Journal of Scientific & Industrial Research
Vol. 82, November 2023, pp. 1165-1176
DOI: 10.56042/jsir.v82i11.1543

A Novel BCI - based Silent Speech Recognition using Hybrid Feature Extraction Techniques and Integrated Stacking Classifier

N Ramkumar* & D Karthika Renuka

Department of Information Technology, PSG College of Technology, Coimbatore 641 004, Tamil Nadu, India

Received 15 May 2023; revised 05 September 2023; accepted 20 September 2023

The Brain Computing Interface (BCI) is a technology that has resulted in the advancement of Neuro-Prosthetics applications. BCI establishes a connection between the brain and a computer system, primarily focusing on assisting, enhancing, or restoring human cognitive and sensory - motor functions. BCI technology enables the acquisition of Electroencephalography (EEG) signals from the human brain. This research concentrates on analyzing the articulatory aspects, including Wernicke's and Broca's areas, for Silent Speech Recognition. Silent Speech Interfaces (SSI) offers an alternative to conventional speech interfaces that rely on acoustic signals. Silent Speech refers to the process of communicating speech in the absence of audible and intelligible acoustic signals. The primary objective of this study is to propose a classifier model for phoneme classification. The input signal undergoes preprocessing, and feature extraction is carried out using traditional methods such as Mel Frequency Cepstrum Coefficients (MFCC), Mel Frequency Spectral Coefficients (MFSC), and Linear Predictive Coding (LPC). The selection of the best features is based on classification accuracy for a subject and is implemented using the Integrated Stacking Classifier. The Integrated Stacking Classifier outperforms other traditional classifiers, achieving an average accuracy of 75% for both thinking and speaking states on the KaraOne dataset and approximately 86.2% and 84.09% for thinking and speaking states on the Fourteen Channel EEG for Imagined Speech (FEIS) dataset.

Keywords: Electroencephalography, Linear predictive coding, Mel frequency cepstrum coefficients, Mel frequency spectral coefficients, Silent speech interface

Introduction

Disability has a broad impact on various aspects of human life. Families caring for disabled individuals often face significant emotional, financial, and sometimes even physical stress. However, proactive efforts in finding resources, foreseeing potential outcomes, and making plans can greatly enhance one's quality of life. Approximately 0.3% of the global population, equivalent to around 20 million people, is affected by conditions such as aphasia resulting from conditions like strokes, severe tetraplegia occurring in the upper spinal cord, and neuromuscular disorders. This underscores the importance of addressing speech disabilities, which is where BCI technology plays a vital role. BCI serves as a communication method for these individuals, acting as an interface between the brain and external devices. It involves the acquisition of various brain signals, their analysis, and conversion into commands displayed on an output device to execute specific tasks. There are two main methods for acquiring these signals: invasive and non-invasive BCI. Invasive BCI requires surgical implantation of electronic devices in the skull, while non-invasive BCI involves placing sensors and electrodes on the scalp to collect signals.[1] Silent Speech Interfaces (SSI) represent an emerging area of research that offers an alternative solution for individuals with communication disabilities. One of the primary focuses in SSI research is phoneme classification, which involves analyzing the sound units in speech. SSIs interpret and route acquired data from the human speech production process, which is obtained in the form of signals from the brain, particularly the vocal cords.[2] Electrodes, positioned using EEG caps, facilitate data collection, followed by further processing.

In complex environments, the efficiency of many existing classification algorithms can be reduced. To address this challenge, this paper introduces an Integrated Stacking classifier that utilizes selected features extracted from various feature selection techniques. The input signals are gathered from two datasets: the KaraOne dataset from the University of

*Author for Correspondence
E-mail: 217rifx01@psgtech.ac.in

Toronto and the FEIS dataset from the University of Edinburgh. Signal preprocessing involves Independent Component Analysis (ICA) to eliminate artifacts from the input signals.[3] Three different feature extraction techniques, involving combinations of MFCC, MFSC, and LPC, are applied to the preprocessed signals. The best features are selected for each subject in the two datasets based on classification accuracy. These selected features are then input into the proposed Integrated Stacking Classification algorithm, leading to performance improvements.[4]

The contributions of this paper can be summarized as follows:

(i) Selection of EEG cap channels relevant to Silent Speech Interface.
(ii) Identification of the best features based on classification accuracy.
(iii) Evaluation of the proposed architecture, providing a benchmark for the Integrated Stacking Classifier model using benchmark datasets.

Related Work

Recently, Bhuvaneshwari *et al*. introduced an innovative optimization approach called Red-Fox Sine Cosine Optimization (RFO-SCA) for feature selection. They applied classifiers like K-Nearest Neighbour (KNN), decision trees, and Random Forest to the selected features, reducing feature dimensionality by 50% using RFO-SCA.[5] In another study, Edla *et al.* detected deceit using EEG signals and deep neural networks. They employed Bandpass filters and Wavelet packet transforms for noise removal and feature extraction, achieving a remarkable 95% accuracy for 30 subjects using auto encoders and softmax networks.[6] Mini *et al.* proposed a Multimodal approach for Automatic Speech Recognition, initially investigating single modality and then incorporating Multimodality. They explored various feature extraction methods, including Discrete Wavelet Transform (DWT), Wavelet Packet Decomposition (WPD), and a combination of both, followed by applying Artificial Neural Networks (ANN) and information fusion concepts. Their study concluded that WPD performed better for Multiclass classification.[7]

Sharon & Murthy presented a multimodal approach for imagined speech recognition, employing both Handcrafted and CorrNets feature extraction methods. They evaluated the KaraOne dataset using the Kaldi toolkit and achieved an accuracy of 35.82% for 11 prompts using Gaussian and Hidden Markov models with deep neural networks.[8] Clayton *et al.* discussed Decoding Imagined Speech. They utilized Independent Component Analysis, bandpass filters, and Hilbert transforms for preprocessing, working with the FEIS dataset. Support Vector Machine (SVM) achieved 69% accuracy for thinking, and Convolutional Neural Network (CNN) achieved 49%, while for speaking, SVM outperformed CNN with 63.7% compared to 49.4%.[9] Mansoor *et al.* discussed Brain-Computer Interface using Deep Learning Algorithms, recommending Short-term Fourier transforms and Notch filters for data preprocessing. They found that adaptive classifiers, particularly CNN, achieved the best accuracy with various speech signal datasets.[10] Dash *et al.* focused on non-invasive devices, collecting signals from eight adults using Magnetoencephalography (MEG). They applied statistical features and CNN to achieve impressive accuracy, around 93% for the imagined state and 96% for the speaking state.[11] Rusnac & Grigore investigated Multi-Class classification using CNN, employing bandpass and notch filters for frequency selection and MFCC for feature extraction. They achieved approximately 24.19% accuracy on the KaraOne dataset.[12] Sharon *et al.* studied speech in three modes: imagined, audition, and production states, focusing on three datasets. They applied bandpass filters and EEGLAB for preprocessing, extracting temporal, spectral, and spatial features. They measured classification accuracy using a unit error rate and adopted the selection-by-exclusion method for optimization.[13] Saha *et al.* contributed to Hierarchical Deep Learning for Imagined Speech recognition, emphasizing phoneme classification using models like CNN, Temporal Convolutional Neural Network (TCNN), Autoencoder, and Long Short Term Memory (LSTM). The CNN was used for spatial feature extraction, and deep auto encoders reduced noise. Their results showed that CNN with LSTM and Deep Auto Encoders (DAE) outperformed other models.[14] Mahapatra & Bhuyan addressed multi-class classification, employing filters like band pass, Laplace, and ICA for noise removal and MFCC feature extraction.[15] They applied machine learning classifiers such as Decision Trees and SVM to the KaraOne dataset, achieving accuracy ranging from 15.81% to 20.80%.

The review of existing research identifies several gaps, including the need for more research on Silent Speech Interfaces, a lack of learning systems for human-computer interaction, and limited exploration

of human-computer interaction in the healthcare domain. To address these gaps, the proposed methodology utilizes various feature extraction techniques and applies them to the Integrated Stacking Classifier, which outperforms traditional classifiers in phoneme classification based on EEG signals acquired from the brain. The proposed model's architecture is illustrated in Fig. 1.

Proposed Methodology

In the previous section, the literature review primarily focused on the analysis of a single type of dataset. However, in our proposed approach, the analysis of two distinct datasets is taken into account. In Fig. 1, the initial stage involves data collection, which is obtained from EEG caps placed on the Wernicke's and Broca's areas. The diagram illustrates the collection of phoneme data from both the KaraOne dataset and the FEIS dataset. Subsequently, in the second stage, the data preprocessing was performed, wherein Independent Component Analysis (ICA) is employed to eliminate noise from the signals. Butterworth filters are applied to select the desired frequency range, and appropriate channels are chosen for further processing. Moving on to the third stage, feature extraction is conducted on both datasets utilizing three different techniques. In the fourth stage, the features extracted are inputted into three distinct traditional classifiers, namely SVM, CNN, and MLP. The selection of the best features is based on the classification accuracy achieved by these three different classifiers. Finally, in the last stage, the selected features are fed into our proposed Integrated Stacking Classifier Model for further analysis and decision-making.

Data Acquisition

Electroencephalography (EEG) signals are obtained by positioning electrodes on the human scalp, a non-invasive procedure that does not necessitate surgical electrode implantation. These electrodes capture brain signals associated with various mental states. EEG signals are categorized into five types, each characterized by distinct frequency ranges: alpha, beta, gamma, theta, and delta. The descriptions of these signal types are provided in the table below Table 1.

The Table 1 describes the frequency range of an individual and their description of which state it could be used. The analysis incorporates two benchmark datasets: the KaraOne dataset and the Fourteen Channel Imagined dataset (FEIS). The KaraOne dataset was developed by the University of Toronto and comprises 14 participants, all of whom are right-handed. Among these participants, ten have English as their native language, and two are highly fluent

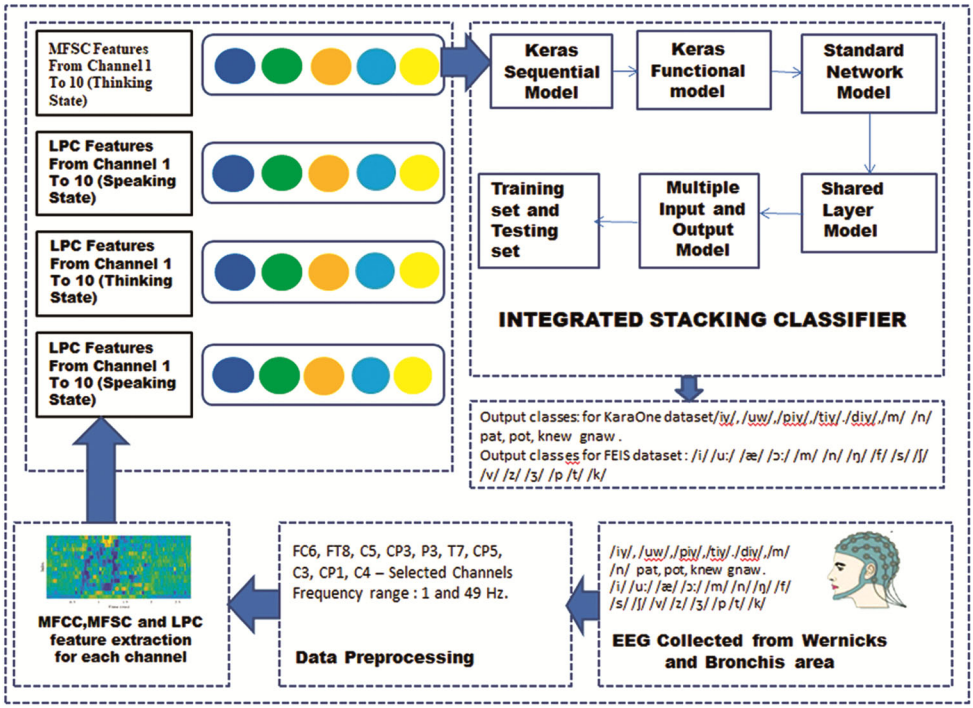

Fig. 1 — Diagram of proposed integrated stacking classifier for silent speech recognition

Table 1 — Type of Signals

S. No	Signals	Frequency Range	Description
1.	Alpha	8-12 Hz	Alpha waves are slow and large. Alpha waves are obtained during the relaxation state.
2.	Beta	13-38 Hz	Beta waves are smaller, but faster. They are linked with mental state and more focus on concentration. These brainwaves are a state of alertness.
3.	Gamma	39-42 Hz	Gamma waves are the fastest waves. Gamma waves are more into consciousness.
4.	Delta	1-3 Hz	Delta waves are the slowest, but it has the highest amplitude. These waves are linked with the asleep state
5.	Theta	4-7 Hz	Theta waves are very slow and are acquired in a relaxed state. They are linked with mental inefficiency

English speakers. The dataset was collected using a 64-channel EEG Cap and encompasses four different mental states: Resting, Stimuli, Imagined, and Speaking. These states are divided into 5-second epochs. The dataset includes seven phonemes: /iy/, /uw/, /piy/, /tiy/, /diy/, /m/, and /n/. Additionally, four distinct words—pat, pot, knew, and gnaw—are considered for analysis. The FEIS dataset is derived from the KaraOne dataset and features a 14-channel EEG cap. It was compiled with the participation of 21 individuals, including 17 right-handed individuals and three who can comfortably use both their right and left hands. In the FEIS dataset, 16 phonemes are used for analysis: /i/, /u:/, /æ/, /ɔ:/, /m/, /n/, /ŋ/, /f/, /s/, /ʃ/, /v/, /z/, /ʒ/, /p/, /t/, and /k/. Both the KaraOne and FEIS datasets were utilized in the experiment to evaluate the effectiveness of the proposed approach.

KaraOne Dataset

The KaraOne dataset is composed of EEG recordings collected using a 64-channel EEG Cap with a sampling frequency of approximately 1000 Hz. The duration of each recording ranges from 30 to 40 minutes. For the experiment, a specific set of channels was selected, including FC6, FT8, C5, CP3, P3, T7, CP5, C3, CP1, and C4.

FEIS Dataset

The FEIS dataset comprises EEG recordings obtained from a 14-channel EEG Cap, with a sampling frequency of approximately 256 Hz. Each recording has duration of approximately 60 min. The selected channels for analysis include F3, FC5, AF3, F7, T7, P7, O1, O2, P8, T8, F8, FC6, and F4. For reference, the locations of these channels are depicted in Fig. 2. It represents the Channel location for the KaraOne dataset and FEIS dataset. The black color represents the KaraOne dataset and the red color indicates the FEIS dataset.

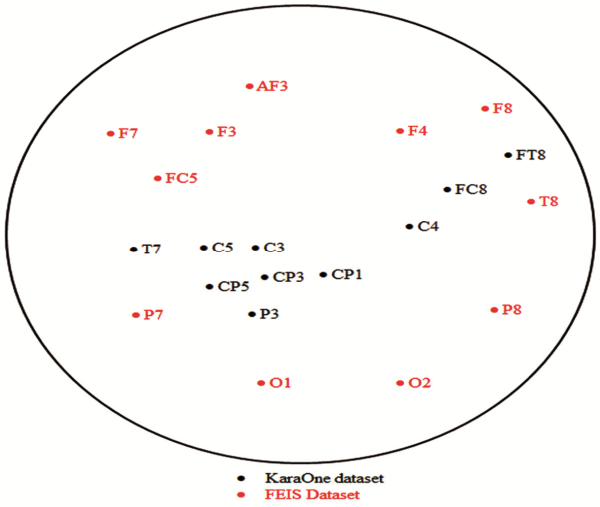

Fig. 2 — Channel location of KaraOne and FEIS dataset

Preprocessing

The preprocessing stage is carried out using the open-source tool EEGLAB. In this process, the data is filtered within a frequency range of 1 to 49 Hz, and the mean value is computed. For each channel, the mean values are then subtracted. To achieve this, a Butterworth band-pass filter is applied. In the low-pass filter, signals with frequencies lower than the specified cut-off frequency are extracted, while signals with frequencies higher than this cut-off frequency are processed accordingly.

Design of the Butterworth Filter

The Butterworth filter is employed with the specific purpose of choosing the desired frequency range. The Butterworth low-pass filter incorporates two essential parameters: N and W_p, where N signifies the order of the filter, and W_p denotes the cut-off frequency. In this context, the value of N is set to 3, while W_p is set to 0.02. Consequently, the range of frequencies that are selected spans from 1 Hz to 49 Hz. The Eq. (1) for the Butterworth low-pass filter is illustrated as follows:

$$H_{(j\omega)} = \frac{1}{\sqrt{1+\epsilon^2\left[\frac{w}{wp}\right]^{2n}}} \quad \ldots (1)$$

Channel selection plays a pivotal role in this process, with careful consideration given to the EEG cap, which consists of approximately 64 channels. In the context of silent speech, it's essential to focus on Broca's and Wernicke's areas of the brain, which are central to speech production and comprehension.

Therefore, channels were chosen strategically, encompassing the regions around these two critical areas. The selected channels include FC6, FT8, C5, CP3, P3, T7, CP5, C3, CP1, and C4. To enhance data quality and reliability, Independent Component Analysis (ICA) is employed to effectively eliminate artifacts from the input signal.

Feature Extraction

The human neurosystem is incredibly intricate, generating a vast array of signals associated with various activities. Consequently, it is imperative to take into account multiple characteristics to discern the nuances in these brain signals. In the realm of Brain-Computer Interfaces (BCI), feature extraction stands out as a crucial task, given the diverse nature of invasive and non-invasive brain signals. These features can encompass a range of aspects, including time-domain features, frequency-domain features, and regression features, all of which contribute to understanding and interpreting EEG signals effectively. In our proposed method for the silent speech interface, the inputs obtained from two pivotal areas in the brain are focused: Broca's and Wernicke's areas. Electrodes are strategically placed over these regions to capture relevant signals, and the channels corresponding to these electrodes are specifically chosen for our research. These selected channels are FC6, FT8, C5, CP3, P3, T7, CP5, C3, CP1, and C4. In our proposed approach, the thinking and speaking states as part are considered for the analysis.

Mel Frequency Cepstrum Coefficients

The Mel Frequency Cepstral Coefficients (MFCC) represents one of the feature extraction methods applied to signals. The MFCC method operates as follows: It begins by segmenting the input signal into frames using a windowing technique. Discrete Fourier Transform (DFT): The DFT is applied to each frame to transform it into the frequency domain. A logarithmic function is then applied to the magnitude of the DFT values. This step helps emphasize the perceptually relevant lower frequencies in the next step; the frequencies are distorted to follow the Mel scale. This distortion aims to mimic the human auditory system's sensitivity to various frequencies. Inverse Discrete Cosine Transform (DCT): it is used to extract the coefficients. MFCC typically yields 13 static coefficients per frame. Dynamic Coefficients: In addition to the static coefficients, the first derivative represents velocity, and the second derivative represents acceleration. These dynamic coefficients provide information about how the static coefficients change over time. When dealing with the original signal; it's common to remove signal edges, which can introduce high-frequency noise due to abrupt changes in amplitude. To mitigate this, Hamming or Hanning windows are applied during segmentation to maintain a smooth transition between frames and reduce high-frequency noise. Furthermore, the frequency bands are divided equally on the Mel scale, which is a perceptually motivated scale that corresponds more closely to human auditory perception.[16] The typical configuration for MFCC includes a Hamming window size of 25 milliseconds with a 10-millisecond offset. The minimum and maximum values of the MFCC coefficients are typically set to −6.6027 and 21.1053, respectively. The MFCC are particularly useful because they emphasize features of the audio signal that are important for human speech perception removing the less relevant information.

Linear Predictive Coding

Linear Predictive Coding (LPC) is a feature extraction method that finds extensive use in speech synthesis. In our proposed approach, where we analyze silent speech in both thinking and speaking states, we employ LPC to extract relevant features from the human vocal tract. LPC is chosen for its ability to compute both the spectral envelope's concentration and the signal's fundamental frequency. The process of LPC feature extraction involves several steps: Frame Blocking: The input signal is divided into multiple frames, typically following a standard duration of about 10 milliseconds between two adjacent frames. Windowing: To minimize the edge effects, frames are multiplied by a Hamming window. This windowing technique ensures a smooth transition between frames. Auto-Correlation Analysis: After windowing, LPC analysis is performed by estimating the maximum auto-correlation value. This step helps identify and characterize the resonance properties of the vocal tract. LPC is particularly valuable in capturing the spectral characteristics of speech, making it a suitable choice for analyzing silent speech in our research context.[17]

Mel frequency Spectral Coefficients

The Mel Frequency Spectral Coefficients (MFSC) is computed before applying the Discrete Cosine Transform (DCT) in our feature extraction process. These coefficients are derived from frequency ranges that are evenly spaced on the Mel scale, approximating the response of the human auditory

system. The MFSC exhibits some degree of correlation among its features. However, this correlation is not a concern because deep neural networks used in our proposed model can effectively handle internal decorrelation. In this research, 39 mel filterbanks are employed to calculate the MFSC for a sampled size of 16 Hz.[18] to ensure the consistency of these coefficients, a minimum threshold value of 0.0075 and a maximum value of 5.25 are applied. The parameters for the three feature extraction techniques, including MFSC, are summarized in Table 2. The MFSC, MFCC, LPC are used for extracting the features from the input signal. After extracting the features, it is fed into different classifiers SVM, CNN and MLP. From the classification results it is observed that MFSC features over thinking state and LPC features over speaking state of KaraOne dataset gave better accuracy, Henceforth both MFSC and LPC features of KaraOne dataset will be selected by the proposed Integrated Stacking Classifier and it is fed as an input to the proposed classifier

Classification Algorithms

In this research, the classification of phonemes using EEG signals in thinking and speaking states is explored. To achieve this, we employ traditional classification algorithms such as Support Vector Machine (SVM), Convolutional Neural Network (CNN), and Multi-Layer Perceptron (MLP). The optimal features are derived from the earlier section, utilizing various feature extraction techniques. Our analysis encompasses individual subject-based evaluations, where we classify the best feature extraction methods. The algorithms and their associated hyper parameters are as follows:

Support Vector Machine

The Support Vector Machine is the traditional machine learning algorithm used for both linear and non-linear data. It is a classification algorithm. The key concept of SVM is its hyperplane. A Hyperplane is used to separate data from two classes. It identifies the hyperplane with the help of support vectors and margins. This has both small margins and large margins. To find the best hyperplane, it always looks for a larger margin, i.e. Maximum Marginal Hyperplane (MMH). The SVM model is implemented using Hinge loss and the hyper parameters are defined as C ε [1:1000] and Gamma ε [0.001: 0.000001]

Table 2 — Configuration of different parameters used in the feature extraction technique

Parameters	MFCC	MFSC	LPC
Hamming window size	25 ms	25 ms	20 ms
Offset Length	10 ms	10 ms	10 ms
Cepstral coefficients	12	12	NA
Final coefficients	39	39	NA
Filters	0.95	0.95	0.75
Filterbank channels	25	39	13
Cepstral filters	22	22	NA

The larger margins are realized by larger margins by reducing the cost function as given in Eq. (2):

$$\frac{1}{2}\ \|\omega^2\| + A\ \sum_{j=1}^{m} \in i \qquad \ldots (2)$$

Under the condition

$x_{i\,(y^T\ a_i\ +C)} \geq 1\text{-}\ \varepsilon_I$ and

$\varepsilon_I \geq 0\ \forall$ S=1…..m.

Convolutional Neural Network

The CNN is one of the traditional neural networks and it performs the convolution operation which produces the features and places it over the stack. In CNN, Convolution is the base layer. The main purpose of this layer is to retrieve the attributes from the input features. The output of the convolution layer is as represented in Eq. (3):

$$C\ (Z_{a,b}) = \sum_{x=-i/2}^{i/2} \sum_{y=-j/2}^{j/2} f_k\ (x,y)\ Z_{a-i,b-j} \qquad \ldots (3)$$

where, f_k is the filter and it has i, x_j, kernel size fed into the input Z, and xj is the input connections to individual CNN neurons. The filter size used is 25, 50, 100 and 200. The next layer is the Pooling layer, the pooling layer is used to reduce the number of features in the given input. By making the learned functions more resistant to changes in scale and orientation, this layer minimizes the number of features while also enhancing the robustness of the learned functions. The filter length is 5.10.20.40 and the stride length is 3, 6, 9 and 12. The max pooling is used in the design of the CNN architecture and it is depicted in Eq. (4) :

$$A\ (b_i) = \text{maximum}\left\{b_{i+j,i+m} | c| \leq \frac{Z}{2}, |d| \leq \frac{Z}{2} K, d \in N\right\} \qquad \ldots (4)$$

where, b is the input and z is the size of the filter. The activation function used in the architecture is the Rectified Linear Unit (RELU). The operation of RELU is to convert all negative values present in the feature map into zero. The most important functionality of RELU is to make non-linearity of the CNN model. RELU uses the activation function of form A(y) = maximum (z, y) to manipulate its output.

After pooling, the fully connected layer is designed in such a way that all the neurons from the preceding layer are connected to each of the neurons in this

layer. A suitable approach for learning nonlinear combinations of these features is to add a fully connected layer. The optimizer used is Adam with a batch size of 1 and the dropout value assigned as 0.5. The output is given as in Eq. (5):

$$P(y) = \sigma\left(X_{iXj} * y\right) \quad \ldots (5)$$

where, σ is the activation function, j is the input size to y, i is the total number of neurons in the fully connected layer and the resultant matrix is defined as X. The last layer is the output layer, the output layer is depicted as one hot vector that indicates the class of the given vector. The resultant class for the output vector y is given in Eq. (6):

$$\left\{j \middle| \exists\ i\ \forall\ j \neq i : x_j \leq x_i\right\} \quad \ldots (6)$$

Softmax Layer: The Softmax layer propagates the error. Let X be the input vector, the softmax calculates the mapping as ⟶ P(y): X^A [0, 1] ^{A}. For an individual component 1 ≤ x ≤ A, the output is calculated and it is shown in Eq. (7):

$$P(a)_i = \frac{e^{xi}}{\sum_{k=1}^{m} e^{xj}} \quad \ldots (7)$$

MultiLayer Perceptron

The Multi-layer Perceptron is a feed forward neural network that has 3 layers, the input layer, output layer and hidden layer. The number of hidden layers can be designed by the user. The input data will flow in the forward direction between the input and output layers. For our experiment, the hidden layer size is assigned as 100, 50 with a random state as 5, the verbose is true, and the learning rate is 0, 100. The neurons are trained with the back propagation algorithm.

Initially, fix the weights and thresholds to a random number between −1 and 1 and select the pairs from (a^x, b^y), from the training data and assign them as an input variable into the input layer (n = 0) and assume

$Z^o_j = a^x_j$, for every node in i, x is the layer numbered. The signal is passed forward throughout the network as Eq. (8) calculates the output C_k^d of k th node from the initial layer to the last layer

$$C_k^d = F(t_k^q) = F\left(\sum_J V_{lk}^q\ C_l^{q-1} + \theta_k^q\right) \quad \ldots (8)$$

where, F(t) is defined as the Sigmoid function. The error at the output layer is defined in the Eq. (9)

$$\delta_i^a = x_i^a(1 - x_i^a)\left(S_I^b - x_i^a\right), \quad \ldots (9)$$

The error is calculated as the output and the actual value. The error rate of the previous layer is given in the Eq. (10):

$$\partial_i^{a-1} = F(x_i^{a-1})\ \sum_j V_{ji}^a\ \delta_j^a \quad \ldots (10)$$

The weights and thresholds are updated and propagated backward are shown in the Eq. (11) & (12):

$$w_{ab}^n(s+1) = w_{ab}^n(s) + \eta\ \delta_b^a + y_a^{n-1} + \alpha\ [w_{ab}^n(s) - w_{ab}^n(s-1)] \quad \ldots (11)$$

$$\theta_b^n(s+1) = \theta_b^n(s) + \eta\delta_b^n + \alpha[\theta_b^n(s) - \theta_b^n(s-1)] \quad \ldots (12)$$

where, t represents the iteration, η is the learning rate and takes the values 0 and 1, and α is the momentum which takes the value 0 and 1, move back to step -2 and repeat from step -2 to step-7 up till the chosen error criteria are obtained.

$$F = \sum_a \sum_b \left(K_i^n - C_k^d\right)/2, \quad \ldots (13)$$

Once the network is trained, the weights and threshold are determined.

Integrated Stacking Classifier

In neural networks, the integrated stacking classifier has five models Keras sequential model, Keras Functional model, Standard Network Model, Shared Layer Model and Multiple Input and Output Model. It produces the classification accuracy based on the number of epochs, training and testing data. The neural network forms the meta-learner. In the proposed approach, the accuracy of all three classifiers is observed for an individual participant in the KaraOne dataset and FEIS dataset. The experiment is performed for the thinking state and speaking state. The Algorithm for the proposed Integrated Stacking Classifier model is as follows:

Algorithm: **INTEGRATED STACKING CLASSIFIER**

Input :
the training set A: $(p1\ \ , q1), (p2, q2) \ldots \ldots (pn, qn)$,
Datasets D $(d1, d2)$,
Features F: $\{f1, f2, f3 \ldots fn\}$,
Classifiers C: $\{C1, C2, C3\}$.
Output: Performance metrics of the phoneme classification for the dataset D
Begin
Do until all datasets are processed
Call the Preprocess() function;
Call the Feature extraction() function;
Call the Classifier() function;
Call the Integrated Stacking Classifier() function;
Function: Preprocess ()
for each dataset
D= Preprocess_Channel Selection, remove_artifacts
D=Preprocess_Select frequencies
$H_{(j\omega)} = \frac{1}{\sqrt{1+\epsilon^{2\left[\frac{w}{wp}\right]^{2n}}}}$
end for
return the preprocessed data from the dataset D
Function: Feature extraction()
for each dataset in D
Features (F) = Feature Extraction {MFCC, MFSC, LPC}

Calculate MFCC, to take DCT, the formula is as follows,

$C_i(a) = \sum_{i=1}^{M} C_i\,(n) j(n) e^{-k2\pi an/M}$ 1≤ k≤ K

The Periodogram estimate is defined as follows,

$P_i(k) = \frac{1}{N} |C_i(a)|^2$

Calculate MFSC, fourier transform is calculated using

$STFT(\tau,t) = \int_{-\infty}^{\infty} y(\tau) i(\tau - t)\, e^{-k2\pi f\tau}\, d\tau$

Calculate LPC, the equation is ,

$S(n) = -\sum_{i=1}^{q} b_k\, t(n-s)$

end for

return the extracted features from the dataset D

Function: Classifier()

for all the extracted features in dataset D

Extracted features_MFCC_MFSC_LPC = Classifiers {SVM)

$\frac{1}{2} \|\omega^2\| + A \sum_{j=1}^{m} \in i$

Extracted features_MFSC_MFCC_LPC=Classifiers {CNN}

$C\,(Z_{a,b}) = \sum_{x=-i/2}^{i/2} \sum_{y=-j/2}^{j/2} f_k\ (x,y)\ Z_{a-i,b-j}$

$A\,(b_i) = \text{maximum}\left\{ b_{i+j,i+m} |c| \le \frac{Z}{2}, |d| \le \frac{Z}{2} K, d \in N \right\}$

$P(y) = \sigma\left(X_{iXj} * y\right)$

$\left\{ j \middle| \exists\ i\ \forall\ j \ne i : x_j \le x_i \right\}$

$P\,(a)_i = \frac{e^{xi}}{\sum_{k=1}^{m} e^{xj}}$

Extracted features_LPC_MFCC_MFSC=Classifiers {MLP}

$C_k^{\,d} = F\,(t_k^{\,q}) = F\left(\sum_J V_{lk}^{q}\ C_l^{q-1} + \theta_k^{q}\right)$

$\delta_i^{a} = x_i^{a}(1 - x_i^{a})\left(S_l^{b} - x_i^{a}\right)$

$\partial_i^{a-1} = F\left(x_i^{a-1}\right) \sum_j V_{jl}^{a}\ \delta_j^{a}$

$w_{ab}^{n}(s+1) = w_{ab}^{n}(s) + \eta \qquad \delta_b^{a} + y_a^{n-1} + \alpha \qquad [w_{ab}^{n}(s) -$ *wabns− 1*

$\theta_b^{n}(s+1) = \theta_b^{n}\,(s) + \eta\delta_b^{n} + \alpha[\theta_b^{n}(s) - \theta_b^{n}(s-1)]$

$F = \sum_a \sum_b \left(K_l^{n} - C_k^{d}\right)/2,$

end for

return accuracy of the classifier model from the extracted features

Function: Integrated Stacking Classifier()

for each participant

Compare {Classifier_output1, Classifier_output_2, Classifier_output_3}

if best =Classifier_output1

then Retrieve the features from the Classifier_Output1

else if best= Classifier_output2

then Retrieve the features from the Classifier_Output2

else if best=Classifier_output3

then Retrieve the features from the Classifier_Output3

end for

return the best features

for all the best features in D

Best features = Integrated Stacking Classifier

end for

return accuracy of all the models for the dataset D

while all the data samples are processed

return precision, recall, f-measure and accuracy for the dataset D

End

Within the Integrated Stacking Classifier, the best classification accuracy achieved for each participant is assessed. The extracted features are identified as the most effective by the top-performing classifier for each participant. These selected features are then combined and used as input for the Integrated Stacking Classifier. This approach aims to leverage the strengths of individual classifiers and enhance overall classification performance. The algorithm combines the model based on the output of the different classifiers. Hence forth, while combining these models it will take the least amount of execution time and memory usage.

Results and Discussion

This section discusses a detailed explanation of the experiments that have been executed to exert the influence of the Integrated Stacking Classifier. The experiment analysis on two benchmark datasets using a Tesla A100 GPU server with deep learning libraries like Tensorflow and Keras. The training test is split into 80% and the testing set is split into 20%. The features extracted from different techniques for thinking and speaking state using the Karaone dataset and FEIS dataset are shown in Fig. 3.

In the above Fig. 3 the features from the KaraOne dataset are acquired based on the classification accuracy, MFSC and LPC feature extraction gave better accuracy applied to the three classifiers, henceforth, the proposed Integrated Stacking Classifier will take the MFSC features from the thinking state and LPC features from speaking state as an input and yields better performance than other traditional classifiers.

In Fig. 4 mentioned above, the features were obtained from the FEIS dataset. After evaluating the classification accuracy, it was determined that LPC feature extraction consistently yielded the highest accuracy when applied to all three classifiers. Therefore, the LPC features extracted from both the thinking state and speaking state within the FEIS dataset are inputted into our proposed model and the resulting classification accuracy is determined. This approach of using the best features for inputting into the proposed model significantly enhances the performance of the Integrated Stacking Classifier. Further in-depth analysis and discussion regarding

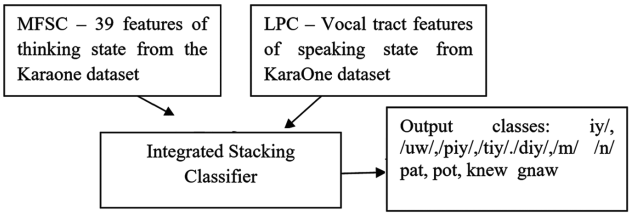

Fig. 3 — Extracted features from the KaraOne dataset

this improvement can be found in the Ablation Study, as presented in Table 4.

Ablation Study

The Ablation study involves an assessment of the three different feature extraction methods. Features are extracted using these techniques and subsequently inputted into the three distinct classifiers. In Tables denoted as 4, 5, 6, and 7, the average accuracy results are presented, considering all three feature extraction methods and all three classifiers. The analysis reveals that, among the various feature extraction techniques, MFSC demonstrates superior performance for the thinking state when applied to the KaraOne dataset. However, for the speaking state in the KaraOne dataset and for both thinking and speaking states in the FEIS dataset, LPC consistently delivers the best performance across all three classifiers. This suggests that LPC is the most effective feature extraction technique compared to the other two methods. Furthermore, the

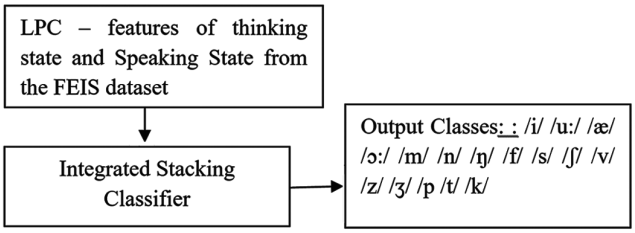

Fig. 4 — Extracted features from the FEIS dataset

Table 4 — Performance analysis of speaking State over KaraOne dataset

Feature Extraction	Classifier	Precision	Recall	F-Measure	Accuracy
MFCC	SVM	30.1	30.1	30.1	30.2
MFCC	CNN	50	50	50	50
MFCC	MLP	31	31	31	31.0
MFCC	Integrated Stacking Classifier	37	38	37	37
MFSC	SVM	29.4	29.3	29.4	29.3
MFSC	CNN	54	54	54	54
MFSC	MLP	30.3	30.4	30.3	30.5
MFSC	Integrated Stacking Classifier	38	37.6	38	38
LPC	SVM	37.9	38	37.9	38.0
LPC	CNN	57	57	57	57
LPC	MLP	35.2	35.2	35.2	35.3
LPC	Integrated Stacking Classifier	75.2	75.6	75	**75.85**

Table 5 — Performance analysis of thinking State over FEIS dataset

Feature Extraction	Classifier	Precision	Recall	F-Measure	Accuracy
MFCC	SVM	31.7	31.9	31.8	32.1
MFCC	CNN	53	53	53	53
MFCC	MLP	32.2	32.4	32.3	32.6
MFCC	Integrated Stacking Classifier	40	40.8	40	40
MFSC	SVM	34.3	34.3	34.3	34.5
MFSC	CNN	55	55	55	55
MFSC	MLP	33.7	33.7	33.7	33.7
MFSC	Integrated Stacking Classifier	41	41.5	41	41
LPC	SVM	36	36	36	36.1
LPC	CNN	58	58	58	58
LPC	MLP	36.5	36.5	36.5	36.7
LPC	Integrated Stacking Classifier	85.9	85.8	86	**86.2**

Table 6 — Performance analysis of speaking State over FEIS dataset

Feature Extraction	Classifier	Precision	Recall	F-Measure	Accuracy
MFCC	SVM	32.4	32.5	32.4	32.7
MFCC	CNN	52	52	52	52
MFCC	MLP	32.5	32.5	32.5	32.6
MFCC	Integrated Stacking Classifier	39	38.8	39	39.1
MFSC	SVM	35.3	35.4	35.4	32.7
MFSC	CNN	55	55	55	55
MFSC	MLP	34.4	34.4	34.4	32.6
MFSC	Integrated Stacking Classifier	40	39.9	39.7	40.1
LPC	SVM	39.2	39.2	39.2	39.3
LPC	CNN	58	58	58	58
LPC	MLP	39	39	39	40.8
LPC	Integrated Stacking Classifier	84.5	84	83.8	**84.09**

Table 7 — The average accuracy of the proposed model

S. No.	State	Dataset	MFCC	MFSC	LPC
1.	Thinking	KaraOne	38.0	**74.92**	41.0
2.	Speaking	KaraOne	37.0	38.0	**75.85**
3.	Thinking	FEIS	40.0	41.0	**86.2**
4.	Speaking	FEIS	39.1	40.1	**84.54**

performance metrics, including Precision, Recall, and F-measure, are detailed in the tables for a comprehensive evaluation of classifier performance.

The above Table 3 presents a comprehensive performance analysis of the thinking state using the KaraOne dataset, considering both feature extraction techniques and classifiers. The results indicate that the MFSC feature extraction method, when combined with all three classifiers, consistently achieves higher accuracy and it is indicated in bold. Specifically, the MFSC features extracted from the thinking state demonstrate superior performance when applied to the Integrated Stacking Classifier.

Table 4 presented above, indicates the performance analysis of the speaking state using the KaraOne dataset, taking into account both feature extraction techniques and classifiers. Among the three classifiers, it is evident that the LPC feature extraction method consistently yields superior accuracy for the speaking state and it is indicated in bold. Consequently, LPC features extracted from the speaking state are employed and inputted into the Integrated Stacking Classifier for improved performance.

The above Tables 5 & 6 provide a comprehensive performance evaluation of both the thinking and speaking states using the FEIS dataset, considering various feature extraction techniques and classifiers. Notably, it is observed that the LPC feature extraction method consistently outperforms the other methods in both states and across all three classifiers and it is represented in bold. Given this observation, the features obtained from LPC are selected and employed as inputs for the Integrated Stacking Classifier, further enhancing the overall performance of the classifier for both thinking and speaking states in the FEIS dataset.

From the information provided in Table 7, it can be deduced that the Integrated Stacking classifier is utilized with specific feature combinations for both the KaraOne and FEIS datasets. For KaraOne, the input features consist of MFSC features from the thinking state and LPC features from the speaking state. Similarly, for the FEIS dataset, LPC features from both thinking and speaking states are used as input for the Integrated Stacking classifier. Notably, this classifier yielded superior accuracy when applied

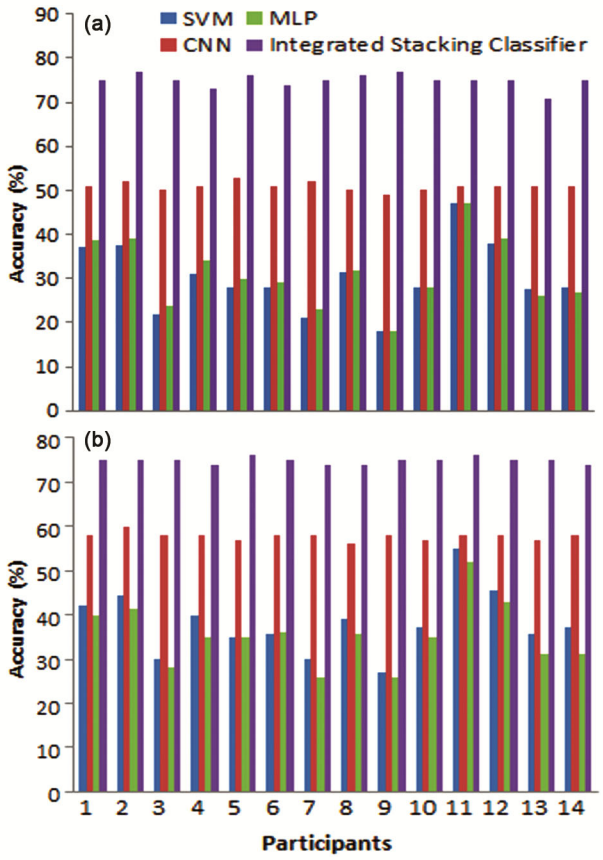

Fig. 5 — Performance evaluation for: (a) Thinking state over KaraOne dataset using MFSC feature extraction, (b) Speaking state over KaraOne dataset using LPC feature extraction

to MFSC features from the thinking state and LPC features from the speaking state compared to other feature combinations. Furthermore, in the case of the FEIS dataset, the LPC features from both thinking and speaking states outperformed other classifiers in terms of performance metrics, including Precision, Recall, F-measure, and Accuracy. These performance evaluations are documented in the ablation study.

In Figs 5 & 6, the performance accuracy of various classifiers was observed, including SVM, CNN, MLP, and the Integrated Stacking Classifier, evaluated on both the KaraOne and FEIS datasets for the thinking and speaking states. The experimental setup involves retrieving features from the best classifier for individual participants and aggregating these features

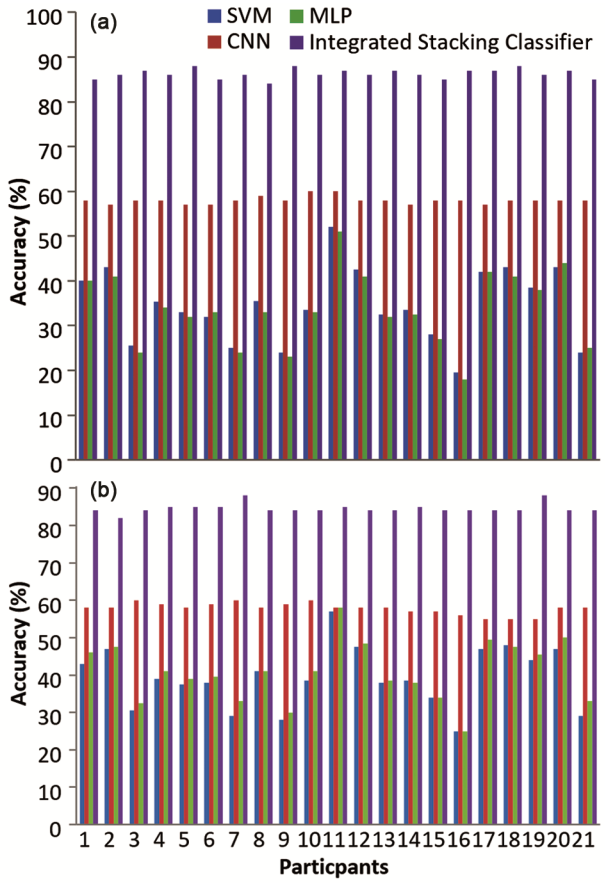

Fig. 6 — Performance evaluation for: (a) thinking state over FEIS dataset using LPC feature extraction (b) Speaking state over FEIS dataset using LPC feature extraction

for all participants, which are then fed into the Integrated Stacking Classifier. The key findings from these figures and experiments are as follows, For the KaraOne dataset: In the thinking state, MFSC provides the best features. In the speaking state, LPC yields superior features. For the FEIS dataset: LPC consistently produces the best features for both thinking and speaking states. The graphical representations in Figs 5 & 6 demonstrate that the proposed model achieves higher accuracy compared to other classifiers. Additionally, various performance metrics, such as Precision, Recall, F-Measure, and Accuracy, are tabulated in the ablation study to provide a more comprehensive assessment of classifier performance.

The Fig. 5(a) illustrates the performance analysis of the proposed model for the thinking state using the KaraOne dataset. Based on the insights drawn from the ablation study and Table 4, it is apparent that among the three feature extraction methods, MFSC feature extraction consistently yields better accuracy compared to the other two traditional classifiers. Additionally, in reference to Table 3 and the observations made from Fig. 5(a), it is evident that the Integrated Stacking Classifier consistently outperforms other classifiers for all participants in the thinking state. This suggests that the proposed model, when utilizing MFSC features, offers superior performance in classifying the thinking state using the KaraOne dataset.

The Fig. 5(b) displays the performance evaluation for the speaking state using the KaraOne dataset. Drawing insights from the ablation study and Table 5, it is evident that among the three feature extraction techniques, LPC consistently delivers better accuracy compared to the other two methods for the speaking state. Furthermore, referring to Table 3 and the observations made from Fig. 5(b), it is clear that the Integrated Stacking Classifier outperforms other traditional classifiers for the speaking state in the KaraOne dataset. This suggests that the proposed model, when utilizing LPC features, excels in classifying the speaking state in this dataset.

The provided Fig. 6(a) illustrates the performance evaluation for the thinking state using the FEIS dataset. Drawing insights from the ablation study and Table 6, it is evident that among the three feature extraction techniques, LPC consistently yields better accuracy when applied to the three classifiers for the thinking state. Additionally, referring to Table 3 and the observations made from Fig. 6(a), it is clear that the Integrated Stacking Classifier consistently offers superior accuracy compared to other traditional classifiers when classifying the thinking state in the FEIS dataset. This suggests that the proposed model, especially when utilizing LPC features, excels in this classification task.

The provided Fig. 6(b) showcases the performance evaluation for the speaking state using the FEIS dataset. Drawing insights from the ablation study and Table 7, it is evident that among the three feature extraction techniques, LPC consistently achieves better accuracy across the three classifiers for the speaking state. Additionally, referring to Table 3 and the observations derived from Fig. 6(b), it is clear that the Integrated Stacking Classifier consistently outperforms other traditional classifiers when classifying the speaking state in the FEIS dataset. This suggests that the proposed model, particularly when utilizing LPC features, excels in this classification task. Furthermore, considering the information from Tables 4, 5, 6, and 7, it can be inferred that the best

features are inputted into the proposed Integrated Stacking Classifier. The results indicate that the proposed model achieves an accuracy of 74.92% using MFSC features from the KaraOne dataset for the thinking state, and for the speaking state, it achieves 75.85% using LPC features from the same dataset. Moreover, for the FEIS dataset, the proposed model achieves impressive accuracy, obtaining 86.2% for the thinking state and 84.54% for the speaking state, both using LPC features.

Conclusions

The proposed model aims to address phoneme classification for individuals with communication disabilities by selecting the most effective features extracted from EEG signals in the context of the Silent Speech Interface. This effort involves utilizing multiple classification algorithms to extract the best features from brain signals. A novel classification approach is introduced, which selects the best features based on individual participant classifier results and feeds these features into the proposed method to achieve enhanced performance, specifically for the thinking state using MFSC features from the KaraOne dataset and for the speaking state using LPC features from the same dataset. For the FEIS dataset, LPC features yielded impressive results for both thinking state and speaking state. In future work, enhancements could be made by employing novel optimization or feature selection algorithms to further improve phoneme classification performance. Additionally, conducting cross-subject analysis may provide valuable insights and potentially lead to even better results.

References

1 Sensinger J W & Dosen S, A review of sensory feedback in upper-limb prostheses from the perspective of human motor control, *Front Neurosci*, **14** (2020) 1–24, https://doi.org/10.3389/fnins.2020.00345.

2 Liu Q, Jiao Y, Miao Y, Zuo C, Wang X, Andrzej C & Jin J, Efficient representations of EEG signals for SSVEP frequency recognition based on deep multiset CCA, *Neurocomputing*, **378** (2020) 36–44, https://doi.org/10.1016/j.neucom.2019.10.049.

3 Chuan-Chih H, Chia-Lung Y, Wai-Keung L, Hao-Teng H, Kuo-Kai S, Lieber Po-Hung Li , Tien-Yu Wui & Po-Lei Lee, Extraction of high-frequency SSVEP for BCI control using iterative filtering based empirical mode decomposition, *Biomed Signal Process*, **61** (2020) 1–12, https://doi.org/10.1016/j.bspc.2020.102022.

4 Wookey, Jessica, Busra, Bong, Azizbek & Suan, Biosignal sensors and Deep learning based speech Recognition – A Review, *Sensors*, **21** (2021) 1–22, https://doi.org/10.3390/s21041399.

5 Bhuvaneshwari M, Kanaga E G M & Anitha J, Bio-inspired Red Fox-Sine cosine optimization for the feature selection of SSVEP-based EEG signals for BCI applications, *Biomed Signal Process Control*, **80** (2022) 1–12, https://doi.org/10.1016/j.bspc.2022.104245.

6 Edla D R, Dodia S, Bablani A & Kuppili V, An efficient deep learning paradigm for deceit identification test on EEG Signals, *ACM Trans Manag Inf Syst*, **12** (2021) 1–20, https://doi.org/10.1145/3458791.

7 Mini P P, Tessamma T & Gopikakumari R, Wavelet feature selection of audio and imagined/vocalized EEG signals for ANN based multimodal ASR system, *Biomed Signal Process Control*, **63** (2021) 1–11, https://doi.org/10.1016/j.bspc.2020.102218.

8 Sharon R A & Murthy H, Correlation based Multi-phasal models for improved imagined speech EEG recognition, *Workshop on Speech, Music and Mind* (2020) 21–25, https://doi.org/10.21437/smm.2020-5.

9 Clayton J, Wellington S, Valentini-Botinhao C & Watts O, *Decoding imagined, heard and spoken speech: Classification and regression of EEG using a 14-channel dry-contact mobile headset* (International speech communication Association) 2020, 4886–4890, https://doi.org/10.21437/Interspeech.2020-2745.

10 Mansoor A, Usman M W, Jamil N & Naeem & M A, Deep learning Algorithm for Brain Computer Interface, *Sci Programm*, **2020** (2020) 1–12, doi:10.1155/2020/5762149.

11 Dash D, Ferrari P & Wang J, Decoding imagined and spoken phrases from non-invasive neural (MEG) signals, *Front Neurosci*, **14** (2020), doi: 10.3389/fnins.2020.00290.

12 Rusnac A L & Grigore O, Imaginary speech recognition using a convolutional network with long-short memory, *Appl Sci*, **12(22)** (2022), 1–20, https://doi.org/10.3390/app122211873.

13 Sharon R A, Narayanan S S, Sur M & Murthy A H, Neural Speech decoding during audition, imagination and production, *IEEE Access*, (2020), 149714–149727, doi: 10.1109/ACCESS.2020.3016756.

14 Saha P, Abdul-Mageed M & Fels S, Towards imagined speech recognition with hierarchical deep learning, *arXiv: 1904.05746v1*, (2019) 1–5.

15 Mahapatra N C & Bhuyan P, Multiclass classification of imagined speech vowels and words of electroencephalography signals using deep learning, *Adv Hum Comput Interact*, **2022** (2022) 1–10, doi:10.1155/2022/1374880.

16 Risanuri H, Agus B, Sujoko S & Anggun W, Denoising speech for MFCC feature extraction using wavelet transformation in speech recognition system, *10th Int Conf Info Technol Electr Eng* (IEEE) 2018, 1–5, doi: 10.1109/ICITEED.2018.8534807.

17 Gupta H & Gupta D, LPC and LPCC method of feature extraction in speech recognition system, *Int Conf Cloud System Big data Eng* (IEEE) 2016, 1–5, doi: 10.1109/confluence.2016.7508171.

18 Ahmad J & Hayat M, MFSC: Multi-voting based feature selection for classification of Golgi proteins by adopting the general form of Chou's PseAAC components, *J Theor Biol*, **463** (2019) 1–29, https://doi.org/10.1016/j.jtbi.2018.12.017.