Iraqi Journal for Computers and Informatics
Information Technology and Communications University
Vol. 51, No. 1, 2025, pp. 1-18
DOI: https://doi.org/10.25195/ijci.v51i1.545
Print ISSN: 2313-190X, Online ISSN: 2520-4912

Research Article

Hybrid Optimized Feature Selection and Deep Learning Method for Emotion Recognition That Uses EEG Data

[1]Asmaa Bashar Hmaza,
Informatics Institute for Postgraduate Studies,
Iraqi Commission for Computer and Informatics,
Baghdad,
Iraq
Ms202220724@iips.edu.iq

[2]Rajaa K. Hasoun,
Department of Information System Management
University of Information Technology and Informatics Communications,
Baghdad,
Iraq

ARTICLEINFO
Article History
Received: 19/01/2025
Accepted: 19/02/2025
Published:28/02/2025

ABSTRACT

Introduction: This study represents an important development in human–machine interactions. It aims to utilize the potential of electroencephalograms (EEGs) in recognizing emotions, which is a complex and variable task. This study presents a complete framework for enhancing emotion identification. It provides an intuitive way for humans to interact with machines emotionally by understanding the emotion machine. The process begins with collecting and preprocessing EEG information to use the data for training and testing the proposed system. Optimization, machine learning, and deep learning algorithms are applied in this study. First, particle swarm optimization (PSO) identifies and optimizes critical functions and reduces feature dimensionality. Thereafter, long short-term memory (LSTM), gated recurrent unit (GRU), and simple recurrent neural network (RNN) architectures are used in emotion identification. All the applied models are evaluated using common evaluation metrics, such as accuracy, precision, and the F1 score. From various implementations of the different models applied to identify EEG emotion, the LSTM model achieved good results with an accuracy of 98.13%, a precision of 98.15%, and an F1 score of 98.13%. Although the GRU and simple RNN models exhibit good performance in emotion identification, their measurements are less than those of LSTM, which outperforms all the other models. This study incorporates the concepts of the PSO algorithm into a feature selection and deep learning model by using LSTM to enhance EEG emotion identification. The proposed model overcomes difficulties and issues related to EEG signals, leading to an accurate emotion detection system and providing enhanced machine understanding of human–machine interactions.

Keywords: *Human–Machine Interaction, Gated Recurrent Unit (GRU), Electroencephalogram (EEG), Recurrent Neural Network (RNN) , Emotion Identification, Long Short-term Memory (LSTM) , Particle Swarm Optimization (PSO)*

1. INTRODUCTION

A brain–computer interface (BCI) is considered an important instrument in biomedical engineering. Such interface allows for the control of devices through neural impulses [1]. Electroencephalograms (EEGs) are commonly used in BCIs to capture human emotions. EEGs play a significant role in identifying human emotions [2]. Numerous studies have used EEGs in emotion detection. In particular, these studies have focused on improving interactions between humans and machines [3].

Electroencephalography can record brain activities through EEG signals and interpret them as different emotional states [4]. The character of these signals poses many problems with regard to obtaining accurate and consistent emotion identification [5]. The current study aims to address these obstacles by employing powerful

Iraqi Journal for Computers and Informatics
Information Technology and Communications University
Vol. 51, No. 1, 2025, pp. 1-18
DOI: https://doi.org/10.25195/ijci.v51i1.545
Print ISSN: 2313-190X, Online ISSN: 2520-4912

machine learning approaches to decipher the patterns present in an EEG dataset. The primary objective is to enhance the accuracy and improve dependability of emotion identification systems by incorporating sophisticated feature selection and deep learning techniques.

The current study uses a hybrid approach that comprises advanced methods for information collection and preprocessing. The technique ensures that the input for model training is clean and applicable, which is important to cope with the inherent difficulties of EEG data. Particle swarm optimization (PSO) is applied to determine which functions from the EEG records are the most informative. This step substantially reduces the dimensionality of the information and focuses the education of the model on characteristics that can be highly indicative of emotional states, ultimately improving overall prediction performance and accuracy.

In addition, the current study investigates three advanced architectures for recurrent neural networks (RNNs): long short-term memory (LSTM), gated recurrent unit (GRU), and simple RNN [6]. These models are not only highly skilled at managing sequential data but also particularly adept at capturing temporal dynamics that are occurring within EEG signals. Every model undergoes a stringent training process on feature sets that have been tuned and then evaluated based on essential performance measures [7].

2. LITERATURE REVIEW

Many new approaches and applications have contributed to the rapid development of EEG-based emotion recognition. The current study summarizes significant advancements in this field, focusing on the most important methods and their effectiveness.

For support vector machine classifiers, the usage of noninvasive EEG in BCIs has been thoroughly investigated. A variety of real-time emotion analysis techniques reached an accuracy of up to 78.41% [8]. In addition to EEG and galvanic skin reaction, another innovative method involved combining several modalities [9]. A multimodal technique attained an impressive 91.5% accuracy for four emotional states on the Dataset for Emotion Analysis Using Physiological Signals (DEAP), along with an average accuracy of 74.2% and a maximum accuracy of 81.2% on the Loughborough University Multimodal Emotion Dataset-2. The advantages of multimodal integration were further demonstrated via a deep autoencoder model that integrated EEG signals with facial emotions, providing an average identification rate of 85.71% [10]. HybridEEGNet applied a deep learning model and used EEG as input data to distinguish among normal individuals and medicated and unmedicated depression patients [11]. This model achieved 68.78% sensitivity, 84.45% specificity, and 79.08% accuracy. The aforementioned studies demonstrated how deep learning can provide benefits to healthcare. In addition, a neural network design that combined an efficient network with a semi-skipping layered gated unit was presented [12]. This model used Internet of Things (IoT) applications to help improve EEG-based emotion classification by reducing computing needs. To extract features from EEG data, EEGFuseNet combines generative adversarial networks with convolutional neural networks (CNNs) and RNNs; it exhibits strong performance across several datasets and focuses on unsupervised learning [13]. Another study investigated how utilizing fewer EEG channels could affect classification accuracy in a CNN-based model. It determined that learning was considerably faster without a huge drop in accuracy. Swarm intelligence methods have been used to select features for EEG-based emotion identification by leveraging DEAP. The results presented an accuracy range of 56.27% to 60.29% [14]. By using the ResNet50 model and the Adam optimizer, CNN achieved good classification accuracy on the Shanghai Jiao Tong University Emotion EEG Dataset (SEED) across three emotional states [15]. Emotion recognition was achieved using a CNN–bidirectional LSTM hybrid technique, where testing on SEED and DEAP revealed significant accuracy gains [16]. In addition, a hybrid model that was tailored for DEAP, achieving a remarkable 92.38% classification accuracy by combining fast Fourier transform with CNN, LSTM, and attention processes [17]. Compared with other conventional approaches, the binary chaotic evolutionary algorithm for optimum feature selection performed better on emotion recognition tasks [18]. Furthermore, the EEG Brain Wave dataset was used to assess a hybrid LSTM model with multiple classifiers, achieving a high level of accuracy, and thus, demonstrating the efficacy of hybrid techniques [19]. The authors of [20] used deep neural networks that were improved using a harmony search technique, providing an accuracy of around 97% in EEG-based motor imagery [20]. Some studies have used recently developed techniques, such as attention mechanisms, to identify emotions from EEG data. The multimodal emotion identification technique is effective, as demonstrated by its excellent accuracy on DEAP and MAHNOB-HCI instances [21, 22].

Iraqi Journal for Computers and Informatics
Information Technology and Communications University
Vol. 51, No. 1, 2025, pp. 1-18
DOI: https://doi.org/10.25195/ijci.v51i1.545
Print ISSN: 2313-190X, Online ISSN: 2520-4912

3. Theoretical Background

This section discusses recurrent deep learning algorithms and PSO for feature selection in the emotion recognition of EEG data.

3.1 Deep Learning

Deep learning, a subset of machine learning within artificial intelligence, utilizes networks that learn autonomously from unstructured data. It operates through supervised or unsupervised learning modes, as shown in Figure 1, which outlines several machine learning models. Deep learning can process and analyze large amounts of complex data, which will be labor-intensive for humans to interpret. These algorithms progressively transform input data into increasingly abstract representations [25]. Advances in machine learning have expanded opportunities for deep learning applications in the disease prediction emotion recognition of EEG data. Deep learning has been proven effective in image processing, classification, and pattern recognition tasks, becoming essential for identifying patterns within vast gene expression data from sources, such as microarray and RNA-Seq datasets. The emotion recognition of EEG data cells based on gene expression levels remains challenging. The current study addresses this issue by using unsupervised recurrent deep learning techniques [26].

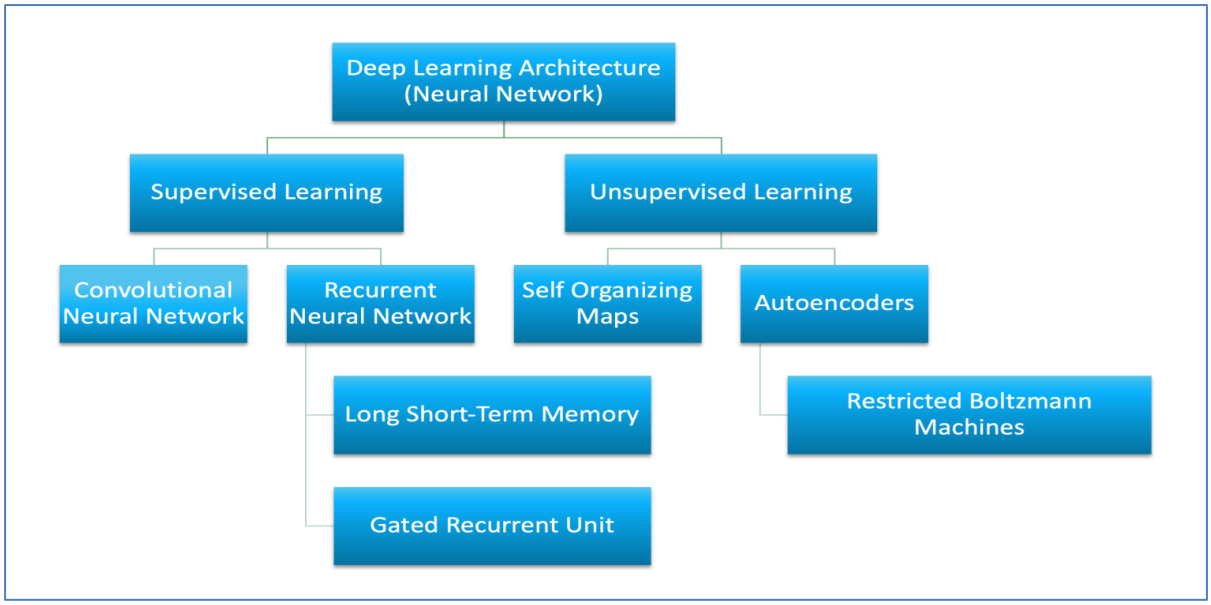

Fig. 1: *Types of deep learning [26].*

3.1.1 RNN

RNNs are deep learning models that specialize on sequential data given their ability to remember previous input through an internal memory. This feature enables RNNs to recycle output from one step back as input for the next, and thus, RNNs are ideal for tasks that involve sequences, such as speech, natural patterns, and DNA data [27]. In contrast with traditional neural networks, RNNs store information from past computations, allowing them to process current and past input simultaneously. An RNN typically has three layers: input, hidden, and output, as illustrated in Figure 2. Their recurrent nature allows them to process each sequence element with an output that depends on earlier calculations [28]. At any time step (t), the input is (x(t)), which is a sequence from (x(1)) to (x(T)), where (T) is the sequence length. The hidden state (h(t)) acts as memory, calculated based on (x(t)) and the previous hidden state (h(t−1)), as shown in Equation (1). The activation function (f) is frequently a nonlinear transformation, such as tanh (Equation 2) or a rectified linear unit (ReLU) (Equation 3). The weight matrices (W), (V), and (U) manage connections within the network: (W) for input-to-hidden, (V) for hidden-to-hidden, and (U) for hidden-to-output. The final output, (y(t)), is generated as shown in Equation (4) [29, 30].

Iraqi Journal for Computers and Informatics
Information Technology and Communications University
Vol. 51, No. 1, 2025, pp. 1-18
DOI: https://doi.org/10.25195/ijci.v51i1.545
Print ISSN: 2313-190X, Online ISSN: 2520-4912

$$h(t) = f(w\,x(t) + v\,h(t-1)) \quad (1)$$

$$\text{Tanh}(x) = (e^{x} - e^{(-x)}) / (e^{x} + e^{(-x)}) \quad (2)$$

$$\text{ReLU}(x) = \max(0, x) \quad (3)$$

$$y(t) = g\,(\,U\,h(t)\,) \quad (4)$$

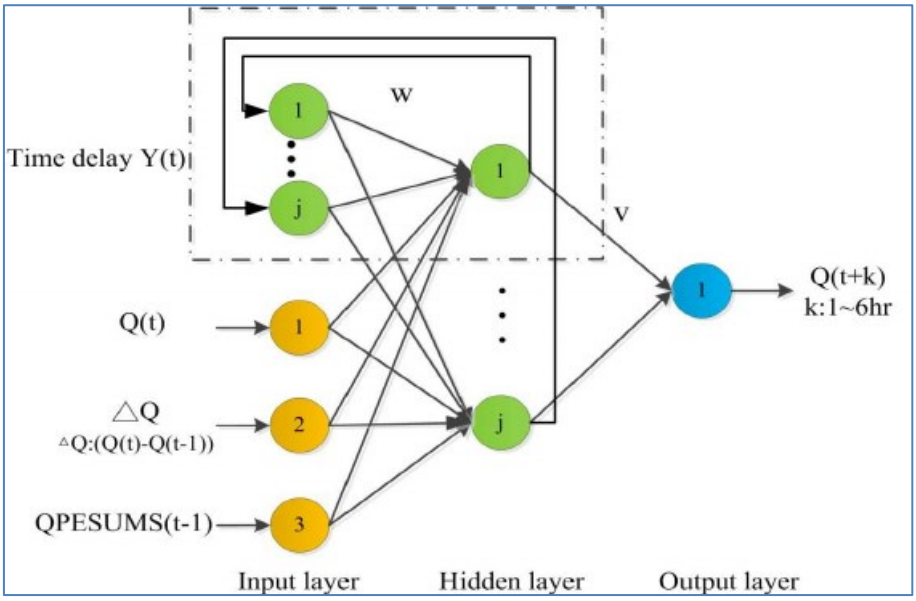

***Fig. 2**: RNN architecture [29].*

3.1.2 LSTM

LSTM is an advanced form of RNN that is designed to deal with the vanishing gradient issue by integrating memory units that retain long-term dependencies. Their architecture consists of memory cells and gates that manage the retention or exclusion of information, making them highly suitable for tasks that require sequential data processing [31]. Each LSTM cell receives three input: the current input (X_t), previous output (h_{t-1}), and previous cell memory (C_{t-1}). These cells produce two output: the new output (h_t) and updated memory (C_t). The LSTM architecture includes three types of gates, namely, input, forget, and output, which control data flow through sigmoid and pointwise multiplication functions (Figure 3) [32]. The input gate manages which activations will enter the memory cell, while the output gate controls output flow. The forget gate decides which information will be discarded from the memory via the sigmoid function. Each time step incorporates previous time step data (h_t−1), (C_{t-1}), and current input (X_t) [33].

$$f_t = \sigma\,(\,W_f\,.\,[h_{t-1}, X_t] + b_f) \quad (5)$$

$$i_t = \sigma\,(W_i\,.\,[\,X_t\,, h_{t-1}\,] + b_i\,) \quad (6)$$

$$\tilde{C} = \tanh\,(\,W_c\,.\,[\,X_t\,, h_{t-1}\,] + b_c\,) \quad (7)$$

Then, the input gate and (tanh) layer update the cell's state, making adjustments based on input relevance. The combined effect updates cell state with Equations 6 and 7 for the input gate and candidate values (C_t). The state update occurs as follows:

$$C_t = f_t * C_{t-1} + \tilde{C}_t * i_t. \quad (8)$$

Finally, the output is calculated by selecting relevant cell states via a sigmoid layer, followed by (tanh) transformation, as shown in Equations 9 and 10:

$$O_t = \sigma\,(\,W_o\,[\,h_{t-1}, X_t\,] + b_o\,), \quad (9)$$

Iraqi Journal for Computers and Informatics
Information Technology and Communications University
Vol. 51, No. 1, 2025, pp. 1-18
DOI: https://doi.org/10.25195/ijci.v51i1.545
Print ISSN: 2313-190X, Online ISSN: 2520-4912

$$h_t = O_t * \tanh(C_t). \quad (10)$$

At the end of each cycle, the updated (C_t) and (h_t) are prepared for the next step, with all the gates trained to optimize memory retention. Multiple memory cells can be grouped to use shared gates, optimizing parameter efficiency [34, 35, 36].

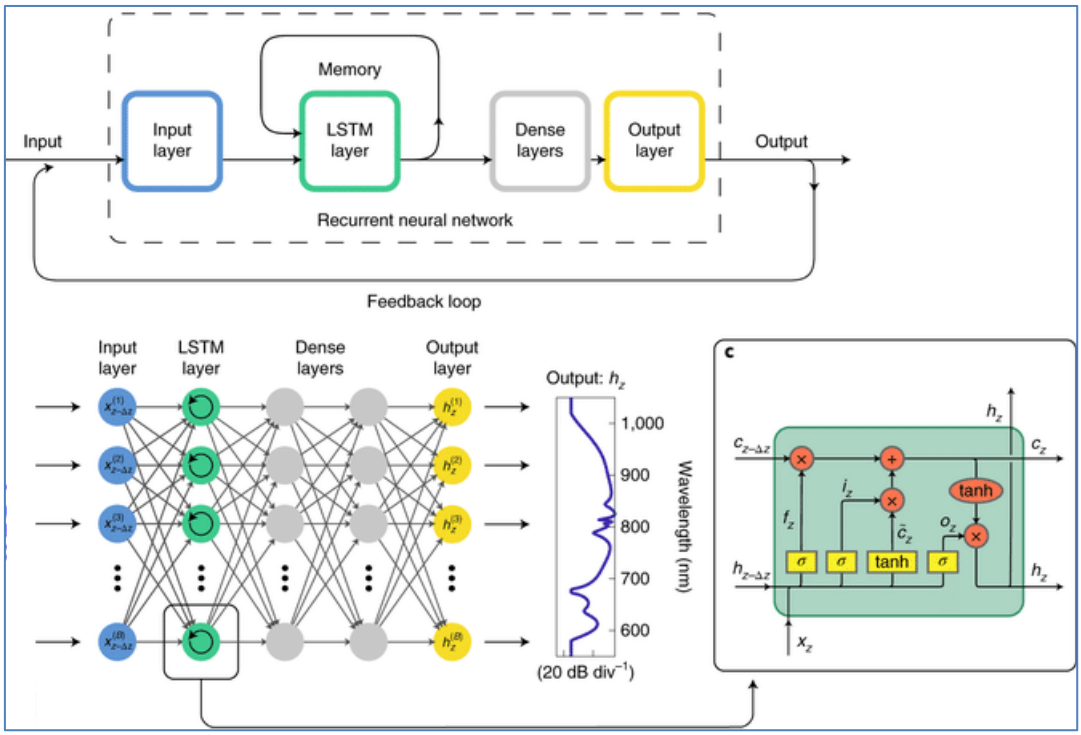

Fig. 3: *Illustration of the LSTM architecture [35].*

3.1.3 GRU

The GRU is an advanced RNN variant that is streamlined from LSTM to address the vanishing gradient problem. GRUs are designed with two gates, i.e., the update and reset gates, making them efficient for processing sequential or temporal data and modeling complex nonlinear relationships in time series as shown in **Figure (4)** [37]. In contrast with LSTMs, GRUs lack a cell state and use only the update and reset gates, combining functions similar to LSTM's forget and input gates. These gates control data flow by using sigmoid layers and pointwise multiplication, determining how much of past data affect the current state. The reset gate selectively combines prior memory (h_{t-1}) with the current input (x_t), as shown in Equation (12) [38].

The reset gate function is calculated as

$$r_t = \sigma\,(W_r \,.\, [h_{t-1}, x_t]). \quad (12)$$

The update gate (z_t) manages the degree to which (h_{t-1}) influences the new hidden state (h_t) in accordance with Equation (13):

$$z_t = \sigma\,(W_z \,.\, [h_{t-1}, x_t]). \quad (13)$$

The candidate hidden state (tanh) is computed using the reset gate, as shown in Equation (14):

$$\tilde{h}_t = \tanh\,(W \,.\, [\, r_t * h_{t-1}, x_t\,]). \quad (14)$$

Iraqi Journal for Computers and Informatics
Information Technology and Communications University
Vol. 51, No. 1, 2025, pp. 1-18
DOI: https://doi.org/10.25195/ijci.v51i1.545
Print ISSN: 2313-190X, Online ISSN: 2520-4912

Finally, the new hidden state (h_t) combines past states with the current update, as shown in Equation (15), while the model output (Y_t) is derived from the hidden state in Equation (16):

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t, \quad (15)$$

$$Y_t = \sigma (w_o . h_t). \quad (16)$$

These equations illustrate the ability of GRUs to retain essential information and efficiently manage sequential data [39].

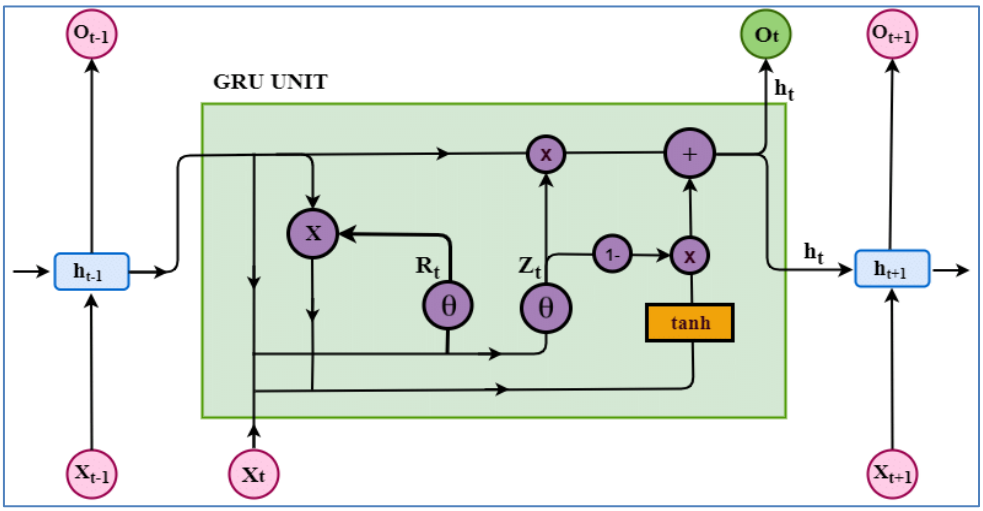

Fig. 4: Illustration of the GRU architecture [39].

3.2 PSO

PSO is inspired by the collective movement of bird or fish swarms, wherein each particle adjusts its position based on its own experience and those of neighboring particles. PSO has been proven effective in finding optimal solutions by balancing the exploration and exploitation of the search space [40]. The steps of the PSO algorithm are as follows.

1. Initialization of Parameters: Initial values for particle positions, velocities, and other PSO parameters are set as indicated in Table 1.

2. Updating of Position and Velocity: Each particle updates its position and velocity based on its own best-known position and the swarm's global best position.

$$V^i_{t+1} = w\, V^i_t + c_1\, r_1\, (P_i - X^i_t) + c_2\, r_2\, (G - X^i_t), \quad (16)$$

$$X^i_{t+1} = X^i_t + V^i_{t+1}, \quad (17)$$

where (V^i_{t+1}) and (X^i_{t+1}) represent the velocity and position of particle (i) at time (t+1), respectively. Parameters (c_1) and (c_2) are learning factors, (r_1) and (r_2) are random values in [0,1], and (w) is the inertia weight.

3. Evaluation and Updating of Best Positions: A particle's fitness is evaluated after calculating a new position. If the current position yields a better fitness value, then it updates the personal best (P_i) and global best (G) positions.

4. Adjusting Inertia Weight: The inertia weight (w) is reduced over time to allow the algorithm to converge, facilitating exploration at the start and focusing on exploitation toward the end.

5. Termination Condition: The process is repeated until a stopping criterion, such as maximum iteration or a convergence threshold, is reached. The best global solution is then outputted as (G) [41].

Iraqi Journal for Computers and Informatics
Information Technology and Communications University
Vol. 51, No. 1, 2025, pp. 1-18
DOI: https://doi.org/10.25195/ijci.v51i1.545
Print ISSN: 2313-190X, Online ISSN: 2520-4912

TABLE I: Group of parameters utilized in PSO

Parameter	Description
M	Number of particles in the population
	Global best solution found by the swarm
P_i	Best-known position for each particle
www	Inertia weight
c_1,c_2	Learning factors for position update
r_1,r_2	Random numbers within the [0,1] range
V_i	Velocity of the iiith particle
X_i	Position of the iiith particle
f(X)	Fitness function

4. DATA COLLECTION

The MUSE EEG headband has four dry extracranial electrodes placed at TP9, AF7, AF8, and TP10 (as illustrated in Figure 5). Two volunteers (a male and a female, aged 20–22 years) had their EEG signals recorded while viewing six separate 60 s video segments. The end outcome of the procedure was 12 min of EEG data that pertained to different emotional reactions. The overall amount of collected EEG data was 36 min, including 6 min of neutral brain activities [42].

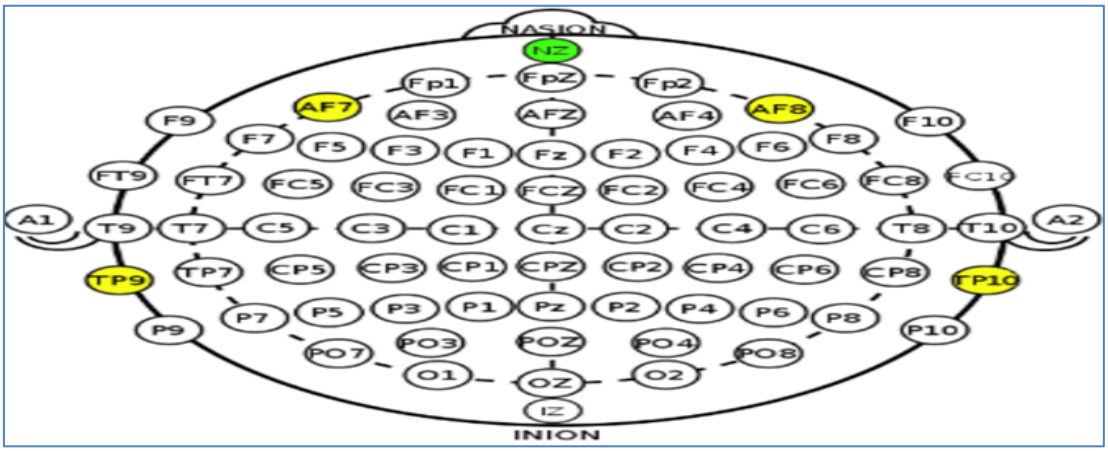

Fig. 5: *Placement of Muse headband EEG sensors TP9, AF7, AF8, and TP10 in accordance with the international standard system for EEG electrode positioning.*

The collection contained 324,000 data points, indicating brain wave activities because the EEG recordings were sampled at a frequency of 150 Hz. In accordance with **Table 2**, the selected film snippets were meant to evoke emotions that lean toward the positive or negative sides. The EEG dataset was divided into three emotional indicators: normal, positive, and negative. The neutral data were recorded before the emotionally charged sessions, and negative emotions included shyness, embarrassment, fear, and anger when the selected videos were used. Meanwhile, positive emotions included pleasure, happiness, interest, and excitement. To set a baseline or resting state for comparison, neutral and unstimulated brain wave data were obtained before the emotionally charged sessions. To minimize any potential emotional effect on the EEG readings, recordings were limited to 3 min daily to guarantee the integrity of resting state data [42].

TABLE II: *Chart that depicts Lövheim's categorization of emotions, each was labeled with different attitudes [42]*

Emotion Category	Emotion/Valence
A	Shame (Negative) Embarrassment (Negative)
B	Contempt (Negative) Revulsion (Negative)
C	Fear (Negative) Horror (Negative)
D	Enjoyment (Positive) Happiness (Positive)

Iraqi Journal for Computers and Informatics
Information Technology and Communications University
Vol. 51, No. 1, 2025, pp. 1-18
DOI: https://doi.org/10.25195/ijci.v51i1.545
Print ISSN: 2313-190X, Online ISSN: 2520-4912

E	Distress (Negative) Sorrow (Negative)
F	Surprise (Negative) (Deficiency of Dopamine)
G	Anger (Negative) Fury (Negative)
H	Interest (Positive) Thrill (Positive)

5. METHODOLOGY

The process of processing EEG brain wave data for emotion detection is organized through several steps that can be described as follows. The process begins with data preparation that involves loading, cleaning, and organizing EEG data. Then, the data are divided into predefined features and numerically encoded labels. PSO is used to select the best features. This algorithm helps determine which features are the best for emotion recognition. The process of training a model leads to preparing the data so that they can be used with LSTM, GRU, and simple RNN models. To decrease overtraining issues, an early-stopping approach is implemented during the training process. The procedure for analyzing EEG data in the context of emotion detection is divided into five distinct phases, as illustrated in detail in Algorithm 1 and Figure 6.

Algorithm 1: Emotion Detection from EEG Data by Using Enhanced Feature Selection and Deep Learning Techniques

Input: Raw EEG brain wave dataset.
Output: Models trained and set for performance assessment.

Step 1: **Preparation of Data:** Import the dataset, clean it by eliminating any missing entries, then segregate it into features (X) and labels (y). Convert labels into integer values and normalize the feature data [23].

- **Import Dataset**: Let the dataset be represented as $D=(x_i,y_i)_{i=1}^{N}$, where $x_i \in R^d$ is the feature vector of the i-th sample, $y_i \in C$ is the corresponding label, N is the total number of samples, and dd is the number of features.
- **Data Cleansing**: Remove any samples (x_i,y_i) where x_i contains missing values:

  $D'=\{(x_i,y_i) | \forall j \in [1,d], x_i[j] \neq NaN\}$.
  Here, D' is the cleansed dataset.

- **Convert Labels into Integer Values**:
  Map each label $y_i \in C$ to an integer $\hat{y}_i \in Z$:

  $\hat{y}_i = LabelEncoder(yi), \forall yi \in y.$
  The encoded labels form the vector $\hat{y}=[\hat{y}_1, \hat{y}_3,\ldots, \hat{y}_M]^T$

- **Normalize Feature Data**:
  Scale each feature x_i to have zero mean and unit variance:

  $$x_i^{norm} = \frac{x_i-\mu}{\sigma} \text{ where } \mu=\frac{1}{M}\sum_{i=1}^{M} x_i, \sigma = \sqrt{\frac{1}{M}\sum_{i=1}^{M} (x_i-\mu)^2}$$

  The normalized feature matrix is X^{norm}.

Step 2: **Feature Selection with PSO**

1. **Initialization**: Define the feature matrix F with dimensions n×d, where n is the number of rows and d = 50 is the number of features (cells).
   Initialize P_i (position) and V_i (velocity) for each particle ii in the PSO algorithm.
2. **Feature Constraints**:
   Ensure no repetition of features within the selected subset:

Iraqi Journal for Computers and Informatics
Information Technology and Communications University
Vol. 51, No. 1, 2025, pp. 1-18
DOI: https://doi.org/10.25195/ijci.v51i1.545
Print ISSN: 2313-190X, Online ISSN: 2520-4912

If $S_k=\{f_1,f_2,\ldots,f_m\}$, then $f_i\neq f_j\ \forall i\neq j, S_k\subseteq\{F_1,F_2,\ldots,F_d\}$.

3. **Metrics Calculation**:
   For each feature f_j:
4. **Correlation**:

$$Corr(f_j)=\frac{\sum_{i=0}^{n}(f_j^i-\underline{f_J})(\underline{F^i}-\underline{F})}{\sqrt{\sum_{i=0}^{n}(f_j^i-\underline{f_J})^2}\cdot\sqrt{\sum_{i=0}^{n}(\underline{F^i}-\underline{F})^2}}$$ (where $\underline{F}$ is the mean of all features).

Lower Corr(f_j) is preferred.

Variance: $Var(f_J)=\frac{1}{n}\sum_{i=0}^{n}(f_j^i-\underline{f_J})^2$ Higher Var(f_j) is preferred.

5. **Fitness Function**: Train an RNN model on the selected feature subset S_k and compute the classification accuracy $A(S_k)$ as the fitness score:

   Fitness(S_k)=A(S_k).

6. **PSO Update Equations**:
   For each particle ii:

   Update velocity: $V_i^{t+1}=w\cdot V_i^t+c1\cdot r1\cdot(P_i^{best}-P_i^t)+c2\cdot r2\cdot(G^{best}-P_i^t)$

   Update position: $P_i^{t+1}=P_i^t+V_i^{t+1}$.

7. **Feature Selection Criteria**:
   Prioritize features with:

   Minimize: Corr(f_j)andMaximize: Var(f_j),subject to maximizing A(S_k).

   Convergence: Stop when the change in Fitness(S_k) over iterations is below a predefined threshold ϵ, or a maximum number of iterations T_{max} is reached.

$$|A(S_k^{t+1})-A(S_k^t)|<\epsilon\ ort>Tmax.$$

8. **Output**:
   The optimal feature subset S^*

$$S^*=arg\ max_{S_k}\ A(S_k), S_k\subseteq\{F_1,F_2,\ldots,F_d\}.$$

Step 3: Dividing the dataset into 80% training and 20% testing ensures that the model learns EEG patterns effectively while maintaining a fair evaluation on unseen data, preventing overfitting and ensuring generalization. $D_{train}=(x_i,y_i)_{i=1}^{0.8M}$, $D_{test}=(x_i,y_i)_{i=0.8M+1}^{M}$

Step 4: Building and Training Deep Learning Models for EEG Data Classification

1. Initial parameters are epochs: 100, batch size: 32
2. Define three types of models: LSTM, GRU, and simple RNN.
3. LSTM Model Construction
   - Initialize a sequential container for the LSTM model.
   - Add a first LSTM layer with 128 units, returning sequences, and specify the input shape.
   - Add a second LSTM layer with 64 units.
   - Append a Dense output layer with a number of units equal and softmax activation. y_{output}=softmax(W·h2+b) where Wand b are trainable parameters.

Iraqi Journal for Computers and Informatics
Information Technology and Communications University
Vol. 51, No. 1, 2025, pp. 1-18
DOI: https://doi.org/10.25195/ijci.v51i1.545
Print ISSN: 2313-190X, Online ISSN: 2520-4912

- Compile the LSTM model with the 'Adam' optimizer and 'sparse_categorical_crossentropy' loss function, tracking 'accuracy' as a metric. $LSTM(X_{input})=Dense(LSTM_2(LSTM_1(X_{input})))$

4. **GRU Model Construction**
   - Initialize a sequential container for the GRU model.
   - Add a first GRU layer with 128 units, returning sequences, and specify the input shape.
   - Add a second GRU layer with 64 units.
   - Append a Dense output layer with a number of units equal to 'unique_labels' and 'softmax' activation $y_{output}=softmax(W\cdot h2+b)$.
   - Compile the GRU model with the 'Adam' optimizer and 'sparse_categorical_crossentropy' loss function, tracking 'accuracy' as a metric. $GRU(X_{input})=Dense(GRU_2(GRU_1(X_{input})))$
5. **Simple RNN Model Construction**
   - Initialize a sequential container for the simple RNN model.
   - Add a first simple RNN layer with 128 units, returning sequences, and specify the input shape.
   - Add a second simple RNN layer with 64 units.
   - Append a Dense output layer with a number of units equal to 'unique_labels' and 'softmax' activation $y_{output}=softmax(W\cdot h2+b)$ where W and b are trainable parameters.
   - Compile the simple RNN model with the 'Adam' optimizer and 'sparse_categorical_crossentropy' loss function, tracking 'accuracy' as a metric $RNN(X_{input})=Dense(GRU_2(GRU_1(X_{input})))$.
6. **Training the Models**

   - Train each model on 'X_train' and 'y_train' for the specified 'epochs' and 'batch_size', using the defined 'callback' for early stopping based on validation loss. Train each model on (X_{train},y_{train}):

$$\Theta = arg\frac{1}{|D_{train}|}\sum_{i\in Dtrain} Loss(y_i,\hat{y_i})$$

   where:

   - Θ Model parameters.
   - Loss function: Sparse categorical cross-entropy.
   - Validate each model using 'X_test' and 'y_test'. $Accuracy = \frac{1}{Dtest}\sum_{i\in Dtest} (y_i = \hat{y_i})$
7. **Outputs**
   - LSTM, GRU, and RNN Histories: Training histories that contain information on the training and validation accuracies and losses for each epoch.
   - Each model's performance can then be further analyzed and compared based on the recorded training histories.

Step 5: **Evaluation of Predictions:** Use the test data to generate predictions and compute key performance indicators, including precision, recall, and F1 score, for each model to determine its efficiency.

The PSO algorithm begins by initializing a matrix that consists of 50 cells, wherein each cell corresponds to a feature or column in the dataset being used. Each feature represents a potential data point that can be selected for classification. A key condition is enforced, i.e., no repetition of selected features within the

Iraqi Journal for Computers and Informatics
Information Technology and Communications University
Vol. 51, No. 1, 2025, pp. 1-18
DOI: https://doi.org/10.25195/ijci.v51i1.545
Print ISSN: 2313-190X, Online ISSN: 2520-4912

table, ensuring that the selection process evaluates unique combinations of features. For each iteration, the algorithm evaluates the features by calculating two key metrics: **correlation** and **variance**. Correlation measures the dependency of a feature on others, wherein lower correlation values are preferred, because they indicate features that provide unique and independent information. Meanwhile, variance measures the spread or variability of a feature, and higher variance values are better because they indicate features with meaningful and distinguishable data points. After calculating these metrics for all the features, the algorithm proceeds to feed the selected features into an RNN-based model. The RNN model is trained on the subset of features, and the classification accuracy of the model is computed. Accuracy acts as the fitness score for the selected features. The PSO algorithm then iterates, updating the selection of features by maximizing classification accuracy while continuing to prioritize features with low correlation and high variance. The process of selecting, evaluating, and refining features continues until the algorithm converges on the optimal subset of features that yields the highest classification accuracy. This condition ensures that the final selected features are not only optimal for the RNN model but also provide the most independent and informative representation of data.

The PSO algorithm iteratively refines feature selection to enhance model performance, reducing redundancy and improving classification accuracy for emotion detection by using EEG data.

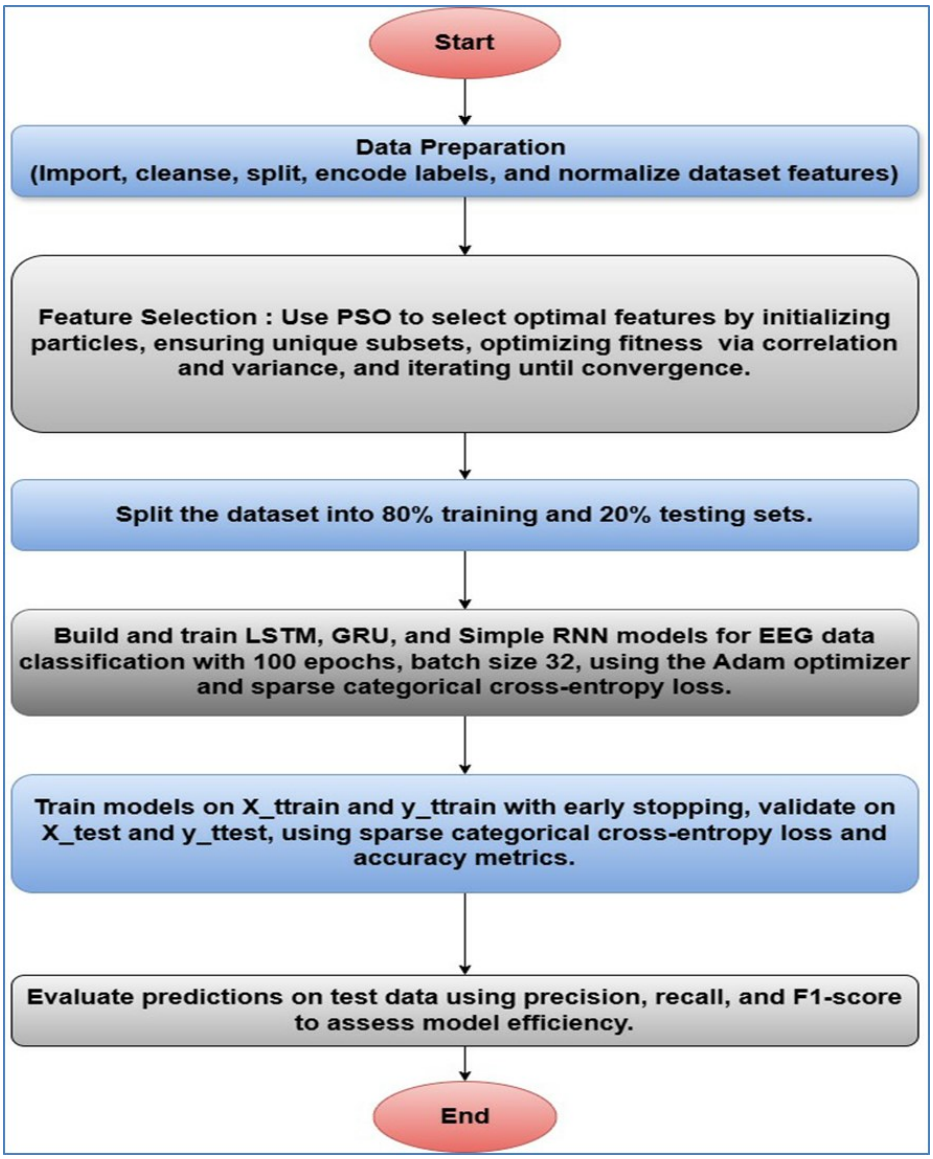

Fig. 6: *Flowchart of the Emotion Recognition EEG Brain Wave dataset.*

Iraqi Journal for Computers and Informatics
Information Technology and Communications University
Vol. 51, No. 1, 2025, pp. 1-18
DOI: https://doi.org/10.25195/ijci.v51i1.545
Print ISSN: 2313-190X, Online ISSN: 2520-4912

6. Results and Discussion

The present study exhibits that the capacity to discern emotions from EEG signals is considerably improved by the integration of advanced deep learning techniques and optimized feature selection. The investigation used the EEG Brain Wave Dataset for Emotion Recognition, which was obtained from Kaggle [23]. As illustrated in Figure 7, diverse dimensions of EEG information, including multiple sign channels, frequency domain names, and statistical attributes derived from EEG measurements, are captured in these columns. A comprehensive examination and graphical illustration of the dataset elucidated critical fact characteristics, along with the distribution of values and correlations between the emotional nation target variable and numerous variables. The next information preprocessing, function engineering, and improvement of deep studying frameworks that are particularly designed for EEG-based total emotion detection are all dependent on the preliminary evaluation [24].

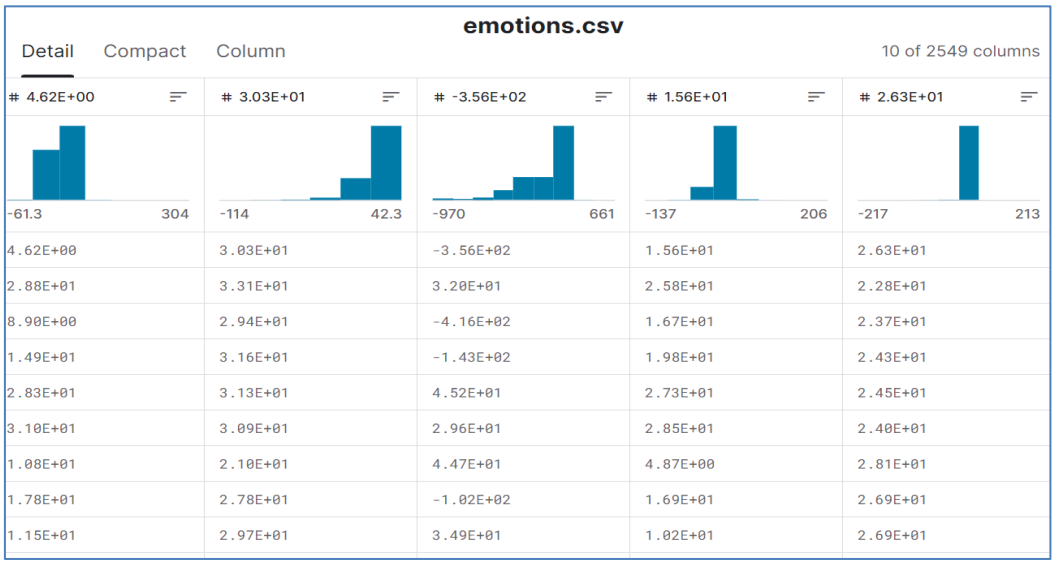

Fig. 7: *Sample EEG Brain Wave for Emotion Recognition dataset.*

The initial processing step produced a clear, normalized, and feature-engineered dataset prepared for the development of machine learning models. A low-pass filter is implemented to reduce high-frequency noise and enhance the clarity of seizure-related patterns in EEG signals. **Figure 8** illustrates the histograms derived from the data, revealing outliers and specific EEG data distributions. The analysis is essential for influencing the selection of additional features and informing the training of predictive models.

# mean_0_	mean_1_a	mean_2_a	mean_3_a	mean_4_a	mean_d_0_a	mean_d_1_a	mean_d_2_a	mean_d_3_a	mean_d_4_a	mean_d_0_a	mean_d_1	mean_d_2_	mean_d_3_	mean_d_4
4.62E+00	3.03E+01	-3.56E+02	1.56E+01	2.63E+01	1.07E+00	4.11E-01	-1.57E+01	2.06E+00	3.15E+00	2.15E+00	2.95E+01	-3.53E+02	1.44E+01	2.15E+01
2.88E+01	3.31E+01	3.20E+01	2.58E+01	2.28E+01	6.55E+00	1.68E+00	2.88E+00	3.83E+00	-4.82E+00	2.56E+01	3.28E+01	2.96E+01	2.15E+01	1.74E+01
8.90E+00	2.94E+01	-4.16E+02	1.67E+01	2.37E+01	7.99E+01	3.36E+00	9.02E+01	8.99E+01	2.03E+00	7.75E+00	3.01E+01	-4.41E+02	9.89E+00	2.53E+01
1.49E+01	3.16E+01	-1.43E+02	1.98E+01	2.43E+01	-5.84E-01	-2.84E-01	8.82E+00	2.30E+00	-1.97E+00	1.73E+01	3.20E+01	-1.48E+02	2.04E+01	2.28E+01
2.83E+01	3.13E+01	4.52E+01	2.73E+01	2.45E+01	3.48E+01	-5.79E+00	3.06E+00	4.14E+01	5.52E+00	2.61E+01	3.43E+01	4.37E+01	2.37E+01	2.06E+01
3.10E+01	3.09E+01	2.96E+01	2.85E+01	2.40E+01	1.65E+00	1.54E+00	3.83E+00	1.87E+00	-1.21E+00	2.80E+01	3.17E+01	2.90E+01	2.51E+01	1.75E+01
1.08E+01	2.10E+01	4.47E+01	4.87E+00	2.81E+01	2.14E+00	1.02E+00	1.32E+01	1.16E+00	-4.39E+00	1.07E+01	2.00E+01	2.47E+01	2.32E+00	3.33E+01
1.78E+01	2.78E+01	-1.02E+02	1.69E+01	2.69E+01	-3.21E+00	-1.95E+00	9.80E+00	-3.24E+00	-9.55E-01	1.96E+01	2.58E+01	-1.04E+02	1.84E+01	3.20E+01
1.15E+01	2.97E+01	3.49E+01	1.02E+01	2.69E+01	-3.80E+01	-1.65E+00	3.89E+00	-3.35E+01	-3.30E+00	2.86E+01	2.86E+01	3.88E+01	2.69E+01	2.35E+01
8.91E+00	2.92E+01	-3.14E+02	6.51E+00	3.09E+01	-1.88E+00	1.90E+00	1.19E+01	-3.60E+00	5.70E+00	9.85E+00	2.98E+01	-3.22E+02	8.43E+00	2.93E+01
5.21E+00	2.84E+01	1.85E+01	3.66E+00	2.26E+01	-1.19E-01	-2.35E+00	5.07E+00	4.61E-01	-4.26E+00	6.18E+00	3.15E+01	1.52E+01	3.11E+00	1.98E+01
1.33E+01	3.04E+01	-1.49E+02	1.18E+01	2.83E+01	3.03E+00	8.95E-01	-4.52E+00	1.72E+00	-6.33E-01	1.14E+01	2.94E+01	-1.58E+02	1.17E+01	2.78E+01
3.01E+01	3.27E+01	2.94E+01	2.83E+01	2.43E+01	-6.05E+00	-3.94E+00	1.69E+00	-3.57E+00	6.02E+00	3.37E+01	3.51E+01	3.08E+01	3.03E+01	1.77E+01
1.93E+01	3.17E+01	-4.56E+00	2.38E+01	3.29E+01	-3.41E+00	6.77E-01	4.66E+01	6.80E+00	3.89E+00	8.92E+00	3.05E+01	-6.95E+01	1.76E+01	2.60E+01
2.88E+01	3.18E+01	3.01E+01	2.69E+01	2.18E+01	-3.55E-01	4.65E-01	3.82E+00	4.42E-01	-1.83E+00	2.64E+01	3.07E+01	3.05E+01	2.37E+01	2.10E+01
1.27E+01	2.38E+01	-1.93E+02	1.11E+01	2.76E+01	-3.72E+00	2.05E+00	1.98E+01	-8.04E-01	-2.87E+00	1.49E+01	2.07E+01	-1.84E+02	1.27E+01	2.56E+01
4.25E-01	3.49E+01	5.39E+01	-1.19E+01	2.26E+01	1.24E+01	-2.72E+00	-4.86E+00	5.29E+00	-1.04E+00	-1.29E+01	3.86E+01	6.36E+01	-1.45E+01	2.35E+01
2.89E+01	3.08E+01	2.94E+01	2.74E+01	2.51E+01	-7.86E+00	1.02E+00	4.99E-01	-8.50E+00	-1.26E+00	3.31E+01	3.11E+01	2.96E+01	3.08E+01	3.23E+01
-1.41E+01	2.19E+01	-9.46E+02	8.93E+00	2.26E+01	2.89E-02	1.28E+00	1.64E+01	9.20E+00	4.26E+00	-1.66E+01	2.16E+01	-9.62E+02	3.29E+00	1.53E+01
8.74E+00	2.76E+01	-3.99E+02	1.50E+01	2.81E+01	-5.86E+00	-1.88E+00	9.10E+00	2.07E+00	-6.29E+00	3.62E+01	2.94E+01	-4.09E+02	1.71E+01	2.41E+01

Fig. 8: *Processed EEG data for epileptic seizure detection: a visualization of normalized and filtered signal features*

Iraqi Journal for Computers and Informatics
Information Technology and Communications University
Vol. 51, No. 1, 2025, pp. 1-18
DOI: https://doi.org/10.25195/ijci.v51i1.545
Print ISSN: 2313-190X, Online ISSN: 2520-4912

Table 3 provides the training evolution of the LSTM, GRU, and RNN models. All the models reached 100% training accuracy by the ninth epoch, and accuracy improved constantly. Moreover, validation accuracy generally improved as indicated in **Table 3**, with the LSTM model demonstrating better accuracy than the GRU and RNN models.

TABLE III: *Training process of the LSTM, RNN, and GRU models*

Model	Epoch	Training Loss	Training Accuracy	Validation Loss	Validation Accuracy
LSTM	1	0.4049	86.69%	0.1886	92.97%
LSTM	5	0.0151	99.88%	0.0687	97.89%
LSTM	10	0.0009866	100%	0.0676	97.42%
GRU	1	0.3210	88.50%	0.1762	92.51%
GRU	5	0.0163	99.77%	0.0963	96.25%
GRU	10	0.0007387	100%	0.0837	97.19%
RNN	1	0.3275	89.56%	0.1926	93.21%
RNN	5	0.0541	98.42%	0.1035	96.49%
RNN	10	0.0026	100%	0.1006	97.66%

Table 3 indicates that all the models attained full training accuracy early in their training cycles, possibly suggesting overfitting. Nonetheless, the excellent validation accuracy indicates that the model can be effectively generalized to fresh data. This feature is crucial for addressing complicated EEG signals used in emotion categorization tasks [25].

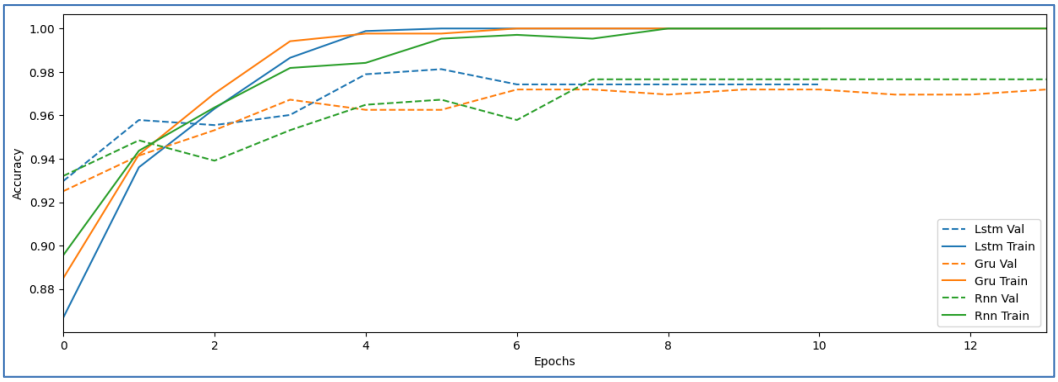

Fig. 9: *Training and validation accuracy of the LSTM, GRU, and RNN models over epochs.*

Figure 9 shows minimal overfitting, as training and validation accuracies remain consistent through applied normalization and early stopping to reduce overfitting, ensuring better generalization.

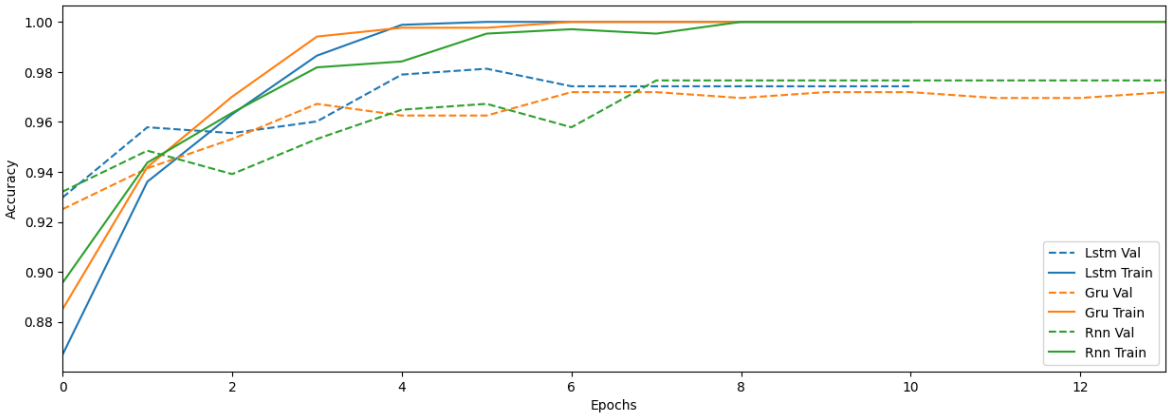

Fig. 10: *Training and validation loss the of LSTM, GRU, and RNN models over epochs.*

Iraqi Journal for Computers and Informatics
Information Technology and Communications University
Vol. 51, No. 1, 2025, pp. 1-18
DOI: https://doi.org/10.25195/ijci.v51i1.545
Print ISSN: 2313-190X, Online ISSN: 2520-4912

Figures 9 and **10** show the accuracy and loss error of the LSTM, GRU, and RNN models. These models efficiently recognize patterns from EEG data, demonstrating quick competence improvements with significant accuracy and minimum loss. The LSTM model exhibits a slight advantage, probably because it can better deal with long-term data dependencies. A lack of overfitting is highlighted by the fact that the training and validation scores of all the models are consistent, indicating strong generalization ability. With the LSTM model exhibiting the best overall metrics, these observations confirm that the models are suitable for identifying emotions by using EEG data. LSTM demonstrates exceptional efficiency in emotion categorization from EEG data, with specific results of 98.13% accuracy, 98.15% precision, and 98.13% F1 score. **Table 4** provides details for the excellent RNN performance with 97.66% accuracy and GRU performance with 96.96% accuracy.

TABLE IV: Results of the testing process

Model	Accuracy	Precision	F1 Score
LSTM	98.13%	98.15%	98.13%
GRU	96.96%	97.03%	96.96%
RNN	97.66%	97.68%	97.66%

7. COMPARISON WITH OTHER WORK

The section compares the proposed work with other studies in the field of EEG-based emotion recognition. The comparison highlights the datasets used, methods employed, results achieved, and the accuracy of each approach. The proposed work demonstrates the effectiveness of integrating advanced deep learning models in emotion detection, as indicated in Table V.

Table V : *Comparison of EEG-based and Emotion Recognition Studies*

Reference	Research Name	Dataset	Methods	Result	Accuracy
[43] Zeng, Y., Wu, Q., Yang, K., Tong, L., Yan, B., Shu, J., & Yao, D. (2018)	EEG-based Identity Authentication Framework That Uses Face RSVP	EEG signals from 45 participants, including users and imposters	HDCA with GA optimization	Achieved robust authentication and high stability over time	94.26% (6 s), 88.88% (30 days interval)
[44] Wang, M., El-Fiqi, H., Hu, J., & Abbass, H. A. (2019	CNN That Uses Dynamic Functional Connectivity for EEG-based Person Identification	PhysioNet BCI dataset (64 electrodes) and self-collected dataset (46 electrodes)	Functional connectivity (PLV) + graph CNN (GCNN)	Achieved high accuracy in identifying individuals across diverse human states	99.96% (Dataset I), 98.94% (Dataset II)
[45] Ketola, Ellen C., et al. 2023	Channel Reduction for an EEG-based Authentication System While Performing Motor Movements	Public WAY_EEG_GAL dataset (3913 trials from 12 participants, 32 channels at 500 Hz sampling)	Feature extraction (12 statistical features), filtering (Butterworth filter), random forest classifier, and channel ranking with Gini Impurity for channel reduction	Successfully reduced EEG channels to 14 while maintaining authentication performance within a 1% accuracy drop from the full 32-channel system	82.25%

Iraqi Journal for Computers and Informatics
Information Technology and Communications University
Vol. 51, No. 1, 2025, pp. 1-18
DOI: https://doi.org/10.25195/ijci.v51i1.545
Print ISSN: 2313-190X, Online ISSN: 2520-4912

[46] TajDini, M., Sokolov, V., Kuzminykh, I., & Ghita, B. (2023)	Brainwave-based Authentication That Uses Feature Fusion	EEG signals from 50 participants (OpenBCI, 16 electrodes)	Feature extraction (spectral info, coherence, mutual correlation, mutual info) + SVM + AdaBoost	Achieved low EER (0.52%) and high classification rates through optimized feature selection	99.06%
Proposed method	EEG-based Emotion Recognition That Uses Deep Learning Models	EEG Brain Wave Dataset from Kaggle [42]	Feature selection (PSO) + deep learning models (LSTM, GRU, RNN)	LSTM achieved the highest performance for emotion classification	98.13% (LSTM), 96.96% (GRU), 97.66% (RNN)

8. CONCLUSION

This study emphasizes the remarkable effectiveness of hybrid optimized feature selection integrated with deep learning models in detecting emotions from EEG data. The LSTM network outperforms all the other evaluated models, rendering it an exceptional option for managing the complicated nature of EEG datasets, wherein precision is essential. The substantial development paves the way for the creation of human–machine interfaces that are more sensitive and responsive, and capable of accurately interpreting and responding to human emotional states.

The superior performance of the LSTM model is primarily due to its capability to efficiently manage long-term reliance within subsequent data. The LSTM model exhibits the ability to accurately identify subtle emotional nuances, which is crucial for improving interactions between humans and machines.

Although the GRU and RNN models also achieve commendable results, their practical application may be affected by different factors, including the computational demands and real-time processing requirement of GRUs. Given their less complex architecture, GRUs maybe preferable in situations wherein simplicity and efficiency are prioritized. Conversely, RNNs, which are distinguished by their simplicity and rapidity, may be appropriate for scenarios that necessitate immediate responses.

In the future, research initiatives will focus on the exploration of real-time applications and further refinement of these models to optimize the benefits of EEG-based emotion recognition technologies. The objective is to improve the real-world applicability of these technologies by increasing their efficiency and effectiveness in real-time emotion detection.

Declarations
Funding: Unfunded project
Ethical approval: Not needed
AI statement: AI tools were employed to aid in the literature review, data analysis, and generation of initial drafts.
Conflict of interest: The study has no conflict with any other work or any kind of interest.
Data availability statement: We can provide data upon request.
Author contributions: The authors contributed equally in the study.

References

[1] Z. Li *et al.*, "Enhancing BCI-based emotion recognition using an improved particle swarm optimization for feature selection," *Sensors*, vol. 20, no. 11, p. 3028, 2020.

[2] M. Dannhauer *et al.*, "Modeling of the human skull in EEG source analysis," *Human Brain Mapping*, vol. 32, no. 9, pp. 1383–1399, 2011.

Iraqi Journal for Computers and Informatics
Information Technology and Communications University
Vol. 51, No. 1, 2025, pp. 1-18
DOI: https://doi.org/10.25195/ijci.v51i1.545
Print ISSN: 2313-190X, Online ISSN: 2520-4912

[3] E. H. Houssein, A. Hammad, and A. A. Ali, "Human emotion recognition from EEG-based brain–computer interface using machine learning: a comprehensive review," *Neural Computing and Applications*, vol. 34, no. 15, pp. 12527–12557, 2022.

[4] I. Ul Hassan *et al.*, "Towards effective emotion detection: A comprehensive machine learning approach on EEG signals," *BioMedInformatics*, vol. 3, no. 4, pp. 1083–1100, 2023.

[5] H. Lian *et al.*, "A survey of deep learning-based multimodal emotion recognition: Speech, text, and face," *Entropy*, vol. 25, no. 10, p. 1440, 2023.

[6] Z. Zainuddin, P. A. EA, and M. Hasan, "Predicting machine failure using recurrent neural network-gated recurrent unit (RNN-GRU) through time series data," *Bulletin of Electrical Engineering and Informatics*, vol. 10, no. 2, pp. 870–878, 2021.

[7] B. Taha, *Eye Biometrics Signal Analysis and Potential Applications in User Authentication and Affective Computing*. Toronto, Canada: University of Toronto, 2023.

[8] P. R. Bhise, S. B. Kulkarni, and T. A. Aldhaheri, "Brain-computer interface based EEG for emotion recognition system: A systematic review," in *2020 2nd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA)*, 2020, IEEE.

[9] Y. Cimtay, E. Ekmekcioglu, and S. Caglar-Ozhan, "Cross-subject multimodal emotion recognition based on hybrid fusion," *IEEE Access*, vol. 8, pp. 168865–168878, 2020.

[10] H. Zhang, "Expression-EEG based collaborative multimodal emotion recognition using deep autoencoder," *IEEE Access*, vol. 8, pp. 164130–164143, 2020.

[11] Z. Wan *et al.*, "HybridEEGNet: A convolutional neural network for EEG feature learning and depression discrimination," *IEEE Access*, vol. 8, pp. 30332–30342, 2020.

[12] M. A. Asghar *et al.*, "Semi-skipping layered gated unit and efficient network: hybrid deep feature selection method for edge computing in EEG-based emotion classification," *IEEE Access*, vol. 9, pp. 13378–13389, 2021.

[13] Z. Liang *et al.*, "EEGFuseNet: Hybrid unsupervised deep feature characterization and fusion for high-dimensional EEG with an application to emotion recognition," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 29, pp. 1913–1925, 2021.

[14] E. Yildirim, Y. Kaya, and F. Kilic, "A channel selection method for emotion recognition from EEG based on swarm-intelligence algorithms," *IEEE Access*, vol. 9, pp. 109889–109902, 2021.

[15] I. S. Ahmad *et al.*, "Deep learning based on CNN for emotion recognition using EEG signal," 2021.

[16] A. Samavat *et al.*, "Deep learning model with adaptive regularization for EEG-based emotion recognition using temporal and frequency features," *IEEE Access*, vol. 10, pp. 24520–24527, 2022.

[17] H. Jiang *et al.*, "EEG emotion recognition using an attention mechanism based on an optimized hybrid model," *Computers, Materials & Continua*, vol. 73, no. 2, 2022.

Iraqi Journal for Computers and Informatics
Information Technology and Communications University
Vol. 51, No. 1, 2025, pp. 1-18
DOI: https://doi.org/10.25195/ijci.v51i1.545
Print ISSN: 2313-190X, Online ISSN: 2520-4912

[18] M. Tahir *et al.*, "A novel binary chaotic genetic algorithm for feature selection and its utility in affective computing and healthcare," *Neural Computing and Applications*, pp. 1–22, 2022.

[19] M. d. M. Haque *et al.*, "EEG-based multi-class emotion recognition using hybrid LSTM approach," *International Journal of Innovative Research in Computer Science & Technology*, vol. 11, no. 3, pp. 1–6, 2023.

[20] A. Nakra and M. Duhan, "Deep neural network with harmony search-based optimal feature selection of EEG signals for motor imagery classification," *International Journal of Information Technology*, vol. 15, no. 2, pp. 611–625, 2023.

[21] S. Bagherzadeh *et al.*, "Developing an EEG-based emotion recognition using ensemble deep learning methods and fusion of brain effective connectivity maps," *IEEE Access*, 2024.

[22] S. Wang *et al.*, "Multimodal emotion recognition from EEG signals and facial expressions," *IEEE Access*, vol. 11, pp. 33061–33068, 2023.

[23] J. Bird, "EEG Brainwave Dataset: Feeling Emotions," Kaggle, [Online]. Available: https://www.kaggle.com/datasets/birdy654/eeg-brainwave-dataset-feeling-emotions, 2019.

[24] H. A. Gonzalez, J. Yoo, and I. M. Elfadel, "EEG-based emotion detection using unsupervised transfer learning," in *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2019, IEEE.

[25] M. Algarni *et al.*, "Deep learning-based approach for emotion recognition using electroencephalography (EEG) signals using bi-directional long short-term memory (Bi-LSTM)," *Sensors*, vol. 22, no. 8, p. 2976, 2022.

[26] X. Li *et al.*, "Application of gated recurrent unit (GRU) neural network for smart batch production prediction," *Energies*, vol. 13, no. 22, p. 6121, 2020.

[27] I. D. Mienye, T. G. Swart, and G. Obaido, "Recurrent neural networks: A comprehensive review of architectures, variants, and applications," Information, vol. 15, no. 9, p. 517, 2024.

[28] S. F. Ahmed et al., "Deep learning modelling techniques: Current progress, applications, advantages, and challenges," Artificial Intelligence Review, vol. 56, no. 11, pp. 13521–13617, 2023.

[29] A. Tashakkori et al., "Enhancing stock market prediction accuracy with recurrent deep learning models: A case study on the CAC40 index," World Journal of Advanced Research and Reviews, vol. 23, no. 1, pp. 2309–1, 2024.

[30] S. Krishnan, P. Magalingam, and R. Ibrahim, "Hybrid deep learning model using recurrent neural network and gated recurrent unit for heart disease prediction," International Journal of Electrical & Computer Engineering (2088-8708), vol. 11, no. 6, 2021.

[31] M. Waqas and U. W. Humphries, "A critical review of RNN and LSTM variants in hydrological time series predictions," MethodsX, 2024, Art. no. 102946.

Iraqi Journal for Computers and Informatics
Information Technology and Communications University
Vol. 51, No. 1, 2025, pp. 1-18
DOI: https://doi.org/10.25195/ijci.v51i1.545
Print ISSN: 2313-190X, Online ISSN: 2520-4912

[32] I. D. Mienye, T. G. Swart, and G. Obaido, "Recurrent neural networks: A comprehensive review of architectures, variants, and applications," Information, vol. 15, no. 9, p. 517, 2024.

[33] T. Fadziso, "Overcoming the vanishing gradient problem during learning recurrent neural nets (RNN)," Asian Journal of Applied Science and Engineering, vol. 9, no. 1, pp. 197–208, 2020.

[34] C. B. Vennerød, A. Kjærran, and E. S. Bugge, "Long short-term memory RNN," arXiv preprint, arXiv:2105.06756, 2021.

[35] N. Oluwafemi, "Recurrent Neural Networks-Architectures and Applications: Analyzing architectures and applications of recurrent neural networks (RNNs) for modeling sequential data and time-series prediction," Australian Journal of Machine Learning Research & Applications, vol. 3, no. 2, pp. 239–249, 2023.

[36] M. Lechner and R. Hasani, "Learning long-term dependencies in irregularly-sampled time series," arXiv preprint, arXiv:2006.04418, 2020.

[37] W. Zheng and G. Chen, "An accurate GRU-based power time-series prediction approach with selective state updating and stochastic optimization," *IEEE Transactions on Cybernetics*, vol. 52, no. 12, pp. 13902–13914, 2021.

[38] B. Becker, "Advanced recurrent neural networks to study the temporal sequence modeling," *International Journal of Research and Review Techniques*, vol. 3, no. 2, pp. 36–46, 2024.

[39] M. Waqas and U. W. Humphries, "A critical review of RNN and LSTM variants in hydrological time series predictions," *MethodsX*, 2024, Art. no. 102946.

[40] A. G. Gad, "Particle swarm optimization algorithm and its applications: A systematic review," *Archives of Computational Methods in Engineering*, vol. 29, no. 5, pp. 2531–2561, 2022.

[41] J. Nayak *et al.*, "25 years of particle swarm optimization: Flourishing voyage of two decades," *Archives of Computational Methods in Engineering*, vol. 30, no. 3, pp. 1663–1725, 2023.

[42] J. Bird, "EEG Brainwave Dataset: Feeling Emotions," Kaggle, 2019. [Online]. Available: https://www.kaggle.com/datasets/birdy654/eeg-brainwave-dataset-feeling-emotions/dat

[43] Zeng, Y., Wu, Q., Yang, K., Tong, L., Yan, B., Shu, J., & Yao, D. (2018). " EEG-based identity authentication framework using face rapid serial visual presentation with optimized channels". Sensors, 19(1), 6.

[44] Wang, M., El-Fiqi, H., Hu, J., & Abbass, H. A. (2019). "Convolutional neural networks using dynamic functional connectivity for EEG-based person identification in diverse human states". IEEE Transactions on Information Forensics and Security, 14(12), 3259-3272.

[45] Ketola, Ellen C., et al. "Channel Reduction for an EEG-Based Authentication System While Performing Motor Movements." Sensors 22.23 (2022): 9156.

[46] TajDini, M., Sokolov, V., Kuzminykh, I., & Ghita, B. (2023). "Brainwave-based authentication using features fusion". Computers & Security, 129, 103198.