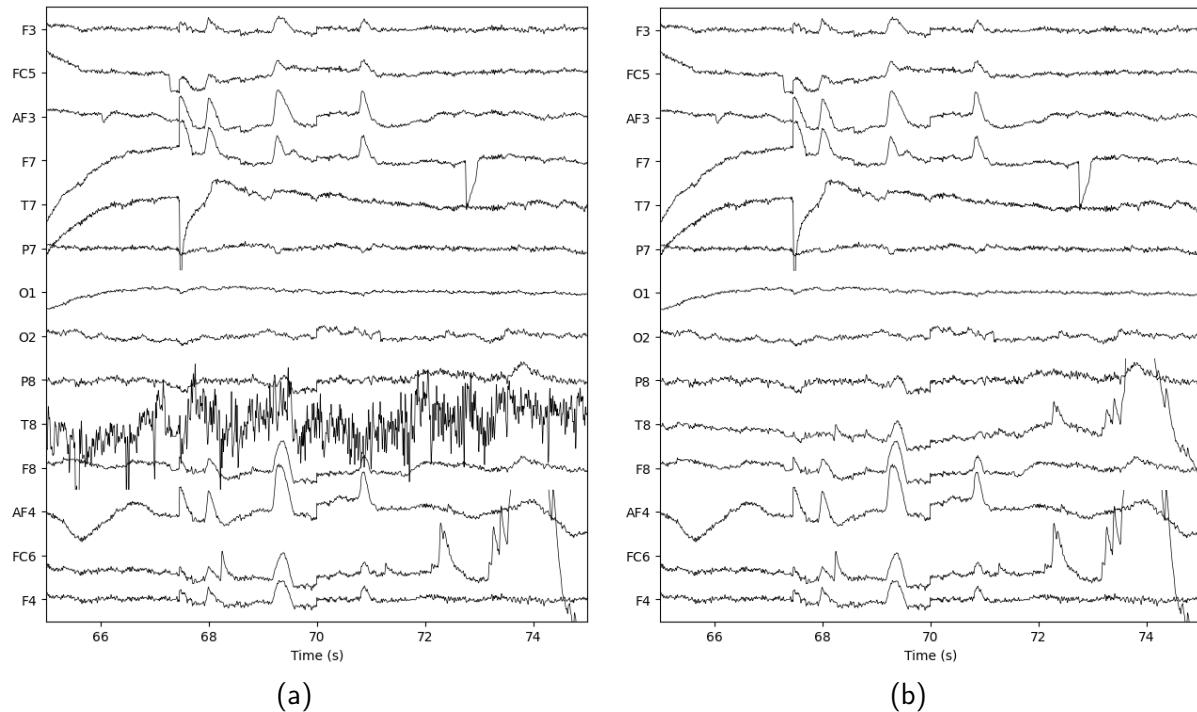


Figure 3.5: Time domain graphs of P03 imagined speech raw EEG signal between 65 and 75 seconds at identical scale, where the x-axis is time in seconds and the y-axis is the scaled signal amplitude in volts. (a) Bad channel T8 before interpolation. (b) Bad channel T8 after interpolation.

3.4.2 Digital Filtering

The next stage of preprocessing is the digital filtering of signal frequencies containing artefacts. As discussed in Section 2.2, low-frequency electrode drift is high-pass filtered at 1 Hz, shown by the vertical grey dotted line in Figure 3.4c.

The Nyquist frequency of the signal in this study is 128 Hz, half of the headset's sampling rate, allowing us to reconstruct the EEG signal with artefacts removed. The full Nyquist frequency range can be seen in the PSD plots in Figure 3.4. As shown in Figure 2.4, this frequency still captures the frequency ranges relevant to this study.

3.4.3 Individual Component Analysis

ICA performs blind source separation (BSS). During EEG execution, each electrode placed upon the scalp collects multiple signals from various brain regions as well as EOG artefacts. ICA identifies EOG signals using designated EEG channels as individual components, allowing EOG data to be filtered out from the remaining EEG channels. However, ICA assumes individual EEG signals are unrelated as it does not incorporate knowledge of neural signals. [71]

ICA is performed on continuous, filtered and interpolated, where applicable, data. Visual inspection of the time domain EEG plots can be used to locate EOG artefacts in the signal,

one such artefact is demonstrated by the reference green line in Figure 3.6a. Two signal peaks are visible at channels AF3 and AF4, which were selected to be the EOG channels, with these artefacts visibly affecting the adjacent channels. Figure 3.6b demonstrates the result of applying ICA to remove the EOG components.

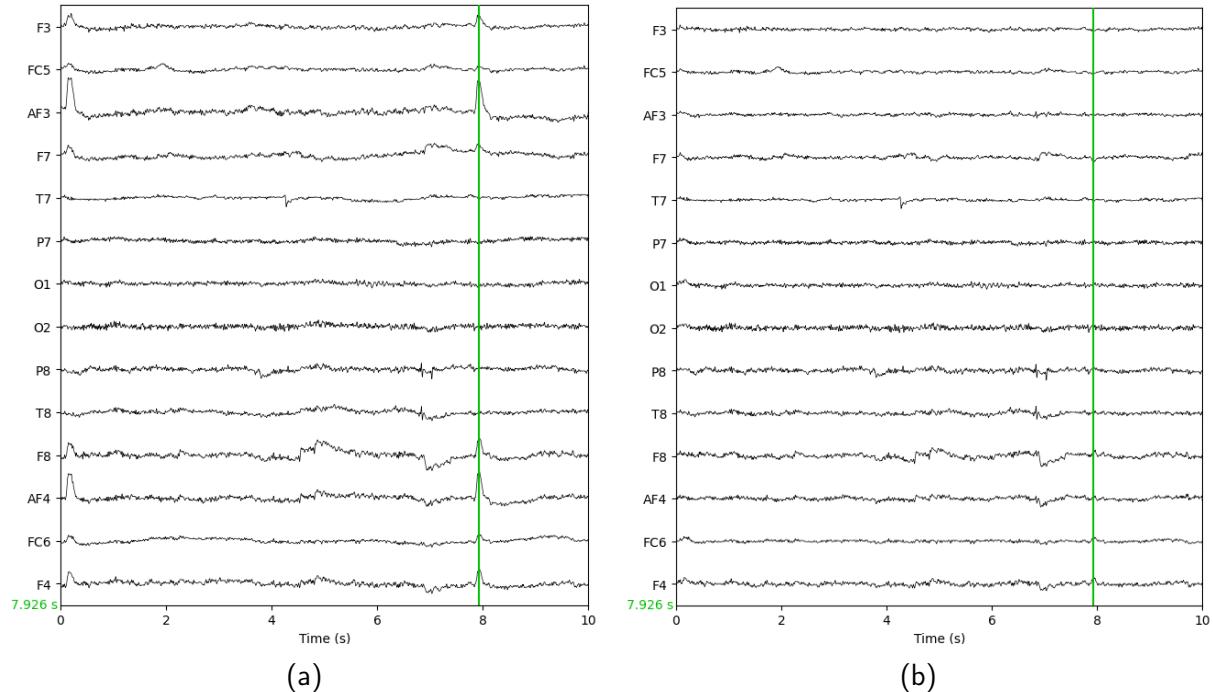


Figure 3.6: Time domain P04 inner speech raw EEG signal, where the x-axis is time in seconds and the y-axis is the scaled signal amplitude in volts. Vertical green line highlights an EOG artefact. (a) Pre-ICA raw signal. (b) Post-ICA raw signal.

Upon executing ICA decomposition, MNE contains an automatic algorithm for detecting EOG ICA components, used in this study to identify the potential EOG components for exclusion. Subsequently, a plot of the individual components' topographical maps is generated. We apply visual expertise to confirm the exclusion of a component. As the Emotiv EPOC+ headset has a low density of electrodes and does not facilitate dedicated EOG channels, the automatic algorithm often proposes too many or no components to exclude. Figure 3.7 presents a topographic map plot of the P04 dataset, where the number of the component is inversely proportional to the amount of variance in the component's signal. The MNE algorithm proposes components ICA000 and ICA002 for removal. We only remove a single component from each analysis, to ensure minimum amounts of useful information are lost from the signal during preprocessing. ICA002 closely represents a typical EOG component. ICA006 is also a possible candidate as an EOG component, however, component ICA002 has a higher variance, representing EOG artefacts more closely and therefore is a more suitable candidate for removal. Despite ICA000 having a higher variance than ICA002, its topographic map does not represent EOG behaviour, making it an invalid choice for exclusion. Therefore, component ICA002 is excluded from the P04 signal. This process is manually repeated for each dataset, ensuring the majority of the signal is retained.

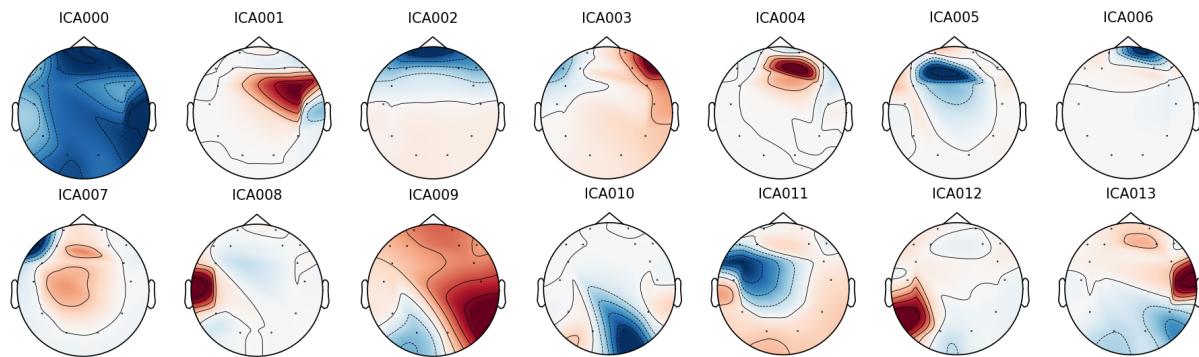


Figure 3.7: P04 imagined speech topographic map plot of the individual components.

3.4.4 Data Epoching

The final stage of the preprocessing pipeline is epoching the continuous data. Since we only focus on classifying EEG data during the thinking stage, the other stages are omitted from the continuous recordings and an MNE Epochs object is created for each dataset, where MNE automatically applies baseline correction to the signal. Lastly, we save the newly discontinuous thinking data to the `data_preprocessed` directory, found in the project's repository [64].

3.5 Feature Extraction

Feature extraction process following the general intrachannel methodology discussed in Section 2.3.1 is employed, consisting of two steps:

1. Window EEG signal.
2. Extract statistical features from each window.

Each 3-second EEG epoch is split into five 600 ms non-overlapping windows, aligning with the mental activity rhythm set during the experiment.

The three main types of features, often found in literature and discussed in 2.3.1, are extracted from each window:

1. Eleven linear features (LF) in the time domain: *mean, absolute mean, standard deviation, sum, variance, maximum, absolute maximum, minimum, absolute minimum, minimum + maximum and maximum - minimum*. 11 features \times 14 channels = 154 features per window.
2. Seven non-linear features (NLF) in the frequency domain: *Higuchi Fractal Dimension and Katz Fractal Dimension, spectral entropy, Hurst Exponent, spectral power, phase and magnitude*. 7 features \times 14 channels = 98 features per window.
3. Thirteen MFCCs. 13 features \times 14 channels = 182 features per window.

AntroPy helper package [72] was used to calculate the *Higuchi Fractal Dimension*, the *Katz Fractal Dimension* and the *spectral entropy*; Hurst package [73] to calculate the *Hurst Exponent*; librosa [74] to calculate the MFCCs.

As such, five types of data decomposition are compared: minimally processed raw EEG signal, preprocessed signal, LF, NLF and MFCC.

3.6 Classification

Following the discussions in Section 2.3.2, we employ five conventional machine learning models with the help of the Scikit-learn Python package [75], namely: KNN, Gaussian NB, LDA, SVM and DT with Adaptive Boosting (AdaBoost). Furthermore, we employ a single CNN-based deep learning classifier — EEGNet [50].

We perform every experiment using five-fold cross-validation. For conventional classifiers, data is split into five blocks; four are used for training and one for testing, resulting in an 80/20 data split. For the CNN model, the fifth block is further split equally into validation and testing sets, resulting in an 80/10/10 data split.

Since the Emotiv EPOC+ headset has 14 EEG channels and no dedicated EOG channels, we run all algorithms on raw, preprocessed and feature extracted data. We can thus make a quantified analysis on whether it is worth dedicating EEG channels to remove EOG artefacts and extracting statistical features to get an alternative raw data representation. Furthermore, we perform the above classification on a data subset from the FEIS dataset [23], to gain a direct comparison between the experimental paradigms. We also perform a baseline binary classification using a phonetically distinct phonemes pair /æ/ and /f/.

Lastly, all experiments are performed on the same desktop computer, equipped with Intel Core i5-4690K CPU, 16 GB DDR3 RAM and NVIDIA GeForce RTX 3050 GPU. Conventional classifiers utilise the CPU to perform computations, whereas EEGNet utilises the GPU.

3.6.1 K-Nearest Neighbours

KNN is a non-parametric, supervised learning algorithm, utilising a metric of distance between the data points in the n -dimensional feature space, to make a prediction. Based on this distance, the algorithm finds the k nearest points to a given query point and classifies it based on the majority of the surrounding classes. KNN assumes points of the same class are clustered near each other. The value of k is proportional to the classifier bias and inversely proportional to its variance. [76]

A binary classification example plot in Figure 3.8a shows a query data point and its three nearest neighbours, where their majority belongs to Class 2. Figure 3.8b shows the query point accordingly classified.

We employ the Euclidean distance metric in our implementation of KNN. When faced with equidistant points of different classes, Scikit-learn breaks ties by choosing the first point in the order of the original dataset. In our experiments, we choose exponentially growing values of k , starting with $k = 1$.

3.6.2 Gaussian Naïve Bayes

The NB algorithm builds upon conditional probability, where X is a feature and Y is the corresponding label. The training dataset provides the algorithm with the probabilities of features, given the appropriate labels ($P(X|Y)$). The Bayes rule is applied to find the probability of each label given the testing data ($P(Y|X)$). In a multi-class classification problem, such as ours, this probability is calculated for each class. When the label is dependent on multiple features, X_1, X_2, \dots, X_n , NB assumes each feature is independent of all other features, simplifying the probability to $P(Y|X_1, X_2, \dots, X_n)$. EEG signals from each of the 14 channels are continuous

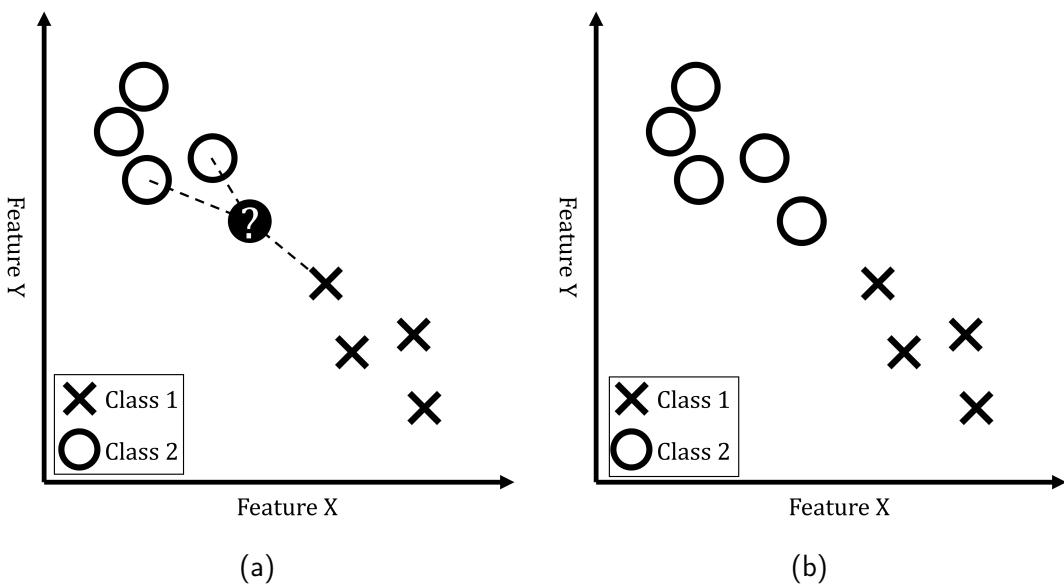


Figure 3.8: Binary classification with two features using KNN algorithm, where $k = 3$. (a) New query point, marked with symbol ?, where the dotted lines indicate the three nearest data points. (b) Query point assigned to Class 2 as the majority.

variables, X_1, X_2, \dots, X_{14} , rather than distinct features. To approach this issue, each value of X is assumed to follow a Gaussian distribution. [77]

3.6.3 Linear Discriminant Analysis

LDA, similarly to Gaussian NB, assumes a Gaussian distribution of each variable for each class. The algorithm calculates each class's vector of variable means and covariance matrix. New observations are classified by constructing a discriminant function $h(X)$ from the means vectors and covariance matrices. [78]

3.6.4 Support Vector Machine

SVM's main objective is to approximate a classification hyperplane through an n -dimensional space, where n corresponds to the number of features. A further objective is to ensure the hyperplane maximises the margin distance between all classes. The support vectors are referred to as the points in space closest to the hyperplane, influencing its position. The basic SVM algorithm only supports binary classification. Scikit-learn handles this limitation via a one-vs-one scheme; a binary SVM classifier is developed for each class combination pair. For m classes, $\frac{1}{2}m(m - 1)$ binary SVM classifiers are constructed, resulting in the computational complexity of $O(m^2)$. [79]

A binary classification with two features example plot in Figure 3.9a shows potential one-dimensional hyperplanes constructed. Figure 3.9b shows the optimal hyperplane equidistant (w) from both classes' support vectors.

In our experiments, we implement the three SVM kernel functions available within Scikit-learn, including the Radial Basis Function (RBF), the polynomial function and the sigmoid function.

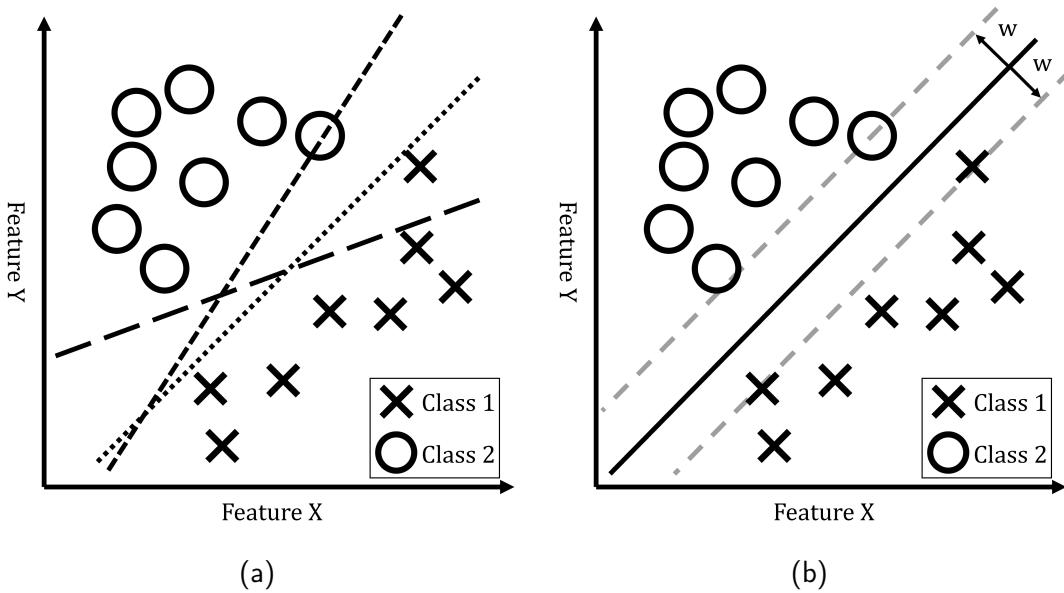


Figure 3.9: Binary classification with two features using SVM algorithm, where w is the margin distance. (a) Potential classification hyperplanes constructed. (b) Optimal classification hyperplane found.

3.6.5 Decision Trees with Adaptive Boosting

DT algorithms are non-parametric, supervised methods. A DT begins with a data root node at the top, where each subsequent node splits the data by a decision rule. Leaf nodes, without any subsequent decision rules, are terminal nodes classifying the data. AdaBoost is a type of ensemble method proposed by Freund and Schapire [80], which combines a sequence of weaker estimators. As data is iteratively processed through the sequence, correctly classified training data points are decreased in weight, whereas incorrectly classified points receive a weight increase. This method forces the subsequent estimators to concentrate on the data misclassified by their predecessors. The final prediction is a weighted sum of predictions from all estimators. In this study, we use a sequence of weak DTs with Scikit-learn's Multi-class AdaBoost model [81]. Throughout experimentation, we vary the estimators' number and learning rate of the classifier. In the remainder of this work, DTs with AdaBoost classifier is referred to as AB.

3.6.6 EEGNet

Our motivation behind selecting EEGNet [50] as the deep learning CNN classifier is built upon the model's past success in similar EEG tasks. Chang *et al.* investigated its performance for motor imagery tasks [82], whilst Vahid *et al.* used event-related EEG potentials with EEGNet [83]. Furthermore, EEGNet outperformed other CNNs, recurrent neural networks (RNNs) and conventional algorithms, on the benchmark dataset in the P300-based BCI competition [84].

The model gains its success by utilising a combination of temporal, depthwise and separable convolutions with a low number of trainable parameters, making it suitable for learning from datasets with a limited amount of data. The architecture of EEGNet is made configurable with parameters controlling the number of temporal (F_1), spatial (D) and pointwise (F_2) filters to learn. We employ the default parameters proposed by the authors, where $F_1 = 8$, $D = 2$ and $F_2 = 16$. The recommended kernel size of the temporal convolution is half of the EEG

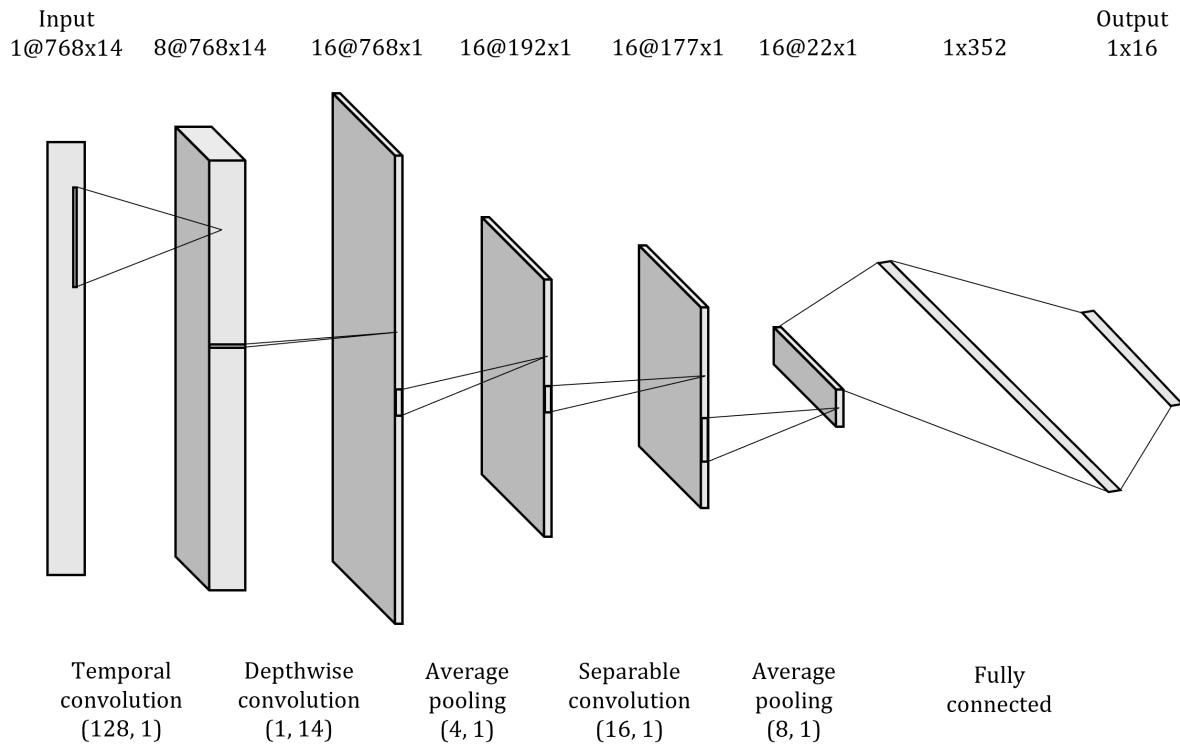


Figure 3.10: EEGNet architecture, where the layer dimensions correspond to a 3-second, at 256 Hz, across 14 channels, EEG signal input.

sampling rate.

We use epochs of EEG signal as our input to EEGNet. With raw and preprocessed data, the two-dimensional input corresponds to 3 seconds at a 256 Hz sampling rate across 14 EEG channels (768×14). The network architecture under these conditions is presented in Figure 3.10 (full EEGNet architecture can be found in Appendix D). The extracted features, discussed in Section 3.5, are also treated as two-dimensional inputs (features \times channels).

The authors of EEGNet openly published the relevant source code on GitHub [85]; originally written in the TensorFlow framework [86]. For this work, the original code was rewritten using PyTorch [87].

3.6.7 Hyperparameter Sweeps

To achieve optimal learning with EEGNet, sweeps of the available hyperparameters are performed. This includes the common: batch size, number of epochs, learning rate, weight decay and the dropout rate in the corresponding layers of EEGNet. Furthermore, we evaluate the common loss functions and optimisers found in the literature for EEG signal classification. Loss functions include negative log-likelihood (NLL) and cross-entropy (CE) [2, 40, 48, 50]. Optimisers include Adam and stochastic gradient descent (SGD) [40, 41, 48, 50, 83].

The sweeps are performed using the Weights & Biases (WandB) [88] Python library. We utilise the Bayes search algorithm to maximise the validation accuracy and to efficiently search for the optimal configuration. Since one of the aims of EEGNet is to generalise to different EEG datasets, we perform a single sweep with 200 iterations across P01-P04 datasets. We then perform individual sweeps of 100 iterations each, on the P00, binary and FEIS datasets.

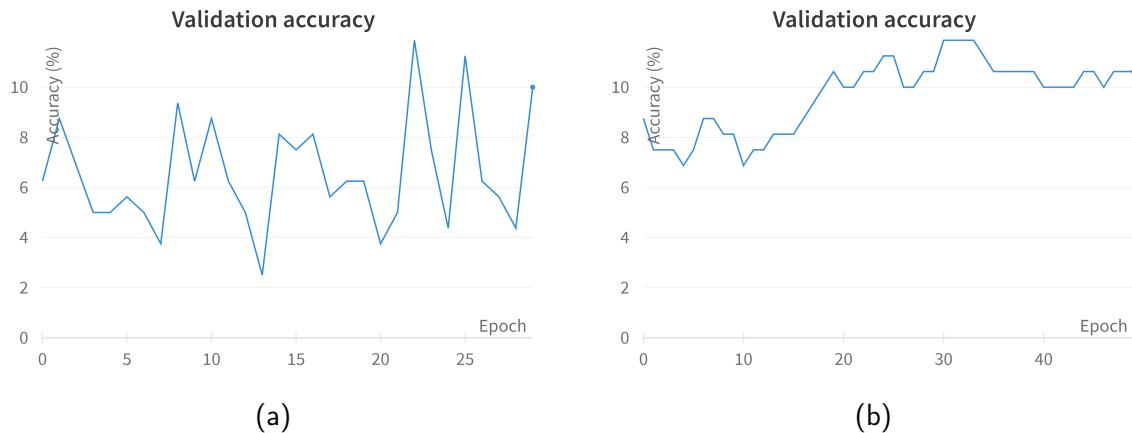


Figure 3.11: Examples of validation accuracy graph generated during a hyperparameter sweep.
(a) Poor accuracy graph. (b) Desired accuracy graph.

Upon completion of each sweep, WandB automatically generates performance reports of the various hyperparameter configurations. We select configurations showing the highest performance and further investigate them for stability in validation accuracy. Figure 3.11 demonstrates the performance of two different sweeps. Figure 3.11a shows a run with high final accuracy. However, the overall trend is unstable, indicating the model did not learn. Figure 3.11b showcases a desirable pattern of a run with a stable increase in validation accuracy.

After performing the sweeps, each model is trained using the corresponding hyperparameters. The final hyperparameter configuration can be found in Appendix E.

3.7 Summary

This chapter began with a detailed introduction of the experimental paradigm used for collecting imagined and inner speech modalities, consecutively discussing the potential limitations of the setup. We described how multiple software packages link together to bring EEG signal from the Emotiv EPOC+ headset into a readable CSV file, whilst pointing out the discovered drawbacks. The dataset structure is then presented and described. Subsequently, we detailed the data preprocessing methodology, including the analysis of the raw signal, digital filtering, ICA and data epoching. Lastly, we demonstrated employed feature extraction methods, including LFs, NLFs and MFCCs; followed by an overview of utilised classification algorithms. These include KNN, Gaussian NB, LDA, AB and EEGNet. We concluded by describing the hyperparameter sweep process for EEGNet.

Chapter 4

Results

This chapter is dedicated to presenting the classification accuracy results of conventional and deep learning methods on the data collected during this study and the FEIS dataset. Table 4.1 shows the conventional classification test results on P00 data. Table 4.2 shows classification results on the P01 dataset. Since P01 results represent the general trends in data, the remainder of the conventional classifiers testing results for P02, P03 and P04 can be found in Appendix F.1. Table 4.3 shows the results for binary and FEIS dataset classification. The training accuracy results for P00-P04, binary and FEIS dataset classification can be found in Appendix F.2. Lastly, Table 4.4 shows the accuracy results using EEGNet. The results for AB, Gaussian NB, LDA and EEGNet are summarised in the bar charts in Figures 4.1, 4.2, 4.3 and 4.4 respectively.

All multiclass classification results are compared to the baseline of $1/16 \times 100 = 6.25\%$ chance accuracy, and 50% chance accuracy on binary classification.

In all tables, we emphasise the top and bottom scoring mean accuracy for each classifier configuration. Where an identical mean accuracy is present, the mean accuracy with a lower standard deviation is chosen as the better scoring.

Raw and preprocessed data are excluded from multiclass classification when using SVM due to computational complexity (refer to Section 3.6.4). With the available hardware for our experiments, where the CPU is the limiting factor, a single run of SVM on raw or preprocessed data took up to six hours, which yielded to be unpractical.

Table 4.1: P00 testing accuracy classification results on conventional algorithms, where e stands for the number of estimators in AB, lr stands for the learning rate of AB and k stands for the number of nearest neighbours in KNN.

Method	Mean imagined speech percentage accuracy (std. dev)	Mean inner speech percentage accuracy (std. dev)
	Raw	Raw
AB ($e=50, lr=0.1$)	6.21 (0.09)	6.15 (0.02)
AB ($e=50, lr=0.5$)	6.26 (0.14)	6.18 (0.15)
AB ($e=50, lr=1.0$)	6.19 (0.11)	6.25 (0.05)
AB ($e=50, lr=1.5$)	6.38 (0.04)	6.14 (0.05)
AB ($e=50, lr=2.0$)	6.31 (0.06)	6.32 (0.12)
AB ($e=100, lr=1.0$)	6.44 (0.06)	6.30 (0.15)
AB ($e=200, lr=1.0$)	6.31 (0.05)	6.25 (0.14)
AB ($e=400, lr=1.0$)	6.33 (0.10)	6.22 (0.13)
Gaussian NB	6.25 (0.14)	6.20 (0.14)
KNN ($k=1$)	6.33 (0.22)	6.17 (0.12)
KNN ($k=5$)	6.31 (0.06)	6.20 (0.03)
KNN ($k=13$)	6.28 (0.05)	6.26 (0.01)
KNN ($k=25$)	6.34 (0.05)	6.23 (0.04)
KNN ($k=53$)	6.33 (0.13)	6.20 (0.11)
KNN ($k=89$)	6.22 (0.09)	6.10 (0.09)
KNN ($k=139$)	6.36 (0.09)	6.25 (0.03)
LDA	6.26 (0.11)	6.12 (0.02)

Table 4.2: P01 testing accuracy results on conventional algorithms, where cyan and red colours highlight the best and the worst mean classifier accuracy respectively, e stands for the number of estimators in AB, lr stands for the learning rate of AB and k stands for the number of nearest neighbours in KNN.

Method	Mean imagined speech percentage accuracy (std. dev)					Mean inner speech percentage accuracy (std. dev)				
	Raw	Prep	LF	NLF	MFCC	Raw	Prep	LF	NLF	MFCC
AB ($e=50$, $lr=0.1$)	14.2 (0.19)	10.5 (0.05)	7.77 (0.47)	6.58 (1.34)	5.96 (0.49)	13.5 (0.10)	10.1 (0.06)	10.0 (0.47)	8.21 (1.31)	5.71 (0.79)
AB ($e=50$, $lr=0.5$)	16.5 (0.29)	12.3 (0.12)	9.59 (0.82)	7.08 (0.63)	5.89 (0.70)	15.6 (0.34)	11.2 (0.09)	8.15 (0.31)	9.03 (2.14)	6.83 (0.47)
AB ($e=50$, $lr=1.0$)	15.8 (0.19)	12.2 (0.07)	8.90 (1.12)	6.90 (0.79)	6.96 (1.56)	15.5 (0.41)	11.5 (0.11)	9.28 (1.32)	7.34 (1.08)	6.33 (0.33)
AB ($e=50$, $lr=1.5$)	13.6 (0.19)	11.6 (0.12)	6.65 (0.62)	5.96 (1.13)	6.14 (0.89)	13.3 (0.47)	11.0 (0.06)	8.09 (1.87)	7.02 (0.32)	4.45 (1.47)
AB ($e=50$, $lr=2.0$)	7.67 (1.39)	6.65 (0.15)	7.34 (0.86)	7.27 (0.84)	6.21 (0.46)	8.14 (0.42)	7.46 (0.16)	8.21 (0.70)	6.21 (0.26)	6.83 (0.47)
AB ($e=100$, $lr=1.0$)	17.1 (0.17)	13.1 (0.07)	8.15 (0.18)	7.02 (0.87)	6.40 (1.08)	16.6 (0.29)	12.2 (0.00)	8.02 (0.44)	6.65 (0.99)	6.21 (1.20)
AB ($e=200$, $lr=1.0$)	17.5 (0.16)	13.9 (0.33)	7.40 (0.63)	7.52 (1.25)	6.39 (0.77)	16.7 (0.15)	12.4 (0.09)	9.28 (0.71)	7.59 (1.50)	7.27 (0.54)
AB ($e=400$, $lr=1.0$)	17.0 (0.03)	13.7 (0.15)	9.15 (1.56)	7.02 (0.23)	6.96 (1.37)	15.5 (0.15)	12.2 (0.09)	8.53 (0.39)	6.08 (0.88)	6.33 (0.50)
Gaussian NB	9.62 (0.08)	9.16 (0.09)	6.96 (1.49)	7.02 (0.98)	7.27 (0.78)	10.9 (0.12)	8.35 (0.06)	6.96 (2.12)	6.77 (1.53)	6.52 (0.32)
KNN ($k=1$)	99.5 (0.02)	81.4 (0.12)	9.34 (0.50)	7.77 (0.31)	6.21 (0.41)	97.6 (0.05)	74.1 (0.03)	7.02 (1.16)	6.77 (1.46)	6.46 (0.79)
KNN ($k=5$)	98.4 (0.05)	69.8 (0.18)	7.34 (1.08)	7.02 (0.87)	7.46 (1.31)	93.2 (0.20)	58.1 (0.16)	6.71 (0.17)	6.77 (0.27)	6.33 (0.36)
KNN ($k=13$)	96.8 (0.08)	62.4 (0.03)	7.84 (1.02)	6.96 (0.01)	8.09 (0.31)	88.9 (0.20)	47.7 (0.16)	8.09 (0.70)	7.65 (0.50)	6.77 (0.55)
KNN ($k=25$)	94.7 (0.08)	56.8 (0.06)	7.52 (0.82)	6.71 (0.84)	5.89 (1.31)	85.0 (0.02)	41.5 (0.11)	7.52 (1.01)	6.90 (0.36)	7.71 (0.85)
KNN ($k=53$)	90.8 (0.06)	50.0 (0.15)	6.83 (0.78)	7.34 (0.46)	7.40 (1.83)	79.4 (0.21)	35.1 (0.09)	7.21 (0.44)	7.08 (1.02)	6.33 (0.69)
KNN ($k=89$)	86.5 (0.17)	44.9 (0.12)	6.83 (0.49)	6.96 (0.95)	6.33 (0.24)	74.7 (0.18)	31.4 (0.14)	6.65 (0.44)	7.34 (0.94)	6.46 (1.07)
KNN ($k=139$)	81.1 (0.10)	40.7 (0.09)	7.59 (1.05)	6.46 (0.38)	6.14 (0.09)	69.8 (0.14)	28.5 (0.19)	7.21 (0.58)	7.52 (0.27)	5.27 (1.01)
LDA	12.2 (0.13)	11.2 (0.07)	11.0 (0.55)	10.3 (1.97)	7.15 (0.46)	13.6 (0.13)	10.5 (0.11)	9.78 (1.71)	9.15 (1.48)	8.15 (0.39)
SVM (RBF)	n/a	n/a	5.14 (0.23)	4.95 (0.64)	4.26 (0.36)	n/a	n/a	5.20 (0.17)	4.64 (0.08)	4.76 (0.18)
SVM (polynomial)	n/a	n/a	4.26 (0.17)	4.14 (0.54)	4.51 (0.41)	n/a	n/a	4.77 (0.64)	4.14 (0.23)	4.58 (0.32)
SVM (sigmoid)	n/a	n/a	5.45 (0.40)	4.64 (0.78)	4.51 (0.15)	n/a	n/a	5.02 (0.50)	4.26 (0.47)	4.45 (0.62)

Table 4.3: Binary and FEIS dataset classification testing accuracy results on conventional algorithms, where cyan and red colours highlight the best and the worst mean classifier accuracy respectively, e stands for the number of estimators in AB, lr stands for the learning rate of AB and k stands for the number of nearest neighbours in KNN.

Method	Binary dataset mean percentage accuracy (std. dev)					FEIS dataset mean percentage accuracy (std. dev)				
	Raw	Prep	LF	NLF	MFCC	Raw	Prep	LF	NLF	MFCC
AB ($e=50$, $lr=0.1$)	66.6 (0.14)	59.8 (0.49)	60.1 (8.10)	59.5 (1.99)	45.5 (3.05)	14.0 (0.33)	12.6 (0.18)	9.43 (1.11)	9.18 (0.89)	7.17 (0.53)
AB ($e=50$, $lr=0.5$)	72.1 (0.30)	63.0 (0.55)	57.0 (2.86)	57.0 (2.81)	50.0 (2.22)	16.7 (0.15)	13.7 (0.14)	10.1 (2.39)	8.81 (1.25)	5.16 (0.64)
AB ($e=50$, $lr=1.0$)	74.1 (0.28)	63.9 (0.41)	55.5 (4.44)	53.0 (4.04)	53.5 (5.37)	16.2 (0.43)	13.6 (0.08)	9.18 (2.05)	7.80 (1.86)	5.79 (2.05)
AB ($e=50$, $lr=1.5$)	73.7 (0.37)	63.7 (0.50)	62.0 (4.97)	51.0 (5.19)	50.5 (3.67)	14.9 (0.22)	12.9 (0.22)	7.80 (1.25)	5.66 (0.62)	7.42 (1.58)
AB ($e=50$, $lr=2.0$)	49.8 (0.27)	50.0 (0.07)	47.0 (6.48)	55.5 (7.71)	53.0 (6.63)	8.19 (0.94)	8.86 (0.69)	10.9 (4.08)	8.18 (1.25)	6.54 (1.08)
AB ($e=100$, $lr=1.0$)	77.0 (0.33)	65.4 (0.51)	59.0 (7.40)	54.0 (5.57)	52.0 (4.33)	17.3 (0.23)	14.4 (0.10)	10.6 (0.31)	7.67 (1.46)	6.04 (1.07)
AB ($e=200$, $lr=1.0$)	78.4 (0.25)	66.6 (0.57)	48.0 (3.67)	46.6 (9.53)	54.5 (3.70)	17.6 (0.27)	14.7 (0.12)	8.43 (2.00)	7.17 (2.02)	5.41 (1.46)
AB ($e=400$, $lr=1.0$)	79.6 (0.41)	67.0 (0.53)	57.0 (9.76)	53.0 (2.82)	47.0 (2.65)	16.4 (0.12)	14.4 (0.12)	8.81 (0.47)	7.17 (1.07)	3.90 (0.64)
Gaussian NB	50.0 (0.46)	53.7 (0.42)	50.5 (4.61)	50.0 (9.35)	58.5 (1.29)	12.8 (0.17)	11.5 (0.16)	7.92 (1.41)	8.43 (1.30)	6.04 (1.11)
KNN ($k=1$)	99.9 (0.00)	94.9 (0.14)	49.1 (8.97)	50.0 (3.05)	49.5 (4.17)	94.6 (0.11)	57.3 (0.28)	9.18 (1.52)	8.30 (0.81)	7.42 (0.64)
KNN ($k=5$)	99.7 (0.07)	91.2 (0.45)	48.5 (8.10)	50.1 (8.91)	47.0 (2.65)	90.3 (0.15)	46.6 (0.19)	8.30 (0.92)	7.30 (0.99)	6.16 (0.64)
KNN ($k=13$)	99.5 (0.01)	87.6 (0.30)	49.5 (5.13)	56.5 (3.04)	51.5 (3.81)	87.1 (0.14)	40.8 (0.10)	6.67 (1.55)	7.67 (0.18)	6.79 (0.62)
KNN ($k=25$)	99.0 (0.12)	84.7 (0.07)	47.5 (3.37)	51.0 (5.98)	54.0 (9.47)	84.2 (0.06)	37.6 (0.18)	7.04 (1.08)	6.16 (3.21)	6.16 (0.89)
KNN ($k=53$)	97.8 (0.15)	80.8 (0.42)	48.0 (1.27)	50.0 (1.62)	52.5 (0.92)	79.8 (0.13)	33.5 (0.10)	6.04 (1.41)	5.91 (0.78)	5.66 (1.60)
KNN ($k=89$)	95.8 (0.05)	78.0 (0.30)	47.5 (3.36)	49.5 (0.35)	51.5 (2.21)	75.4 (0.07)	30.9 (0.06)	5.03 (1.88)	5.16 (0.89)	5.03 (1.08)
KNN ($k=139$)	93.7 (0.30)	75.5 (0.60)	50.0 (2.22)	45.5 (1.60)	47.5 (1.27)	71.2 (0.21)	28.8 (0.13)	5.41 (2.16)	6.42 (1.11)	5.53 (1.28)
LDA	61.9 (0.18)	57.5 (0.68)	47.0 (8.44)	57.5 (5.24)	54.0 (4.65)	13.0 (0.22)	12.4 (0.23)	16.2 (1.41)	11.8 (1.92)	6.92 (1.25)
SVM (RBF)	75.5 (0.45)	68.7 (0.14)	46.5 (1.80)	46.0 (0.92)	47.5 (1.27)	n/a	n/a	3.77 (0.81)	4.28 (0.18)	4.28 (0.36)
SVM (polynomial)	71.1 (0.25)	55.8 (0.55)	44.0 (1.59)	43.0 (1.59)	43.5 (3.14)	n/a	n/a	4.53 (0.53)	3.77 (0.40)	4.03 (1.17)
SVM (sigmoid)	49.6 (0.14)	49.8 (0.91)	44.5 (3.12)	43.5 (3.96)	46.0 (1.57)	n/a	n/a	4.15 (0.31)	4.03 (1.23)	3.52 (0.71)

Table 4.4: EEGNet classification results, where cyan and red colours highlight the best and the worst mean classifier accuracy respectively.

		Mean percentage accuracy (std. dev)				
Dataset	Speech	Raw	Prep	LF	NLF	MFCC
P00	Imagined	3.12 (3.42)	n/a	n/a	n/a	n/a
	Inner	3.12 (2.80)	n/a	n/a	n/a	n/a
P01	Imagined	7.50 (3.75) 6.25 (6.56)	5.00 (1.53) 6.25 (3.42)	6.25 (1.48) 9.00 (2.46)	6.62 (1.75) 7.25 (2.08)	6.38 (1.08) 8.00 (1.65)
	Inner	10.6 (4.24) 8.75 (8.48)	6.88 (4.15) 6.25 (5.23)	7.12 (2.58) 5.75 (1.21)	5.75 (2.07) 6.50 (2.29)	6.38 (1.83) 7.00 (1.65)
P02	Imagined	8.12 (3.19)	6.25 (1.98)	7.75 (0.50)	7.25 (1.46)	8.00 (0.73)
	Inner	8.12 (3.19)	8.75 (6.06)	8.00 (1.33)	7.25 (1.46)	6.25 (1.19)
P04	Imagined	7.50 (3.19)	8.12 (5.08)	6.88 (1.19)	5.50 (2.28)	7.62 (2.22)
	Inner	6.88 (3.64)	8.75 (3.64)	7.62 (1.91)	6.12 (1.87)	5.12 (0.73)
Binary	Imagined	90.0 (12.3)	90.0 (12.3)	100 (0.00)	99.0 (2.00)	100 (0.00)
FEIS	Imagined	8.75 (5.00)	7.50 (4.68)	9.50 (2.69)	5.00 (0.79)	7.00 (2.81)

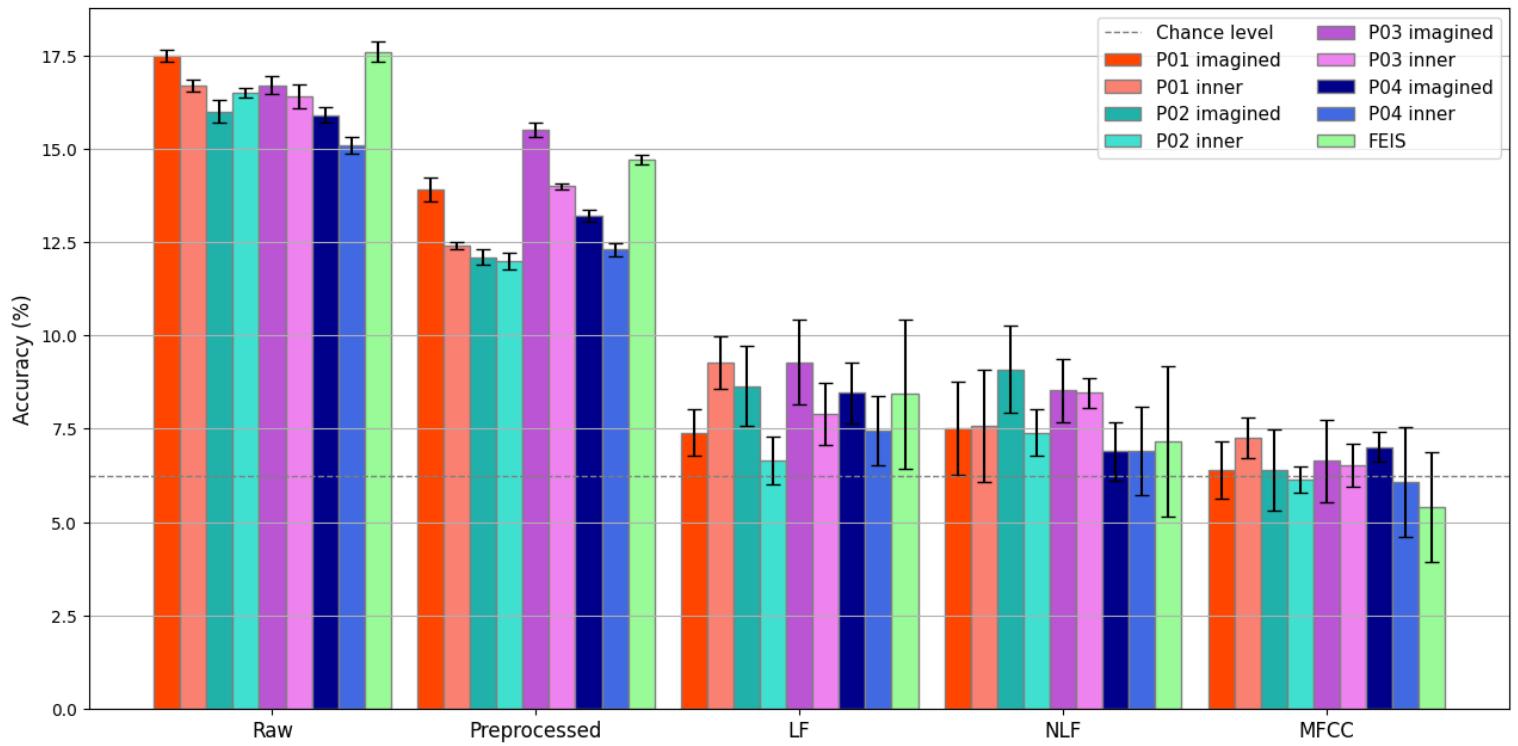


Figure 4.1: AB classification accuracy using different datasets.

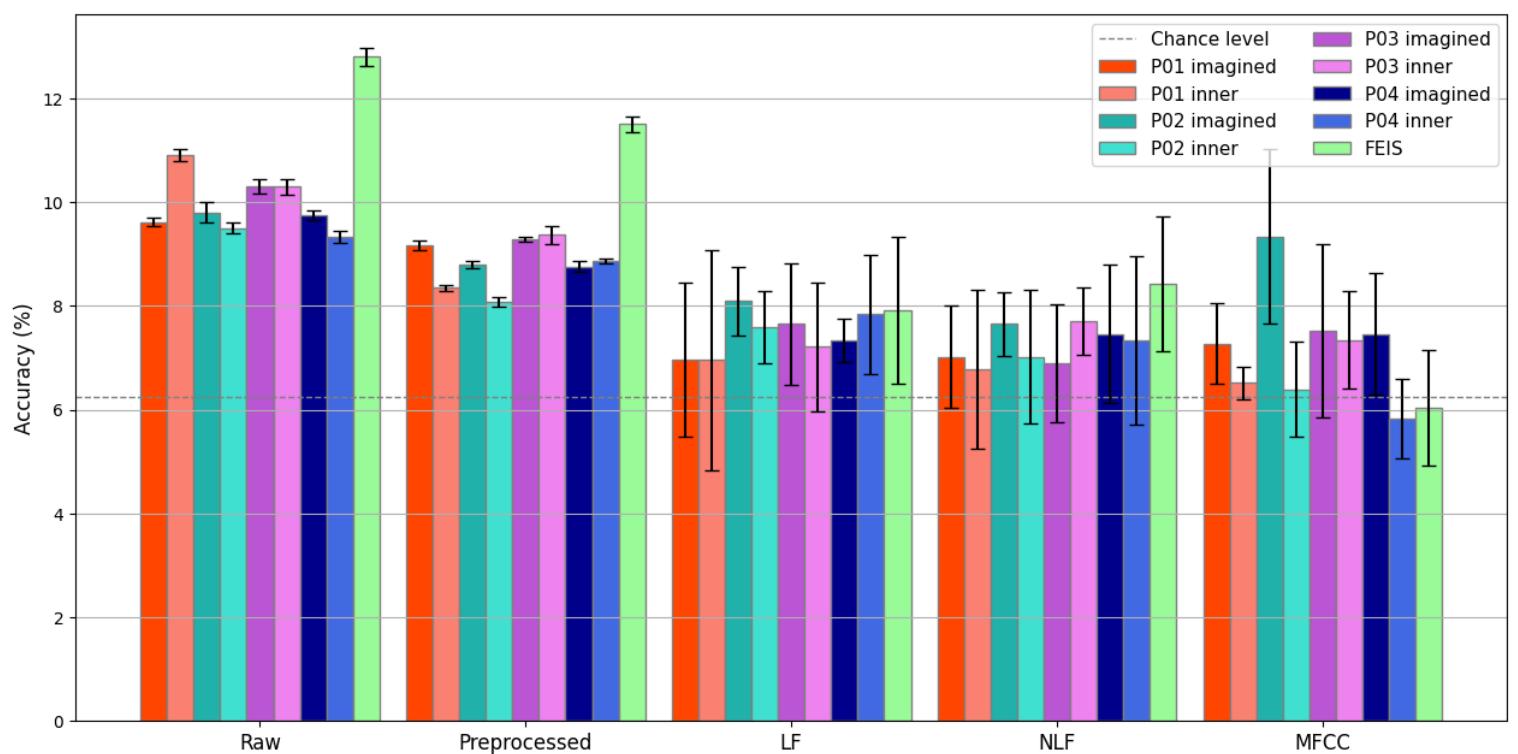


Figure 4.2: Gaussian NB classification accuracy using different datasets.

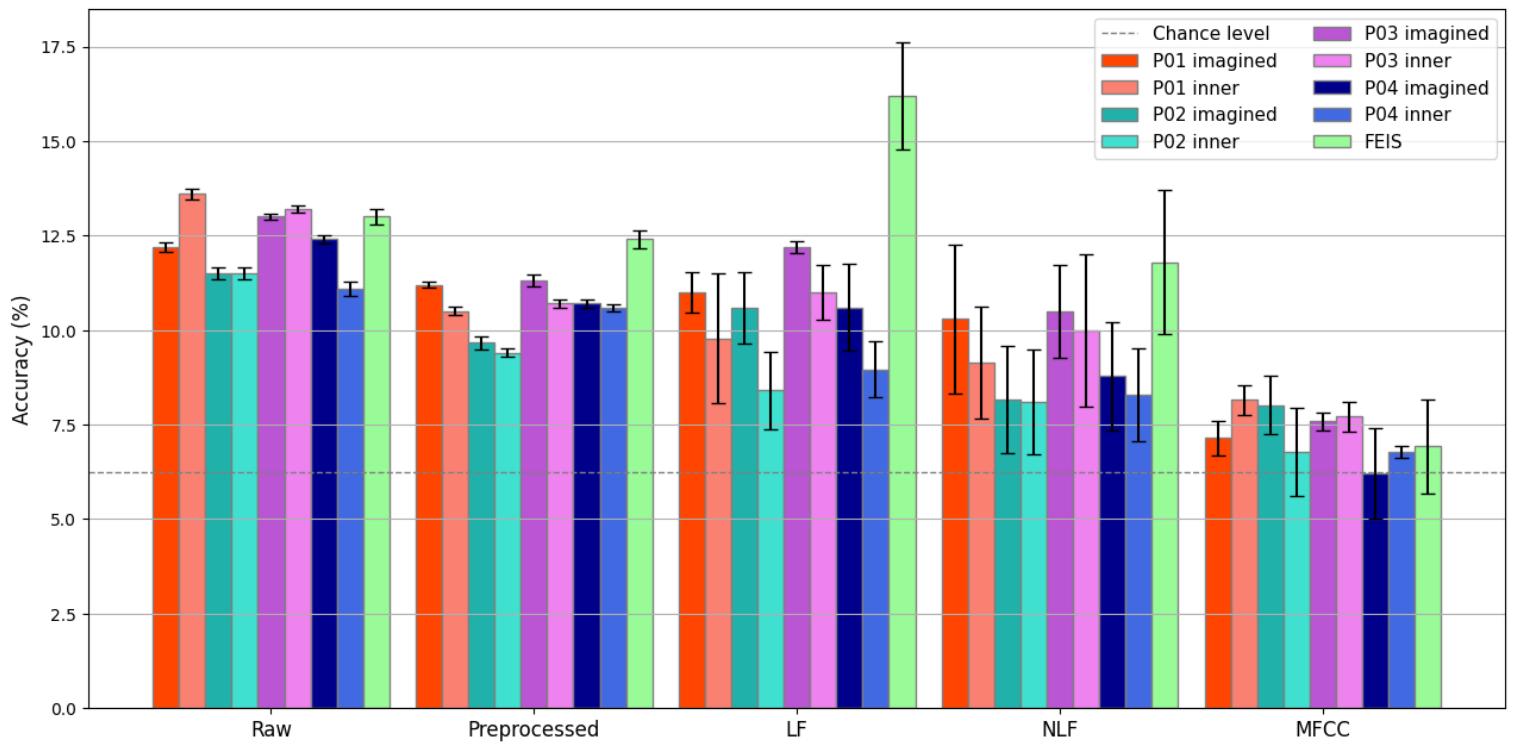


Figure 4.3: LDA classification accuracy using different datasets.

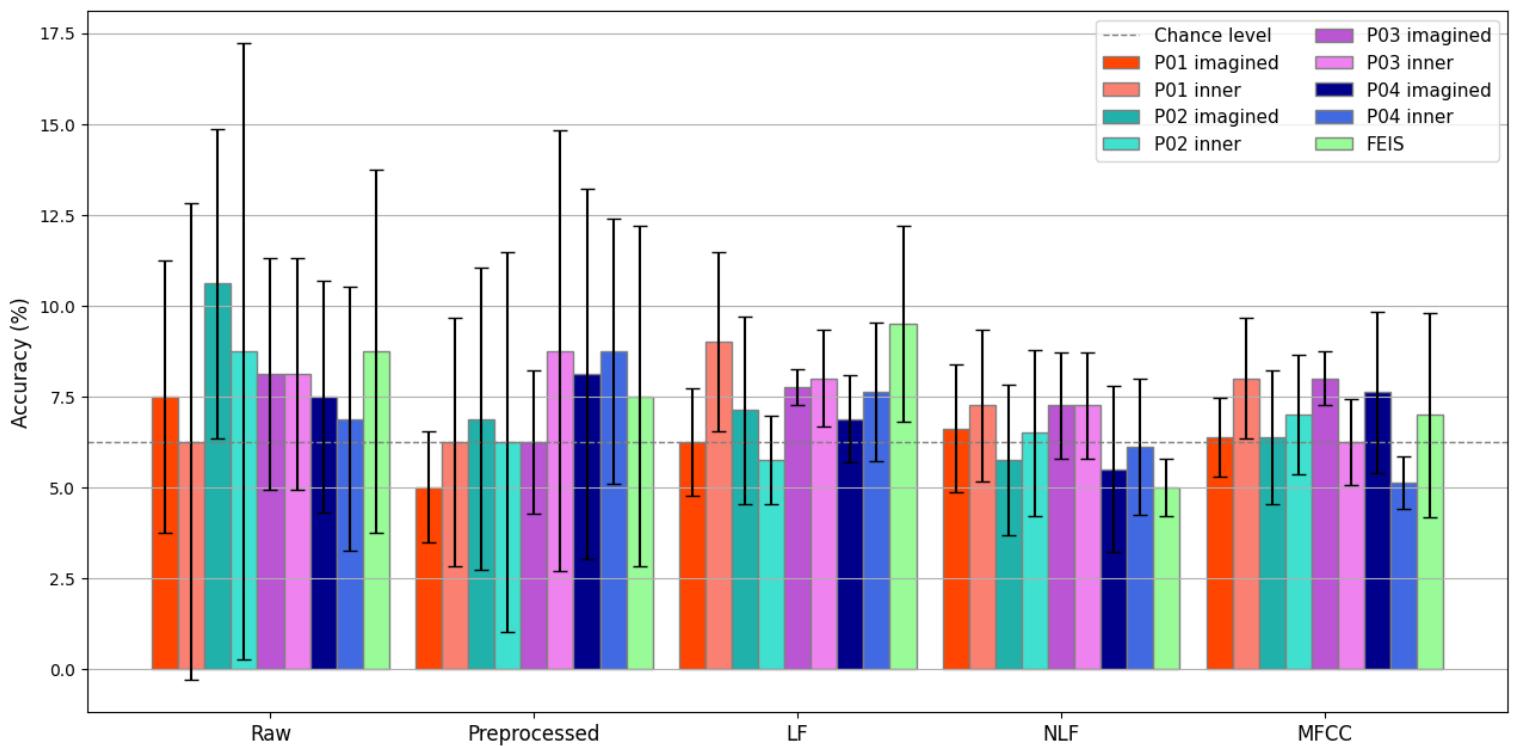


Figure 4.4: EEGNet classification accuracy using different datasets.

Chapter 5

Discussion

Section 5.1 discusses the arisen data correlation issues between training and testing datasets. Examined in Section 5.2 is the dataset acquired with the corrupted Emotiv EPOC device. Section 5.3 compares conventional algorithms and summarises optimal parameters. EEGNet performance analysis is provided in Section 5.4, followed by raw, preprocessed, LF, NLF and MFCC types of data decomposition comparisons in Section 5.5. Section 5.6 discusses differences in the accuracy of results when performing inner and imagined speech.

5.1 Training and Testing Data Correlation

As mentioned by Shen *et al.* [45], time-series data is naturally highly correlated. Combined with feature extraction from overlapping windows, datasets result in high correlations between adjacent window features. In this study, the correlation is reduced by employing zero overlapping between windows, although this alone is insufficient. During experimentation, feature windows are split randomly between training and testing samples, resulting in dependent windows skewing results. Whilst preparing raw and preprocessed data for conventional classifiers, individual data points were randomly split into training and testing sets, thereby leading to the high correlation between the two. As mitigation, we propose individual data instances be placed into containers such as the corresponding original EEG epochs; uncorrelated by the experimental paradigm. This methodology is employed to prepare data for EEGNet; utilising the robust API provided by PyTorch's Dataset and DataLoader data primitives. EEGNet demonstrates poor performance relative to conventional alternatives, however, data utilised for predictions is uncorrelated to data used for training, thereby indicating results obtained with EEGNet are more reliable. Despite the compromised data preparation technique used for conventional classifiers, it was utilised for all conventional classifiers. Obtained results and findings are thus analysed and discussed in subsequent sections.

5.2 Participant 00 Dataset

Although the raw P00 dataset was omitted from further decomposition, it was still classified. Conventional methods achieve around chance mean accuracy between 6.10% and 6.44%, as indicated in Table 4.1, whilst EEGNet achieved a mean accuracy of 3.12% for both inner and imagined speech paradigms, thus failing to reach the chance level. The observed higher accuracy for conventional classifiers indicates a high correlation between the training and testing

data used. Furthermore, as opposed to highlighted patterns observed with other datasets, the variation between algorithms and their parameters does not yield significant differences between accuracy values, suggesting the algorithms solely rely on data correlation. EEGNet yields the anticipated below chance accuracy, confirming the correlated data speculation. P00 data is disregarded in further discussion.

5.3 Conventional Classifiers

From the P01-04, FEIS and binary classification data, it is apparent that KNN exceeds other algorithms in accuracy by up to 82%, thus KNN is discussed in the subsequent section.

SVM is a common benchmark algorithm across EEG classification literature [2, 23, 26, 42, 53, 54, 56, 57]. However, our results do not suggest that SVM is superior to other classifiers. All datasets indicate SVM achieves below chance accuracy. This is a subpar performance compared to the binary classification performed by Clayton *et al.*, who reach 69.0% accuracy on the FEIS dataset [23]. RBF and polynomial kernels demonstrate comparable binary accuracy to other conventional methods for raw and preprocessed data. However, SVM and other conventional classifiers have more adjustable parameters not covered by this study. They could benefit from hyperparameter sweeps, similarly to EGGNet, producing higher results.

Analogous to results obtained by Balaji *et al.* [57], AB offers consistently higher accuracy scores than SVM. This also extends to Gaussian NB and LDA in all experiments, except binary classification. AB, Gaussian NB and LDA, unlike SVM, are not majorly affected by increasing computational time on multiclass classification tasks. Researchers in reviewed literature often apply overlapping sliding windows for feature extraction. Their methods for mitigating the high correlation between training and testing sets are undetailed. The methods used may have led to the higher SVM classification accuracy than observed by us.

AB performs best with 200 estimators and a 1.0 learning rate across all multiclass classification tasks, with the highest accuracy of 17.6% achieved on the raw FEIS dataset. AB achieves 79.6% accuracy on raw data for binary classification, with 400 estimators and a 1.0 learning rate.

5.3.1 K-Nearest Neighbours

KNN demonstrates exceptionally high accuracy results on raw and preprocessed datasets across all experiments. When $k = 1$, KNN reaches up to 99.9% and 99.5% mean accuracy in binary and multiclass classification respectively. These values decrease as k increases. Further experiments are performed with growing k values, limited by the sample point number. Figure 5.1a demonstrates the common accuracy trend between P01-04 and FEIS datasets. The accuracy rapidly drops with increasing k values, converging towards 21.0% and 14.0% accuracy for raw and preprocessed data respectively. Figure 5.1b demonstrates the binary classification trend, with raw and preprocessed data converging at 93% and 75% accuracy respectively. KNN behaviour on raw and preprocessed data is proposed to be due to high clustering of data points within and across classes in the same region. It is likely the correlated testing data points are located close to the points of the correct class, however, as k value increases, points from nearby class clusters negatively affect the accuracy.

The same pattern is not replicated when performing KNN on extracted features. Although

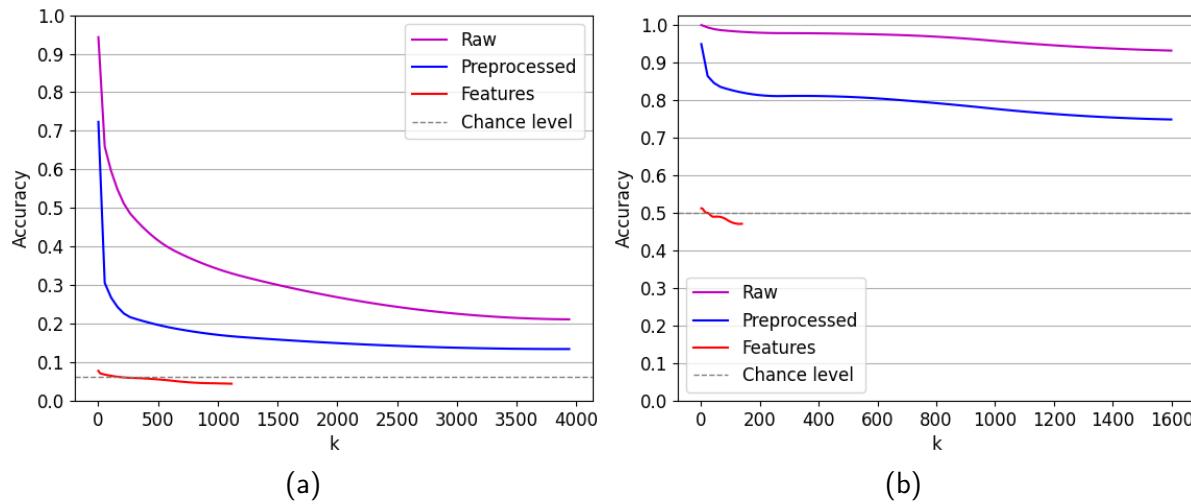


Figure 5.1: Graphs of KNN accuracy against the value of k across datasets. (a) Graph representing the accuracy trends in datasets for P01-P04 and FEIS. (b) Graph representing the accuracy trends in binary classification.

lower k values demonstrate higher accuracy, they do not exhibit an extreme accuracy peak. These accuracy results are in the same range as for other classifiers. Increasing k values to the maximum reaches below chance results. The lower performance for preprocessed and featured data, compared to raw, is proposed to be due to the dispersion of highly clustered raw data points by the performed data manipulation.

The high dependence of KNN on data correlation deems it an unreliable method. With this and the impracticality of SVM, all further analysis is performed using AB with 200 estimators and 1.0 learning rate, Gaussian NB and LDA.

5.4 EEGNet

EEGNet was designed for classifying raw EEG signals, although it is a robust network able to generalise across different BCI paradigms [50]. Thus, the model was trained using all decomposition types. However, due to the statistical correlation of feature windows, accuracy results in binary classification were observed to be 99%, 100% and 100% for NLF, LF and MFCC respectively, thereby outperforming raw and preprocessed data at 90%. EEGNet equally suffers from featured data correlation as conventional algorithms, hence multiclass results can be directly compared with similar classification accuracy observed.

5.5 Data Decomposition Types

A primary aim was to identify the optimal data decomposition method. Figure 4.1 indicates a significant difference favouring raw and preprocessed data with the AB classifier. LF, NLF and MFCC show inferior results. Furthermore, raw and preprocessed data demonstrate low standard deviation. Gaussian NB, Figure 4.2, and LDA, Figure 4.3, results further demonstrate this standard deviation pattern, without as significant accuracy difference between decomposition methods. We speculate this pattern emerges due to high testing and training correlation within raw and preprocessed data. We, therefore, do not use conventional classifiers to compare raw

EEG signals with their extracted features.

The results obtained with EEGNet, where the data correlation issue within raw and preprocessed data was mitigated, supply evidence for our speculation. Raw and preprocessed data no longer demonstrate significant classification performance over LF, NLF and MFCC, as shown in Figure 4.4. EEGNet also indicates comparable standard deviation across all decomposition types.

Cyan bars in Figures 4.1, 4.2 and 4.3 indicate the FEIS dataset demonstrates significantly superior classification performance with conventional classifiers, whilst a comparable performance is observed with EEGNet in Figure 4.4. Similarly, Clayton *et al.* [23] achieved better classification with SVM than CNN, thereby indicating the rapid experiment pace employed in this work; 3 seconds per EEG epoch, as opposed to 5 seconds, with the same phoneme set and headset, does not show inferior performance. However, further investigation into the causes, many of which are described in Section 3.1.2, of difference within conventional models may be necessary.

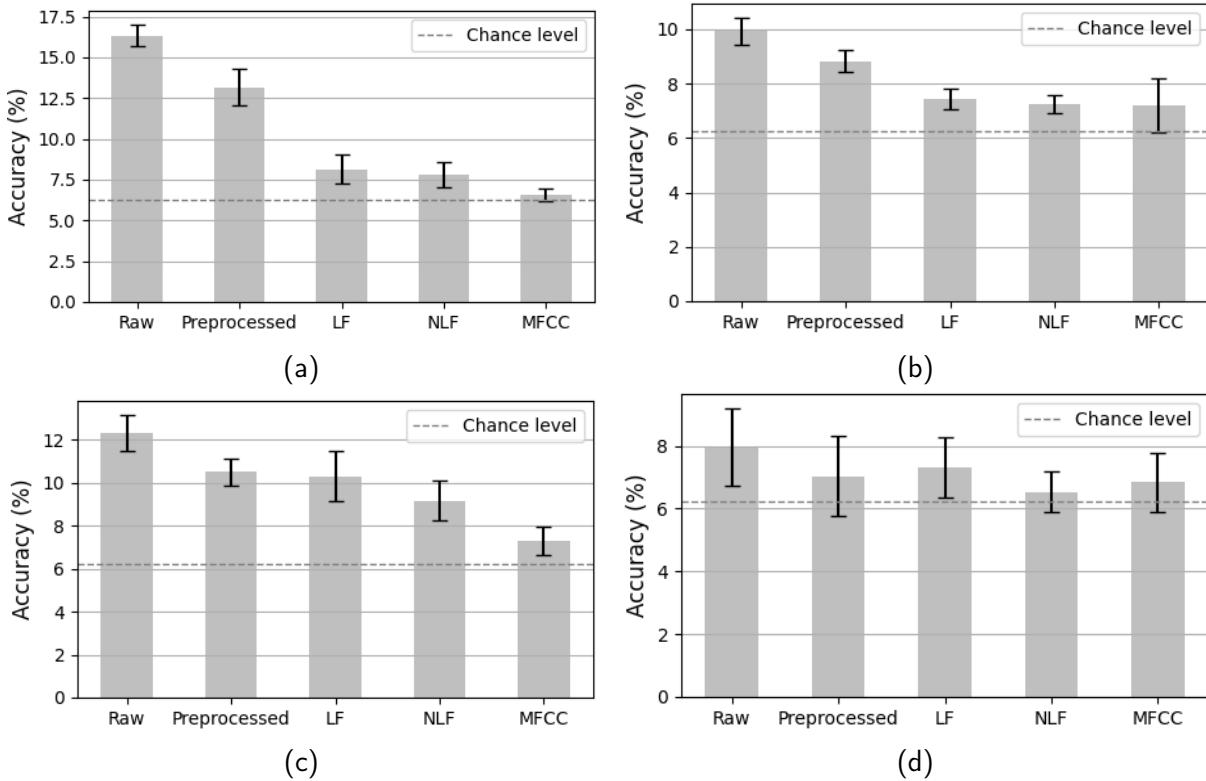


Figure 5.2: Mean accuracy across datasets on each data type using different classifiers. (a) AB. (b) Gaussian NB. (c) LDA. (d) EEGNet.

To further analyse which decomposition method yields superior classification performance, the mean accuracy is found between P01-04 datasets, for all five decomposition types across the four classifiers. Results are illustrated in Figure 5.2. Figures 5.2a, 5.2b and 5.2c indicate the presence of training and testing data correlation, resulting in higher raw data classification accuracy. Nevertheless, meeting the expectation of EEGNet, originally designed to classify raw EEG data, shown in Figure 5.2d, raw data demonstrates the highest average performance of 8%, followed by LF at 7%. However, manipulating network architecture to encourage work with extracted statistical features would likely increase accuracy with LFs, NLFs and MFCCs. Cooney *et al.* [42] report that MFCCs are superior to LFs and NLFs, however, our results oppose their claim; MFCCs show significantly lower results with AB and LDA classifiers, and

insignificant differences with Gaussian NB and EEGNet. Thus rejecting the first hypothesis (H1). LFs demonstrated superior performance with LDA and EEGNet, whilst only comparable with other classifiers.

Figure 5.2 demonstrates accuracy achieved by preprocessed data is inferior to raw data in all instances. The reasons for this require further investigation, however, we anticipate this could be a result of a sub-optimal use of preprocessing techniques described in Section 3.4. Possibly, an excessive amount of information was excluded during ICA.

5.6 Imagined Speech and Inner Speech

To derive a valid quantification supporting either imagined or inner speech during BCI tasks, raw and preprocessed accuracy results on AB, Gaussian NB and LDA are omitted from further analysis, whilst maintaining full EEGNet results. Thus, we minimise the correlation effect between training and testing datasets.

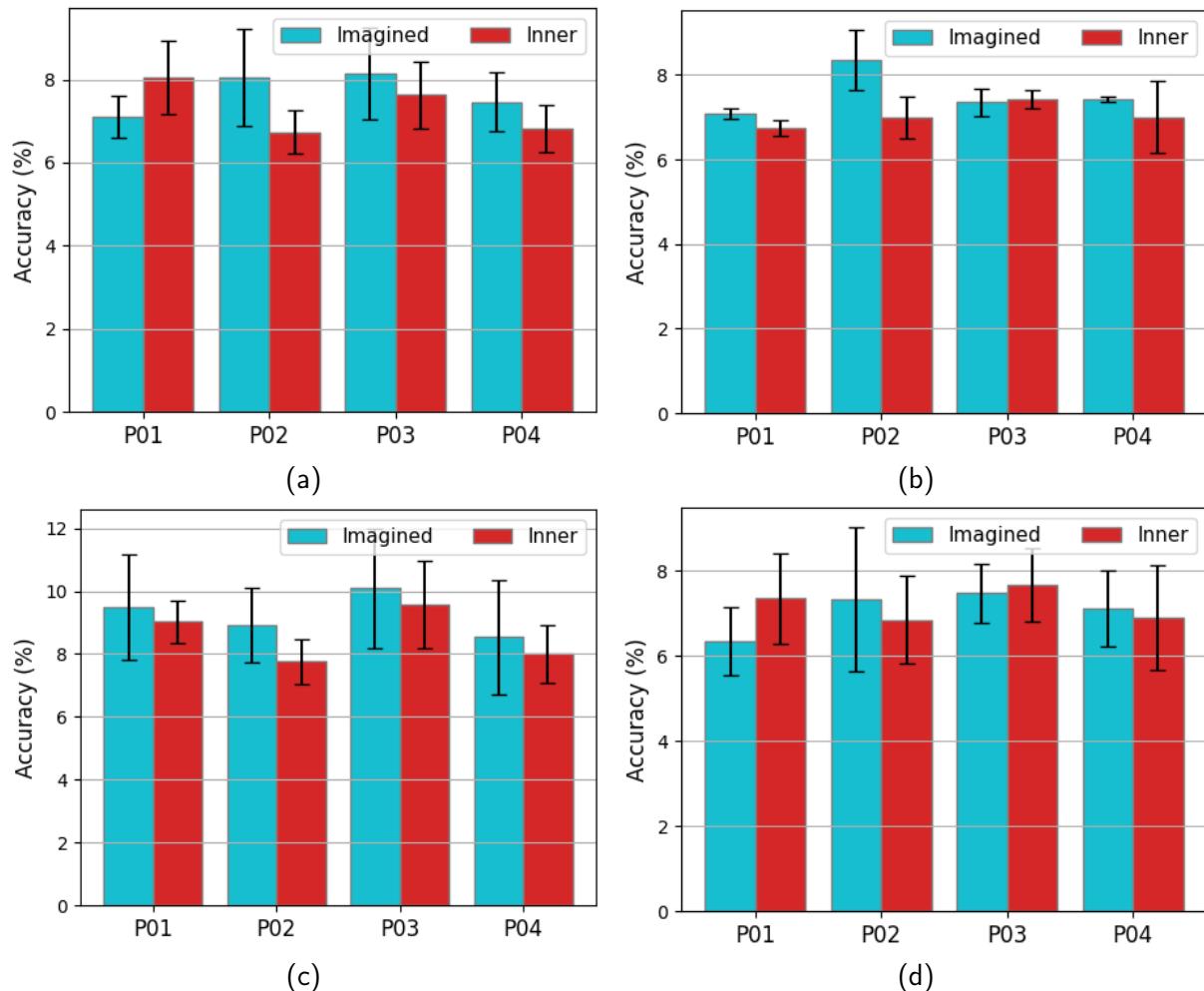


Figure 5.3: Mean accuracy of different classifiers on imagined speech and inner speech between different participants. (a) AB. (b) Gaussian NB. (c) LDA. (d) EEGNet.

The mean across the selected decomposition types for each classifier, for each speech type, and for each participant is calculated, as demonstrated in Figure 5.3.

We find no consistent evidence towards imagined or inner speech between the classifiers and datasets. Although all classifiers indicate imagined speech to be the superior paradigm for P02 and P04, there are discrepancies between classifiers for P01 and P03. As such, we cannot deduce either of the speech modalities is favourable for BCI applications, thus rejecting the second hypothesis (H2). Further investigations are required, but the results observed indicate the limitations discussed in Chapter 3.1.2: it is uncertain whether participants performed tasks correctly, and whether the understanding of imagined and inner speech definitions agreed between participants.

Moreover, we anticipate the possibility of variation in individuals' ability to comprehend vivid imagination of their articulators moving or maintaining inner speech in pure thought, perhaps relating to their talents in creativity and abstract thinking.

5.7 Summary

The chapter began with an analysis of the results for correlation between training and testing datasets across all classifiers, with the subsequent sections confirming the presence of this correlation. The corrupted dataset P00 is discussed, yielding confirmation of the correlation. The trends in results for conventional and deep learning classifiers were analysed, selecting the methods suitable for further analysis. Performing this analysis on data decomposition types demonstrated MFCCs' subpar performance, rejecting H1. Lastly, it was shown that the methodology employed in this study was not sufficient to support either imagined or inner speech, thus rejecting H2.

Chapter 6

Conclusions and Future Work

This study demonstrates the successful use of a commercially available, low SNR 14-channel EEG headset, Emotiv EPOC+, to solve the complex problem of decoding imagined and inner speech. Clear theoretical differentiation between the speech modalities was drawn with a novel dataset publicly released. The work attempted to establish a practical difference between the speech modalities, concluding the employed methodology was insufficient. The research also derived various methods for performing EEG data decomposition. Overall, the best reliable 16-class accuracy of 8% and binary accuracy of 90% were achieved on minimally processed data with EEGNet.

Our novel dataset brings the opportunity for researchers to perform further studies defining the practical use of imagined and inner speech processes with an affordable EEG headset, whilst experimenting with data decomposition and classification techniques. Meanwhile, the experimental paradigm could benefit from a more robust method of preparing participants for imagined and inner speech tasks through training, whilst also keeping participants engaged for longer; rewarding researchers with more data.

Despite the discussed limitations, the field would benefit by improving the understanding of the ability required to produce imagined and inner speech. We suspect a practical preference towards either modality was not found in this study, as some creative individuals may excel at engaging their motor neurons for imagined speech, whilst individuals with an analytical mindset may have stronger inner speech abilities. Further research in establishing this link when considering design choices for speech decoding BCI systems is required. A system designed for decoding imagined speech may prove ineffective in decoding inner speech. Furthermore, certain statistical features may be better at representing each speech type, leading to superior different ML algorithms or ANN architectures.

The findings of this research do not support the use of preprocessing and statistical feature decomposition for decoding speech. We propose deep learning may be more powerful at learning to discriminate useful information from noise given a sufficient amount of training data is provided. However, we suggest further work towards maintaining useful information during signal preprocessing, as well as tailoring deep learning classifiers to use extracted features.

Word count: 9899

Bibliography

- [1] Tanja Schultz et al. "Biosignal-Based Spoken Communication: A Survey". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 25.12 (2017), pp. 2257–2271. DOI: 10.1109/TASLP.2017.2752365.
- [2] Jerrin Thomas Panachakel and Angarai Ganesan Ramakrishnan. "Decoding Covert Speech From EEG-A Comprehensive Review". In: *Frontiers in Neuroscience* 15 (2021). ISSN: 1662-453X. DOI: 10.3389/fnins.2021.642251.
- [3] Jonathan R Wolpaw et al. "Brain–computer interfaces for communication and control". In: *Clinical Neurophysiology* 113.6 (2002), pp. 767–791. ISSN: 1388-2457. DOI: 10.1016/S1388-2457(02)00057-3.
- [4] Christian Herff and Tanja Schultz. "Automatic Speech Recognition from Neural Signals: A Focused Review". In: *Frontiers in Neuroscience* 10 (2016). ISSN: 1662-453X. DOI: 10.3389/fnins.2016.00429.
- [5] B. Denby et al. "Silent speech interfaces". In: *Speech Communication* 52.4 (2010). Silent Speech Interfaces, pp. 270–287. ISSN: 0167-6393. DOI: 10.1016/j.specom.2009.08.002.
- [6] Jeffrey J. Berry et al. "Speech sensorimotor learning through a virtual vocal tract". In: *Proceedings of Meetings on Acoustics* 19.1 (2013), p. 060099. DOI: 10.1121/1.4800666.
- [7] William F. Katz and Sonya Mehta. "Visual Feedback of Tongue Movement for Novel Speech Sound Learning". In: *Frontiers in Human Neuroscience* 9 (2015). ISSN: 1662-5161. DOI: 10.3389/fnhum.2015.00612.
- [8] Luis Fernando Nicolas-Alonso and Jaime Gomez-Gil. "Brain computer interfaces, a review". en. In: *Sensors (Basel)* 12.2 (Jan. 2012), pp. 1211–1279.
- [9] Emotiv. *Emotiv EPOC+*. 2022. URL: <https://www.emotiv.com/epoc/> (visited on 01/04/2022).
- [10] Andrea Biasiucci, Benedetta Franceschiello, and Micah M. Murray. "Electroencephalography". In: *Current Biology* 29.3 (2019), R80–R85. ISSN: 0960-9822. DOI: 10.1016/j.cub.2018.11.052.
- [11] Julie A Fiez. "Neuroimaging studies of speech: An overview of techniques and methodological approaches". In: *Journal of Communication Disorders* 34.6 (2001), pp. 445–454. ISSN: 0021-9924. DOI: [https://doi.org/10.1016/S0021-9924\(01\)00064-8](https://doi.org/10.1016/S0021-9924(01)00064-8).
- [12] Fabio Massimo Zanzotto and Danilo Croce. "Comparing EEG/ERP-Like and fMRI-Like Techniques for Reading Machine Thoughts". In: *Brain Informatics*. Ed. by Yiyu Yao et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 133–144. ISBN: 978-3-642-15314-3.
- [13] Rebecca L. Billingsley-Marshall, Panagiotis G. Simos, and Andrew C. Papanicolaou. "Reliability and Validity of Functional Neuroimaging Techniques for Identifying Language-

- Critical Areas in Children and Adults". In: *Developmental Neuropsychology* 26.2 (2004), pp. 541–563. DOI: 10.1207/s15326942dn2602_1.
- [14] Mia Illman et al. "Comparing MEG and EEG in detecting the 20-Hz rhythm modulation to tactile and proprioceptive stimulation". In: *NeuroImage* 215 (2020), p. 116804. ISSN: 1053-8119. DOI: 10.1016/j.neuroimage.2020.116804.
- [15] Guangye Li et al. "A preliminary study towards prosthetic hand control using human stereo-electroencephalography (SEEG) signals". In: *2017 8th International IEEE/EMBS Conference on Neural Engineering (NER)*. 2017, pp. 375–378. DOI: 10.1109/NER.2017.8008368.
- [16] Bernhard Gaimann et al. "A comparison between using ECoG and EEG for direct brain communication". In: *Proceedings of the EMBC05* (2005).
- [17] Alisa Berger et al. "Current State and Future Prospects of EEG and fNIRS in Robot-Assisted Gait Rehabilitation: A Brief Review". In: *Frontiers in Human Neuroscience* 13 (2019). ISSN: 1662-5161. DOI: 10.3389/fnhum.2019.00172.
- [18] Paola Pinti et al. "The present and future use of functional near-infrared spectroscopy (fNIRS) for cognitive neuroscience". In: *Annals of the New York Academy of Sciences* 1464.1 (2020), pp. 5–29. DOI: <https://doi.org/10.1111/nyas.13948>.
- [19] Darren J. Leamy, Rónán Collins, and Tomas E. Ward. "Combining fNIRS and EEG to Improve Motor Cortex Activity Classification during an Imagined Movement-Based Task". In: *Foundations of Augmented Cognition. Directing the Future of Adaptive Systems*. Ed. by Dylan D. Schmorow and Cali M. Fidopiastis. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 177–185. ISBN: 978-3-642-21852-1.
- [20] J. Artieda, M. Alegre, and M. Valencia. In: *Clinical Neurophysiology* 117.9 (2006), pp. 2109–2110. ISSN: 1388-2457. DOI: 10.1016/j.clinph.2006.06.001.
- [21] Michael Scherg et al. "Taking the EEG Back Into the Brain: The Power of Multiple Discrete Sources". In: *Frontiers in Neurology* 10 (2019). ISSN: 1664-2295. DOI: 10.3389/fneur.2019.00855.
- [22] M. A. Lopez-Gordo, D. Sanchez-Morillo, and F. Pelayo Valle. "Dry EEG Electrodes". In: *Sensors* 14.7 (2014), pp. 12847–12870. ISSN: 1424-8220. DOI: 10.3390/s140712847.
- [23] J. Clayton et al. "Decoding imagined, heard, and spoken speech: Classification and regression of EEG using a 14-channel dry-contact mobile headset". In: vol. 2020-October. cited By 4. 2020, pp. 4886–4890. DOI: 10.21437/Interspeech.2020-2745.
- [24] Akshi and Madhav Rao. "Decoding Imagined Speech Using Wearable EEG Headset For a Single Subject". In: *2021 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. 2021, pp. 2622–2627. DOI: 10.1109/BIBM52615.2021.9669361.
- [25] Ryota Sakai, Atsuhiko Kai, and Seiichi Nakagawa. "Classification of Imagined and Heard Speech Using Amplitude Spectrum and Relative Phase of EEG". In: *2021 IEEE 3rd Global Conference on Life Sciences and Technologies (LifeTech)*. 2021, pp. 373–375. DOI: 10.1109/LifeTech52111.2021.9391883.
- [26] Shunan Zhao and Frank Rudzicz. "Classifying phonological categories in imagined and articulated speech". In: *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2015, pp. 992–996. DOI: 10.1109/ICASSP.2015.7178118.
- [27] Charles S. DaSalla et al. "Single-trial classification of vowel speech imagery using common spatial patterns". In: *Neural Networks* 22.9 (2009). Brain-Machine Interface, pp. 1334–1339. ISSN: 0893-6080. DOI: <https://doi.org/10.1016/j.neunet.2009.05.008>.

- [28] Michael D'Zmura et al. "Toward EEG Sensing of Imagined Speech". In: *Human-Computer Interaction. New Trends*. Ed. by Julie A. Jacko. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 40–48. ISBN: 978-3-642-02574-7.
- [29] Anne Porbadnik et al. "EEG-based speech recognition". In: *Impact of Temporal Effects* (2009).
- [30] Krzysztof Kotowski et al. "Validation of Emotiv EPOC+ for extracting ERP correlates of emotional face processing". In: *Biocybernetics and Biomedical Engineering* 38.4 (2018), pp. 773–781. ISSN: 0208-5216. DOI: <https://doi.org/10.1016/j.bbe.2018.06.006>.
- [31] Irwin B. Levitan and Leonard K. Kaczmarek. *The NeuronCell and Molecular Biology: Cell and Molecular Biology*. Oxford, UK: Oxford University Press, July 2015. ISBN: 9780190239114. DOI: 10.1093/med/9780199773893.001.0001.
- [32] Nicolas Nieto et al. "Thinking out loud, an open-access EEG-based BCI dataset for inner speech recognition". In: *Scientific Data* 9.1 (Feb. 2022), p. 52. ISSN: 2052-4463. DOI: 10.1038/s41597-022-01147-2.
- [33] Timothée Proix et al. "Imagined speech can be decoded from low- and cross-frequency intracranial EEG features". In: *Nature Communications* 13 (2022). ISSN: 2041-1723. DOI: 10.1038/s41467-021-27725-3.
- [34] Zahra Jafari, Bryan E. Kolb, and Majid H. Mohajerani. "Neural oscillations and brain stimulation in Alzheimer's disease". In: *Progress in Neurobiology* 194 (2020), p. 101878. ISSN: 0301-0082. DOI: <https://doi.org/10.1016/j.pneurobio.2020.101878>.
- [35] S. Gupta and H. Singh. "Preprocessing EEG signals for direct human-system interface". In: *Proceedings IEEE International Joint Symposia on Intelligence and Systems*. 1996, pp. 32–37. DOI: 10.1109/IJSIS.1996.565048.
- [36] Gabriella Tamburro et al. "A new ICA-based fingerprint method for the automatic removal of physiological artifacts from EEG recordings". In: *PeerJ* 6 (2018), e4380.
- [37] Chuong H Nguyen, George K Karavas, and Panagiotis Arتمiadis. "Inferring imagined speech using EEG signals: a new approach using Riemannian manifold features". In: *Journal of Neural Engineering* 15.1 (Nov. 2017), p. 016002. DOI: 10.1088/1741-2552/aa8235. URL: <https://doi.org/10.1088/1741-2552/aa8235>.
- [38] Alexander Craik, Yongtian He, and Jose L Contreras-Vidal. "Deep learning for electroencephalogram (EEG) classification tasks: a review". In: *Journal of Neural Engineering* 16.3 (2019), p. 031001. DOI: 10.1088/1741-2552/ab0ab5.
- [39] Zitong Wan et al. "A review on transfer learning in EEG signal analysis". In: *Neurocomputing* 421 (2021), pp. 1–14. ISSN: 0925-2312. DOI: <https://doi.org/10.1016/j.neucom.2020.09.017>.
- [40] Ciaran Cooney, Raffaella Folli, and Damien Coyle. "Optimizing Layers Improves CNN Generalization and Transfer Learning for Imagined Speech Decoding from EEG". In: *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*. 2019, pp. 1311–1316. DOI: 10.1109/SMC.2019.8914246.
- [41] Pramit Saha and Sidney Fels. "Hierarchical Deep Feature Learning for Decoding Imagined Speech from EEG". In: *Proceedings of the AAAI Conference on Artificial Intelligence* 33.01 (July 2019), pp. 10019–10020. DOI: 10.1609/aaai.v33i01.330110019. URL: <https://ojs.aaai.org/index.php/AAAI/article/view/5146>.
- [42] Ciaran Cooney, Rafaella Folli, and Damien Coyle. "Mel Frequency Cepstral Coefficients Enhance Imagined Speech Decoding Accuracy from EEG". In: *2018 29th Irish Signals and Systems Conference (ISSC)*. 2018, pp. 1–7. DOI: 10.1109/ISSC.2018.8585291.

- [43] Todor Ganchev. "Contemporary Methods for Speech Parameterization". In: *Contemporary Methods for Speech Parameterization*. New York, NY: Springer New York, 2011, pp. 1–106. ISBN: 978-1-4419-8447-0. DOI: 10.1007/978-1-4419-8447-0_1.
- [44] Diah Risqiwati et al. "Feature Selection for EEG-Based Fatigue Analysis Using Pearson Correlation". In: *2020 International Seminar on Intelligent Technology and Its Applications (ISITIA)*. 2020, pp. 164–169. DOI: 10.1109/ISITIA49792.2020.9163760.
- [45] Kai-Quan Shen et al. "A Feature Selection Method for Multilevel Mental Fatigue EEG Classification". In: *IEEE Transactions on Biomedical Engineering* 54.7 (2007), pp. 1231–1237. DOI: 10.1109/TBME.2007.890733.
- [46] Omair Ali et al. "Enhancing the decoding accuracy of EEG signals by the introduction of anchored-STFT and adversarial data augmentation method". In: *Scientific Reports* 12 (2022), p. 4245. ISSN: 2045-2322. DOI: 10.1038/s41598-022-07992-w.
- [47] Miguel Angrick et al. "Speech synthesis from ECoG using densely connected 3D convolutional neural networks". In: *Journal of neural engineering* 16.3 (2019), p. 036019.
- [48] Ciaran Cooney et al. "Evaluation of Hyperparameter Optimization in Machine and Deep Learning Methods for Decoding Imagined Speech EEG". en. In: *Sensors (Basel)* 20.16 (Aug. 2020).
- [49] Lorenzo Brigato and Luca Iocchi. "A Close Look at Deep Learning with Small Data". In: *2020 25th International Conference on Pattern Recognition (ICPR)*. 2021, pp. 2490–2497. DOI: 10.1109/ICPR48806.2021.9412492.
- [50] Vernon J Lawhern et al. "EEGNet: a compact convolutional neural network for EEG-based brain-computer interfaces". en. In: *J Neural Eng* 15.5 (June 2018), p. 056013.
- [51] Nieto Nicolas et al. "*Inner Speech*". 2022. DOI: doi:10.18112/openneuro.ds003626.v2.1.2.
- [52] Germán A. Pressel Coretto, Iván E. Gareis, and H. Leonardo Rufiner. "Open access database of EEG signals recorded during imagined speech". In: *12th International Symposium on Medical Information Processing and Analysis*. Ed. by Eduardo Romero et al. Vol. 10160. International Society for Optics and Photonics. SPIE, 2017, p. 1016002. DOI: 10.1117/12.2255697. URL: <https://doi.org/10.1117/12.2255697>.
- [53] Dipti Pawar and Sudhir Dhage. "Multiclass covert speech classification using extreme learning machine". In: *Biomedical Engineering Letters* 10.2 (May 2020), pp. 217–226. ISSN: 2093-985X. DOI: 10.1007/s13534-020-00152-x.
- [54] Koji Koizumi, Kazutaka Ueda, and Masayuki Nakao. "Development of a Cognitive Brain-Machine Interface Based on a Visual Imagery Method". In: *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. 2018, pp. 1062–1065. DOI: 10.1109/EMBC.2018.8512520.
- [55] Alejandro Antonio Torres García., Carlos Alberto Reyes García., and Luis Villaseñor Pineda. "TOWARD A SILENT SPEECH INTERFACE BASED ON UNSPOKEN SPEECH". In: *Proceedings of the International Conference on Bio-inspired Systems and Signal Processing - BIOSIGNALS, (BIOSTEC 2012)*. INSTICC. SciTePress, 2012, pp. 370–373. ISBN: 978-989-8425-89-8. DOI: 10.5220/0003769603700373.
- [56] Alborz Rezazadeh Sereshkeh et al. "EEG Classification of Covert Speech Using Regularized Neural Networks". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 25.12 (2017), pp. 2292–2300. DOI: 10.1109/TASLP.2017.2758164.
- [57] Advait Balaji et al. "EEG-based classification of bilingual unspoken speech using ANN". In: *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. 2017, pp. 1022–1025. DOI: 10.1109/EMBC.2017.8037000.

- [58] Alycia Cummings, Amebu Seddoh, and Brianna Jallo. "Phonological code retrieval during picture naming: Influence of consonant class". en. In: *Brain Res* 1635 (Jan. 2016), pp. 71–85.
- [59] Gregory Hickok. "Computational neuroanatomy of speech production". en. In: *Nat Rev Neurosci* 13.2 (Jan. 2012), pp. 135–145.
- [60] Inge Timmers, Bernadette M Jansma, and M Estela Rubio-Gozalbo. "From mind to mouth: event related potentials of sentence production in classic galactosemia". en. In: *PLoS One* 7.12 (Dec. 2012), e52826.
- [61] Philip Lieberman. "The Evolution of Human Speech: Its Anatomical and Neural Bases". In: *Current Anthropology* 48.1 (2007), pp. 39–66. DOI: 10.1086/509092.
- [62] Ben Alderson-Day and Charles Fernyhough. "Inner Speech: Development, Cognitive Functions, Phenomenology, and Neurobiology". en. In: *Psychol Bull* 141.5 (May 2015), pp. 931–965.
- [63] Patrick Suppes, Zhong-Lin Lu, and Bing Han. "Brain wave recognition of words". In: *Proceedings of the National Academy of Sciences* 94.26 (1997), pp. 14965–14969. DOI: 10.1073/pnas.94.26.14965.
- [64] Vladislav Shishkin. *An investigation into decoding imagined and inner speech processes from commercial EEG devices*. Version 1.0.0. Aug. 2022. URL: https://github.com/vshis/emotiv_epoc_eeg_decoding_imagined_and_inner_speech.
- [65] EMOTIV. *EMOTIV Launcher*. Version 3.3.0. Oct. 2021. URL: <https://www.emotiv.com/emotiv-launcher/>.
- [66] CymatiCorp. *CyKIT*. Version 3.0. Dec. 27, 2018. URL: <https://github.com/CymatiCorp/CyKit>.
- [67] Lindgren Jussi et al. *OpenViBE*. Version 3.3.0. Apr. 20, 2022. URL: <http://openvibe.inria.fr/>.
- [68] Loki Software. *OpenAL*. Version 1.1. Nov. 2009. URL: <https://www.openal.org/>.
- [69] Alexandre Gramfort et al. "MEG and EEG Data Analysis with MNE-Python". In: *Frontiers in Neuroscience* 7.267 (2013), pp. 1–13. DOI: 10.3389/fnins.2013.00267.
- [70] Matthieu Duvinage et al. "Performance of the Emotiv EPOC headset for P300-based applications". In: *BioMedical Engineering OnLine* 12.1 (June 2013), p. 56. ISSN: 1475-925X. DOI: 10.1186/1475-925X-12-56.
- [71] James V. Stone. "Independent component analysis: an introduction". In: *Trends in Cognitive Sciences* 6.2 (2002), pp. 59–64. ISSN: 1364-6613. DOI: [https://doi.org/10.1016/S1364-6613\(00\)01813-1](https://doi.org/10.1016/S1364-6613(00)01813-1).
- [72] Raphael Vallat. *AntroPy*. Version 0.1.4. Apr. 2021. URL: <https://github.com/raphaelvallat/antropy>.
- [73] Dmitry Mottl. *hurst*. Version 0.0.5. Feb. 7, 2019. URL: <https://github.com/Mottl/hurst>.
- [74] Brian McFee et al. *librosa/librosa*: 0.9.2. Version 0.9.2. June 2022. DOI: 10.5281/zenodo.6759664. URL: <https://doi.org/10.5281/zenodo.6759664>.
- [75] F. Pedregosa et al. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.
- [76] Kashvi Taunk et al. "A Brief Review of Nearest Neighbor Algorithm for Learning and Classification". In: May 2019, pp. 1255–1260. DOI: 10.1109/ICCS45141.2019.9065747.
- [77] "Bayesian Networks as Classifiers". In: *Bayesian Networks and Decision Graphs: February 8, 2007*. New York, NY: Springer New York, 2007, pp. 265–276. ISBN: 978-0-387-68282-2. DOI: 10.1007/978-0-387-68282-2_8.

- [78] Keinosuke Fukunaga. "Chapter 10 - FEATURE EXTRACTION AND LINEAR MAPPING FOR CLASSIFICATION". In: *Introduction to Statistical Pattern Recognition (Second Edition)*. Ed. by Keinosuke Fukunaga. Second Edition. Boston: Academic Press, 1990, pp. 441–507. ISBN: 978-0-08-047865-4. DOI: <https://doi.org/10.1016/B978-0-08-047865-4.50016-8>.
- [79] William S Noble. "What is a support vector machine?" en. In: *Nat Biotechnol* 24.12 (Dec. 2006), pp. 1565–1567.
- [80] Yoav Freund and Robert E Schapire. "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting". In: *Journal of Computer and System Sciences* 55.1 (1997), pp. 119–139. ISSN: 0022-0000. DOI: <https://doi.org/10.1006/jcss.1997.1504>.
- [81] Trevor Hastie et al. "Multi-class adaboost". In: *Statistics and its Interface* 2.3 (2009), pp. 349–360.
- [82] Chang Gao, Wenchao Liu, and Xian Yang. "Convolutional neural network and riemannian geometry hybrid approach for motor imagery classification". In: *Neurocomputing* 507 (2022), pp. 180–190. ISSN: 0925-2312. DOI: <https://doi.org/10.1016/j.neucom.2022.08.024>.
- [83] Amirali Vahid et al. "Deep Learning Based on Event-Related EEG Differentiates Children with ADHD from Healthy Controls". In: *Journal of Clinical Medicine* 8.7 (2019). ISSN: 2077-0383. DOI: 10.3390/jcm8071055.
- [84] Marco Simões et al. "BCIAUT-P300: A multi-session and multi-subject benchmark dataset on autism for P300-based brain-computer-interfaces". In: *Frontiers in Neuroscience* 14 (2020), p. 568104.
- [85] Vernon J Lawhern and Alexandre Gramfort. *Army Research Laboratory EEGModels*. 2022. URL: <https://github.com/vlawhern/ar1-eegmodels>.
- [86] Martín Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Software available from tensorflow.org. 2015. URL: <https://www.tensorflow.org/>.
- [87] Adam Paszke et al. "PyTorch: An Imperative Style, High-Performance Deep Learning Library". In: *Advances in Neural Information Processing Systems* 32. Ed. by H. Wallach et al. Curran Associates, Inc., 2019, pp. 8024–8035. URL: <http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf>.
- [88] Lukas Biewald. *Experiment Tracking with Weights and Biases*. Software available from wandb.com. 2020. URL: <https://www.wandb.com/>.

Appendix A

Participant Information Sheet



PARTICIPANT INFORMATION SHEET

An investigation into decoding speech and language processes from commercial EEG devices

Name of Researcher: Vladislav Shishkin
Contact details of Researcher: vs556@bath.ac.uk

Name of Supervisors: Dr James Laird Scott Wellington
Contact details of Supervisors: jl317@bath.ac.uk sdlw20@bath.ac.uk

This information sheet forms part of the process of informed consent. It should give you the basic idea of what the research is about and what your participation will involve. Please read this information sheet carefully and ask one of the researchers named above if you are not clear about any details of the project.

1. What is the purpose of the project:

You are about to participate in a study which involves wearing an Emotiv 16-channel mobile EEG device. During the session, you will be asked to sit as comfortably and as still as possible while carrying out attention-based tasks. These tasks do not require you to move any part of your body throughout the recording session.

The session will take place here. Once the experiment is underway, your brain activity will be continuously monitored to record your EEG signals in the tasks.

We wish to investigate the extent to which it is possible to decode task-elicited information from your EEG data. Your session should last for 3 hours. You will be given full instructions shortly and will be able to ask any questions you may have.

2. Who can be a participant?

All persons are eligible to participate in this research.

3. Do I have to take part?

No, it is completely up to you to decide if you would like to participate. Before you decide to take part, we will describe the project and go through this information sheet with you. If you agree to take part, we will then ask you to sign a consent form.

If at any time you decide you no longer wish to take part in this project you are free to withdraw, without giving a reason.

This document explains what kind of study we're doing, what your rights are, and what will be done with your data. You should keep this page for your records.

4. What will I be asked to do?

Brain activity will be recorded of using Emotiv EPOC device for the total of 120 minutes. The experiment will be recording two modalities of brain activity: imagined speech and inner speech. To help the participants with the understanding of the two modalities, the following definitions are provided:

1. Imagined speech: a person imagining themselves saying something, such that they mentally carry out the process of how they would be moving their articulators (tongue, lips, vocal cords, etc,) in the production of that speech, without moving those organs.
2. Inner speech: a person imagining hearing their own voice (often referred to as the 'inner voice') saying something, such as when 'silent reading' at the same speed that they would speak those words.

60 minutes of recording will be allocated to imagined speech and the remaining 60 minutes to inner speech. However, the participants will be given a 5-to-10-minute break after every 20 minutes of recording. The experiment will be conducted by the participants alone, sitting on a comfortable chair, in front of a laptop monitor. The main procedure will obey the following steps:

1. Phone prompt appears on screen for three seconds, and the respective audio is played five times.
2. Blank white screen appears on screen for one second, allowing participants time to prepare for the imagination state.
3. Imagined speech state, where the participant repeats the phone five times using imagined speech / inner speech, with a symbol moving across the screen to set the rhythm.
4. Three second rest state, to give the participant time to clear mind prior to the next iteration of data collection.

The above steps will be repeated for the duration of the experiments in a randomised order, to keep the participants engaged.

5. Are there reasons why I should not take part?

There are no exclusion criteria for this research.

6. What are the possible benefits of taking part?

There are no direct benefits to you taking part in this research. However, you will be contributing to our knowledge about the utility of EEG in future remediation devices,

which may be used by individuals with acquired functional loss due to, for example, neurodegenerative diseases.

7. What are the possible disadvantages and risks of taking part?

There are no anticipated disadvantages or risks to you taking part in this research.

8. Will my participation involve any discomfort or embarrassment?

There is no anticipated discomfort or embarrassment during the process of collecting your data (i.e. the EEG recordings).

However, the data gathering will require you to sit still for a long period of time: if at any point during the data gathering you feel uncomfortable, or require a break, please alert the researcher who will stop the process. It will remain your choice whether or not to continue, with no penalty if you choose to discontinue.

9. Who will have access to the information that I provide?

During the research period, only the researcher and supervisors named at the start and end of this document will have access to the data you provide.

The only identifiable personal data in this research are the informed consent forms that are provided to you alongside this participant information document. Only the named supervisors will have the key to the locked cabinet where these are securely stored.

Non-identifiable data (e.g., EEG recordings) will be used for the training and evaluation of signal processing/machine learning pipelines only.

10. What will happen to the data collected and results of the project?

All the information we collect during the course of the research will be processed in accordance with Data Protection Law (GDPR). In order to safeguard your privacy, we will never share personal information (like names or dates of birth) with anyone outside the research team. Recorded data will not be kept for any longer than 5 years. Your name or other identifying information will not be disclosed in any presentation or publication of the research.

Your data will be referred to by a unique participant number rather than by name. During the research period, we will store any non-identifiable data (e.g., EEG recordings) securely as password protected files on the University of Bath's secure server (X drive) and on a password protected, encrypted hard drive. After the research period, we will store non-identifiable data securely using the University of Bath's Research Data Archive, alongside secure encrypted storage services. Personal data (e.g., signed forms) will be stored securely in a locked filing cabinet at the

University of Bath, and will not be kept for any longer than 5 years (after which they will be securely destroyed).

The anonymised data collected during this study will be used for research purposes. With your permission, your non-identifiable data such as EEG recordings may also be used for research or teaching purposes, and may be shared with other researchers or with the general public (e.g., we may make it available through the world wide web, or use it in TV or radio broadcasts). Your name or other identifying information will not be disclosed in any presentation or publication of the research.

Once this project is completed, other researchers at the University of Bath may conduct related research projects which would benefit from the use of the non-identifiable data (e.g., EEG recordings) that you have provided.

11. Who has reviewed the project?

This project has been reviewed through the EIRA1 process within the Department of Computer Science.

12. How can I withdraw from the project?

Your participation is voluntary, and you may withdraw from the study at any time without providing a reason for doing so and without any repercussions. If you withdraw from the study during data gathering, you can inform one of the above identified researchers in person. If you withdraw from the study after data gathering, you can inform one of the above identified researchers by email.

If you withdraw from the study during or after data gathering, we will delete your data and there is no penalty or loss of benefits to which you are otherwise entitled.

13. University of Bath privacy notice

The University of Bath privacy notice can be found here:

<https://www.bath.ac.uk/corporate-information/university-of-bath-privacy-notice-for-research-participants/>.

14. What happens if there is a problem?

If you have any concerns about what you've just read, or any questions about any aspect of the project, or if you have any concerns at any stage during or after the data gathering, please feel free to ask, or contact us later using the contact details at the start and end of this document. We will do our best to answer any questions.

15. If I require further information who should I contact and how?

Thank you for expressing an interest in participating in this project. Please do not hesitate to get in touch with us if you would like some more information.

Name of Researcher: Vladislav Shishkin
Contact details of Researcher: vs556@bath.ac.uk

Name of Supervisors: Dr James Laird Scott Wellington
Contact details of Supervisors: jl317@bath.ac.uk sdlw20@bath.ac.uk

Appendix B

Participant Consent Form

CONSENT FORM

An investigation into decoding speech and language processes from commercial EEG devices

Name of Researcher: Vladislav Shishkin
Contact details of Researcher: vs556@bath.ac.uk

Name of Supervisors: Dr James Laird Scott Wellington
Contact details of Supervisors: jl317@bath.ac.uk sdlw20@bath.ac.uk

Please initial box if you agree with the statement

1. I have been provided with information explaining what participation in this project involves.
2. I have had an opportunity to ask questions and discuss this project.
3. I have received satisfactory answers to all questions I have asked.
4. I have received enough information about the project to make a decision about my participation.
5. I understand that I am free to withdraw my consent to participate in the project at any time without having to give a reason for withdrawing.
6. I understand that I am free to withdraw my data within two weeks of my participation.
7. I understand the nature and purpose of the procedures involved in this project. These have been communicated to me on the information sheet accompanying this form.
8. I understand that the University of Bath may use the data collected for this study in future research project(s) but that the conditions on this form under which I have provided the data will still apply.
9. I understand the data I provide will be processed in accordance with Data Protection Law (GDPR), and that on completion of the project my name or other identifying information will not be disclosed in any presentation or publication of the research.
10. I agree to the University of Bath keeping and processing the data that I provide during the course of this study and my consent is conditional upon the University complying with its duties and obligations under the Data Protection Act.

11. I hereby fully and freely consent to my participation in this project.

Participant's signature: _____ Date: _____

Participant name in BLOCK Letters: _____

Researcher's signature: ____ Vladislav Shishkin____ Date: ____09.05.2022____

Researcher name in BLOCK Letters: _____ VLADISLAV SHISHKIN_____

If you have any concerns or complaints related to your participation in this project please direct them to the researchers named above.

Appendix C

Dataset File Tree

The file tree in Figure C.1 depicts the dataset folder structure. Full dataset can be accessed from this project's GitHub repository [64].

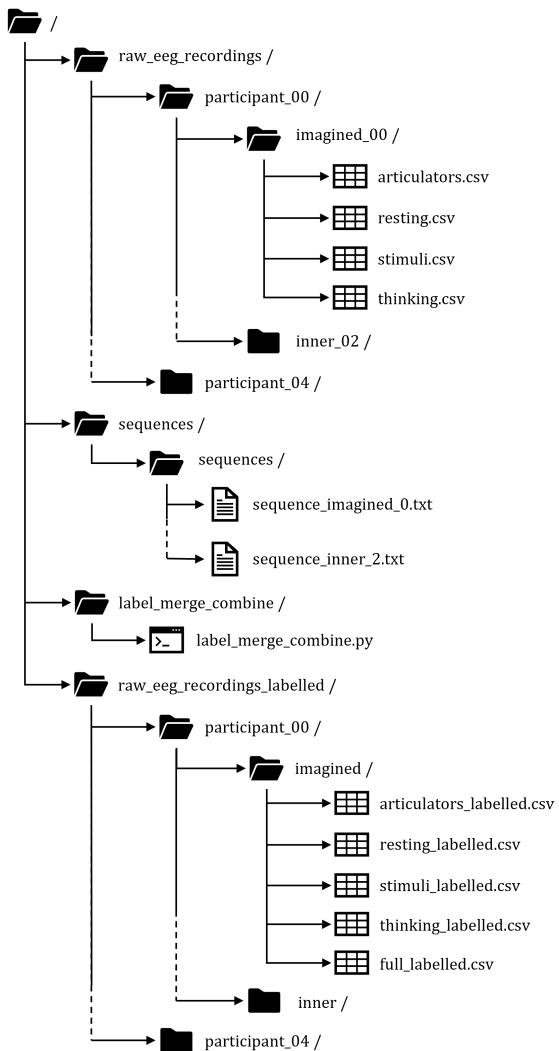


Figure C.1: A portion of the study's dataset [64] file hierarchy, depicting how the raw and the labelled EEG recording files are structured with the relevant sequence and Python files.

Appendix D

EEGNet Architecture

Table D.1 demonstrates the EEGNet architecture.

Table D.1: EEGNet architecture, where F_1 is the number of temporal filters, D is the number of spatial filters, F_2 is the number of pointwise filters, S is the EEG device sampling rate, C is the number of EEG channels and p is the probability of dropout.

Block	Layer	Number of input channels	Number of output channels	Kernel size	Options
1	Conv2D	1	F_1	$(S/2, 1)$	Padding = same
	BatchNorm	F_1	F_1		
	DepthConv2D	F_1	$D \times F_1$	$(1, C)$	Groups = F_1
	BatchNorm	$D \times F_1$	$D \times F_1$		
	ELU				
	AvgPool2D			$(4, 1)$	
	Dropout				p
2	SeparableConv2D	$D \times F_1$	F_2	$(16, 1)$	Groups = F_2
	BatchNorm	F_2	F_2		
	ELU				
	AvgPool2D			$(8, 1)$	
	Dropout				p
	Flatten				
3	Fully connected				
	Softmax				

Appendix E

EEGNet Hyperparameters

Table E.1 demonstrates the EEGNet hyperparameters used for each dataset.

Table E.1: EEGNet hyperparameter configurations.

Dataset	Datatype	Batch size	Loss function	Dropout rate	Epochs	Learning rate	Optimiser	Weight decay
P00	Raw	64	NLL	0.0	30	0.001	Adam	0.0
P01-04	Raw	128	NLL	0.0	50	0.00001	Adam	0.1
	Prep	32	NLL	0.0	40	0.0001	Adam	10
	LF	64	NLL	0.0	10	0.01	Adam	0.0
	NLF	64	CE	0.25	50	0.001	Adam	0.0
	MFCC	128	CE	0.0	50	0.0001	Adam	0.0
Binary	Raw	16	NLL	0.0	40	0.1	SGD	1.0
	Prep	64	NLL	0.75	30	0.1	SGD	0.0
	LF	64	NLL	0.0	20	0.1	Adam	0.1
	NLF	16	NLL	0.5	50	0.01	Adam	0.0
	MFCC	32	NLL	0.75	30	0.1	SGD	0.0
FEIS	Raw	128	NLL	0.0	10	0.001	SGD	0.0
	Prep	16	NLL	0.0	50	0.01	SGD	0.0
	LF	128	CE	0.25	50	0.001	Adam	0.0
	NLF	32	CE	0.25	10	0.001	Adam	0.0
	MFCC	64	NLL	0.0	10	0.0001	SGD	0.0

Appendix F

Full Results

Tables F.1, F.2 and F.3 show results for all conventional classifiers across the P02, P03 and P04 datasets respectively.

F.1 Testing Accuracy results

Table F.1: P02 testing accuracy classification results on conventional algorithms, where cyan and red colours highlight the best and the worst mean classifier accuracy respectively, e stands for the number of estimators in AB, lr stands for the learning rate of AB and k stands for the number of nearest neighbours in KNN.

Method	Mean imagined speech percentage accuracy (std. dev)					Mean inner speech percentage accuracy (std. dev)				
	Raw	Prep	LF	NLF	MFCC	Raw	Prep	LF	NLF	MFCC
AB ($e=50$, $lr=0.1$)	12.7 (0.44)	10.1 (0.19)	8.59 (0.70)	6.83 (1.32)	5.02 (1.52)	12.7 (0.10)	9.32 (0.39)	6.90 (0.44)	6.02 (0.56)	5.71 (0.54)
AB ($e=50$, $lr=0.5$)	14.6 (0.23)	11.2 (0.23)	8.84 (0.82)	7.96 (0.33)	6.27 (0.39)	14.4 (0.29)	10.4 (0.16)	8.65 (1.01)	7.96 (0.57)	7.40 (1.45)
AB ($e=50$, $lr=1.0$)	14.2 (0.14)	11.0 (0.21)	7.71 (0.95)	6.46 (1.70)	8.27 (1.68)	14.2 (0.12)	10.6 (0.05)	7.84 (0.64)	8.02 (1.13)	7.08 (0.62)
AB ($e=50$, $lr=1.5$)	12.9 (0.40)	10.3 (0.32)	9.84 (0.85)	7.33 (0.55)	6.71 (0.85)	13.0 (0.38)	10.0 (0.21)	7.65 (0.63)	7.71 (1.51)	6.71 (0.78)
AB ($e=50$, $lr=2.0$)	7.54 (0.47)	7.19 (0.53)	7.90 (0.86)	7.21 (0.36)	7.34 (0.01)	7.98 (0.20)	7.31 (0.33)	7.46 (1.62)	5.08 (0.82)	8.21 (0.85)
AB ($e=100$, $lr=1.0$)	16.0 (0.23)	11.9 (0.19)	8.40 (0.65)	7.71 (0.40)	6.52 (0.64)	15.7 (0.38)	11.5 (0.16)	7.27 (0.48)	6.46 (1.03)	7.02 (0.54)
AB ($e=200$, $lr=1.0$)	16.0 (0.31)	12.1 (0.20)	8.65 (1.08)	9.09 (1.17)	6.40 (1.08)	16.5 (0.13)	12.0 (0.22)	6.65 (0.64)	7.40 (0.61)	6.14 (0.35)
AB ($e=400$, $lr=1.0$)	15.8 (0.11)	12.0 (0.17)	7.21 (1.14)	6.84 (1.29)	6.27 (0.65)	15.7 (0.23)	12.0 (0.13)	6.96 (1.07)	5.89 (0.62)	5.45 (0.67)
Gaussian NB	9.80 (0.19)	8.79 (0.07)	8.09 (0.67)	7.65 (0.61)	9.34 (1.69)	9.50 (0.11)	8.08 (0.10)	7.59 (0.69)	7.02 (1.29)	6.39 (0.92)
KNN ($k=1$)	93.9 (0.08)	71.3 (0.14)	7.84 (0.64)	8.09 (0.56)	7.65 (1.16)	92.9 (0.05)	67.9 (0.11)	7.77 (0.84)	7.02 (0.55)	6.52 (0.45)
KNN ($k=5$)	85.1 (0.11)	54.1 (0.11)	8.53 (1.58)	7.90 (0.85)	7.08 (0.32)	83.3 (0.09)	51.6 (0.10)	6.33 (0.54)	6.40 (0.62)	6.08 (0.77)
KNN ($k=13$)	78.1 (0.14)	42.9 (0.13)	7.96 (1.16)	8.53 (0.47)	6.71 (0.44)	76.3 (0.10)	40.8 (0.13)	7.96 (0.80)	7.65 (0.72)	7.02 (1.03)
KNN ($k=25$)	72.8 (0.04)	36.6 (0.08)	7.21 (1.31)	7.90 (1.19)	7.08 (1.63)	71.0 (0.02)	34.8 (0.10)	6.46 (0.94)	7.84 (0.90)	5.96 (0.44)
KNN ($k=53$)	66.1 (0.19)	30.7 (0.07)	7.46 (1.11)	6.77 (0.46)	7.59 (0.98)	64.9 (0.06)	29.6 (0.13)	7.27 (0.63)	5.52 (0.79)	5.83 (0.94)
KNN ($k=89$)	61.1 (0.14)	27.5 (0.19)	7.46 (1.27)	6.14 (0.63)	7.27 (1.03)	60.2 (0.05)	26.2 (0.07)	7.21 (0.75)	6.71 (0.47)	6.21 (0.80)
KNN ($k=139$)	56.3 (0.33)	24.9 (0.16)	6.46 (1.31)	7.02 (1.08)	7.15 (0.26)	55.5 (0.15)	23.6 (0.15)	6.96 (1.11)	6.14 (0.09)	5.58 (0.39)
LDA	11.5 (0.15)	9.66 (0.18)	10.6 (0.95)	8.15 (1.42)	8.02 (0.77)	11.5 (0.15)	9.40 (0.11)	8.40 (1.03)	8.09 (1.39)	6.77 (1.17)
SVM (RBF)	n/a	n/a	5.58 (0.54)	5.02 (0.79)	4.89 (0.15)	n/a	n/a	4.95 (0.18)	4.83 (0.23)	4.83 (0.76)
SVM (polynomial)	n/a	n/a	5.27 (0.27)	4.51 (0.47)	4.89 (0.15)	n/a	n/a	5.02 (0.78)	4.51 (0.55)	4.76 (0.32)
SVM (sigmoid)	n/a	n/a	5.08 (0.41)	4.20 (0.68)	4.89 (0.16)	n/a	n/a	5.08 (0.26)	4.20 (0.58)	4.89 (0.26)

Table F.2: P03 testing accuracy classification results on conventional algorithms, where cyan and red colours highlight the best and the worst mean classifier accuracy respectively, e stands for the number of estimators in AB, lr stands for the learning rate of AB and k stands for the number of nearest neighbours in KNN.

Method	Mean imagined speech percentage accuracy (std. dev)					Mean inner speech percentage accuracy (std. dev)				
	Raw	Prep	LF	NLF	MFCC	Raw	Prep	LF	NLF	MFCC
AB ($e=50$, $lr=0.1$)	14.5 (0.03)	13.2 (0.06)	10.4 (0.33)	7.71 (1.08)	6.58 (0.95)	13.5 (0.45)	11.9 (0.20)	8.40 (0.85)	7.40 (0.53)	6.40 (0.94)
AB ($e=50$, $lr=0.5$)	15.8 (0.26)	15.4 (0.40)	10.7 (0.40)	8.53 (1.02)	6.27 (0.79)	15.0 (0.34)	13.7 (0.13)	9.66 (0.18)	8.40 (1.31)	5.33 (0.73)
AB ($e=50$, $lr=1.0$)	15.6 (0.60)	15.1 (0.12)	8.65 (0.31)	7.84 (0.89)	6.71 (0.99)	14.3 (0.22)	13.2 (0.12)	8.53 (1.80)	7.77 (0.85)	6.40 (0.80)
AB ($e=50$, $lr=1.5$)	13.7 (0.55)	12.9 (0.39)	8.59 (1.17)	8.65 (1.33)	7.84 (1.02)	13.3 (0.26)	11.5 (0.13)	8.34 (0.88)	7.59 (1.49)	6.33 (0.47)
AB ($e=50$, $lr=2.0$)	9.69 (0.41)	8.14 (0.77)	7.52 (1.00)	7.02 (0.79)	6.02 (0.27)	9.50 (2.17)	7.28 (0.61)	8.78 (1.08)	7.59 (1.10)	6.33 (0.32)
AB ($e=100$, $lr=1.0$)	16.3 (0.32)	15.5 (0.18)	10.7 (1.61)	9.40 (0.56)	6.27 (0.84)	15.7 (0.01)	14.1 (0.09)	8.34 (1.11)	7.34 (0.80)	6.71 (0.32)
AB ($e=200$, $lr=1.0$)	16.7 (0.24)	15.5 (0.19)	9.28 (1.13)	8.53 (0.85)	6.65 (1.10)	16.4 (0.33)	14.0 (0.08)	7.90 (0.82)	8.46 (0.40)	6.52 (0.58)
AB ($e=400$, $lr=1.0$)	16.2 (0.29)	15.1 (0.07)	9.28 (0.79)	8.97 (0.62)	6.52 (0.33)	15.6 (0.23)	13.9 (0.14)	7.59 (1.40)	6.58 (0.86)	6.58 (1.60)
Gaussian NB	10.3 (0.13)	9.29 (0.05)	7.65 (1.17)	6.90 (1.14)	7.52 (1.67)	10.3 (0.15)	9.37 (0.17)	7.21 (1.24)	7.71 (0.65)	7.34 (0.94)
KNN ($k=1$)	97.0 (0.05)	84.5 (0.17)	7.08 (1.02)	7.71 (0.70)	7.59 (0.70)	99.7 (0.01)	92.1 (0.07)	8.34 (1.51)	7.77 (0.32)	5.96 (0.24)
KNN ($k=5$)	94.1 (0.04)	74.0 (0.07)	7.21 (0.79)	8.28 (0.41)	8.03 (0.73)	98.8 (0.05)	82.0 (0.19)	7.65 (0.33)	6.46 (0.98)	7.77 (1.14)
KNN ($k=13$)	90.3 (0.14)	66.3 (0.13)	8.34 (1.01)	7.46 (0.33)	7.46 (0.84)	96.9 (0.07)	73.6 (0.11)	7.27 (0.58)	6.46 (0.36)	7.52 (1.56)
KNN ($k=25$)	86.4 (0.11)	60.3 (0.06)	6.71 (0.76)	6.65 (0.17)	7.15 (1.01)	94.6 (0.04)	67.1 (0.04)	7.34 (1.16)	7.40 (1.59)	7.27 (1.03)
KNN ($k=53$)	80.2 (0.13)	52.8 (0.23)	8.09 (2.18)	7.02 (1.48)	7.02 (0.87)	89.8 (0.13)	58.6 (0.29)	8.15 (0.35)	7.52 (0.93)	5.83 (0.15)
KNN ($k=89$)	74.6 (0.16)	47.3 (0.29)	8.28 (1.63)	7.46 (0.75)	6.71 (0.64)	84.7 (0.08)	51.9 (0.17)	6.40 (0.54)	8.28 (1.71)	5.89 (0.73)
KNN ($k=139$)	69.0 (0.17)	42.7 (0.10)	7.15 (1.48)	7.59 (0.38)	7.21 (0.45)	78.6 (0.13)	46.2 (0.14)	6.33 (0.33)	6.27 (0.32)	6.77 (1.33)
LDA	13.0 (0.09)	11.3 (0.16)	12.2 (0.15)	10.5 (1.23)	7.59 (0.23)	13.2 (0.10)	10.7 (0.12)	11.0 (0.73)	10.0 (2.02)	7.71 (0.40)
SVM (RBF)	n/a	n/a	5.20 (1.08)	4.77 (0.58)	4.70 (0.15)	n/a	n/a	5.27 (0.27)	4.45 (0.87)	4.70 (0.16)
SVM (polynomial)	n/a	n/a	5.08 (0.27)	5.20 (0.42)	4.76 (0.17)	n/a	n/a	5.33 (0.32)	4.20 (1.17)	4.20 (0.58)
SVM (sigmoid)	n/a	n/a	4.64 (0.84)	4.45 (1.06)	4.45 (0.78)	n/a	n/a	4.51 (0.70)	4.20 (0.61)	4.58 (0.35)

Table F.3: P04 testing accuracy results on conventional algorithms, where cyan and red colours highlight the best and the worst mean classifier accuracy respectively, e stands for the number of estimators in AB, lr stands for the learning rate of AB and k stands for the number of nearest neighbours in KNN.

Method	Mean imagined speech percentage accuracy (std. dev)					Mean inner speech percentage accuracy (std. dev)				
	Raw	Prep	LF	NLF	MFCC	Raw	Prep	LF	NLF	MFCC
AB ($e=50$, $lr=0.1$)	13.5 (0.18)	11.3 (0.03)	8.21 (0.44)	6.90 (1.25)	7.08 (0.84)	12.1 (0.23)	10.1 (0.18)	6.90 (0.89)	7.21 (0.69)	5.39 (0.62)
AB ($e=50$, $lr=0.5$)	14.9 (0.10)	12.2 (0.12)	7.27 (1.15)	7.96 (0.18)	5.39 (0.47)	13.7 (0.34)	11.7 (0.09)	7.71 (0.95)	7.15 (0.94)	5.96 (1.77)
AB ($e=50$, $lr=1.0$)	14.4 (0.22)	11.7 (0.13)	8.02 (0.88)	6.71 (0.62)	6.39 (0.31)	13.6 (0.20)	11.4 (0.09)	7.96 (0.99)	7.46 (0.54)	5.14 (0.87)
AB ($e=50$, $lr=1.5$)	12.5 (0.07)	11.0 (0.30)	7.71 (1.86)	6.27 (0.58)	5.64 (1.08)	12.8 (0.28)	10.9 (0.09)	7.08 (0.58)	7.08 (1.42)	6.21 (0.61)
AB ($e=50$, $lr=2.0$)	7.43 (0.01)	7.85 (0.48)	8.59 (2.18)	6.33 (0.09)	5.71 (1.20)	7.30 (0.35)	7.01 (0.12)	7.15 (0.92)	7.21 (0.69)	6.14 (1.07)
AB ($e=100$, $lr=1.0$)	15.8 (0.17)	12.6 (0.25)	7.71 (1.88)	7.40 (1.01)	5.89 (1.78)	14.8 (0.35)	12.5 (0.22)	7.02 (0.65)	6.65 (0.69)	6.77 (0.82)
AB ($e=200$, $lr=1.0$)	15.9 (0.21)	13.2 (0.17)	8.46 (0.82)	6.90 (0.78)	7.02 (0.39)	15.1 (0.22)	12.3 (0.18)	7.46 (0.93)	6.90 (1.19)	6.08 (1.46)
AB ($e=400$, $lr=1.0$)	15.2 (0.20)	13.1 (0.09)	7.84 (0.94)	6.83 (1.00)	5.52 (0.39)	14.4 (0.08)	12.3 (0.14)	6.65 (0.09)	6.33 (1.56)	6.14 (0.47)
Gaussian NB	9.74 (0.09)	8.76 (0.11)	7.34 (0.41)	7.46 (1.34)	7.46 (1.17)	9.33 (0.11)	8.87 (0.04)	7.84 (1.15)	7.34 (1.62)	5.83 (0.77)
KNN ($k=1$)	98.2 (0.02)	86.6 (0.08)	7.71 (0.41)	7.34 (0.41)	4.83 (1.55)	97.2 (0.03)	78.2 (0.13)	8.28 (0.41)	7.27 (0.76)	7.34 (0.41)
KNN ($k=5$)	94.6 (0.08)	73.5 (0.13)	7.40 (0.35)	7.46 (0.32)	6.27 (0.91)	91.5 (0.16)	61.9 (0.06)	8.09 (0.70)	6.58 (0.16)	6.96 (1.35)
KNN ($k=13$)	90.6 (0.13)	63.6 (0.13)	6.52 (1.29)	8.09 (1.38)	5.45 (1.00)	86.5 (0.16)	51.6 (0.06)	6.33 (0.89)	6.96 (0.16)	5.39 (0.94)
KNN ($k=25$)	86.6 (0.21)	56.9 (0.17)	6.27 (1.03)	6.39 (0.61)	6.96 (1.19)	82.2 (0.16)	45.4 (0.07)	6.71 (0.33)	6.58 (0.40)	6.14 (0.44)
KNN ($k=53$)	80.1 (0.15)	48.8 (0.16)	7.21 (1.52)	7.34 (0.46)	6.71 (1.39)	75.8 (0.06)	39.0 (0.22)	6.46 (0.62)	5.96 (0.47)	5.89 (1.16)
KNN ($k=89$)	74.1 (0.23)	42.9 (0.06)	7.02 (0.85)	5.89 (0.24)	7.65 (1.64)	70.2 (0.23)	34.7 (0.04)	5.96 (0.63)	6.14 (0.85)	5.20 (0.39)
KNN ($k=139$)	67.7 (0.03)	38.0 (0.21)	6.77 (1.52)	6.77 (0.67)	6.90 (0.98)	64.6 (0.15)	31.3 (0.19)	6.08 (0.77)	5.33 (0.18)	5.20 (0.50)
LDA	12.4 (0.10)	10.7 (0.10)	10.6 (1.16)	8.78 (1.44)	6.21 (1.20)	11.1 (0.18)	10.6 (0.10)	8.96 (0.75)	8.28 (1.23)	6.77 (0.15)
SVM (RBF)	n/a	n/a	5.83 (0.01)	4.76 (0.32)	4.70 (0.31)	n/a	n/a	4.58 (0.45)	4.89 (1.10)	5.20 (0.62)
SVM (polynomial)	n/a	n/a	4.58 (0.47)	4.70 (0.82)	4.76 (0.38)	n/a	n/a	5.02 (0.17)	4.26 (0.47)	5.08 (0.27)
SVM (sigmoid)	n/a	n/a	4.89 (0.30)	4.89 (0.25)	4.58 (0.38)	n/a	n/a	5.45 (0.41)	4.58 (0.78)	4.70 (0.31)

F.2 Training Accuracy results

Tables F.4, F.5, F.6, F.7, F.8, F.9 and F.10 show results for all conventional classifiers across the P00, P01, P02, P03, P04, binary and FEIS datasets respectively.

Table F.4: P00 training accuracy classification results on conventional algorithms, where e stands for the number of estimators in AB, lr stands for the learning rate of AB and k stands for the number of nearest neighbours in KNN.

Method	Mean imagined speech percentage accuracy (std. dev)	Mean inner speech percentage accuracy (std. dev)
	Raw	Raw
AB ($e=50$, $lr=0.1$)	7.19 (0.07)	7.14 (0.05)
AB ($e=50$, $lr=0.5$)	7.52 (0.01)	7.38 (0.06)
AB ($e=50$, $lr=1.0$)	7.40 (0.06)	7.32 (0.14)
AB ($e=50$, $lr=1.5$)	7.41 (0.05)	7.28 (0.04)
AB ($e=50$, $lr=2.0$)	6.34 (0.12)	6.34 (0.09)
AB ($e=100$, $lr=1.0$)	7.79 (0.09)	7.70 (0.15)
AB ($e=200$, $lr=1.0$)	8.07 (0.12)	8.05 (0.06)
AB ($e=400$, $lr=1.0$)	8.67 (0.10)	8.56 (0.08)
Gaussian NB	7.01 (0.06)	6.99 (0.06)
KNN ($k=1$)	100 (0.00)	100 (0.00)
KNN ($k=5$)	30.7 (0.11)	30.7 (0.19)
KNN ($k=13$)	20.9 (0.03)	20.6 (0.20)
KNN ($k=25$)	16.5 (0.12)	16.4 (0.18)
KNN ($k=53$)	13.1 (0.08)	12.8 (0.14)
KNN ($k=89$)	11.5 (0.08)	11.4 (0.10)
KNN ($k=139$)	10.5 (0.14)	10.3 (0.09)
LDA	6.89 (0.04)	6.85 (0.08)

Table F.5: P01 training accuracy classification results on conventional algorithms, where cyan and red colours highlight the best and the worst mean classifier accuracy respectively, e stands for the number of estimators in AB, lr stands for the learning rate of AB and k stands for the number of nearest neighbours in KNN.

Method	Mean imagined speech percentage accuracy (std. dev)					Mean inner speech percentage accuracy (std. dev)				
	Raw	Prep	LF	NLF	MFCC	Raw	Prep	LF	NLF	MFCC
AB ($e=50$, $lr=0.1$)	14.4 (0.12)	10.6 (0.03)	19.6 (0.81)	19.9 (1.50)	20.3 (0.93)	13.6 (0.10)	10.3 (0.05)	22.0 (1.00)	21.4 (1.53)	20.4 (1.38)
AB ($e=50$, $lr=0.5$)	16.6 (0.16)	12.6 (0.17)	21.1 (1.19)	21.1 (1.26)	17.0 (1.44)	15.8 (0.18)	11.4 (0.16)	22.1 (0.69)	21.6 (1.53)	19.7 (1.59)
AB ($e=50$, $lr=1.0$)	15.9 (0.24)	12.4 (0.09)	18.9 (0.81)	19.3 (1.12)	15.2 (2.69)	15.8 (0.48)	11.8 (0.08)	18.7 (1.30)	18.7 (1.82)	15.4 (1.95)
AB ($e=50$, $lr=1.5$)	13.8 (0.19)	11.6 (0.13)	17.1 (1.72)	14.0 (0.28)	12.9 (2.90)	13.4 (0.48)	11.2 (0.07)	15.4 (1.94)	16.4 (1.12)	16.0 (1.21)
AB ($e=50$, $lr=2.0$)	7.79 (1.44)	6.75 (0.08)	11.6 (0.83)	11.4 (1.19)	11.6 (2.21)	7.99 (0.44)	7.49 (0.04)	13.0 (1.64)	11.3 (2.47)	12.9 (1.43)
AB ($e=100$, $lr=1.0$)	17.5 (0.17)	13.4 (0.04)	18.8 (0.75)	17.8 (3.29)	15.5 (1.85)	16.8 (0.34)	12.5 (0.02)	19.6 (1.07)	18.5 (2.03)	14.6 (0.66)
AB ($e=200$, $lr=1.0$)	17.7 (0.29)	14.2 (0.30)	21.5 (3.27)	16.2 (2.24)	14.6 (0.49)	17.1 (0.20)	12.8 (0.07)	20.1 (5.17)	16.8 (0.93)	14.4 (1.41)
AB ($e=400$, $lr=1.0$)	17.1 (0.18)	14.2 (0.09)	30.2 (2.34)	28.3 (9.48)	19.2 (5.66)	15.9 (0.09)	12.8 (0.09)	22.1 (9.31)	18.4 (6.15)	16.6 (1.47)
Gaussian NB	9.63 (0.05)	9.14 (0.03)	7.93 (0.29)	7.76 (0.07)	30.5 (0.63)	10.9 (0.08)	8.36 (0.05)	8.53 (0.12)	7.90 (0.39)	33.6 (1.69)
KNN ($k=1$)	100 (0.00)	100 (0.00)	100 (0.00)	100 (0.00)	100 (0.00)	100 (0.00)	100 (0.00)	100 (0.00)	100 (0.00)	100 (0.00)
KNN ($k=5$)	99.3 (0.02)	84.6 (0.07)	32.8 (0.26)	33.1 (0.61)	32.9 (1.57)	97.2 (0.02)	79.2 (0.06)	32.9 (1.71)	31.8 (0.30)	32.2 (1.63)
KNN ($k=13$)	97.7 (0.02)	72.9 (0.05)	22.2 (1.34)	23.1 (0.28)	22.7 (0.93)	92.5 (0.03)	63.6 (0.09)	23.4 (0.91)	24.1 (0.26)	22.0 (1.04)
KNN ($k=25$)	95.6 (0.03)	64.4 (0.06)	17.6 (0.43)	17.6 (0.66)	17.4 (0.90)	88.0 (0.04)	52.4 (0.04)	18.2 (0.68)	18.7 (0.56)	17.0 (1.61)
KNN ($k=53$)	91.7 (0.05)	55.0 (0.03)	14.4 (0.24)	14.5 (0.91)	15.3 (0.63)	81.7 (0.06)	42.0 (0.03)	14.4 (0.28)	15.5 (1.11)	14.6 (1.12)
KNN ($k=89$)	87.3 (0.05)	48.7 (0.06)	12.6 (0.33)	12.0 (0.61)	12.3 (0.85)	76.6 (0.04)	36.2 (0.06)	12.5 (0.50)	11.9 (0.78)	12.9 (0.39)
KNN ($k=139$)	82.0 (0.05)	43.4 (0.15)	11.4 (0.22)	10.7 (0.43)	10.8 (0.87)	71.3 (0.04)	32.0 (0.10)	11.7 (1.25)	11.8 (0.57)	11.0 (1.22)
LDA	12.3 (0.02)	11.2 (0.03)	36.8 (0.22)	32.9 (1.77)	50.8 (1.29)	13.6 (0.06)	10.6 (0.04)	36.1 (0.31)	32.7 (0.50)	50.2 (0.18)
SVM (RBF)	n/a	n/a	8.43 (0.43)	8.59 (0.25)	7.27 (0.18)	n/a	n/a	7.55 (0.22)	7.84 (0.18)	7.02 (0.09)
SVM (polynomial)	n/a	n/a	8.46 (0.40)	7.96 (0.61)	7.46 (0.47)	n/a	n/a	7.81 (0.21)	7.41 (0.14)	7.12 (0.16)
SVM (sigmoid)	n/a	n/a	7.37 (0.12)	7.27 (0.18)	7.40 (0.29)	n/a	n/a	7.18 (0.12)	6.91 (0.12)	7.18 (0.31)

Table F.6: P02 training accuracy classification results on conventional algorithms, where cyan and red colours highlight the best and the worst mean classifier accuracy respectively, e stands for the number of estimators in AB, lr stands for the learning rate of AB and k stands for the number of nearest neighbours in KNN.

Method	Mean imagined speech percentage accuracy (std. dev)					Mean inner speech percentage accuracy (std. dev)				
	Raw	Prep	LF	NLF	MFCC	Raw	Prep	LF	NLF	MFCC
AB ($e=50$, $lr=0.1$)	12.8 (0.46)	10.3 (0.05)	20.5 (0.59)	20.2 (1.09)	19.5 (1.58)	12.8 (0.05)	9.52 (0.28)	19.0 (0.81)	20.3 (0.76)	20.6 (0.95)
AB ($e=50$, $lr=0.5$)	14.7 (0.13)	11.4 (0.16)	21.0 (1.47)	20.7 (0.61)	19.6 (1.43)	14.6 (0.23)	10.6 (0.15)	21.7 (1.27)	21.6 (1.57)	20.2 (1.16)
AB ($e=50$, $lr=1.0$)	14.5 (0.09)	11.2 (0.18)	17.9 (1.46)	16.8 (1.31)	15.9 (0.49)	14.4 (0.17)	10.8 (0.12)	17.0 (1.57)	17.0 (1.85)	16.6 (0.80)
AB ($e=50$, $lr=1.5$)	13.1 (0.42)	10.5 (0.18)	16.5 (1.19)	14.7 (1.32)	14.6 (1.81)	13.2 (0.22)	10.3 (0.15)	16.9 (0.23)	15.0 (1.79)	14.4 (1.05)
AB ($e=50$, $lr=2.0$)	7.55 (0.51)	7.21 (0.58)	12.4 (1.43)	11.8 (1.23)	13.0 (2.36)	8.09 (0.30)	7.37 (0.29)	12.7 (1.96)	11.4 (1.62)	12.3 (2.55)
AB ($e=100$, $lr=1.0$)	16.2 (0.19)	12.2 (0.14)	16.9 (3.66)	15.8 (1.21)	14.3 (1.62)	16.1 (0.40)	11.9 (0.02)	16.6 (1.16)	16.4 (1.24)	13.9 (1.61)
AB ($e=200$, $lr=1.0$)	16.4 (0.34)	12.6 (0.09)	14.8 (2.25)	14.9 (1.27)	14.4 (0.46)	16.9 (0.03)	12.6 (0.17)	16.3 (2.63)	17.5 (1.66)	16.7 (0.70)
AB ($e=400$, $lr=1.0$)	16.2 (0.19)	12.6 (0.12)	20.4 (6.24)	14.6 (0.98)	14.3 (1.19)	16.1 (0.14)	12.6 (0.12)	17.1 (1.65)	18.8 (6.99)	19.5 (5.53)
Gaussian NB	9.80 (0.05)	8.81 (0.01)	9.37 (0.62)	8.81 (0.45)	37.1 (0.79)	9.52 (0.06)	8.12 (0.07)	8.46 (0.27)	8.46 (0.33)	37.4 (0.78)
KNN ($k=1$)	100 (0.00)	100 (0.00)	100 (0.00)	100 (0.00)	100 (0.00)	100 (0.00)	100 (0.00)	100 (0.00)	100 (0.00)	100 (0.00)
KNN ($k=5$)	93.9 (0.02)	77.5 (0.04)	32.4 (1.16)	33.5 (0.62)	33.6 (1.58)	93.1 (0.12)	75.9 (0.11)	33.1 (0.94)	33.2 (0.90)	31.8 (0.04)
KNN ($k=13$)	85.3 (0.04)	60.7 (0.04)	23.5 (0.11)	23.1 (0.74)	22.3 (0.69)	84.0 (0.07)	59.2 (0.11)	23.1 (0.05)	22.9 (0.95)	21.0 (0.43)
KNN ($k=25$)	78.1 (0.04)	48.8 (0.04)	17.7 (0.24)	17.7 (0.90)	17.3 (0.58)	76.8 (0.05)	47.4 (0.01)	16.4 (0.77)	16.6 (1.06)	15.9 (1.39)
KNN ($k=53$)	69.8 (0.06)	38.1 (0.10)	14.1 (0.51)	14.1 (0.40)	15.1 (0.50)	68.7 (0.09)	36.9 (0.09)	13.3 (0.80)	13.9 (0.89)	13.5 (0.78)
KNN ($k=89$)	63.9 (0.18)	32.6 (0.06)	11.9 (0.62)	11.9 (0.59)	12.1 (0.32)	62.9 (0.11)	31.5 (0.03)	11.8 (1.13)	12.2 (0.35)	12.4 (0.84)
KNN ($k=139$)	58.5 (0.13)	28.7 (0.06)	11.2 (0.47)	10.5 (0.67)	11.6 (0.21)	57.8 (0.07)	27.6 (0.13)	11.2 (0.93)	11.0 (0.87)	9.84 (1.08)
LDA	11.6 (0.08)	9.68 (0.05)	35.7 (0.83)	31.3 (1.18)	53.3 (0.31)	11.6 (0.04)	9.47 (0.03)	34.5 (0.40)	31.7 (0.62)	51.1 (0.96)
SVM (RBF)	n/a	n/a	8.46 (0.28)	8.53 (0.04)	6.96 (0.07)	n/a	n/a	8.31 (0.29)	8.50 (0.25)	7.56 (1.18)
SVM (polynomial)	n/a	n/a	8.24 (0.36)	7.74 (0.21)	7.74 (1.18)	n/a	n/a	8.31 (0.22)	7.65 (0.08)	7.02 (0.16)
SVM (sigmoid)	n/a	n/a	7.43 (0.00)	7.15 (0.19)	7.43 (0.60)	n/a	n/a	7.37 (0.20)	7.27 (0.06)	6.96 (0.13)

Table F.7: P03 training accuracy classification results on conventional algorithms, where cyan and red colours highlight the best and the worst mean classifier accuracy respectively, e stands for the number of estimators in AB, lr stands for the learning rate of AB and k stands for the number of nearest neighbours in KNN.

Method	Mean imagined speech percentage accuracy (std. dev)					Mean inner speech percentage accuracy (std. dev)				
	Raw	Prep	LF	NLF	MFCC	Raw	Prep	LF	NLF	MFCC
AB ($e=50$, $lr=0.1$)	14.6 (0.05)	13.4 (0.02)	21.9 (1.09)	20.2 (1.61)	21.4 (1.26)	13.7 (0.35)	12.1 (0.06)	22.3 (1.72)	22.1 (1.90)	19.6 (0.56)
AB ($e=50$, $lr=0.5$)	16.0 (0.34)	15.6 (0.24)	23.0 (0.57)	19.3 (1.57)	19.3 (1.54)	15.1 (0.28)	13.9 (0.06)	23.5 (1.81)	21.6 (1.77)	20.7 (2.79)
AB ($e=50$, $lr=1.0$)	15.7 (0.55)	15.4 (0.11)	17.9 (1.80)	16.6 (1.76)	14.9 (2.81)	14.5 (0.18)	13.5 (0.23)	17.6 (1.17)	15.1 (2.45)	15.9 (1.02)
AB ($e=50$, $lr=1.5$)	13.9 (0.71)	13.2 (0.20)	15.3 (1.96)	14.5 (1.32)	16.7 (1.09)	13.4 (0.17)	11.8 (0.16)	16.0 (0.89)	16.2 (0.77)	13.7 (0.81)
AB ($e=50$, $lr=2.0$)	9.69 (0.34)	8.16 (0.76)	12.8 (1.58)	11.2 (0.89)	10.4 (1.38)	9.56 (2.16)	7.37 (0.72)	10.8 (0.66)	13.0 (1.41)	13.4 (1.61)
AB ($e=100$, $lr=1.0$)	16.6 (0.39)	15.8 (0.23)	17.4 (0.76)	13.6 (0.14)	14.6 (1.46)	16.0 (0.11)	14.4 (0.04)	17.6 (1.89)	16.0 (2.22)	14.8 (0.73)
AB ($e=200$, $lr=1.0$)	16.9 (0.14)	15.8 (0.15)	17.3 (0.77)	14.1 (0.59)	16.1 (3.24)	16.7 (0.06)	14.5 (0.13)	19.0 (3.03)	14.5 (0.51)	16.1 (1.85)
AB ($e=400$, $lr=1.0$)	16.5 (0.42)	15.5 (0.16)	17.8 (1.16)	15.1 (1.04)	14.9 (0.29)	15.8 (0.24)	14.3 (0.08)	25.2 (7.85)	15.8 (1.41)	14.5 (2.78)
Gaussian NB	10.4 (0.07)	9.30 (0.05)	8.31 (0.23)	8.13 (0.35)	36.2 (0.49)	10.3 (0.04)	9.39 (0.04)	9.06 (0.43)	8.70 (0.16)	37.6 (0.37)
KNN ($k=1$)	100 (0.00)	100 (0.00)	100 (0.00)	100 (0.00)	100 (0.00)	100 (0.00)	100 (0.00)	100 (0.00)	100 (0.00)	100 (0.00)
KNN ($k=5$)	96.8 (0.01)	87.2 (0.07)	32.5 (1.33)	32.6 (0.83)	33.2 (0.31)	99.5 (0.01)	92.2 (0.02)	32.2 (1.62)	33.5 (1.33)	30.9 (0.71)
KNN ($k=13$)	92.6 (0.06)	76.0 (0.08)	22.4 (0.86)	22.6 (0.23)	22.3 (0.53)	97.9 (0.01)	82.0 (0.02)	22.4 (0.64)	22.4 (0.82)	21.1 (1.08)
KNN ($k=25$)	88.3 (0.07)	67.5 (0.06)	17.5 (0.51)	17.1 (1.58)	17.0 (0.56)	95.6 (0.02)	73.4 (0.08)	18.8 (0.11)	18.0 (1.32)	17.7 (1.50)
KNN ($k=53$)	81.8 (0.07)	57.6 (0.05)	14.1 (1.19)	14.8 (1.51)	13.1 (0.76)	90.8 (0.02)	63.0 (0.09)	14.2 (0.16)	14.6 (0.57)	13.9 (0.73)
KNN ($k=89$)	75.9 (0.01)	50.7 (0.08)	12.6 (0.73)	13.2 (0.68)	12.2 (1.26)	85.7 (0.09)	55.4 (0.06)	12.0 (0.40)	12.8 (1.12)	11.7 (0.67)
KNN ($k=139$)	70.0 (0.08)	45.2 (0.10)	11.8 (0.70)	11.9 (0.36)	11.7 (0.50)	79.6 (0.09)	48.9 (0.07)	11.1 (0.59)	10.3 (0.89)	11.1 (1.34)
LDA	13.1 (0.04)	11.3 (0.09)	35.4 (0.05)	31.6 (0.59)	51.5 (1.14)	13.3 (0.04)	10.7 (0.05)	35.5 (0.46)	32.9 (1.07)	50.3 (1.35)
SVM (RBF)	n/a	n/a	9.87 (0.27)	9.40 (0.13)	7.05 (0.08)	n/a	n/a	9.12 (0.23)	9.34 (0.50)	7.43 (0.46)
SVM (polynomial)	n/a	n/a	9.31 (0.13)	8.95 (0.34)	7.15 (0.26)	n/a	n/a	8.81 (0.25)	8.92 (0.15)	7.15 (0.08)
SVM (sigmoid)	n/a	n/a	7.43 (0.16)	6.99 (0.51)	7.02 (0.25)	n/a	n/a	7.37 (0.49)	7.51 (0.10)	7.12 (0.18)

Table F.8: P04 training accuracy classification results on conventional algorithms, where cyan and red colours highlight the best and the worst mean classifier accuracy respectively, e stands for the number of estimators in AB, lr stands for the learning rate of AB and k stands for the number of nearest neighbours in KNN.

Method	Mean imagined speech percentage accuracy (std. dev)					Mean inner speech percentage accuracy (std. dev)				
	Raw	Prep	LF	NLF	MFCC	Raw	Prep	LF	NLF	MFCC
AB ($e=50$, $lr=0.1$)	13.7 (0.10)	11.5 (0.14)	22.1 (2.15)	18.9 (0.69)	21.9 (1.04)	12.3 (0.11)	10.2 (0.08)	21.8 (1.49)	21.2 (1.49)	19.3 (1.24)
AB ($e=50$, $lr=0.5$)	15.1 (0.02)	12.4 (0.06)	21.3 (0.84)	19.3 (0.66)	17.2 (3.90)	14.0 (0.28)	11.9 (0.03)	21.9 (0.34)	19.5 (1.05)	17.1 (1.35)
AB ($e=50$, $lr=1.0$)	14.6 (0.17)	12.0 (0.13)	19.4 (0.75)	17.9 (0.90)	13.9 (2.14)	13.7 (0.20)	11.6 (0.05)	18.1 (0.84)	15.4 (1.03)	16.0 (1.28)
AB ($e=50$, $lr=1.5$)	12.7 (0.22)	11.3 (0.36)	18.1 (0.88)	14.0 (0.30)	14.1 (1.57)	12.8 (0.27)	11.1 (0.20)	15.3 (1.53)	14.5 (1.29)	13.7 (1.46)
AB ($e=50$, $lr=2.0$)	7.48 (0.04)	7.93 (0.48)	12.2 (2.98)	11.9 (1.92)	14.0 (1.86)	7.36 (0.36)	7.07 (0.22)	11.8 (1.15)	12.7 (1.12)	13.4 (1.62)
AB ($e=100$, $lr=1.0$)	16.2 (0.06)	12.9 (0.11)	18.7 (1.15)	15.5 (1.70)	15.6 (2.02)	15.0 (0.36)	12.8 (0.06)	17.0 (1.38)	15.1 (2.45)	14.8 (1.63)
AB ($e=200$, $lr=1.0$)	16.1 (0.20)	13.5 (0.11)	17.6 (1.15)	15.1 (1.99)	13.7 (2.66)	15.4 (0.18)	12.7 (0.09)	16.9 (0.80)	13.4 (0.66)	14.9 (1.20)
AB ($e=400$, $lr=1.0$)	15.5 (0.15)	13.5 (0.17)	16.1 (3.05)	15.0 (1.53)	14.2 (1.53)	14.8 (0.09)	12.8 (0.17)	14.0 (0.52)	25.1 (6.86)	14.4 (0.39)
Gaussian NB	9.73 (0.02)	8.79 (0.03)	8.71 (0.83)	8.31 (0.40)	40.4 (0.81)	9.38 (0.03)	8.89 (0.03)	9.47 (0.29)	9.06 (0.74)	40.1 (0.94)
KNN ($k=1$)	100 (0.00)	100 (0.00)	100 (0.00)	100 (0.00)	100 (0.00)	100 (0.00)	100 (0.00)	100 (0.00)	100 (0.00)	100 (0.00)
KNN ($k=5$)	97.9 (0.06)	88.1 (0.04)	31.1 (0.92)	31.3 (0.78)	29.7 (0.67)	96.8 (0.03)	82.7 (0.10)	31.2 (1.12)	32.6 (0.76)	30.3 (0.98)
KNN ($k=13$)	93.7 (0.07)	75.6 (0.05)	22.3 (0.66)	21.7 (0.21)	21.8 (1.30)	91.0 (0.03)	67.8 (0.08)	22.8 (0.79)	21.4 (0.77)	20.9 (0.73)
KNN ($k=25$)	89.3 (0.08)	65.5 (0.06)	16.5 (0.31)	16.6 (0.86)	16.6 (0.23)	85.8 (0.03)	56.8 (0.04)	17.0 (1.08)	17.4 (0.70)	15.4 (0.94)
KNN ($k=53$)	82.4 (0.08)	54.7 (0.10)	13.0 (0.47)	13.7 (0.39)	12.6 (0.60)	78.6 (0.10)	46.0 (0.06)	13.0 (0.12)	13.9 (1.48)	13.3 (0.64)
KNN ($k=89$)	76.1 (0.09)	47.4 (0.14)	11.2 (0.27)	11.5 (0.04)	12.1 (0.42)	72.6 (0.07)	39.9 (0.07)	11.0 (0.89)	11.6 (0.51)	11.3 (0.83)
KNN ($k=139$)	69.4 (0.02)	41.4 (0.03)	11.0 (0.77)	10.1 (0.89)	9.97 (0.54)	66.6 (0.05)	35.1 (0.10)	10.0 (0.60)	9.91 (0.09)	10.6 (0.69)
LDA	12.4 (0.02)	10.7 (0.02)	35.0 (0.99)	30.9 (0.61)	51.0 (0.71)	11.1 (0.06)	10.6 (0.04)	35.5 (0.85)	30.7 (0.37)	49.7 (0.40)
SVM (RBF)	n/a	n/a	9.87 (0.42)	9.97 (0.23)	7.05 (0.15)	n/a	n/a	8.28 (0.28)	8.46 (0.74)	7.34 (0.48)
SVM (polynomial)	n/a	n/a	9.31 (0.47)	8.86 (0.46)	7.74 (0.67)	n/a	n/a	7.96 (0.25)	7.73 (0.24)	7.34 (0.61)
SVM (sigmoid)	n/a	n/a	7.40 (0.32)	7.27 (0.17)	7.12 (0.19)	n/a	n/a	7.02 (0.24)	7.15 (0.24)	7.05 (0.16)

Table F.9: Binary training accuracy classification results on conventional algorithms, where cyan and red colours highlight the best and the worst mean classifier accuracy respectively, e stands for the number of estimators in AB, lr stands for the learning rate of AB and k stands for the number of nearest neighbours in KNN.

Method	Mean percentage accuracy (std. dev)				
	Raw	Prep	LF	NLF	MFCC
AB ($e=50$, $lr=0.1$)	66.9 (0.28)	60.3 (0.37)	91.5 (2.31)	89.5 (1.80)	92.0 (0.38)
AB ($e=50$, $lr=0.5$)	72.8 (0.35)	64.4 (0.14)	100 (0.00)	100 (0.00)	100 (0.00)
AB ($e=50$, $lr=1.0$)	75.2 (0.14)	65.6 (0.12)	100 (0.00)	100 (0.00)	100 (0.00)
AB ($e=50$, $lr=1.5$)	74.7 (1.09)	65.3 (0.41)	100 (0.00)	100 (0.00)	100 (0.00)
AB ($e=50$, $lr=2.0$)	50.1 (0.16)	50.2 (0.13)	87.7 (1.55)	80.5 (3.17)	87.5 (2.52)
AB ($e=100$, $lr=1.0$)	78.3 (0.26)	67.4 (0.28)	100 (0.00)	100 (0.00)	100 (0.00)
AB ($e=200$, $lr=1.0$)	80.0 (0.40)	68.9 (0.05)	100 (0.00)	100 (0.00)	100 (0.00)
AB ($e=400$, $lr=1.0$)	81.3 (0.35)	69.9 (0.10)	100 (0.00)	100 (0.00)	100 (0.00)
Gaussian NB	50.0 (0.13)	53.7 (0.11)	53.3 (2.30)	52.6 (2.54)	79.5 (1.47)
KNN ($k=1$)	100 (0.00)	100 (0.00)	100 (0.00)	100 (0.00)	100 (0.00)
KNN ($k=5$)	99.9 (0.02)	95.3 (0.07)	70.7 (2.77)	70.0 (5.22)	69.3 (2.10)
KNN ($k=13$)	99.7 (0.03)	90.6 (0.19)	60.3 (3.40)	61.7 (2.25)	65.7 (2.66)
KNN ($k=25$)	99.2 (0.05)	87.0 (0.08)	60.5 (3.55)	55.8 (3.08)	58.0 (0.74)
KNN ($k=53$)	98.0 (0.06)	82.4 (0.29)	53.2 (2.04)	53.3 (3.89)	55.5 (4.01)
KNN ($k=89$)	96.1 (0.06)	78.9 (0.30)	53.7 (2.60)	53.5 (1.78)	53.3 (3.28)
KNN ($k=139$)	93.8 (0.11)	76.4 (0.04)	51.0 (0.47)	52.3 (0.82)	51.2 (0.64)
LDA	62.0 (0.17)	57.6 (0.26)	99.5 (0.71)	92.1 (1.16)	98.0 (0.93)
SVM (RBF)	75.9 (0.43)	69.3 (0.15)	54.0 (1.50)	53.3 (1.75)	51.2 (0.64)
SVM (polynomial)	71.1 (0.40)	56.0 (0.09)	54.3 (2.58)	54.5 (2.12)	53.2 (1.57)
SVM (sigmoid)	50.2 (0.07)	49.8 (0.41)	50.5 (0.47)	49.5 (1.79)	52.0 (0.76)

Table F.10: FEIS dataset training accuracy classification results on conventional algorithms, where cyan and red colours highlight the best and the worst mean classifier accuracy respectively, e stands for the number of estimators in AB, lr stands for the learning rate of AB and k stands for the number of nearest neighbours in KNN.

Method	Mean percentage accuracy (std. dev)				
	Raw	Prep	LF	NLF	MFCC
AB ($e=50$, $lr=0.1$)	14.2 (0.20)	12.8 (0.08)	30.4 (3.82)	28.4 (1.40)	24.6 (0.73)
AB ($e=50$, $lr=0.5$)	16.9 (0.18)	14.1 (0.02)	22.8 (2.56)	20.2 (0.94)	17.3 (2.01)
AB ($e=50$, $lr=1.0$)	16.4 (0.42)	13.9 (0.08)	13.0 (1.34)	15.7 (2.63)	15.5 (1.08)
AB ($e=50$, $lr=1.5$)	15.0 (0.25)	13.1 (0.18)	15.8 (2.01)	14.3 (1.20)	14.4 (0.90)
AB ($e=50$, $lr=2.0$)	8.31 (0.89)	8.99 (0.76)	14.6 (1.03)	12.4 (1.18)	14.8 (1.70)
AB ($e=100$, $lr=1.0$)	17.7 (0.15)	14.9 (0.08)	14.1 (0.90)	15.6 (0.36)	17.4 (3.01)
AB ($e=200$, $lr=1.0$)	18.0 (0.19)	15.3 (0.09)	15.3 (2.55)	15.1 (2.24)	10.4 (0.90)
AB ($e=400$, $lr=1.0$)	16.7 (0.14)	14.9 (0.06)	21.7 (8.96)	16.4 (2.39)	14.3 (0.86)
Gaussian NB	12.8 (0.03)	11.6 (0.04)	12.1 (1.65)	11.2 (0.50)	67.0 (1.35)
KNN ($k=1$)	100 (0.00)	100 (0.00)	100 (0.00)	100 (0.00)	100 (0.00)
KNN ($k=5$)	95.6 (0.04)	70.6 (0.04)	33.6 (0.86)	33.6 (0.76)	34.0 (1.18)
KNN ($k=13$)	91.0 (0.04)	57.3 (0.14)	21.9 (0.88)	21.2 (1.69)	20.3 (0.32)
KNN ($k=25$)	87.1 (0.12)	48.5 (0.12)	18.1 (1.20)	18.1 (0.73)	16.7 (0.98)
KNN ($k=53$)	81.8 (0.01)	40.3 (0.15)	13.0 (1.20)	12.0 (0.49)	11.4 (0.99)
KNN ($k=89$)	76.9 (0.17)	35.6 (0.04)	10.2 (0.41)	10.6 (0.41)	11.1 (1.22)
KNN ($k=139$)	72.6 (0.12)	32.5 (0.07)	10.3 (0.54)	9.31 (1.70)	10.6 (1.17)
LDA	13.1 (0.08)	12.5 (0.10)	55.8 (0.89)	54.0 (0.63)	81.4 (0.89)
SVM (RBF)	n/a	n/a	8.81 (0.73)	7.99 (0.18)	7.30 (0.18)
SVM (polynomial)	n/a	n/a	7.99 (0.09)	7.80 (0.23)	8.24 (0.58)
SVM (sigmoid)	n/a	n/a	7.55 (0.15)	7.20 (0.54)	7.36 (0.41)