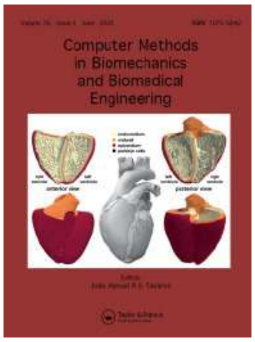

Computer Methods in Biomechanics and Biomedical Engineering

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/gcmb20

EBi-LSTM: an enhanced bi-directional LSTM for time-series data classification by heuristic development of optimal feature integration in brain computer interface

Mala Saraswat & Anil Kumar Dubey

COMPUTER METHODS IN BIOMECHANICS AND BIOMEDICAL ENGINEERING
https://doi.org/10.1080/10255842.2023.2187662

Taylor & Francis
Taylor & Francis Group

Check for updates

EBi-LSTM: an enhanced bi-directional LSTM for time-series data classification by heuristic development of optimal feature integration in brain computer interface

Mala Saraswat[a] and Anil Kumar Dubey[b]

[a]Assistant Professor, School of Computing Science and Engineering, Bennett University, Noida, India; [b]Associate Professor, CSE Department, ABES Engineering College Ghaziabad, Ghaziabad, India

ABSTRACT
Generally, time series data is referred to as the sequential representation of data that observes from different applications. Therefore, such expertise can use Electroencephalography (EEG) signals to fetch data regarding brain neural activities in brain–computer interface (BCI) systems. Due to massive and myriads data, the signals are appealed in a non-stationary format that ends with a poor quality resolution. To overcome this existing issue, a new framework of enhanced deep learning methods is proposed. The source signals are collected and undergo feature extraction in four ways. Hence, the features are concatenated to enhance the performance. Subsequently, the concatenated features are given to probability ratio-based Reptile Search Algorithm (PR-RSA) to select the optimal features. Finally, the classification is conducted using Enhanced Bi-directional Long Short-Term Memory (EBi-LSTM), where the hyperparameters are optimized by PR-RSA. Throughout the result analysis, it is confirmed that the offered model obtains elevated classification accuracy, and thus tends to increase the performance.

ARTICLE HISTORY
Received 19 October 2022
Accepted 1 March 2023

KEYWORDS
Time-series data classification; brain–computer interface; electroencephalography signal; ensemble feature extraction; probability ratio-based reptile search algorithm; enhanced bi-directional long short-term memory

1. Introduction

With the tremendous growth of information technology, the prevalence of sensors has been less affordable over the recent few years. The massive amount of large-scale dimensional time series data is to be gathered from various fields like finance, bioinformatics, etc. (Yeomans et al. 2019). While considering the broadcast range of practical implications, multivariate time series is needed to be classified perfectly as well as appropriately. Nowadays, it is used in the healthcare industries and activity recognition process. Furthermore, the users (Chambon et al. 2018) can easily interpret the computer devices or system, termed a brain–machine interface. Here, the machine eases to forecast the prime objectives of the cognitive state of human brain activities that assist by taking the EEG signals. It is also named BCI (Abenna et al. 2022), where it employs Artificial Intelligence (AI)-aided techniques that are used to harness the brain signals to achieve instrumental gadgets like wheelchairs, speller programs, virtual games, and some other assistive devices (Altaheri et al. 2021). Additionally, the Hilbert transform is mainly used to represent the temporal traits of the EEG signal that helps to maintain the energy level of the brain state. Though it performs beneficially, novel temporal representation is taken for the EEG signal, where it is intact with temporal features (Wang et al. 2020).

With the advancement technique, the BCI constructs with standard methodologies to make a connection establishment between computer devices and the human brain (Radman et al. 2021). The signal commands are to be decoded by the task-based neural activities in BCI based on EEG signal records. Thus, the BCI is more reliable for physically impaired people to easily communicate with the devices and external factors (Amin et al. 2019). Among the EEG signals, Motor imagery (MI) is an essential role in MI-based schemes. The MI information of EEG signals are portraying the image motion of the human body parts, but it is not physically moving (Roy 2022). The MI-assisted BCI techniques are implemented that consider the imaginary movement of feet and hands of every person (Guger et al. 2000). For the past few years, machine learning algorithms are immensely entailed to recognize the oscillatory activity of brain signals (Lv et al.

CONTACT Anil Kumar Dubey anildudenish@gmail.com

2021). The health data is taken from each individual that contains subject variations and physical constraints. It leads to giving an open platform for further improvement in terms of increasing the efficiency (Mini et al. 2021) of the system. Hence, the AI-assisted Deep Neural Networks (DNNs) are examined to determine the healthcare data in BCI models (Nakra and Duhan 2022).

Deep learning is an extended version of machine learning approaches, which are used for implementing Natural Language Processing (NLP) and computer device implications. The main drawback of using EEG signal possesses insufficient data samples for recognizing the activities by the computer system as well as NLP. In general, the recognition is processed with two significant steps feature extraction and classification (Coyle et al. 2005; Dose et al. 2018). For feature extraction, the Common Spatial Pattern (CSP) is classically utilized to retrieve the informative features (Huang et al. 2022). Subsequently, the development of time series data classification model is critical and challenges one to make a response for MI. Some experts have used the Support Vector Machine (SVM) (Tyagi and Nehra 2016) for EEG signal classification in MI tasks. Rather than other approaches, a deep Convolutional Neural Network (CNN) is framed with 210 EEG source data along with cortical regions of primary motor and premotor. Due to the admittance of the convolution network, it renders a promising performance for classifying (Chacon-Murguia and Rivas-Posada 2022) the time series data. Thus, the novel approach is inferring the concept of architectural representation, usage of activation functions, regularization, and optimization processes.

The major aspects of the study are depicted here.

- To implement the latest framework for time series data classification in BCI using an improved heuristic approach and enhanced deep learning technique for delivering better analysis of the brain neural activities.
- To perform the ensemble feature extraction by utilizing four various models. The four distinct feature set is acquired via temporal features, RBM features, CNN features, and autoencoder features. Finally, these features are merged to increase the accuracy value.
- To fetch the optimal features from the resultant fused feature set using a novel PR-RSA method. The selection of optimal features reduces the false rate and dimensional issue and thereby enhancing the classification performance.
- To frame the novel EBi-LSTM for classification purposes. The constraints like several suitable hidden neuron counts and epochs are optimally attained with the help of the proposed PR-RSA for improving the efficacy of the model.
- To analyze the simulation results using different measures. The statistical and comparative analysis is done for the proposed model over classical heuristic and deep structured architectures.

The residual modules of the work are described below. Section 2 provides an investigation of conventional approaches. The description of dataset collection and general information on BCI is given in Section 3. Section 4 illustrates the optimal feature selection using the novel heuristic algorithm in BCI. The demonstration of a novel deep learning approach is provided in Section 5. The result visualization is given in Section 6. In the end, the paperwork is concluded in Section 7.

2. Existing works

2.1. Related works

Sakhavi et al. (2018) have implemented a CNN architecture-aided classification method for MI-related data. It was accomplished by proposing the latest temporal presentation of data. For classification, it has also employed the CNN approach. Hence, a significant representation was created by altering the filter bank relied on spatial features of data. Hence, the given task has achieved better-classified outcomes that were to be enhanced by up to 7%. It has also exhibited the temporal traits of signal that has delivered accurate value.

Vega et al. (2022) have explored the Temporal Convolutional Networks (TCNs) using EEG signals. The novel method was constructed with Long Short-Term Memory (LSTM), which was associated with Fuzzy Neural Block (FNB). In FNB, the fuzzy components were helpful to resist noisy data. Three various architectural representations were utilized to build the BCI. The outcomes of the suggested EEG-TCFNet model have exploited the classification rate as 98.6% concerning the dependent and independent nature of subject strategy, correspondingly. The FNB was used in all three diverse architectures. Therefore, the outcomes have ensured that the improved method has provided an efficient performance.

Ieracitano et al. (2021) have considered the cortical signal representation of EEG for data classification. It has also been used for performing the inverse issue *via* beamforming. The two ways of classification were conducted through deep CNN, where it has composed of 'pre-Hand Close (HC) versus Resting State (RE) and pre-Hand Open (HO) versus RE'. Hence, the simulated results have demonstrated that the improved approach has acquired the appropriate value to do better classification.

Jin et al. (2020) have deployed the Extreme Learning Machine (ELM) for data classification using MI-based EEG recordings. To resolve the existing challenges and enhance the performance, it has also proposed the Sparse Bayesian ELM (SBELM). The beneficial traits of ELM were superimposed with a sparse Bayesian approach. It was used to resolve the complexities and evade the irrelevant hidden neuron counts. The improvements in the method were estimated with publicly available data sources. Thus, the suggested method has acquired the desired value for data classification in BCI systems.

Maweu et al. (2021) have explored the Guided Evolutionary Synthesizer (GES) for time series data classification. The proposed work has influenced the genetic algorithm to perform the data classification process. Other conventional learnings and network models were taken to validate the efficiency of the suggested method. Further, the empirical value was obtained by considering the eight different experimental databases. Here, one of the methods a Residual Network (ResNet) was used. The two different kinds of recordings were collected as EEG and electrocardiogram (ECG). Further, the GES model was trained with synthetic data along with perturbed data. Finally, the performance was estimated and tested with other classical techniques. Thus, the outcomes have illustrated that it has achieved better classification performance.

Sharma et al. (2022) have influenced the multi-layered perceptron (MLP) approach for time series data classification in BCI devices. During the process of MI tasks, SVM was used to lessen the training time. Once the implementation was to be done, a higher classification rate of 90% was obtained. To further increase the results, optimization was used to reach the impressive value. Finally, the given designed method has ensured better data classification performance.

Zheng et al. (2016) have implemented the 'Multi-Channels Deep Convolutional Neural Networks (MC-DCNN)' for multivariate time series data classification. Initially, this technique has learned the features of univariate series data of every channel. At the final stage, it has combined altogether to do the multivariate classification. Further, the relevant features were fed as input to the MLP for data classification. Lastly, the empirical outcomes were executed and their results were analyzed. Thus, the outstanding value was yielded to satisfy the classification of multivariate time series.

Amin et al. (2019) have recommended the various layers of the CNN model. It has also considered the multiple-level features that were integrated. The employed CNN method has extracted the spectral as well as temporal attributes of raw data of EEG recordings. Finally, the validation was implemented, and its outcomes were validated. Thus, the effective results were aided to maximize the classification accuracy for time series data.

Jin et al. (2021) have implemented the latest time filter by promoting the temporally local weighting into the fitness function of the Task-Related Component Analysis (TRCA)-aided approach with the singular value decomposition. It has been utilized to perform a robust similarity metric to enhance recognition capability. At last, the empirical outcomes of the recommended method have demonstrated its effective performance.

Jin et al. (2021) have developed a fusion algorithm according to the Dempster–Shafer theory, with a consideration of the issuing of attributes. The author has used two types of competition data sets, where first validate the efficiency of the enhanced fitness functions regarding categorization embeddability, feature distribution, and accuracy. Finally, the comparison has been taken with other feature selection approaches which were executed in both computational time and accuracy.

Li et al. (2018) explored the special effects of audiovisual inputs in an experimented cocktail party. Here, auditory-only stimuli (voices) and visual-only (faces) were constructed by acquiring the auditory and visual aspects from the synthesized audiovisual stimulation. The neural presentations of the emotion attributes were assessed by computing and decoding accuracy rate and brain pattern-based reversibility index according to the fMRI data.

2.2. Research gaps and challenges

The temporal or time series data are calculated from human beings, which has the limitation of subject difference as well as physical constraints. Due to this, it has difficult to find a unique data generation model effectively and efficiently. Thus, EEG is commonly used in BCI to resolve the issue, which has the main aspect of the high temporal solution, ease to use, and accurate monitoring. Numerous studies are discussed in Table 1 about existing works of time series data classification in BCI. CNN (Sakhavi et al. 2018) provides higher results rather than the existing method and it has less time consumption. However, it does not consider the issue of a small number of samples. Also, it causes overfitting. CNN and Fuzzy Neural Networks (Vega et al. 2022) acquire high decoding accuracy. Moreover, due to the fixed size of the input, it does not outperform the better results. CNN and deep learning (Ieracitano et al. 2021) attain better classification results. Due to the larger number of electrodes, it leads to an inverse problem solution, and it has the limitation of using EEG epochs. ELM and SVM (Jin et al. 2020) obtain more generalization capacity and improve performance. But the limitation of this research is that it utilizes only a single band frequency, and it lacks to examine the different types of sparse prior distributions. DNNs and GES (Maweu et al. 2021) provide better performance with synthetic data and enhanced the better classification rate. Due to the imperfection of samples, overfitting occurs. SVM and machine learning (Sharma et al. 2022) acquire better classification results. However, it cannot be used for real-time applications. K-NN classification (Zheng et al. 2016) obtains better feature representation of time series data. But, it needs more time-consuming and it does not give the solution since it has constant parameters. CNN and multilevel feature extraction (Amin et al. 2019) achieves higher sensitivity and better accuracy value. Due to the multilevel feature, it enhances classification accuracy. But, it only processes with a less quantity of datasets. The researchers found that it mitigates the performance by some artifacts, less spatial resolution, and invariable data.

To facilitate this issue, it is needed to develop a new method to classify the temporal data for BCI using deep learning techniques. The latest PR-RSA-EBi-LSTM method is promoted in this work for optimizing the constraints of deep learning algorithms. It is utilized to resolve overfitting issues and also it is feasible for larger datasets. The developments of the given offered method are helpful for real-time applications such as clinical and medical applications. The simulation result of the offered time-series data classification model proved that it attains elevated precision and accuracy rate and also it resolves the cross-validation issues. The validation of statistical evaluation on the offered method reveals that it is statistically significant.

Table 1. Superiorities and downsides of time-series data classification for BCI.

Author (citation)	Framework	Superiorities	Downsides
Sakhavi et al. (2018)	CNN	• It provides higher results rather than the existing method. • It has less time consumption.	• It does not consider the issue of a small number of samples. • Overfitting occurs.
Vega et al. (2022)	CNN and Fuzzy Neural Network	• It acquires high decoding accuracy.	• Due to the fixed size of the input, it does not outperform the better results.
Ieracitano et al. (2021)	CNN and deep learning	• It attains better classification results.	• Due to the larger number of electrodes, it leads to an inverse problem solution. • It has the limitation of using EEG epochs.
Jin et al. (2020)	ELM and Support Vector Machine (SVM)	• It obtains more generalization capacity. • It improves performance.	• It utilizes only a single band frequency. • It lacks to examine the different types of sparse prior distributions.
Maweu et al. (2021)	Deep neural networks (DNNs) and GES	• It provides better performance with synthetic data. • It attains a better classification rate.	• Overfitting occurs by the imperfection of samples.
Sharma et al. (2022)	SVM and machine learning	• It acquires better classification results.	• It is not suitable for real-world applications.
Zheng et al. (2016)	k-NN classification	• It obtains better feature representation of time series data.	• It needs more time consumption. • It does not give the solution since it has constant parameters.
Amin et al. (2019)	Convolution Neural Networks and Multilevel feature extraction	• It achieves higher sensitivity and better accuracy value. • The multilevel feature enhances classification accuracy.	• It only processes with a less quantity of datasets.

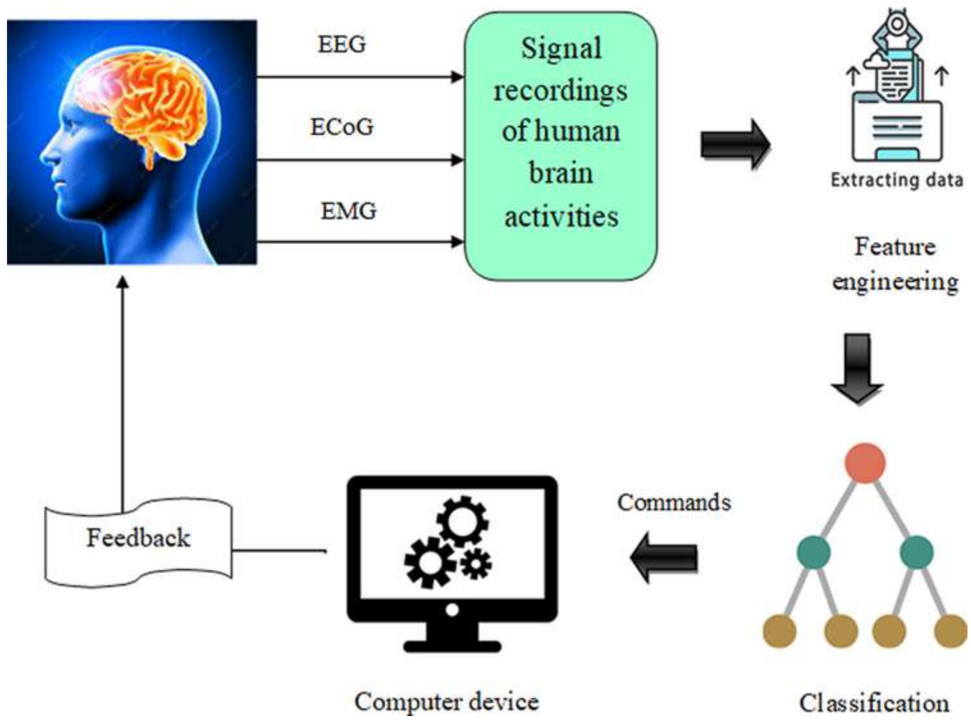

Figure 1. General flow diagram of BCI systems.

3. Time series data classification in the brain–computer interface: data collection procedure

3.1. Brain–computer interface data

The natural paradigm behavior of BCI is to receive the signals of the human brain, which are converted into understandable commands for machines. Over the past few years, the BCI becomes the hotspot area in research and development. While taking into consideration of brain activities, the EEG signal is intensely involved with the interaction process. The development of BCI entails various steps such as signal acquisition, feature identification, and finally classification. Traditionally, BCI is differentiated into three major types 'Invasive BCIs, Partially invasive BCIs, and Non-invasive BCIs'. Most of the implemented model is focussed on the EEG signal-based data. This kind of signal is mainly used in classification to forecast the human behavior of data. With the non-invasive nature of the signal, it helps to detect the various features of signals emitted from the human brain. Other than EEG, the research works use Electrocorticography (ECoG) signals, Electromyography (EMG) signals, and so on.

Merits of using BCI data:

- These devices can assist in the medical industry while diagnosing patients that retrieve brain activities concerning neurological disorders.
- It is widely used and applicable in a rehabilitation center, where physically or mentally impaired people can easily communicate with other individuals
- Due to this interface, the message or activities are appropriately conveyed to the corresponding scholars or medical specialists.
- It is also helpful for monitoring the patients by the devices in an adhesive manner, where the information can be fetched.
- It can also provide an excellent cognitive state of the individuals through the feedback response from the system.

Figure 1 illustrates the general representation of BCI systems.

3.2. Experimented dataset description

The input signals are fetched from the given link as 'https://www.kaggle.com/competitions/neuroml2020eeg/data: Access Date: 2022-10-06'. It is named 'Brain or Neural Computer Interaction (BNCI): Horizon 2020'. The natural paradigm in BCI is to receive the signals of the human brain, which are converted into understandable commands for machines. It is a freely accessible data source, which is comprised of 29 different classes. Every class consists of participants and signals that are in form of EEG, EOG, fNRIS, EMG, etc. This database entails most of the signal recordings as EEG.

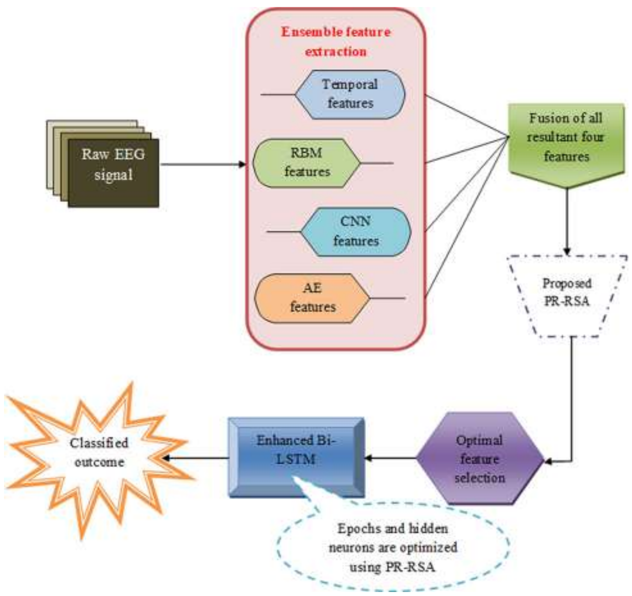

Figure 2. Architectural representation of suggested time series data classification in BCI using the enhanced deep learning method.

Thus, the collected signals for our proposed system are indicated by E_z, where $z = 1, 2, 3, \ldots, Z$, in turn, Z denotes the total gathered signals.

3.3. Proposed time series data classification in BCI

Since the 21st era, the BCI is an essential and significant process in research and development areas of its extensive growth. In general, BCI is defined as the communication establishment between the human brain and computer devices. It is not required to depend on the neuro-muscular activities of human individuals as it leads to many beneficial traits to design a BCI. Hence, it is emerging as an intense idea of communication in a virtual perception of devices or machines. Yet, to implement an effective model, such techniques have the constraints of achieving more robustness and accuracy. Here, the time series manner of data or signals is important to deploy the method. Therefore, several scholars have focused on the classification of time series data in BCI systems (Ambati and El-Gayar 2021). The two most concerns are feature classification and extraction procedure. In all the deployed methods, after the data acquisition is done, the finding of noteworthy features is a challenging factor. As some methods have resolved the dimensional problem in features, it still exists with performance degradation of non-optimal results. Furthermore, the data is represented in a non-stationary format; it keeps on varying to extract the features. Thus, this kind of fluctuation causes an impact on its efficiency. Also, it becomes critical to train the features in certain classifiers. Thus, the training procedure is another challenging issue. Similarly, classification plays a pivotal role in BCI data classification. Years ago, expert systems are utilized to do the classification job. Those models still open the room for further improvement as it needs more handing out time and such complexities. To combat these challenging issues, various deep learning models are employed. Though all of the techniques have provided a promising outcome, it subsists some limitations. Since the time sequential data collection, the overfitting issue leads to mitigating the robustness of the system and also has less generalization capability. This problem confines the real-time implication of BCI. The solution to overcome this issue, is a novel heuristic-aided deep learning method is offered for time series data classification that is illustrated in Figure 2.

The recommended methodology constitutes various stages as (i) Signal collection, (ii) Feature extraction, (iii) Feature concatenation, (iv) Optimal feature selection, and (v) Classification. Firstly, the necessary raw signals are to be taken from the standard datasets. Once the signals are collected, the ensemble feature extraction takes place. Here, the four extraction techniques are utilized. The first set of features is obtained by the temporal feature extraction process. The second set is acquired through the DBN-RBM layer. Simultaneously, the third feature is determined *via* CNN, and the last set of deep features is attained by using an autoencoder. Consequently, all of the resultant features are fused to form a single feature set to increase the classification rate. Due to a large amount of data, there will be an occurrence of more redundant data, and takes more training time. To facilitate this problem, the PR-RSA is proposed for optimal feature selection. Finally, these optimal features are fed as input to the EBi-LSTM model, where the constraints such as several suitable hidden neuron counts and epochs are tuned optimally by the PR-RSA approach. Due to this parameter optimization, it has the potential to reduce the complexities in structure and time. The performance is estimated and compared over other developed approaches. Therefore, the outcomes ensure that it exhibits a higher classification performance.

4. Optimal feature selection using PR-RSA for time series data classification in BCI

4.1. Proposed PR-RSA

The novel algorithm is developed by using the traditional RSA to optimize the parameters to maximize the classification rate. Rather than other optimizations, the RSA mimics the natural mannerism of crocodiles to target the prey. Here, the reptiles as 'crocodiles' are taken for optimization. The advantage of RSA (Abualigah et al. 2022) is increasing the convergence rate, rectifying the local optima issue, and performing in a gradient-free manner that tends to perfectly identify the problems and overcome them. Yet, it is not feasible for real-world data, not supported the multi-objective functionalities. In addition to this, the usage of random values leads to reducing the consistency and efficiency of the optimal value. To facilitate such issues, the latest PR-RSA is promoted by deriving the latest mathematical expression. Hence, the following phases are explaining the proposed optimization process.

Phase 1:- Initialization: Conversely, the RSA is processed with two main search stages exploration phase and the exploitation phase. The foremost step is to initialize the population in terms of candidate solutions and total iterations. Let us assume C the total crocodile population, the dimensional size of the population r, and also R the total candidate solutions. The maximum iteration S is taken. The solution is represented in the matrix format as given in Eq. (1).

$$C = \begin{bmatrix} c_{1,1} & \cdots & c_{1,r} \\ \vdots & \ddots & \vdots \\ c_{R,1} & \cdots & c_{R,r} \end{bmatrix} \tag{1}$$

Here, $c_{a,b}$ signifies the bth position of ath the solution or crocodile. It is derived using Eq. (1).

$$c_{a,b} = rd \times (upB - lwB) + lwB \tag{2}$$

The random value, upper and lower bound value is declared by rd, upB, and lwB, accordingly.

Phase 2:- Encircling the prey: The exploration process is accomplished by two ways of movement that are 'high walking and belly walking'. As the natural disturbances, the crocodiles are not able to provide the best value by the search space. Thus, it considers the wide search area for encircling the prey. To make the transition from the exploration to the exploitation phase, the total iteration is categorized into four conditions.

Therefore, the exploration phase is accomplished by taking into the account first two criteria with the aid of high and belly walking movement. The first half of the process is done by high walking by the condition of $s \le \frac{S}{4}$, whereas the second half is processed between $s \le 2\frac{S}{4}$ and $s > \frac{S}{4}$. Therefore, the location is upgraded in search space for encircling behavior. It is equated by Eqs. (3) and (4).

$$c_{a,b}(s+1) = Bt_b(s) \times -\lambda_{a,b}(s) \times \alpha - F_{a,b}(s) \times rd \quad \text{when, } s \le \frac{S}{4} \tag{3}$$

$$c_{a,b}(s+1) = Bt_b(s) \times c_{r_1,b} \times PR(s) \times rd \quad \text{when, } \frac{S}{4} < s \le 2\,\frac{S}{4} \tag{4}$$

Here, is the bth position concerning s iteration. The arbitrary value is taken among 0 and 1 that is given in rd. Over the iteration, the best value is obtained by $Bt_b(s)$, the term α uses to control the accuracy level, which is fixed as 0.1 and rd as the random value of [0, 1] (Al-Shourbaji et al. 2022). Also, the arbitrary position of ath the solution is noted as $c_{r_1,b}$, where r_1 is the random parameter that lies between 1 and R. In search space, the reducing function is declared by $F_{a,b}$, which is derived using Eq. (5).

$$F_{a,b} = \frac{Bt_b(s) - c_{(r_2,b)}}{Bt_b(s) + \gamma} \tag{5}$$

Here, the random value as r_2 contains the range of 1 to R, and the small value is denoted by γ. The hunting operator $\lambda_{a,b}(s)$ is estimated using Eq. (6).

$$\lambda_{a,b}(s) = Bt_b(s) \times G_{a,b} \tag{6}$$

Here, the term $G_{a,b}$ computes the percentage difference value among the best and current solution of bth position. It is given in Eq. (7).

$$G_{a,b} = \eta + \frac{c_{a,b} - A(c_a)}{Bt_b(s) \times (upB_b - lwB_b) + \gamma} \tag{7}$$

The term, upper bound and lower bound of bth position is indicated by upB_b and lwB_b, respectively. Consequently, the average value is calculated for all the solutions. It is denoted by $A(c_a)$ and expressed in Eq. (8).

$$A(c_a) = \frac{1}{R}\sum_{b=1}^{R} c_{a,b} \tag{8}$$

As the novel BCI system uses time series or sequential data, the baseline RSA cannot tackle the discrete data values. In Eq. (4), the term PR annotates the probability ratio. In the traditional update of RSA, the arbitrary value is used. It is also degrading the exploitation process by having a non-optimal value of probability ratio. To overcome these flaws, a novel PR-RSA is introduced, where the new formulation is done for probability ratio with a fixed value of two sensitive variables. Over the iteration, these parameters are used to provide the best value as it relies on both the exploration and exploitation phase. Thus, the new expression of PR-RSA is modeled by using Eq. (9).

$$PR(s) = 2 \times \left[\left({}^{(\alpha \times C \times S)} \Big/ {}_{2} \right) + \eta \right] * \left[1 - \left(\frac{s}{S} \right) \right] \tag{9}$$

Here, the total population, maximum iteration, and current iteration are represented by C, S, and s, accordingly. The sensitive parameter as η that is set as 0.1, whereas the value α is given as 0.005. Finally, the number '2' defines the correlation rate that is varied from 2 to 0.

Phase 3:- Hunting the prey: As the name implies, after encircling the prey, the crocodiles are in the position to attack the prey finally. The rest of the two predetermined criteria are used for the exploitation phase. For hunting behavior, it employs 'hunting coordination and cooperation'. When the condition as $s \le 3\frac{S}{4}$ and $s > 2\frac{S}{4}$, the hunting coordination is performed, whereas the hunting cooperation is accomplished between $s \le S$ and $s > 3\frac{S}{4}$. Therefore, the position updating *via* the exploitation phase is modeled by Eqs. (10) and (11).

$$c_{a,b}(s+1) = Bt_b(s) \times G_{a,b}(s) \times rd \quad \text{when, } s \le 3\frac{S}{4} \text{ and } s > 2\frac{S}{4} \tag{10}$$

$$c_{a,b}(s+1) = Bt_b(s) \times -\lambda_{a,b}(s) \times \gamma - F_{a,b}(s) \times rd \quad \text{when, } s \le S \text{ and } s > 3\frac{S}{4} \tag{11}$$

Finally, the exploration process is done only on the condition as $s > \frac{S}{2}$. On the other hand, the exploitation phase is performed when the criteria as $s > \frac{S}{2}$. Therefore, the optimal outcome is obtained to tune the parameters. The pseudo-code of offered PR-RSA is elaborated as given below.

Algorithm 1: Novel PR-RSA

Assume C as total population and R as the candidate solution
Set the parameters η and α
Consider the total count of iterations as S
Compute the probability ratio PR
Do while ($s < S$)
  Evaluate the objective function
  By Eq. (9), PR the value is determined
  Do for($a = 1$ *to* R)
    Do for($b = 1$ *to* r)
      Estimate λ, F and G by Eqs. (6), (5), and (7), respectively
      Do if ($s \le \frac{S}{4}$)
        By Eq. (3), the position is updated by high walking
      Else if ($s \le 2\frac{S}{4}$ and $s > \frac{S}{4}$)
        Updating with belly walking in Eq. (4)
      Else if ($s \le 3\frac{S}{4}$ and $s > 2\frac{S}{4}$)
        By Eq. (10), the position is updated with hunting coordination
      else
        Using Eq. (11), the position is updated with hunting cooperation
      End do if
    End do for
  End do for
  $s = s + 1$;
End do while
Obtain the finest value

The flow diagram illustration of the suggested algorithm is given in Figure 3.

4.2. Feature extraction and concatenation

After the source signal is collected, it is undergone feature extraction. The feature extraction procedure is a significant step in a feature selection process. Here, the four types of feature extraction processes are utilized for extracting the features such as temporal features, DBN-RBM features, CNN features, and autoencoder features. In general, the temporal signals hold a limited duration which is transformed into a digital format. Here, the features are extracted directly from the temporal sequence. Moreover, DBN-RBM features are utilized to extract the features in an effective manner, which makes the process easier way. The Euclidean Distance is promoted to measure the similarities between images. Accordingly, CNN features provide astounding results which are utilized to enhance the prediction performance. It is also termed a feature extraction network where it automatically extracts the features. Finally, the Autoencoder can be used as a data preparation approach to perform feature extraction, which helps to train diverse expert systems. By considering these four unique features in the classification phase, enhanced and accurate results can be obtained through this research work. Here the most informative features are acquired through four different extractive processes that are explained below.

Temporal features (Miao et al. 2017): The temporal feature plays a pivotal role while considering the signal as input. From the name itself, it mainly extracts the time domain-related feature information from the gathered input signals for analyzing brain activities. As the signal is represented in time and frequency nature, the temporal features are defined to

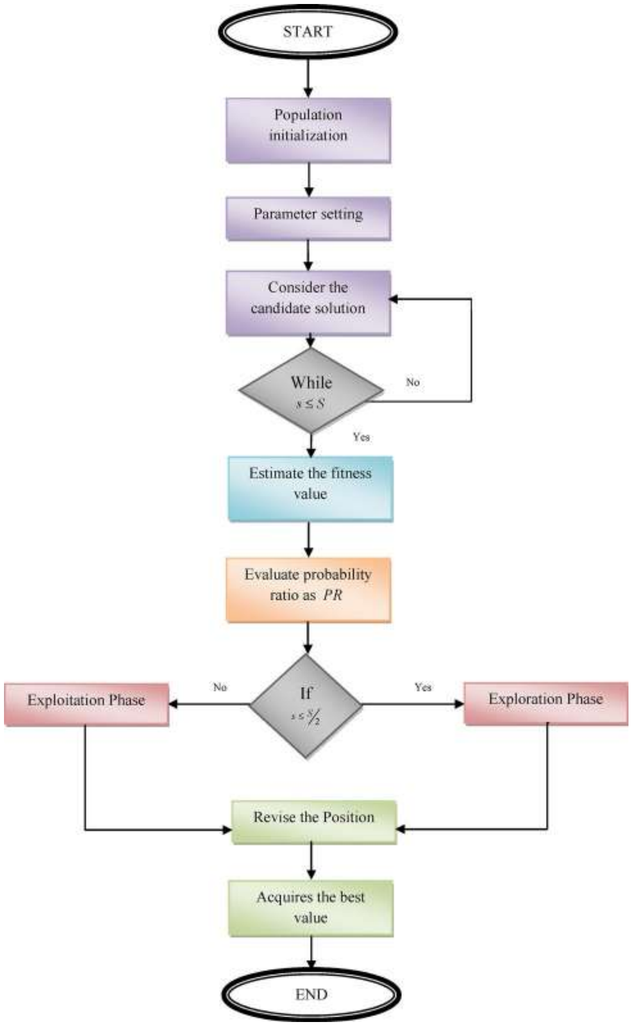

Figure 3. Flow diagram of PR-RSA.

retrieve the signal information regarding the time metric. The signals are collected from electrodes, it has appeared in time format as well. Therefore, the input signal as E_z is given, where the time-related or temporal features are extracted. It is represented using F_f^{Temp}.

DBN-RBM features (Cortez et al. 2020): DBN is one kind of deep learning model for extracting the features. This model is constructed with many stochastic layers and unsupervised layers as RBMs, where the RBM layers are placed one with another. Conversely, the DBN is processed in two different ways as unsupervised learning of every RBM layer and training the entire network. The major functionalities are done *via* the RBM layer, where the features are also obtained. Here, the input as E_z given to find the relevant features.

The RBM is commonly represented as the simple network type. The hidden layer of the first RBM can act as an input layer in succeeding RBM. It is mainly dependent on the visible and hidden neuron units. Assume, x and y as the visible and hidden neurons of the raw signal as E_z to determine the features. Furthermore, the weight z is taken as the weight matrix that makes the connection between visible and hidden neurons. Hence, the energy of joint configuration is expressed using Eq. (12).

$$G(x,y) = -\sum_{p=1}^{X}\sum_{q=1}^{Y} x_p y_q w_{pq} - \sum_{p=1}^{X} i_p x_p - \sum_{q=1}^{Y} j_q y_q \tag{12}$$

Here, the total count of visible and hidden neurons is represented by X and Y, respectively. Also, the two bias terms are indicated by i and j, correspondingly. The expression of the log probability function in RBM is represented with a weight matrix. It is given in Eq. (13).

$$\frac{\partial \log r(x)}{\partial w_{pq}} \infty w_{pq} = \mu\left[\{x_p y_p\}_{dta} - \{x_p y_p\}_{rcnst}\right] \tag{13}$$

The term μ refers to the learning rate and the training data and reconstructed data is noted as $\{x_p y_p\}_{dta}$ and $\{x_p y_p\}_{rcnst}$, accordingly. Further, $r(x)$ the term is used to compute the probability value for visible units with the summation of hidden units. Thus, it is shown in Eq. (14).

$$r(x) = \frac{1}{F}\sum_{y} e^{-G(x,y)} \tag{14}$$

Hence, the RBM features are obtained and it is represented as F_f^{RBM}.

CNN features (Ieracitano et al. 2021): Rather than other models, CNN is used generously to retrieve the feature information for performing the classification task. The CNN is composed of multiple layers as an activation layer, pooling layer, and convolution layer, which is followed by fully connected layers. The convolution layers are processed with a set of L filters, where the convolution operation takes place with the input map E_z as the size of $e_1 \times e_2$. Further, the filter size is mentioned by $l_1 \times l_2$, which is to be performed with weights and strides. Hence, the operation is also mentioned with the feature maps containing the size as $m_1 \times m_2$. It is formulated using Eqs. (15) and (16).

$$m_1 = \frac{e_1 - l_1 + 2 \times pp}{t} + 1 \tag{15}$$

$$m_2 = \frac{e_2 - l_2 + 2 \times pp}{t} + 1 \tag{16}$$

In the above two equations, the padding parameter is marked by pp. In addition to this, the Rectified Linear Unit (ReLU) is deployed to attain good generalization ability and training process. The pooling layer is used to do the downsampling operation concerning the preceding layer. After the activation function, the output of the model is obtained by Eqs. (17) and (18) in a reduced format.

$$o_1 = \frac{e_1 - \bar{l}_1}{\bar{t}} + 1 \tag{17}$$

$$o_2 = \frac{e_2 - \bar{l}_2}{\bar{t}} + 1 \tag{18}$$

Finally, the CNN ends the extraction process, where the resultant feature is indicated as F_f^{CNN}.

Autoencoder features (Mirzaei and Ghasemi 2021): It is one variant of neural networks that are processed with an encoder and decoder concept. It is utilized to retrieve the informative features for performing the classification job. The encoder and decoder are the two main blocks to developing the autoencoder network. In the first process, the encoder aims to transform the high-dimensional features into a less-dimensional feature representation. On the second decoder, it is again converting into high dimensional features and the input and output are driving to learning the autoencoder. The mathematical model of the autoencoder is given as follows.

Like a neural network, it contains three main regions as 'input, hidden, and output'. The first block is called an encoder, which combines all the latent features of the input signal as E_z ends with a hidden layer process. It is given in Eq. (19).

$$E_{hl} = \eta(\Im(E_z)) \tag{19}$$

Here, the hidden layer is declared by E_{hl}. Further, the expression of the decoder is represented in Eq. (20).

$$E_{ot} = \eta(\Im(E_{hl})) \tag{20}$$

To estimate the best value, the loss function can be calculated between the input and output layer using Eq. (21).

$$LF = \|E_{ot} - E_{hl}\|_2 \tag{21}$$

Hence, the deep features are yielded and it is indicated as F_f^{AE}. With the achievement of four various features, it is concatenated then to find the optimal features. Thus, the concatenation is happened by the annotation of $FF_g = \left\{F_f^{Temp}, F_f^{RBM}, F_f^{CNN}, F_f^{AE}\right\}$. The total numbers of features extracted from the above 4 kinds of methods are 71. It is used for the further subsequent section as optimal feature selection.

4.3. Optimal feature selection

The fused feature FF_g is taken as input in this section. Conversely, feature selection is the process of mitigating the total input variables while implementing the model. Since this model fetches some noteworthy features for classification, it contains the limitation of dimensional issues. Owing to high dimensionality, it impacts structural or computation complexities; also it requires more training time and performance degradation. To alleviate such problems, it determines only the optimal features for the learning model. The selection of accurate features is useful to evade the falsely retrieved features, thus it sustains the true positive value. Therefore, the accurate features are to be chosen by the PR-RSA, and the resultant features are indicated by FF_s^{opt}, which is used for the EBi-LSTM model. Figure 4 visualizes the optimal feature selection using the PR-RSA method.

5. Implementing enhanced bidirectional LSTM for intelligent time series data classification model

5.1. Bi-LSTM framework

As the name implies, the Bi-LSTM (Alwasiti et al. 2020) is constructed with two LSTMs, where one LSTM is used for forward propagation and the other LSTM is aided for backward propagation. Since LSTM consists of different gates such as 'memory cell, input gate forgets gate, and output gate'. The foremost cell as a memory unit is used for storing data over a long time manner. The rest of the gates are doing their classifier process. Here, the optimal features as FF_s^{opt} is given as input. The numerical model of Bi-LSTM is explored below.

The input as the optimal feature is processed *via* the Bi-LSTM in three ways. Firstly, the sigmoid function is employed to select which kind of feature is taken from the cell state. This process is done by forgetting the gate. It is given in Eq. (22).

$$fg_n = \sigma\left(w_{fg} \cdot \left[hl_{n-1}, FF_n^{opt}\right] + bs_{fg}\right) \tag{22}$$

Once it is entered, the input layer is processed with the tanh function that is to be updated in the cell state. It is expressed using Eqs. (23) and (24).

$$ig_n = \sigma\left(w_{ig} \cdot \left[hl_{n-1}, FF_n^{opt}\right] + bs_{ig}\right) \tag{23}$$

$$CS'_n = \tanh\left(w_{cs} \cdot \left[hl_{n-1}, FF_n^{opt}\right] + bs_{cs}\right) \tag{24}$$

Further, the cell state is upgraded using Eq. (25).

$$CS_n = fg_n \cdot CS_{n-1} + ig_n \cdot CS'_n \tag{25}$$

Lastly, the outcome region of the model is processed with the sigmoid layer of cell state and tanh functionalities. It is shown in Eqs. (26) and (27).

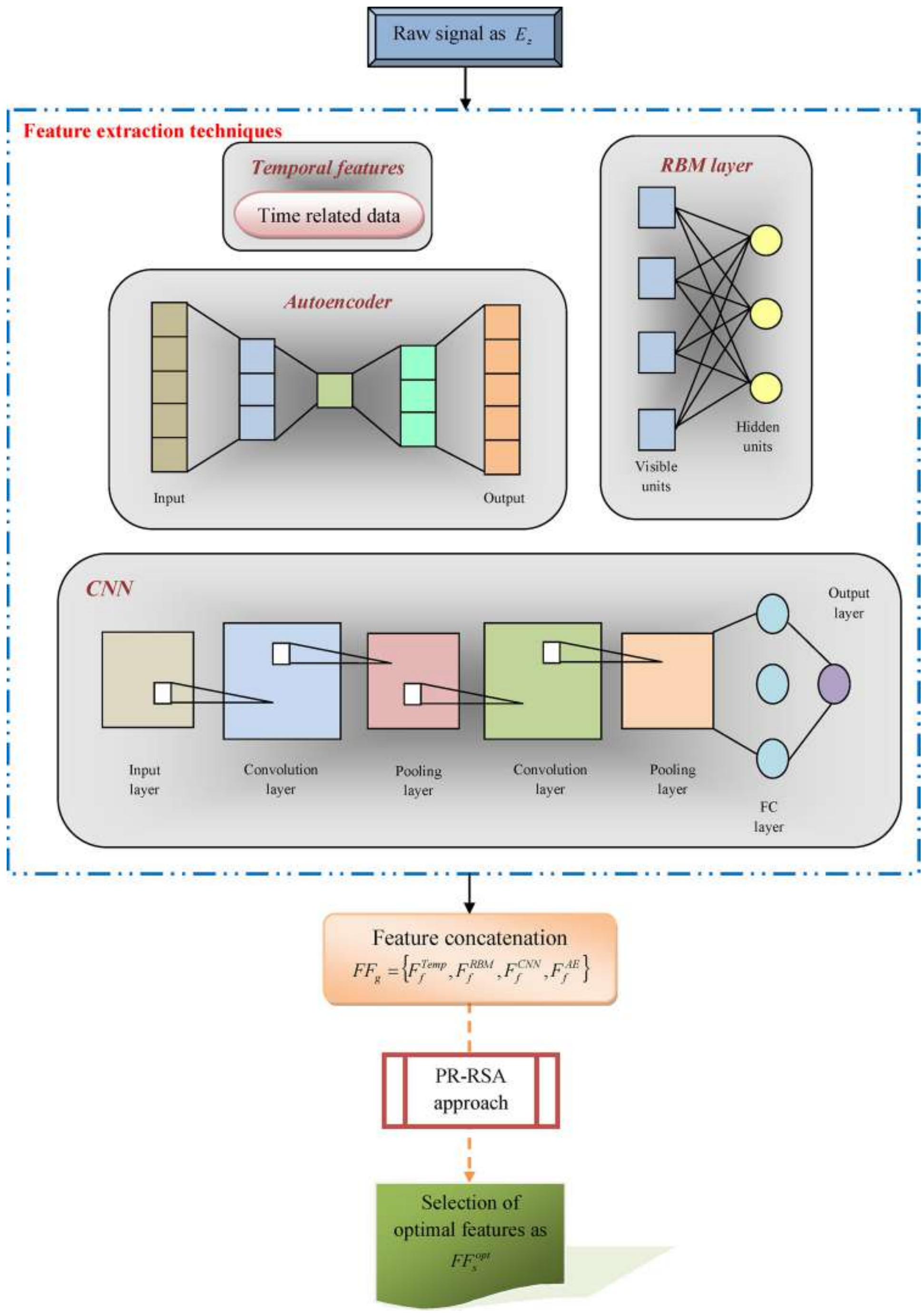

Figure 4. Depiction of optimal feature selection using PR-RSA heuristic approach.

$$og_n = \sigma\left(w_{og} \cdot \left[hl_{n-1}, FF_n^{opt}\right] + bs_{og}\right) \tag{26}$$

$$hl_n = og_n * \tanh(CS_n) \tag{27}$$

From Eqs. (22–27), the utilized terms are denoted as below

σ and tanh = sigmoid and tangent activation function.
FF_n^{opt} = input as optimally selected features.
w and bs- weight and bias terms.
hl_{n-1} = previously hidden level.

With the help of these gate functionalities, the Bi-LSTM performs both backward propagation and forward propagation to render the classified results. Thus, it is formulated using Eqs. (28) and (29).

$$hl_n^{fr} = \sigma\left(w_{xy} \cdot FF_n^{opt} + w_{yy} \cdot hl_{n-1}^{fr}\right) \tag{28}$$

$$hl_n^{bd} = \sigma\left(w_{xy} \cdot FF_n^{opt} + w_{yy} \cdot hl_{n+1}^{bd}\right) \tag{29}$$

The weight between input and hidden is marked by w_{xy} as well as among the two hidden states as w_{yy}.

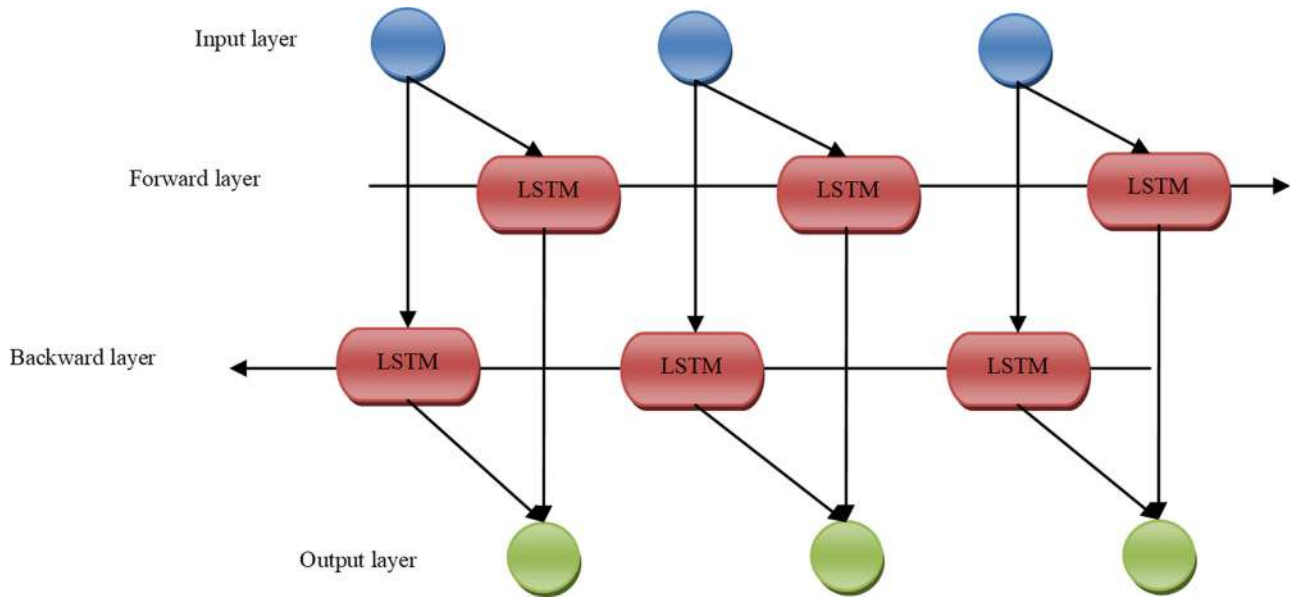

Figure 5. Diagrammatic illustration of the Bi-LSTM technique.

Finally, the classified result is obtained from the output layer that is expressed in Eq. (30).

$$Op_n = \sigma\left(hl_n^{fr}, hl_n^{bd}\right) \tag{30}$$

Finally, the classified results are obtained through the Bi-LSTM network, which is elucidated in Figure 5.

5.2. Enhanced Bi-LSTM framework for classification

The traditional Bi-LSTM model contains beneficial characteristics for increasing performance. Contrary to other models, the Bi-LSTM avoids the gradient vanishing problem, storing the data, requires minimal features, and so on. On the other hand, it poses some constraints to overcome as more processing time. Since it contains more gates and relied on the previous state, it causes computational and time complexity issues. It also paves the way for occurring overfitting problems. To facilitate this issue, the hidden neuron count and epochs range are optimally tuned by using an objective of PR-RSA in the EBi-LSTM. This parameter optimization assists to increase the classification accuracy.

Hence, the cost function of the novel Bi-LSTM is illustrated in Eq. (31).

$$Obj = \underset{\left\{FF_s^{opt}, Ep^{Bl}, Hn^{BL}\right\}}{\text{argmax}} [AY] \tag{31}$$

The term FF_s^{opt} denotes the optimal features that vary from 1 to 5. The epochs and hidden neuron counts are indicated using Ep^{BL} and Hn^{BL}, respectively. Here, the epoch count has the range of [50, 100] and the number of suitably hidden neuron counts lies between 5 and 255. Further, the variable AY signifies the term accuracy, which is 'defined as the state or quality of representing the appropriate or precise value'. It is estimated *via* Eq. (32).

$$AY = \frac{I+J}{I+J+K+L} \tag{32}$$

The true positive and false positive value is denoted by I and K, as well as true negative and false negative rate is indicated using J and L. Hence, the schematic representation of the enhanced Bi-LSTM model for classification is shown in Figure 6.

6. Discussion of results

6.1. Experimental setup

The offered classification framework was simulated in the Python platform and its simulated outcomes were executed. For our experimentation, the proposed model has taken the 10 number of population, total iteration as 25, and chromosome length as 7. Distinct parameters such as 'Accuracy, Sensitivity, Specificity, Precision, Negative Predictive Value (NPV), F1Score and Mathews Correlation Coefficient (MCC), False Positive Rate (FPR), False Negative Rate (FNR) and False Discovery Rate (FDR)' were employed for analyzing the performance. Finally, the performance of the novel method was compared with classical heuristic algorithms such as (Grey Wolf Optimizer) GWO-EBi-LSTM (Mirjalili et al. 2014), (Electric Fish Optimization) EFO-EBi-LSTM (Yilmaz and Sen 2020), (Salp Swarm Optimization) SSA-EBi-LSTM

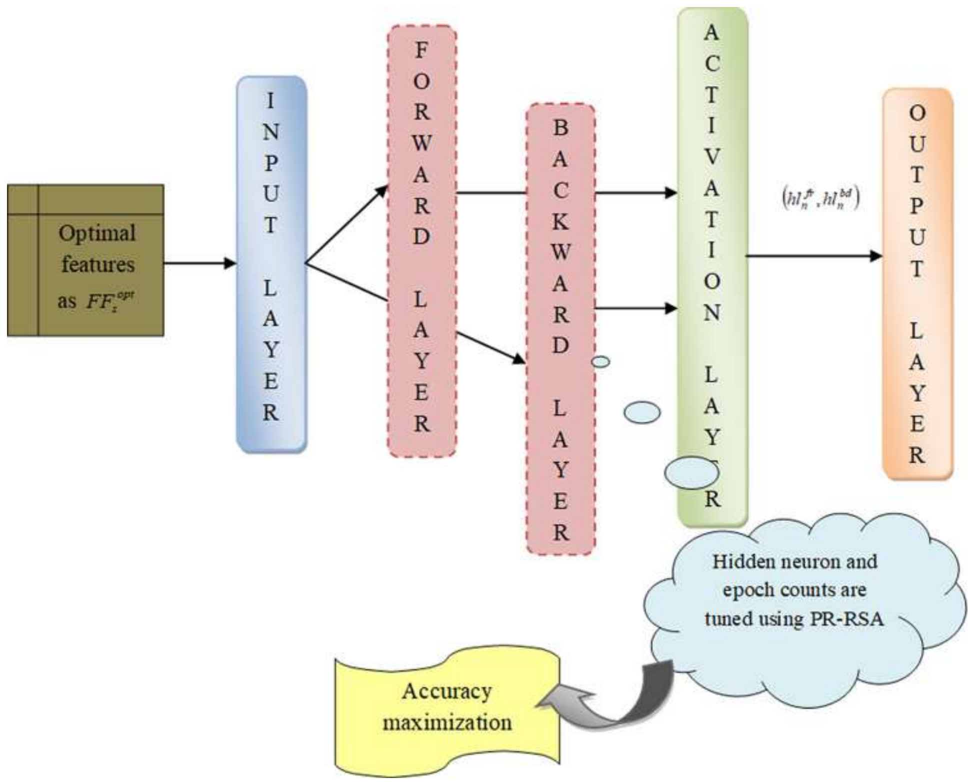

Figure 6. Depiction of optimal feature selection using PR-RSA heuristic approach.

(Mirjalili et al. 2017), and RSA-EBi-LSTM (Abualigah et al. 2022). Similarly, the traditional classifiers like SVM (Sharma et al. 2022), KNN (Zheng et al. 2016), CNN (Sakhavi et al. 2018), LSTM (Zhou et al. 2018), and Bi-LSTM (Alwasiti et al. 2020) are used for comparison.

6.2. Evaluation parameters

Diverse metrics are considered for analyzing the development level that is listed below.

Accuracy: It computes *via* Eq. (32).

Precision: It provides the closeness rate of time series data.

$$Pn = \frac{I}{I + K} \tag{33}$$

FPR and FNR: 'The false positive rate provides the error rate, in which the results are obtained incorrectly presence of signals. On the second hand, the false negative is used to determine the absence of time series data incorrectly actually when the signal is present'.

$$FPR = \frac{K}{K + J} \tag{34}$$

$$FNR = \frac{L}{L + I} \tag{35}$$

Sensitivity and Specificity: Sensitivity is explained as the 'probability of actual positive test' and specificity is measured *via* the 'probability of negative test'.

$$Sensitivity = \frac{I}{I + L} \tag{36}$$

$$Specificity = \frac{J}{J + K} \tag{37}$$

F1-Score: 'It is defined as the harmonic mean value of precision and recall'.

$$F1Score = 2 * \frac{Pn * \mathrm{Re}}{Pn + \mathrm{Re}} \tag{38}$$

FDR: 'It is defined as the ratio between the false positive and total number of both true and false positive'.

$$FDR = \frac{K}{I + K} \tag{39}$$

NPV: "The negative predictive rate is measured by the ratio of true negative to the total value of true and false negative".

$$NPV = \frac{J}{J + L} \tag{40}$$

MCC: It is computed by the 'difference between the classified value and actual value'.

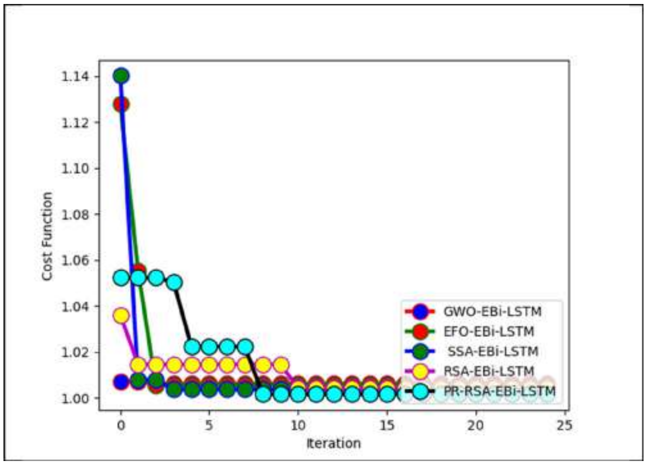

Figure 7. Convergence analysis of proposed time series data classification using EEG signal compared over different algorithms.

$$MCC = \frac{I \times J - K \times L}{\sqrt{(I + K)(I + L)(J + K)(J + L)}} \tag{41}$$

6.3. Convergence analysis on the suggested method

Figure 7 shows the evaluation of the convergence rate of the offered classification model by differing the iteration and compared with distinct algorithms. From the graph analysis, if the number of iterations increases then the convergence rate decreases. The cost function of the designed PR-RSA-EBi-LSTM method is varied according to the count of iterations. At 15th iteration, the cost function of the offered PR-RSA-EBi-LSTM method is attained as 0.5, 1, 0.5, and 0.7% elevated than GWO-EBi-LSTM, EFO-EBi-LSTM, SSA-EBi-LSTM, and RSA-EBi-LSTM. The better convergence rate is attained at the 15th, 20th, and 25th iterations. Hence, the lower cost function value assists to increase the convergence rate. Thus, the model proves its efficiency for classification.

6.4. k-fold analysis of the suggested method

In this section, Figure 8 demonstrates the k-fold analysis of the suggested classification method and compared it with distinct heuristic algorithms. From Figure 8(h), the precision value of the offered method is attained as 8.9% elevated than GWO-EBi-LSTM, 7% progressed than EFO-EBi-LSTM, 4% elevated than SSA-EBi-LSTM, 2.3 and 1.4% better than RSA-EBi-LSTM when the k-fold level is 2. Simultaneously, the k-fold analysis of the work is compared with traditional classifiers as shown in Figure 9. The FDR analysis is represented in Figure 9(h). The FDR value for offered model is attained as 3.5% lesser than SVM, 3.7% lesser than KNN, 3% lesser than CNN, 1.7%, and 1.4% lesser than LSTM and Bi-LSTM, accordingly when the k-fold level is 1. Hence, the high true and low false value tends to improve the classification results.

6.5. Validation of the suggested method

Estimation of the recommended model over classical algorithms and classifiers is elucidated in Figures 10 and 11. Figure 10(h) depicts the precision analysis for proposed techniques. At the 55th learning percentage, the precision value is acquired as 14.4% of GWO-EBi-LSTM, 18.3% of EFO-EBi-LSTM, 10% of SSA-EBi-LSTM, and 7.2% of RSA-EBi-LSTM, correspondingly that is lesser than proposed PR-RSA-EBi-LSTM. Similarly, Figure 11(i) explains the evaluation of the sensitivity of the offered method. At the 65th learning percentage, the sensitivity value is yielded as 8.6% of SVM, 6.45% of KNN, 5.37% of CNN, 4.3%, and 2.15% of LSTM and Bi-LSTM is less value than PR-RSA-EBi-LSTM. Thus, the extensive results declare that the proposed classification model enhances its accuracy level.

6.6. Validation of suggested method over traditional optimizations

Table 2 evaluates the value for the proposed work by varying the learning percentage and k-folds. While taking the learning percentage, the MCC value of proposed PR-RSA-EBi-LSTM is acquired as 12.7% more than GWO-EBi-LSTM, 13.6% more than EFO-EBi-LSTM, 7.7% more than SSA-EBi-LSTM and 8.84% of RSA-EBi-LSTM, accordingly. From the table results, the proposed work ensures to the achievement of a better-classified outcome.

6.7. Validation of the offered method over traditional classifiers

The overall evaluation for the proposed work is explored in Table 3 compared with distinct classifiers using learning percentages and k-folds. While taking the k-fold, the low false rate is obtained by PR-RSA-EBi-LSTM that is compared with 3.5, 3.6, 3, 1.7, and 1.2% higher than SVM, KNN, CNN, LSTM, and Bi-LSTM. Thus, the extensive results indicate the efficacy of the classification work.

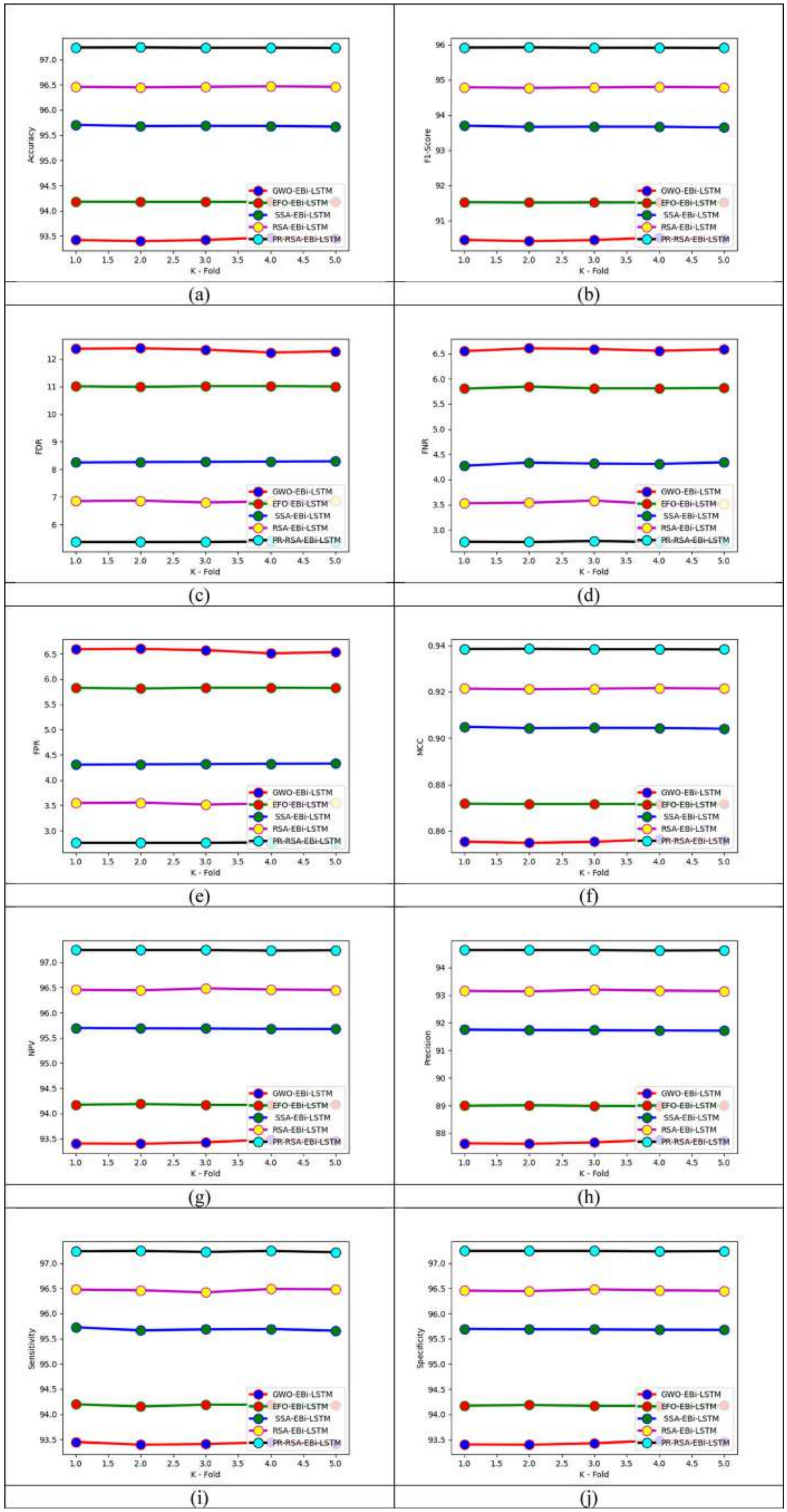

Figure 8. Evaluation of k-fold for the recommended time series data classification using EEG signal regarding (a) Accuracy, (b) F1-score, (c) FDR, (d) FNR, (e) FPR, (f) MCC, (g) NPV, (h) Precision, (i) Sensitivity, and (j) Specificity.

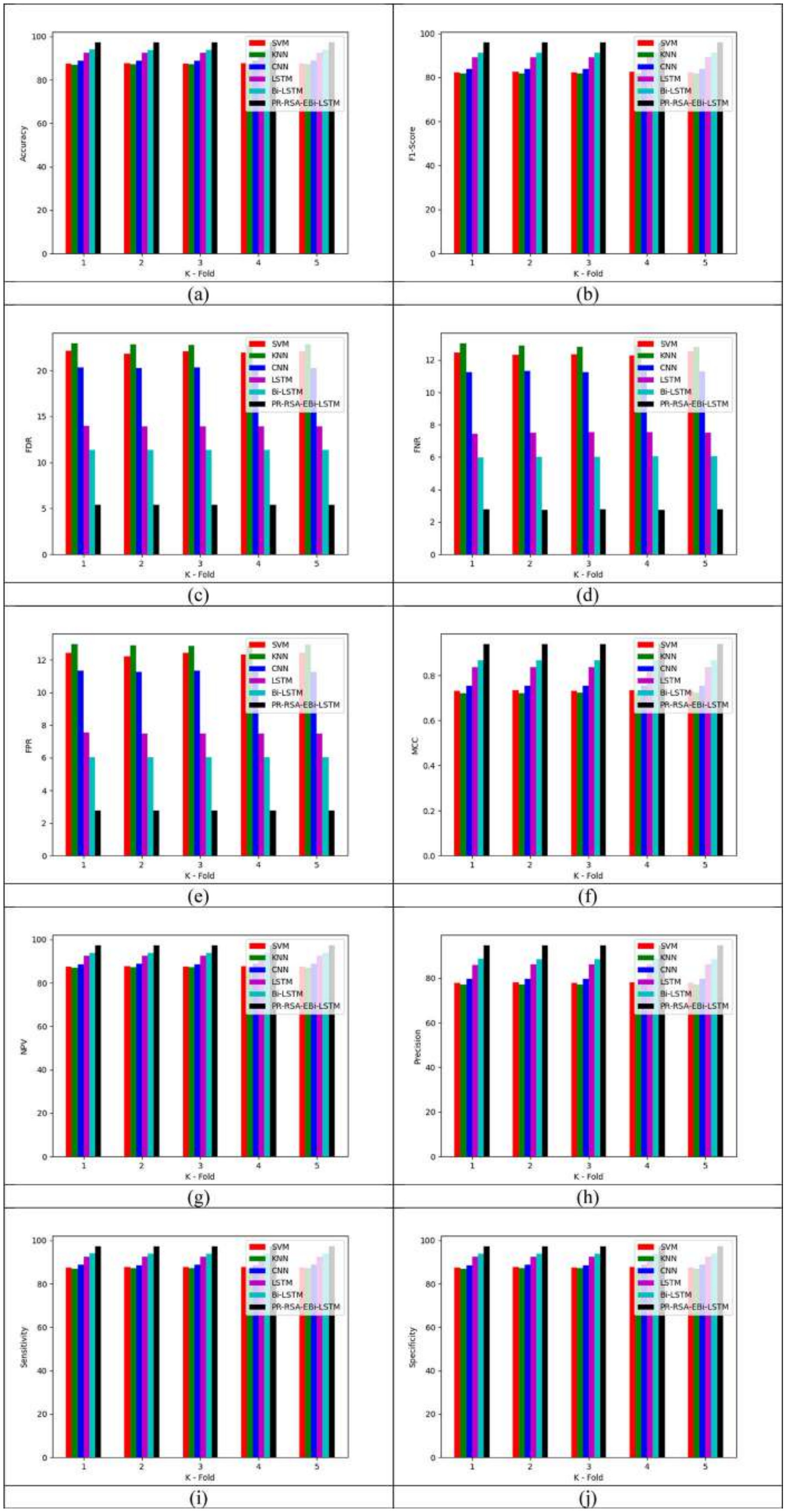

Figure 9. Evaluation of k-fold for the recommended time series data classification using EEG signal regarding (a) Accuracy, (b) F1-score, (c) FDR, (d) FNR, (e) FPR, (f) MCC, (g) NPV, (h) Precision, (i) Sensitivity, and (j) Specificity.

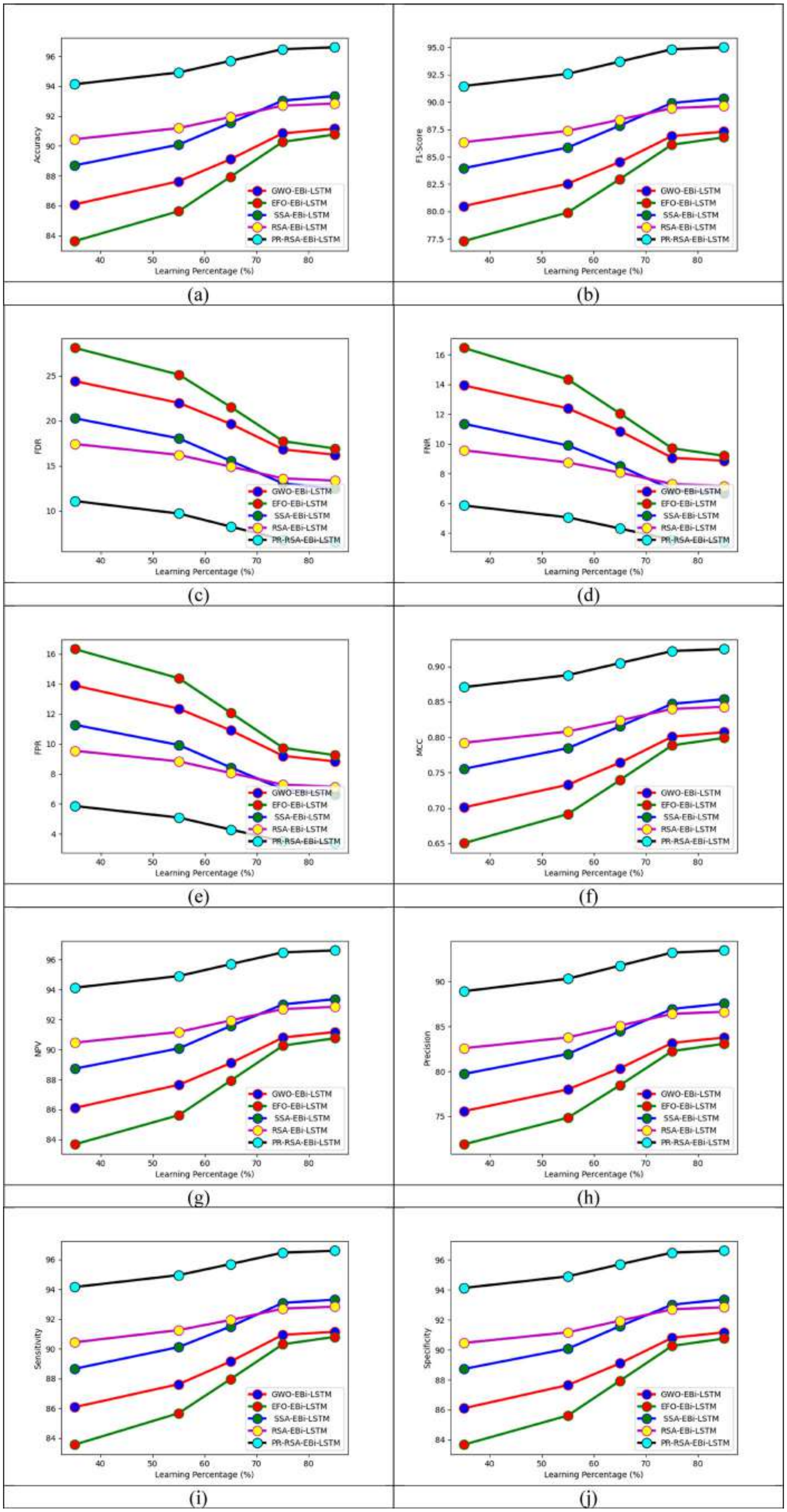

Figure 10. Performance analysis of recommended time series data classification using EEG signal regarding (a) Accuracy, (b) F1-score, (c) FDR, (d) FNR, (e) FPR, (f) MCC, (g) NPV, (h) Precision, (i) Sensitivity and (j) Specificity.

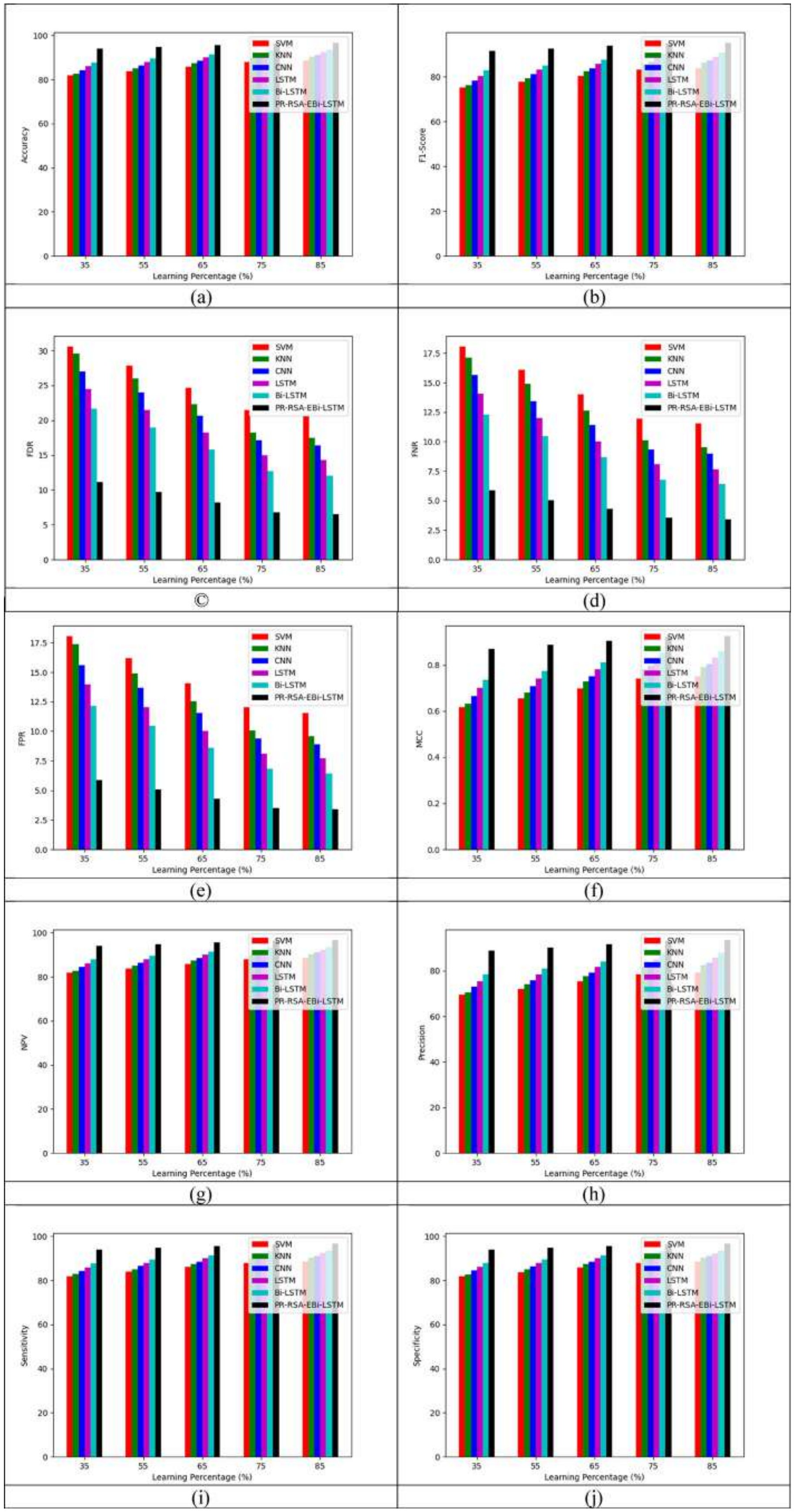

Figure 11. Evaluation of the recommended time series data classification using EEG signal regarding (a) Accuracy, (b) F1-score, (c) FDR, (d) FNR, (e) FPR, (f) MCC, (g) NPV, (h) Precision, (i) Sensitivity and (j) Specificity.

Table 2. Overall estimation of proposed time series data classification over other optimizations using learning percentage and k-fold.

Metrics	GWO-EBi-LSTM (Mirjalili et al. 2014)	EFO-EBi-LSTM (Yilmaz and Sen 2020)	SSA-EBi-LSTM (Mirjalili et al. 2017)	RSA-EBi-LSTM (Abualigah et al. 2022)	PR-RSA-EBi-LSTM
Learning percentage					
"Accuracy"	91.16141	90.77275	93.34705	92.84179	96.6118
"Sensitivity"	91.13855	90.80247	93.3059	92.83265	96.59808
"Specificity"	91.17284	90.75789	93.36763	92.84636	96.61866
"Precision"	83.77254	83.08648	87.5531	86.64618	93.4572
"FPR"	8.82716	9.242112	6.6323	7.1536	3.3813
"FNR"	8.861454	9.197531	6.694102	7.167353	3.40192
"NPV"	91.17284	90.75789	93.36763	92.84636	96.61866
"FDR"	16.22746	16.91352	12.4469	13.35382	6.5428
"F1-Score"	87.30044	86.77328	90.338	89.63279	95.00169
"MCC"	0.807091	0.798951	0.853732	0.842932	0.92469
k-fold					
"Accuracy"	93.44765	94.17924	95.66987	96.46091	97.23137
"Sensitivity"	93.41564	94.18381	95.65844	96.48148	97.21536
"Specificity"	93.46365	94.17695	95.67558	96.45062	97.23937
"Precision"	87.72382	88.99546	91.70831	93.1466	94.62581
"FPR"	6.536351	5.8230	4.3244	3.5593	2.76063
"FNR"	6.584362	5.816187	4.341564	3.518519	2.784636
"NPV"	93.46365	94.17695	95.67558	96.45062	97.23937
"FDR"	12.27618	11.00454	8.291689	6.853397	5.374191
"F1-Score"	90.4803	91.51616	93.64173	94.78472	95.90311
"MCC"	0.855907	0.871724	0.904072	0.921403	0.938324

Table 3. Overall estimation of proposed time series data classification over conventional classifiers using learning percentage and k-fold.

Metrics	SVM (Sharma et al. 2022)	KNN (Zheng et al. 2016)	CNN (Ieracitano et al. 2021)	LSTM (Zhou et al. 2018)	Bi-LSTM (Alwasiti et al. 2020)	PR-RSA-EBi-LSTM
Learning percentage						
"Accuracy"	88.46822	90.42981	91.06996	92.32968	93.57796	96.6118
"Sensitivity"	88.45679	90.45267	91.02195	92.34568	93.57339	96.59808
"Specificity"	88.47394	90.41838	91.09396	92.32167	93.58025	96.61866
"Precision"	79.3271	82.51783	83.63373	85.74158	87.93426	93.4572
"FPR"	11.52606	9.58161	8.91603	7.67832	6.4197	3.3813
"FNR"	11.54321	9.547325	8.978052	7.654321	6.426612	3.40192
"NPV"	88.47394	90.41838	91.09396	92.32167	93.58025	96.61866
"FDR"	20.6729	17.48217	16.36627	14.25842	12.06574	6.5428
"F1-Score"	83.64356	86.30325	87.17157	88.92118	90.66622	95.00169
"MCC"	0.750438	0.791692	0.805109	0.832015	0.858747	0.92469
k-fold						
"Accuracy"	87.54001	87.11477	88.72428	92.51029	93.96205	97.23137
"Sensitivity"	87.45542	87.18107	88.71742	92.48971	93.95062	97.21536
"Specificity"	87.5823	87.08162	88.72771	92.52058	93.96776	97.23937
"Precision"	77.88297	77.13922	79.73739	86.07813	88.62004	94.62581
"FPR"	12.4179	12.9183	11.2722	7.4794	6.0322	2.760631
"FNR"	12.54458	12.81893	11.28258	7.510288	6.049383	2.784636
"NPV"	87.5823	87.08162	88.72771	92.52058	93.96776	97.23937
"FDR"	22.11703	22.86078	20.26261	13.92187	11.37996	5.374191
"F1-Score"	82.39209	81.85331	83.98805	89.16882	91.20751	95.90311
"MCC"	0.730937	0.722457	0.755798	0.83582	0.867016	0.938324

Table 4. A statistical estimation of the offered time series data classification compared with different algorithms.

Metrics	GWO-EBi-LSTM (Mirjalili et al. 2014)	EFO-EBi-LSTM (Yilmaz and Sen 2020)	SSA-EBi-LSTM (Mirjalili et al. 2017)	RSA-EBi-LSTM (Abualigah et al. 2022)	PR-RSA-EBi-LSTM
"Best"	1.006176	1.005642	1.003685	1.004222	1.001652
"Worst"	1.00728	1.1282	1.14021	1.036185	1.052351
"Mean"	1.006264	1.012536	1.009494	1.00928	1.013007
"Median"	1.006176	1.005642	1.003685	1.004222	1.001652
"STD"	0.00029	0.0255	0.0267	0.00741	0.018533

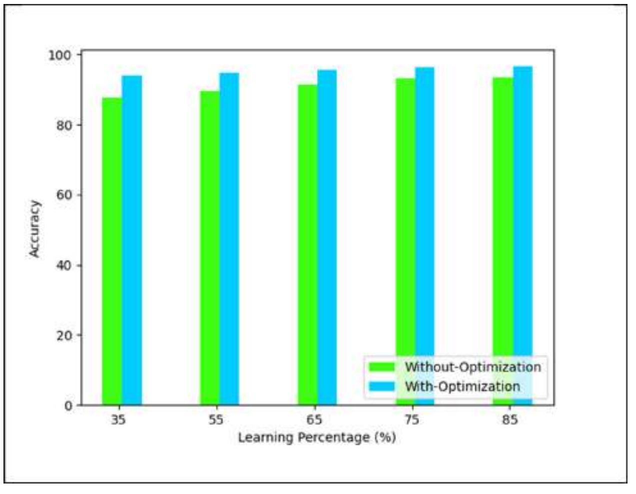

Figure 12. Comparison of the proposed time series data classification model for feature optimization with ensemble feature extraction.

6.8. Statistical validation of the proposed approach

The statistical evaluation of the proposed approach is shown in Table 4. The evaluation is done with five factors 'best, worst, median, mean, and standard deviation'. The mean is the average value of the best and worst values and the median is referred to as the center point of the best and worst values whereas the standard deviation is represented as the 'degree of deviation between each execution'. Thus, the statistical value aids to improve the better performance.

6.9. Comparison of the proposed method for feature optimization with ensemble feature extraction

The comparison of the proposed time series data classification model for feature optimization with ensemble feature extraction is shown in Figure 12.

7. Conclusion and future work

This paper has explored the novel data classification model using an enhanced deep learning method with heuristic development. At first, the necessary signal was to be collected from publicly available sources. Further, it was given into the ensemble feature extraction, where the four different features were retrieved *via* temporal features, DBN-RBM layer features, CNN features, and autoencoder features, respectively. Consequently, these resultant features were fused altogether to form one feature set. To achieve optimal results, the novel PR-RSA was developed to choose only the optimal features. It was then fed into the final EBi-LSTM classification model, where the counts of hidden neurons and epochs were tuned optimally with the help of the PR-RSA approach. The efficiency was computed with diverse metrics, which were then compared over former approaches. Hence, the accuracy value of offered method was attained as elevated than 8.42% of SVM, 6.4% of KNN, 5.73% of CNN, 4.43% of LSTM, and 3.14% of Bi-LSTM, respectively, by differing the learning percentage. Therefore, the empirical outcome has exhibited a higher classification performance. Yet, the limitation of the work is facing time complexity issues. To further improve the results, it recommends using other signal formats and hybrid intelligence optimization algorithms that direct to future development.

Disclosure Statement

No potential conflict of interest was reported by the author(s).

Funding

The author(s) reported there is no funding associated with the work featured in this article.

References

Chacon-Murguia MI, Rivas-Posada E. 2022. A CNN-based modular classification scheme for motor imagery using a novel EEG sampling protocol suitable for IoT healthcare systems. Neural Comput Appl.

Abenna S, Nahid M, Bajit A. 2022. Motor imagery based brain-computer interface: improving the EEG classification using Delta rhythm and LightGBM algorithm. Biomed Signal Process Control. 71:103102.

Abualigah L, Elaziz MA, Sumari P, Geem ZW, Gandomi AH. 2022. Reptile Search Algorithm (RSA): a nature-inspired meta-heuristic optimizer. Expert Syst Appl. 191:116158.

Al-Shourbaji I, Helian N, Sun Y, Alshathri S, Elaziz MA. 2022. Boosting ant colony optimization with reptile search algorithm for churn prediction. Mathematics. 10(7):1031.

Altaheri H, Muhammad G, Alsulaiman M, Amin SU, Altuwaijri GA, Abdul W, Bencherif MA, Faisal M. 2021. Deep learning techniques for classification of electroencephalogram (EEG) motor imagery (MI) signals: a review. Neural Comput Appl.

Alwasiti H, Yusoff MZ, Raza K. 2020. Motor imagery classification for brain-computer interface using deep metric learning. IEEE Access. 8:109949–109963.

Ambati LS, El-Gayar O. 2021. Human activity recognition: a comparison of machine learning approaches. J Midwest Assoc Inf Syst. 1:49–60.

Amin SU, Alsulaiman M, Muhammad G, Bencherif MA, Hossain MS. 2019. Multilevel weighted feature fusion using convolutional neural networks for EEG motor imagery classification. IEEE Access. 7:18940–18950.

Chambon S, Galtier MN, Arnal PJ, Wainrib G, Gramfort A. 2018. A deep learning architecture for temporal sleep stage classification using multivariate and multimodal time series. IEEE Trans Neural Syst Rehabil Eng. 26(4):758–769.

Cortez SA, Flores C, Andreu-Perez J. 2020. Single-trial P300 classification using deep belief networks for a BCI system. 2020 IEEE XXVII International Conference on Electronics, Electrical Engineering and Computing (INTERCON), p. 1–4.

Coyle D, Prasad G, McGinnity TM. 2005. A time-series prediction approach for feature extraction in a brain-computer interface. IEEE Trans Neural Syst Rehabil Eng. 13(4):461–467.

Dose H, Moller JS, Iversen HK, Puthusserypady S. 2018. An end-to-end deep learning approach to MI-EEG signal classification for BCIs. Expert Syst Appl. 114:532–542.

Guger C, Ramoser H, Pfurtscheller G. 2000. Real-time EEG analysis with subject-specific spatial patterns for a brain-computer interface (BCI). IEEE Trans Rehabil Eng. 8(4):447–456.

Huang Y, Jin J, Xu R, Miao Y, Liu C, Cichocki A. 2022. Multi-view optimization of time-frequency common spatial patterns for brain-computer interfaces. J Neurosci Methods. 365:109378.

Ieracitano C, Mammone N, Hussain A, Morabito FC. 2021. A novel explainable machine learning approach for EEG-based brain-computer interface systems. Neural Comput Appl. 34:11347–11360.

Jin J, Wang Z, Xu R, Liu C, Wang X, Cichocki A. 2021. Robust similarity measurement based on a novel time filter for SSVEPs detection. IEEE Trans Neural Netw Learning Syst. 1–10.

Jin J, Xiao R, Daly I, Miao Y, Wang X, Cichocki A. 2021. Internal feature selection method of CSP Based on L1-Norm and Dempster-Shafer theory. IEEE Trans Neural Netw Learn Syst. 32(11):4814–4825.

Jin Z, Zhou G, Gao D, Zhang Y. 2020. EEG classification using sparse Bayesian extreme learning machine for brain–computer interface. Neural Comput Appl. 32(11):6601–6609.

Li Y, Wang F, Chen Y, Cichocki A, Sejnowski T. 2018. The effects of audiovisual inputs on solving the cocktail party problem in the human brain: an fMRI study. Cereb Cortex. 28(10):3623–3637.

Lv Z, Qiao L, Wang Q, Piccialli F. 2021. Advanced machine-learning methods for brain-computer interfacing. IEEE/ACM Trans Comput Biol Bioinform. 18(5):1688–1698.

Maweu BM, Shamsuddin R, Dakshit S, Prabhakaran B. 2021. Generating healthcare time series data for improving diagnostic accuracy of deep neural networks. IEEE Trans Instrum Meas. 70.

Miao M, Zeng H, Wang A, Zhao C, Liu F. 2017. Discriminative spatial-frequency-temporal feature extraction and classification of motor imagery EEG: an sparse regression and Weighted Naïve Bayesian Classifier-based approach. J Neurosci Methods. 278:13–24.

Mini PP, Thomas T, Gopikakumari R. 2021. EEG-based direct speech BCI system using a fusion of SMRT and MFCC/LPCC features with ANN classifier. Biomedical Signal Process Control. 68:102625.

Mirjalili S, Gandomi AH, Mirjalili SZ, Saremi S, Farisd H, Mirjalili SM. 2017. Salp Swarm Algorithm: a bio-inspired optimizer for engineering design problems. Adv Eng Softw. 114:163–191.

Mirjalili S, Mirjalili SM, Lewis A. 2014. Grey wolf optimizer. Adv Eng Softw. 69:46–61.

Mirzaei S, Ghasemi P. 2021. EEG motor imagery classification using dynamic connectivity patterns and convolutional autoencoder. Biomed Signal Process Control. 68:102584.

Nakra A, Duhan M. 2022. Deep neural network with harmony search based optimal feature selection of EEG signals for motor imagery classification. Int J Inf Technol. 15:611–625.

Radman M, Chaibakhsh A, Nariman-Zadeh N, He H. 2021. Feature fusion for improving performance of motor imagery brain-computer interface system. Biomed Signal Process Control. 68:102763.

Roy AM. 2022. An efficient multi-scale CNN model with intrinsic feature integration for motor imagery EEG subject classification in brain-machine interfaces. Biomed Signal Process Control. 74:103496.

Sakhavi S, Guan C, Yan S. 2018. Learning temporal information for brain-computer interface using convolutional neural networks. IEEE Trans Neural Netw Learn Syst. 29(11):5619–5629.

Sharma R, Kim M, Gupta A. 2022. Motor imagery classification in brain-machine interface with machine learning algorithms: classical approach to multi-layer perceptron model. Biomed Signal Process Control. 71:103101.

Tyagi A, Nehra V. 2016. Classification of motor imagery EEG signals using SVM, k-NN and ANN. CSI Trans ICT. 4:135–139.

Vega CF, Quevedo J, Escandon E, Kiani M, Ding W, Andreu-Perez J. 2022. Fuzzy temporal convolutional neural networks in P300-based brain–computer interface for smart home interaction. Appl Soft Comput. 117:108359.

Wang L, Huang W, Yang Z, Zhang C. 2020. Temporal-spatial-frequency depth extraction of brain-computer interface based on mental tasks. Biomed Signal Process Control. 58:101845.

Yeomans J, Thwaites S, Robertson WSP, Booth D, Ng B, Thewlis D. 2019. Simulating time-series data for improved deep neural network performance. IEEE Access. 7:131248–131255.

Yilmaz S, Sen S. 2020. Electric fish optimization: a new heuristic algorithm inspired by electrolocation. Neural Comput Appl. 32:11543–11578.

Zheng Y, Liu Q, Chen E, Ge Y, Zhao JL. 2016. Exploiting multi-channels deep convolutional neural networks for multivariate time series classification. Front Comput Sci. 10:96–112.

Zhou J, Meng M, Gao Y, Ma Y, Zhang Q. 2018. Classification of motor imagery EEG using wavelet envelope analysis and LSTM networks. 2018 Chinese Control and Decision Conference (CCDC), pp. 5600–5605.