



Enhanced Emotion Recognition Through the Integration of Gated Recurrent Unit and Convolutional Neural Networks Using MindWave Mobile EEG Device

Mahdi Hamdi^{ID}, Timur Inan^{ID}

Information Technology Department, Altinbas University, Istanbul 34083, Turkey

Corresponding Author Email: 213721503@ogr.altinbas.edu.tr

<https://doi.org/10.18280/mmp.100514>

ABSTRACT

Received: 12 June 2023

Revised: 20 August 2023

Accepted: 5 September 2023

Available online: 27 October 2023

Keywords:

emotion classification, EEG Signals, MindWave Mobile, emotion recognition, deep learning methods neural networks algorithms for EEG, GRU method, CNN algorithm

Emotion recognition utilizing MindWave signals and neural networks presents a substantial challenge due to the inherent complexity of human emotions and the variability of individual brainwaves. The selection of the appropriate algorithm, dictated by the problem and available data, necessitates an understanding of each algorithm's unique strengths and weaknesses. Previous studies have predominantly focused on the classification of emotions through EEG signals employing various standalone neural network algorithms. However, our study fills a notable research gap by integrating Convolutional Neural Networks (CNN) and Gated Recurrent Units (GRU). This innovative combination yields improved testing performance and accuracy, setting a benchmark in the realm of emotion recognition. The process encompasses the collection of MindWave data, the elimination of noise through preprocessing, the extraction of features indicative of emotional states, and the training of a neural network using labeled data. Finally, the network's accuracy is evaluated on novel data. By addressing the unique challenges and complexities associated with emotion classification using EEG signals, this study provides a promising and advanced approach towards the understanding and recognition of human emotions, paving the way for potential real-world applications.

1. INTRODUCTION

Emotions, fundamental to human expression, greatly influence daily decisions and normal activities. In the current era, artificial intelligence (AI) techniques are being harnessed for recognizing human emotions, thereby enhancing the progression of human-computer interaction. The brain, the central organ for information processing and management within the human body, generates physiological signals that are captured and scrutinized via electroencephalography (EEG). The limbic system, the brain's control center, is a four-tiered system responsible for emotion regulation and motivation. In most instances, emotions are intricately linked with behavioral patterns, and an individual's genuine thoughts can provide insight into their feelings, emotional states, and psychological conditions [1]. Emotions are integral to decision-making in daily life and physiological activities [2]. Consequently, EEG signals hold the potential to provide real-time insights into an individual's current emotional state.

Electroencephalogram (EEG) devices are primarily designed to record the electrical activities that transpire in the brain during various physical and chemical brain functions. These brain-generated signals vary in frequency and amplitude depending on the individual's state, whether awake, asleep, or in different mood conditions. Deemed as aperiodic signals, brain waves are segregated into five distinct frequency bands of EEG signals: Theta, Alpha, Delta, Beta, and Gamma. Each band is confined within specific frequency values measured in Hertz [3].

Numerous methods exist for emotion detection,

identification, and feature extraction, with EEG signals being one of the most significant [4]. In the twentieth century, psychologist Paul Ekman categorized feelings into emotions closely associated with human physiological responses. Hence, through brain wave emotion classification, categories of feelings can be established [5]. For EEG data collection, devices equipped with electrodes are positioned on the scalp. In this study, the MindWave Mobile device was utilized to acquire EEG signals. The device comprises a headset, ear clip sensors, and a forehead clip, the latter grounding the EEG electrode. Figure 1 illustrates the device's structure [6].

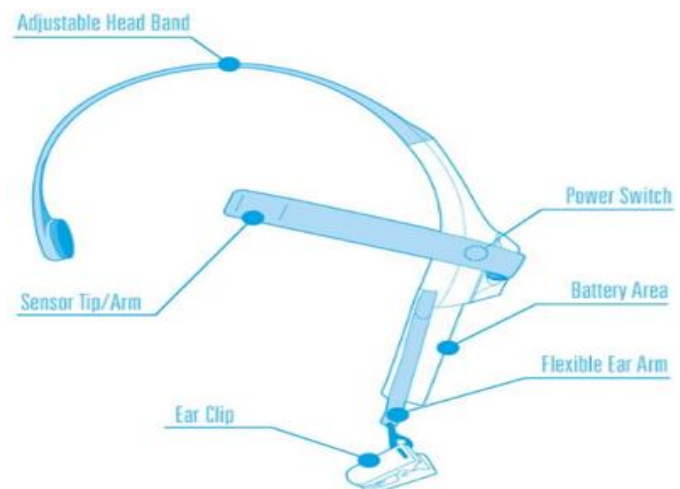


Figure 1. MindWave device [7]

The primary objective of this study is to discern emotions utilizing EEG signals, specifically through the employment of the MindWave Mobile device. The creation of a unique dataset for this research not only facilitates practical comparison of algorithms and methods but also enables the integration of various methods to yield enhanced results during analysis and conclusion phases. This comprehensive approach distinguishes the study, addressing the limitations of previous research that relied solely on pre-existing datasets and implemented certain methods in a restricted manner.

2. LITERATURE REVIEW

In previous studies, various approaches have been employed to address the task of emotion classification using EEG signals. These approaches investigate the relationship between brain activity and emotional responses, utilizing different sensory stimuli such as visual and auditory cues like words, images, and video clips to influence an individual's emotional state. For instance, Tomarken et al. [7] hypothesized that resting forehead asymmetry can predict emotional affect and conducted research using electroencephalography (EEG) to explore this relationship. Similar findings were reported by Davidson, who indicated that brain electrical activity is associated with emotional responses [8].

Numerous studies have utilized EEG signals to classify and examine emotions. Lee and Hsieh focused on communication patterns dependent on EEG signals for classification [9]. Wang et al. [10] conducted an experiment to record emotional states through EEG signals while participants were exposed to emotionally stimulating films. Wang et al. [10] utilized self-assessment by their research (SAM) and created a dataset of six volunteers to investigate emotion state classification based on EEG signals, employing SVM for accuracy calculation. Murugappan et al. [11] employed the discrete wavelet transform to analyze EEG signals and highlighted the challenges researchers face in collecting EEG data using clinical EEG devices. Peining et al. [12] added that while clinical devices offer high accuracy, they pose challenges for participants and patients. Clinical data collection restricts movement, and electrodes placed on the scalp may leave residue on the hair. Moreover, the high cost of medical devices has led to the development of commercial alternatives to address these issues.

The emergence of deep learning techniques has garnered attention in the field of emotion recognition based on brain waves. Convolution Neural Networks (CNNs) have been widely utilized in EEG-based emotion classification tasks due to their ability to automatically learn spatial and temporal patterns in data. Zhang et al. [13] proposed a deep CNN architecture for emotion recognition from EEG signals, outperforming conventional machine learning techniques. In another study, Liu et al. [14] presented an emotion classification approach using Bi-LSTM, which amalgamates both forward and backward information from input sequences. This method utilizes Bi-LSTM by combining forward and backward LSTMs. The study applied this model to improve accuracy on the DEAP dataset.

Recurrent Neural Networks (RNNs), including variants like LSTM and GRU, are suitable for modeling sequential dependencies in time-series data like EEG signals. Researchers have employed RNN architectures for emotions classification. Yang et al. [15] introduced a multi-channel

LSTM network for emotion recognition from EEG, demonstrating competitive accuracy and robustness to artifacts. Phan et al. [16, 17] proposed a hybrid LSTM-CNN architecture that effectively captured both temporal and spatial features in EEG signals, resulting in improved emotion classification performance.

Deep Belief Networks (DBNs) have also been utilized in emotion classification tasks to learn hierarchical representations of EEG data. DBNs consist of stacked Restricted Boltzmann Machines (RBMs) that progressively learn more abstract features. Yao et al. [18] utilized a DBN-based approach for emotion recognition from EEG signals, achieving notable performance across various datasets.

Ensemble methods, which combine multiple classifiers, have been explored for EEG-based emotion classification. Chen et al. [19] proposed an ensemble framework that integrated multiple CNN and LSTM models, leveraging the complementary strengths of each model to enhance accuracy on emotion classification tasks.

These approaches and architectures highlight the ongoing efforts in the field of emotion classification using EEG signals, aiming to improve accuracy and robustness in capturing and understanding human emotions.

In this study, algorithms and classification methods were highlighted and compared in terms of accuracy and loss. Where the model was trained using LSTM alone, then using GRU alone, and finally integrating the GRU with CNN, on a data set that was specially configured in this study using the MindWave device and processing it. After training, the initial results for both LSTM and GRU showed an accuracy rate ranging between 65-73%, while the final results showed preference using the hybrid algorithm between GRU and CNN, where the results were for accuracy 80-83%. Note that the results could be much better if ready-made data sets were used, because the used data set contains fewer shapes and the reason is that the use of commercial intellectual wave devices contains fewer sensors, which limits the possibility of capturing more shapes and signals, unlike EEG devices medical.

3. METHODOLOGY

There are hundreds of thousands of emotions in humans, studies indicate that there are different characteristics in different emotions, classifying emotions is a complex research problem due to the diverse range of expressions that emotions can take in daily life. It can be simply divided into three states: positive, neutral, and negative. But emotion is basically a synthesis of many feelings, and it is considered a physiological state. Therefore, psychologists tend to define emotions from a multidimensional perspective. Some researchers divided feelings into positive, negative, and neutral, in addition, some researchers have categorized emotions into five distinct categories: "happiness, fear, sadness, disgust, and relaxation". Visual stimuli are commonly utilized to evoke these specific emotions for research and analysis purposes [20].

In our study, emotions were divided into five categories, including (joy, sadness, natural state, anger, and calmness). The data was collected through the MindWave Mobile equally for you emotion as shown in Figure 2.

These are the main emotions in humans, in which clarity and observation lie. However, there is some confusion between these feelings, and the data may converge in some

cases. For example, the state of nature and the state of calm are close in that they are closer to being one state. Likewise in the case of anger and sadness. This also adds a challenge in capturing the signals accurately and how to classify each of these five emotions. These cases were named by collecting data for a group of people through a supervised pattern and displaying some external influences to focus on the emotion to be captured first. Among the external influences that stimulate emotion in people are videos, music, virtual game challenges, and verbal influences.

In general, any EEG-based emotion classification model goes through several stages. This flowchart is an overview of the classification process and that the steps and methods can differ depending on the research question and its methodology (Figure 3).

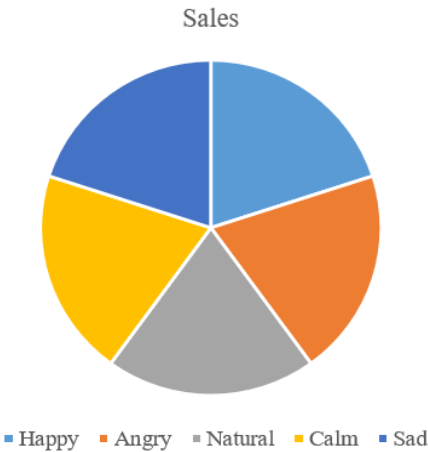


Figure 2. Emotional type

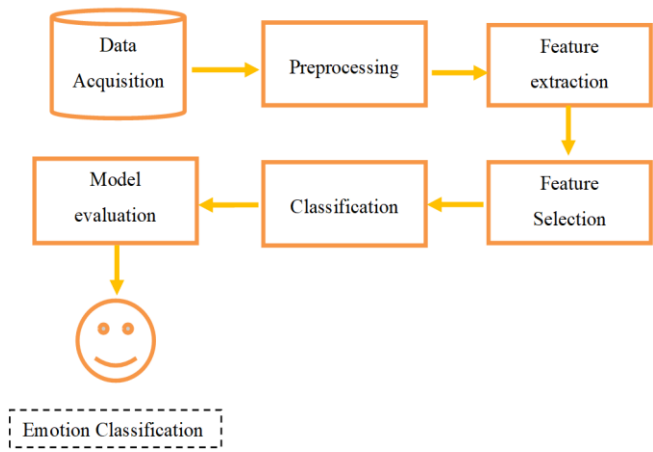


Figure 3. General steps for emotion recognition

3.1 Data collection stage

This model was trained on five types of data extracted by the EEG from three different subjects at different times. Using supervised technology, the MindWave Mobile device was attached to the scalp to collect EEG data for each person. A number of videos with different effects on emotions were played. Data were collected for each case while viewing these clips. Then classifying this data set according to the psychological state of that person while watching the video clip, whether the state was (happiness, anger, calmness, sadness, and neutral state). Things may get mixed up in some

cases, as in cases of sadness and anger, and sometimes this causes problems in categorization. Therefore, the state of anger was affected by stimulating the person emotionally in virtual games and losing him in a game, as well as provoking the person through verbal altercation, which had an effect on capturing the signals of the state of anger. This goes through the stages of preparation and data processing. The MindWave device was first powered on to make sure it was working fine. Then the MindWave library was installed in Python, and this library can be found on Neurosky's official website or through GitHub. Then to connect the MindWave headset to the computer via USB or Bluetooth, a special function must be called from the library to initiate the connection. After successful communication processes, the EEG can be accessed through steps specified in the operating system and programming language. Once this is done, the data reading process can start. This data includes ('Timestamp', 'Raw', 'Attention', 'Meditation', 'delta', 'theta', 'low-alpha', 'high-alpha', 'low-beta', 'high-beta', 'low-gamma', 'mid-gamma').

3.2 Data label

During the data collection process through the MindWave device, the EEG data were labelled with the corresponding feelings during the reading process by means of the supervised technique. Separately, labeling data were created for the divided data for each of the states (happy, sad, calm, angry, and neutral). Then it was loading the all matrixes and combining in one dataset (called all_data) as shown in Table 1.

3.3 Data preparation

Preparing segmented EEG data and corresponding labels for input into the neural network. These include reshaping the data into a 2D or 3D matrix, normalizing the data, and dividing the data into training and test sets. Through machine learning libraries or SKlearn library in the implementation of this pre-processing.

3.4 Pre-processing

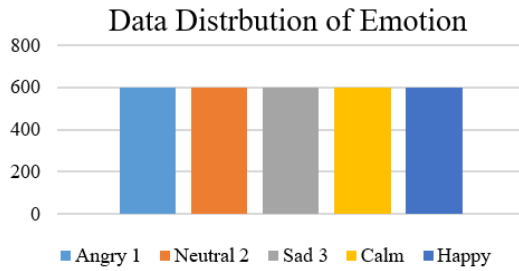
In this study, a comprehensive data preprocessing pipeline was employed to refine the all_data data frame, which is integral to the investigation of emotion recognition from EEG data. First, any rows containing missing (NaN) values were meticulously removed from the dataset, ensuring data integrity. Subsequently, extraneous columns, such as timestamps and channel names, were systematically eliminated to streamline the dataset for analysis. To facilitate a deeper exploration of emotions, the data was further segmented into separate data frames based on distinct emotion categories, achieved through meticulous filtering using the loc method. Additionally, to provide a concise and representative view of the data, "sample" data frames were meticulously crafted for each emotion category, each comprising a single randomly selected row from the respective emotion-specific data frame. This curated dataset and the resultant sample data frames serve as a foundational component of the study, enabling a comprehensive investigation into emotion distribution and patterns within EEG data. These preprocessing steps lay the groundwork for rigorous analysis and hold potential implications for diverse applications, ranging from emotion recognition to EEG-based research in cognitive sciences.

Table 1. EEG dataset after combination and label them

Attention	Meditation	...	Low-Gamma	Mid-Gamma	Emotion
48	24	...	29566	5583	Happy
66	60	...	3361	1444	Angry
93	78	...	2020	2183	Neutral
75	50	...	10090	5472	Neutral
53	20	...	3265	3116	Calm
...	...	...	...	...	...
87	81	...	1988	2203	Neutral
47	56	...	3198	2417	Neutral
54	17	...	527	708	Calm
67	51	...	2270	1145	Happy
66	57	...	3757	6254	Happy

[3000 rows×12 columns]

The distribution of feelings within a data frame after performing the prior operations on it before the input process. Figure 4 shows the emotion distribution process.

**Figure 4.** Emotion distribution process

The operation under consideration represents a quintessential procedure integral to the scientific domain of data preprocessing and transformation. This process is aptly denominated as "data conditioning" within the context of machine learning research. Within this meticulously designed Python function, denoted as `Transform_data`, an array of crucial tasks is orchestrated on the input dataframe, denoted as `all_data`. These encompass the essential conversion of emotion labels into numerical encodings, the segregation of EEG signals and emotion labels into distinct variables, meticulous standardization of EEG signal values, and the consequential one-hot encoding of emotion labels. These orchestrated tasks serve the fundamental purpose of ensuring the dataset's alignment with the prerequisites for efficacious machine learning model training and validation. Subsequently, the dataset is judiciously bifurcated into training and testing partitions, an imperative facet in the scientific inquiry of model evaluation. The focal point of interest lies in ascertaining the count of features, signifying the dimensional complexity of the training dataset, and wielding substantial significance in modeling endeavors. This methodical data preprocessing operation, as elucidated in the academic paper, occupies a seminal role in expediting the precision and effectiveness of machine learning model development, hence warranting meticulous documentation and exposition.

3.5 Architectures used for emotion classification

To evaluate the process of categorizing emotions using EEG signals, accuracy as an indicator of competence is used in many studies and extensively in the literature. Therefore, the quality of the method used for classification is judged to be accurate as the key in how traits are classified.

Various methods have been extensively researched for

emotion classification, including Support Vector Machines (SVM), Linear Discriminant Analysis (LDA), GRU, LSTM, RNN, and CNN.

3.5.1 SVM

SVM is a popular machine learning model used for regression analysis and statistical classification. It is a binary model and a linear classifier, assuming that the classifier objects are linearly separable. SVM defines the most prominent margin in the feature space and utilizes interval maximization as a learning strategy. Guo et al. [21] employed Fuzzy Cognitive Graph (FCM) and SVM in their research for emotion recognition, incorporating both facial expressions and EEG signals. They conducted experiments on a comprehensive dataset and performed a deep analysis of the data. To reduce noise, the researchers divided the data into short time periods and applied data compression techniques to minimize spacing between the data. They utilized the wavelet transform method for feature extraction, which was then used in the classification process. Kumar conducted an empirical comparison using SVM between a deep dataset and a SEED dataset, achieving good accuracy in both cases [22].

SVM has its advantages and disadvantages. One of its advantages is its ability to effectively reflect the brain's state. By employing signal processing techniques in the frequency and time domains, it can analyze static signals and leverage the benefits of irregular movement. It offers high computational efficiency and allows for the extraction of relevant information from the data. However, it may not perform well with unstable signals, and its computational complexity can be a drawback due to the large number of arithmetic operations [23-25].

The mathematical formulation of SVM involves finding an optimal hyperplane that maximizes the margin between the two closest data points from different classes in a training dataset. These data points, known as "support vectors," lie on the boundary of the decision regions.

Solve the optimization problem by the following mathematical formula:

Minimize:

$$\left(\frac{1}{2}\right) * ||w||^2 + C * \sum \xi_i \quad (1)$$

Subject to:

$$Y_i * (w^T * X_i + b) \geq 1 - \xi_i, \text{ for all } i = 1 \text{ to } N \\ \xi_i \geq 0, \text{ for all } i = 1 \text{ to } N \quad (2)$$

SVM can also be applied to classify emotions using EEG

data. Here's a mathematical explanation of SVM for emotion classification:

Data Representation: The data of EEG is typically displayed as an Array X , where each column denotes a feature and each row a sample (e.g., power spectral density, frequency bands, etc.). The corresponding emotion labels are represented as a vector Y .

Formulation for Binary Classification: SVM can be used for binary emotion classification by assigning labels of +1 and -1 to the two emotions of interest. The objective is for finding the hyperplane that separates the two classes with the maximum margin.

Mathematical Formulation: The mathematical formulation for SVM in binary classification can be expressed as:

Minimize:

$$\left(\frac{1}{2}\right) * ||w||^2 + C * \sum \xi_i \quad (3)$$

Subject to:

$$Y_i * (w^T * X_i + b) \geq 1 - \xi_i, \text{ for all } i = 1 \text{ to } N$$

$$\xi_i \geq 0, \text{ for all } i = 1 \text{ to } N \quad (4)$$

In this formulation, the weight vector is denoted by W , perpendicular to the hyper-plane, b is the bias term, ξ_i are slack variables, C is the regularization parameter, (X_i) represents the (i-th) samples in the data matrix X , and Y_{in} is the corresponding emotion label.

Kernel Trick: To handle non-linearly separable data, SVM employs the kernel trick. the incoming data is transformed into a higher-dimensional feature space, the SVM can find a hyperplane that effectively separates the classes. Frequently employed kernels for classifying emotions in EEG data consist of linear, sigmoid, polynomial, and radial basis function kernels.

Multi-Class Classification: SVM is inherently a binary classifier. To extend it for multi-class emotion classification, several strategies can be used, such as “one-vs-one” or “one-vs-rest”. In one-vs-one, multiple SVM models are trained, each comparing pairs of emotions. In one-vs-rest, separate SVM models are trained for each emotion against the rest.

Training and Prediction: Once the SVM model is trained using the labeled EEG data, it can be used to classify new, unseen EEG samples. The sign of $(w^T * X + b)$ determines the predicted emotion label.

3.5.2 CNNs

Numerous traditional “machine learning” techniques have been utilized for emotion classification and have shown Certain achievements. Nonetheless, such approaches are hindered by certain drawbacks, including challenges encompass difficulties in extracting features, achieving low accuracy rates, and encountering limited stability. Research indicates that deep learning provides a more effective approach to emotion detection and is particularly suitable for analyzing and identifying physiological signals. The use of deep learning in emotion recognition has become increasingly prevalent because of its extraordinary capacity for learning and flexibility [26].

CNNs have both advantages and disadvantages when it comes to emotion recognition using EEG signals.

Advantages:

CNNs are effective in identifying patterns in large and complex datasets, which is useful in analyzing the complex

EEG signals for emotion recognition.

They can learn complex feature representations in a hierarchical manner, allowing them to take the raw EEG signals and extract the high-level features.

CNNs can handle the high dimensionality of EEG data effectively, making them suitable for use in emotion recognition tasks.

They are robust to noise and artifacts, which is essential when working with EEG signals that are susceptible to various sources of interference.

Disadvantages:

CNNs require a large amounts of s labeled data for training, which may be challenging to obtain in the case of EEG-based emotion recognition.

The interpretability of CNNs is limited, making it challenging to understand the features that the network has learned from the EEG data.

But when using large data sets and complex networks, it can be expensive in computational operations.

CNNs can overfit the training data, which may result in poor generalization to un-seen EEG signals.

CNNs are a deep learning algorithm that is particularly well-suited for handling large datasets due to its unique architecture. Unlike traditional artificial neural networks with three layers, CNNs incorporate two additional layers: the pooling layer and the con-convolution layer. The convolution layer plays a crucial role in extracting relevant features from EEG data and reducing the influence of noise. The pooling layer performs information filtering and feature selection. Once the features are extracted and selected, the fully connected layer combines them as inputs to the output layer, which employs SoftMax functions for classification. CNNs consist of numerous neurons organized in a “three-dimensional coordinate system”, allowing for efficient processing of large datasets [27, 28]. The CNN structure in Figure 5.

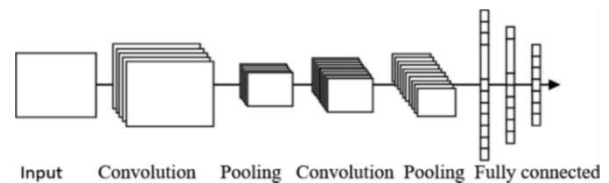


Figure 5. CNN structure [29]

Yang's proposal for sentiment recognition is a multi-column CNN model which includes multiple recognition units whose primary reliance is on one-dimensional clustering and convolution layers CNN. In their study, the researchers used the DEAP dataset as experimental data, which was preprocessed by reducing frequency and analyzing and sampling EEG signals with a bandpass filter multiple time. The preprocessed data was then utilized as input for the recognition module. The model used a weighted averaging of the decisions made by each module to ensure accurate recognition results. The model code was implemented using Python and Pytorch libraries. Experimentally results revealed that the model achieved a high valence rate of 90.01% and arousal rate of 90.65% [15, 29].

Here's a brief explanation of the mathematical components involved in CNNs for EEG-based emotion classification:

Convolution Operation: The convolutional operation is a fundamental component of CNNs. It involves sliding a filter (also known as a kernel) over the input EEG data, performing element-wise multiplication, and summing the results to

generate feature maps. The mathematical calculation for a convolution operation:

$$FeatureMap[i, j] = \text{sum}(Input[i + k, j + l] * Filter[k, l]) \quad (5)$$

Activation Function: After each convolutional operation, an activation function is applied element-wise to introduce non-linearity. Common activation functions used in CNNs include ReLU, sigmoid, or hyperbolic tangent (tanh) functions.

Pooling Operation The pooling process preserves the most prominent features and reduces the spatial dimensions of feature maps. Max pooling is commonly used in CNNs, where the maximum value within each pooling region is selected. The mathematical process of max pooling can be represented as:

$$PooledFeatureMap[i, j] = \max(FeatureMap[i + k, j + l]) \quad (6)$$

In a fully connected layer, the extracted features from the previous layer are passed through a set of neurons, where each neuron is connected to every neuron in the previous layer. The calculation in a fully connected layer can be represented mathematically as follows:

$$Output[i] = \text{activation_function}(\text{sum}(Weight[i, j] * Input[j]) + Bias[i]) \quad (7)$$

The soft magnetization function is commonly used in the final layer of a neural-network for multi-class classification tasks, such as emotion classification. It takes the raw output values from the previous layer and transforms them into a probability distribution over the different classes:

$$P(\text{class } i) = \frac{\exp(Output[i])}{\sum(\exp(Output[j]))} \text{ to all classes} \quad (8)$$

Loss Function and Optimization: The choice of loss function depends on the specific problem. For multi-class classification, cross-entropy loss is commonly used. The CNN is trained by optimizing the weights and biases to minimize the loss using optimization algorithms such as stochastic gradient descent (SGD), Adam, or RMSprop.

The above mathematical components are combined to create a CNN architecture for emotion classification using EEG data. The architecture may vary depending on the specific requirements and complexity of the task. Additionally, data preprocessing, data augmentation, regularization techniques, and hyperparameter tuning are essential aspects to consider for achieving optimal performance in CNN-based emotion classification systems.

3.5.3 RNN

RNN is a deep learning network that has a unique structure distinguishing it from other neural networks. Unlike most networks that only consider the current input, an RNN incorporates a "memory" function, allowing it to take into account inputs at any given time. This enables RNNs to process sequential data by retaining information from previous calculations and utilizing it to influence the Output of the current input. The RNN structure consists of an input layer, a hidden layer, and an output layer, and it can be combined with

other neural networks to extract temporal characteristics from EEG signals. Following feature extraction and selection, the output layer utilizes the SoftMax function for classification. RNNs are particularly suitable for processing sequential data, such as continuous natural language or lengthy text sections, as they can remember past calculations and use that information to predict the next segment of the sequence [29].

Esmeralda Contessa Djamal proposed a model for emotional recognition that combines wavelet transform (WT) and RNN to detect emotions such as sadness, relaxation, and pleasure. The model, developed using Python and TensorFlow, employed EEG signals generated by emotional audio and visual stimuli from ten healthy individuals. The researcher converted the EEG readings using WT into brain waves associated with emotions, which were then fed into an RNN model for recognition. EEG data from four channels were collected to ensure the accuracy of the findings. Experimental results demonstrated recognition rates of 92% for sorrow, 53% for relaxation, and 97% for happiness, indicating that the combination of wavelet transforms and RNN is a reliable technique for emotion categorization [30].

To categorize emotional aspects in EEG data, Tao et al. [31] proposed a model called ACRNN, which combines CNN and RNN with an attention mechanism. RNN is utilized to capture temporal features, while CNN is employed to extract spatial features. The ACRNN model was trained using EEG and ECG signals generated by unemotional stimuli and implemented using Python and Tensor-Flow. Noise in the EEG readings was reduced using blind source separation. The ACRNN model, employing the soft max function in RNN, achieved accurate emotion classification. The research revealed that both the CNN and CNN-LSTM models performed well in extracting emotional features.

The mathematical formulation of a RNN involves capturing sequential information and utilizing recurrent connections. Here's an overview of the key mathematical formulations:

Hidden State Update:

In an RNN, the hidden state captures information from previous time steps and influences the current prediction. Updates are made to the concealed state at time step t , represented by " $h(t)$ ", is updated using the input at time step t , denoted by " $x(t)$ ", and the previous hidden state, denoted by " $h(t-1)$ ". The mathematical calculation for the hidden state update can be expressed as:

$$h(t) = \text{activation_function}(W_{xh} * x(t) + W_{hh} * h(t-1) + b_h) \quad (9)$$

where, W_{xh} represents the weight matrix for the input, W_{hh} represents the weight matrix for the hidden state, and b_h represents the bias term. The activation function is typically a non-linear activation function like tanh or sigmoid.

Output calculation:

The output of the RNN at each time step is calculated based on the current hidden state. The output at time step t , denoted by $y(t)$, is obtained by applying a weight matrix, W_{yh} , to the hidden state and adding a bias term b_y . The mathematical calculation for the output can be expressed as:

$$y(t) = \text{activation_function}(W_{yh} * h(t) + b_y)$$

where, " W_{yh} " is the weight matrix for the output, and b_y represents the bias term.

Sequence unfolding: To process a sequence of inputs, the

RNN is typically unfolded over time, creating a series of interconnected RNN cells. Each RNN cell represents a time step, and the hidden state is updated sequentially based on the previous hidden state and the input.

Backpropagation Through Time (BPTT): Training an RNN involves propagating the error gradient back through time. This process, known as BPTT, adjusts the weights and biases to minimize the loss function. BPTT essentially extends the standard backpropagation algorithm to handle the sequential nature of RNNs.

These are the key mathematical formulations for an RNN. Variations of RNNs, such as LSTM and GRU, introduce additional equations and mechanisms to address issues like the vanishing gradient problem and improve the modeling of long-term dependencies.

3.5.4 LSTM

Traditional machine learning models often face challenges such as complex computations and low efficiency, as mentioned in reference [32]. However, in the field of emotion recognition, the LSTM neural network, which belongs to the domain of deep learning, has gained significant attention due to its unique network architecture [33]. LSTM is a specialized type of recurrent neural network [34] that incorporates a "gate" mechanism, setting it apart from the simple cyclic structure of traditional RNNs. The LSTM architecture consists of an input layer, a recurrent body structure, and an output layer that can selectively retain or discard information using the "gate" mechanism, thus enabling memory storage or forgetting [35]. The structure of an LSTM unit is illustrated in Figure 6.

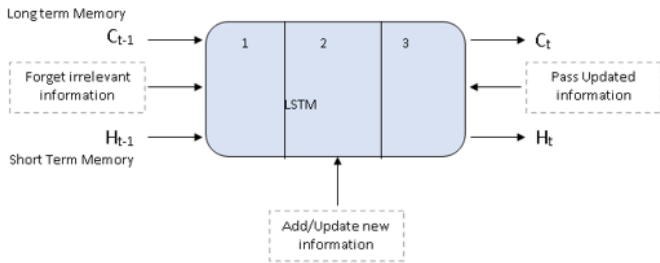


Figure 6. LSTM structure

Oblivion Gate: It is considered the first gate of the gates. All old memories pass through when this portal opens fully. While when closed, Past memories won't be retained. It consists of multiplying one element with another. The forgetting gate equation can be expressed as follows:

$$f_t = \sigma(X_t * U_f + H_{t-1} * W_f) \quad (10)$$

To erase most of the old memory, multiplied by a vector that is nearly zero. Whereas, when setting the forgetting gate to the value of 1, this allows the old memory to pass through, as in the following equation:

$$C_{t-1} * f_t = 0 \text{ (if } f_t = 0 \text{)} \quad (11)$$

$$C_{t-1} * f_t = C_{t-1} \text{ (if } f_t = 1 \text{)} \quad (12)$$

When X_t and H_{t-1} are inserted, they represent the current input timestamp and the hidden state of the previous timestamp, respectively. The weights associated with the input and hidden states are expressed by U_f and W_f .

The input gate is the second gate in the LSTM unit. It determines how much new information should be allowed in. It considers that new and old memories may be influenced differently by adjusting this gate. The input gate measures the significance of the new data that the input carries. The equation for the input gate is as follows:

$$i_t = \sigma(X_t * U_i + H_{t-1} * W_i) \quad (13)$$

where, W_i and U_i represent the weights associated with the current input and the previous hidden state.

The cell state, denoted by C_t , is a combination of the old memory and the current in-put. It combines the elements of the old memory and the current input using element-wise addition.

$$\bar{C}_t = \tanh(X_t * U_c + H_{t-1} * W_c) \text{ (New in formation)} \quad (14)$$

$$C_t = f_t * C_{t-1} + i_t * \bar{C}_t \text{ (Updating Cell State)} \quad (15)$$

The output gate generates the output for the LSTM unit. It is influenced by the new memory, the current input, and the previous output. The output gate controls the amount of new memory that should be passed on to the next LSTM unit.

$$O_t = \sigma(X_t * U_o + H_{t-1} * W_o) \quad (16)$$

The current hidden state is determined by using the modified cell state C_t and ap-plying the tanh function. The output gate O_t is also considered, as its value is between 0 and 1 due to the sigmoid function.

$$H_t = X_t * \tanh(C_t) \quad (17)$$

The function for current output and long term memory (C_t) is the hidden state. The SoftMax activation function is applied to the hidden state H_t if there is a need to get the current timestamp output.

$$\text{Output} = \text{Softmax}(H_t) \quad (18)$$

Liu proposed a model for emotion classification that utilizes Bi-LSTM to incorporate both forward and backward information of input sequences based on LSTM. The model was evaluated using EEG data from 15 channels associated with emotions in the DEAP dataset. The researchers preprocessed the data by filtering and down-sampling before feeding it into the Bi-LSTM for recognition and classification. The SoftMax layer was used to classify the output results of the Bi-LSTM, achieving high classification rates [36].

Lu et al. [37] introduced another model that combines CNN and LSTM for emotion recognition. The model leverages CNN to extract features from the 62-channel signals, and the LSTM network further integrates and extracts features based on the interconnections between the signal features of each channel. The SoftMax function is used to classify the output results of the CNN-LSTM model. The model was implemented using TensorFlow deep learning framework on the Ubuntu environment. Both the CNN and LSTM-based methods demonstrated high classification rates in their respective experiments.

3.5.5 GRU

GRU is similar to LSTM but has fewer gates and is a variant

of the RNN architecture where gates are used to control the flow of information between neurons. It also relies on the hidden state to transfer memory between redundant units. Compared to LSTM, it is new and thus surpasses it with its clearer structure [38].

GRU is a type of recurrent neural network that operates by passing a hidden state from one time step to another. It incorporates gate mechanisms to perform calculations on the input data and the hidden state. One of the distinctive features of GRU is its ability to capture both short-term and long-term dependencies simultaneously. GRU has found applications in various fields such as “speech recognition”, stock price prediction, “sentiment analysis”, “machine translation”, and more. Figure 7 illustrates the structure of a GRU unit.

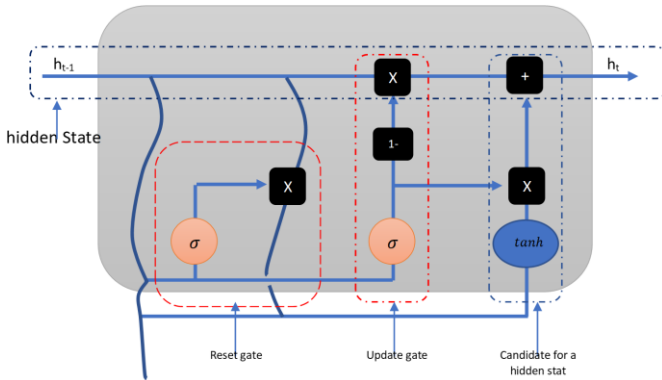


Figure 7. GRU structure

(1-2) Reset gate: The current input (x_t) and the previous hidden state (h_{t-1}) are multiplied by their respective weights and then transferred together through the reset gate. The first step determines which values to ignore (0), remember (1), or partially keep (between 0 and 1) because the sigmoid function has a range of 0 and 1. The previous hidden state is reset in the second step by multiplying it by the results of the first step.

$$r_t = \sigma(X_t * U_r + H_{t-1} * W_r) \quad (19)$$

The hidden case in (3-4-5), although the third step of the update gate may appear similar to the first step in some respects, the weights and biases used to measure these vectors are different, resulting in a characteristic sigmoidal output. As a result, in the fourth step, we subtract the complex vector from the vector containing 1 and multiply it by the sigmoid function (the previous hidden case). (Step 5). This is a component of updating the hidden state with new data.

$$U_t = \sigma(X_t * U_u + H_{t-1} * W_u) \quad (20)$$

(6-7-8) The outputs are mixed with the new inputs (x_t), multiplied by their respective weights, and biases added before going through the tanh activation function (6th step). This is done after resetting a previous hidden state in step two. The state is generated the new hidden state h_t multiplies the hidden state filter by the update gate output (step 7) and combines it with the previously changed hidden state h_{t-1} . Once the iterative unit has processed the complete sequence, the process is then repeated for the time step $t+1$ and subsequent time steps.

$$\bar{h} = \tanh(X_t * U_h + r_t * h_{t-1} * W_h) \quad (21)$$

$$h_t = U_t * h_{t-1} + (1 - U_t) * \bar{h} \quad (22)$$

where, h_t indicates the current hidden state architecture by GRU.

3.6 Compared different algorithms

SVM, CNN, RNN, LSTM, and GRU are all commonly used algorithms for emotion recognition in EEG signals. Here's a comparison of their strengths and weaknesses:

1. SVM:

-Strengths: Can handle high-dimensional data with a limited number of samples, effective in binary classification tasks, can handle non-linear data through the use of kernel functions, good at capturing non-linear decision boundaries.

-Weaknesses: Limited ability to capture temporal dynamics, computationally expensive for large datasets, requires careful selection of kernel functions and hyperparameters.

2. CNN:

-Strengths: Effective in capturing spatial and temporal features, automatically learns relevant features from raw data, reduces the need for manual feature engineering, suitable for large-scale datasets, widely used in image and signal processing tasks.

-Weaknesses: Requires large amounts of labeled data, computationally expensive for complex architectures and large datasets, may overfit with insufficient regularization.

3. RNN:

-Strengths: Can model sequential data and capture temporal dependencies, flexible in handling inputs of varying lengths, suitable for real-time processing, widely used in natural language processing and speech recognition.

-Weaknesses: Suffers from the vanishing/exploding gradient problem, limited ability to capture long-term dependencies, computationally expensive due to sequential processing, difficult to parallelize.

4. LSTM:

-Strengths: Specifically designed to address the vanishing gradient problem, effective in capturing long-term dependencies, retains memory over long sequences, widely used in various sequence modeling tasks, such as speech recognition and language translation.

-Weaknesses: Computationally expensive, requires large amounts of labeled data, may overfit with insufficient regularization, can be difficult to interpret.

5. GRU:

-Strengths: Similar to LSTM, captures long-term dependencies, faster to train due to fewer parameters, uses gating mechanisms to selectively update and forget information, can handle sequence-to-sequence learning tasks.

-Weaknesses: May still suffer from the vanishing gradient problem, requires careful tuning of hyperparameters, may overfit with insufficient regularization.

3.7 Classification

Our custom neural network model featured a multi-layered architecture, encompassing convolutional and recurrent layers, designed to extract intricate patterns and temporal dependencies within the input data. The architecture consisted of the following layers:

- Input Layer: Accepting input data with 11 features.
- Convolutional Layers: Employing 256 filters for feature extraction.
- Gated Recurrent Unit (GRU) Layer: Facilitating the capture of sequential dependencies.
- Flatten Layer: Transforming data into a one-dimensional

format.

- Dense Layer: Producing the final classification output with five emotion categories.

Our model was trained over 100 epochs to refine its internal parameters and optimize its performance.

Table 2. Keras implementation (CNN & GRU) architecture model

Layer (Type)	Output Shape	Param
input_1 (InputLayer)	[(None, 11)]	0
tf.expand_dims (TFOpLambda)	(None, 11)	0
conv1d (Conv1D)	(None,9, 256)	1024
conv1d_1 (Conv1D)	(None,5,256)	327936
gru (GRU)	(None,2560)	1182720
flatten (Flatten)	(None,2560)	0
dense (Dense)	(None,5)	12805
Total params	1,524,485	
Trainable params	1,524,485	
Non-trainable params	0	

The presented neural network model, denoted as "model" in Table 2 comprises a structured architecture designed for Emotion classification by EEG signals. This model is composed of multiple layers, each serving a distinct purpose in the information processing pipeline. The initial layer, named "input_1 (InputLayer)" takes input data characterized by a shape of (None, 11), where "None" signifies a flexible batch size, and the data encompasses 11 distinct features. Subsequently, the "tf.expand_dims (TFOpLambda)" layer extends the input into a three-dimensional format, resulting in a shape of (None, 11, 1), a transformation often employed to facilitate compatibility with convolutional layers designed to handle three-dimensional data.

The model then integrates two consecutive 1D convolutional layers, "conv1d" and "conv1d_1," each comprised of 256 filters. The application of these convolutional layers results in output shapes of (None, 9, 256) and (None, 5, 256), respectively, each contributing varying degrees of complexity to the model's representation. The "gru (GRU)" layer, a Gated Recurrent Unit with 512 units, introduces recurrent processing capabilities to the model, enhancing its ability to capture sequential dependencies within the data.

Following the GRU layer, the data is transformed by a "flatten (Flatten)" layer, effectively converting the output into a one-dimensional format with 2,560 elements. Lastly, the "dense (Dense)" layer, a fully connected component, produces the final classification output with five distinct classes. This corresponds to the number of categories or classes relevant to the classification task. The dense layer encompasses 12,805 trainable parameters, contributing to the model's decision-making process.

In totality, this model architecture encompasses 1,524,485 trainable parameters, with the training process commencing over a span of 100 epochs. Each epoch represents an iterative learning cycle wherein the model refines its internal parameters to optimize its performance with respect to the specified task. This summary of the model architecture provides a comprehensive insight into the neural network's structural intricacies, aiding in understanding its capabilities and complexity in the context of the experimental setup.

3.8 Predict emotion in real time

After training the model and classifying the emotions on the

EEG dataset, this trained deep learning model was used to classify and predict the emotions on new EEG data in real-time. This study focuses also on the real-time prediction of emotions using a pre-trained model. The model, previously trained on a suitable dataset, is loaded to make predictions on incoming data. The real-time data, obtained in CSV format, is preprocessed to remove irrelevant columns and normalize the brain signals. Preprocessing includes encoding emotion labels into numerical values and scaling the brain signals using StandardScaler.

After preprocessing, the pre-trained model is used to predict the emotions associated with the input data. The model's predictions are obtained by applying the argmax function to the model's output, resulting in the most probable emotion for each input. These predicted emotions are then assigned back to the data.

To determine the prevailing emotion from the predictions, the code calculates the frequency of each emotion. The emotion with the highest frequency is considered the most likely emotion and is printed accordingly ("Happy", "Angry", "Calm", "Neutral", or "Sad").

This approach enables real-time emotion prediction by utilizing a pre-trained model and applying it to incoming data. By incorporating appropriate preprocessing techniques and leveraging the model's prediction capabilities, it offers a practical and efficient method for emotion recognition in real-world scenarios.

3.9 The rationale for enhanced CNN-GRU hybrid neural networks

The choice of using a combination of Convolutional Neural Networks (CNN) and Gated Recurrent Units (GRU) for emotion classification using EEG signals is based on leveraging the strengths of both architectures to effectively process and analyze the complex temporal and spatial patterns present in EEG data.

CNN for spatial feature extraction: CNNs are well-suited for spatial feature extraction. EEG data often contains valuable spatial information, as different regions of the brain are associated with various emotional states. By using convolutional layers, CNNs can automatically learn and extract relevant spatial features from the EEG signal. These features can capture the distinct patterns in brain activity associated with different emotions.

GRU for temporal modeling: EEG signals are inherently temporal in nature. They consist of time-series data where the sequence of measurements is critical for understanding brain activity. GRUs, a type of recurrent neural network (RNN), are designed to model sequential data effectively. They capture temporal dependencies and patterns in the EEG signal, enabling the network to consider how brain activity evolves over time, which is crucial for emotion recognition.

Combining Spatial and Temporal Information: Emotion recognition from EEG data often requires considering both spatial and temporal aspects. For instance, specific patterns of brain activity may evolve differently over time for different emotions. By combining CNN and GRU layers in a neural network architecture, you can process the spatial features extracted by CNNs and capture temporal dependencies in the EEG data using GRUs. This hybrid architecture allows the model to make predictions by considering both spatial and temporal information simultaneously.

Improved Generalization: CNNs and RNNs, including

GRUs, have shown success in various machine learning tasks. Combining these architectures can lead to improved generalization because each part of the network focuses on what it does best. CNNs excel at capturing spatial features, while GRUs excel at modeling sequential dependencies. This can result in a more robust model that is better equipped to handle the complexity of EEG data.

In summary, the logic behind using a combination of CNN and GRU for emotion classification using EEG signals is to take advantage of their complementary strengths in processing spatial and temporal information. This approach can lead to more accurate and interpretable models for recognizing emotions from EEG data, which is vital for applications in fields like affective computing, healthcare, and human-computer interaction.

4. RESULTS

Emotion recognition is a complex field in artificial intelligence, and achieving high accuracy in classifying emotions from data is crucial for its successful integration into AI systems. In our study, we focused on enhancing emotion recognition through the implementation of advanced neural network models. We explored the capabilities of a custom-designed model and compared its performance to traditional machine learning classifiers.

Each classifier has its own weaknesses and strengths, the choice depends on the specific requirements of the problem at hand. SVM is suitable for binary classification and handling high-dimensional data. CNN is effective in capturing spatial and temporal features from raw data. RNN and LSTM are suitable for modeling sequential data, while LSTM excels at capturing long-term dependencies. GRU offers a balance between computational efficiency and capturing temporal dynamics. The selection should consider factors such as the nature of the data, available computational resources, and the desired trade-off between model complexity and performance.

The classification was performed using five different machine-learning algorithms. “Gaussian Naïve Bayes classifier, support vector machine, logistic regression, decision tree classifier, and random forest”, addition to algorithms that was mentioned first.

The model has trained for each classifier on (X_train, Y_train) data, and then predictions was made for the test data (X_test) and the classification is printed, which includes accuracy, recall, F1 score, and support for each separately. In addition, the confusion matrix was calculated and drawn using the Confusion Matrix.

Upon closer inspection, we observed the following predictions and actual labels for a subset of test samples:

-Predicted Emotions: [0 3 0 3 2 4 3 2 4 4]
 -Actual Emotions: [0 3 0 3 3 1 3 4 0 4]
 then has been analyzed the confusion matrix to gain insights into the model's performance. Notably, the confusion matrix for our custom model, as well as traditional machine learning classifiers.

Custom Neural Network Model (GRU & CNN)

[90	4	5	7	8]
[6	107	0	5	8]
[6	0	96	10	4]
[8	9	6	105	5]
[7	9	3	4	88]

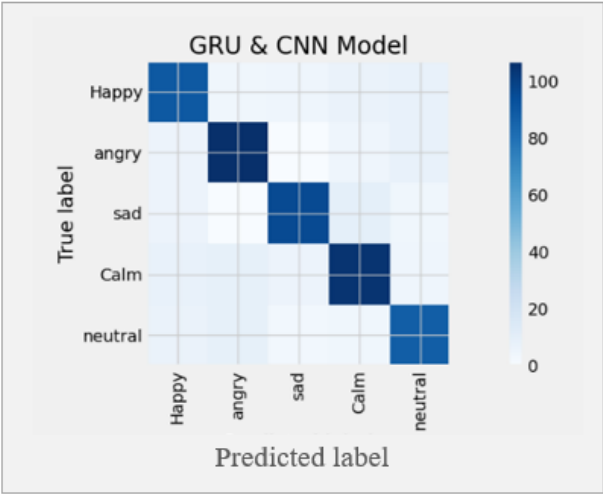


Figure 8. GRU & CNN convolution matrix

Figure 8 shows that the confusion matrix plot visually displays the model's classification performance. Each cell represents the number of correct and incorrect predictions for each category. Where the horizontal labels are the predicted labels and the vertical features are the true labels that are supposed to be correct. The color intensity shows the size of these counts. The greater the intensity of the blue color, the more accurate the model is in classification and prediction. The results in Figure 8 above showed great distinction in classification, as each feature was given a high color intensity and was consistent between the predicted labels and the true labels.

In contrast to the following Figure 9, which shows the extent of dispersion of colors in the confusion matrix, which means that the classification results are weak for this algorithm when working on it, this algorithm was tested to clarify the extent of the difference in the confusion matrix diagram when the algorithm gives high classification accuracy and when the accuracy is low.

Gaussian Naive Bayes

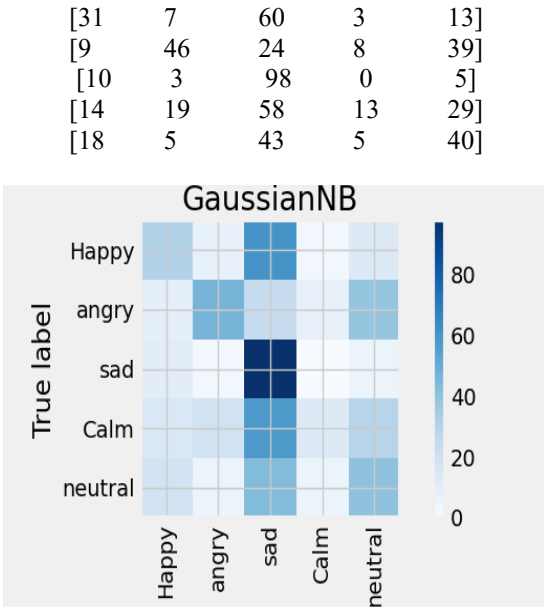


Figure 9. GaussianNB convolution matrix

When the confusion matrix is as in Figure 9, the real-time prediction will give wrong results, as most of the time reading the sentiment data will give the sad emotion. The reason is that the dominant characteristic and intensity of the color is due to emotion of sad.

But we notice in Figure 10 that the confusion matrix of the SVM algorithm in classification results better than GaussianNB, but we also notice from the figure that the happy state in true label gives similar results to the sad state when predicting. This is called confusion when predictive label.

This is also the case in Figure 11, which describes the Logistic Regression technique, where the results are similar to SVC.

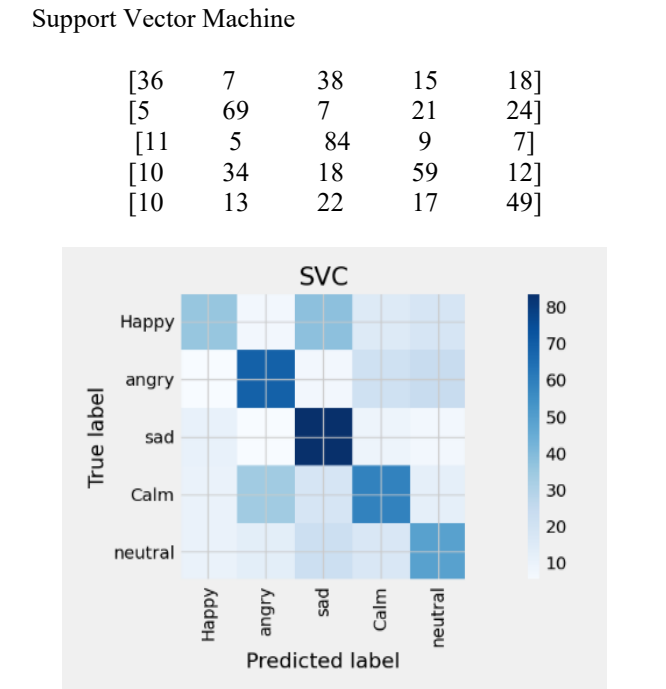


Figure 10. SVC convolution matrix

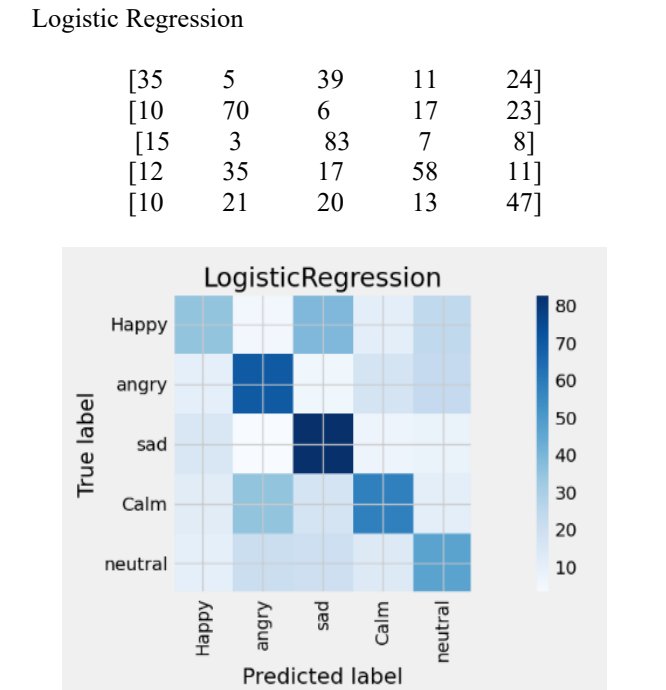


Figure 11. LR Convolution matrix

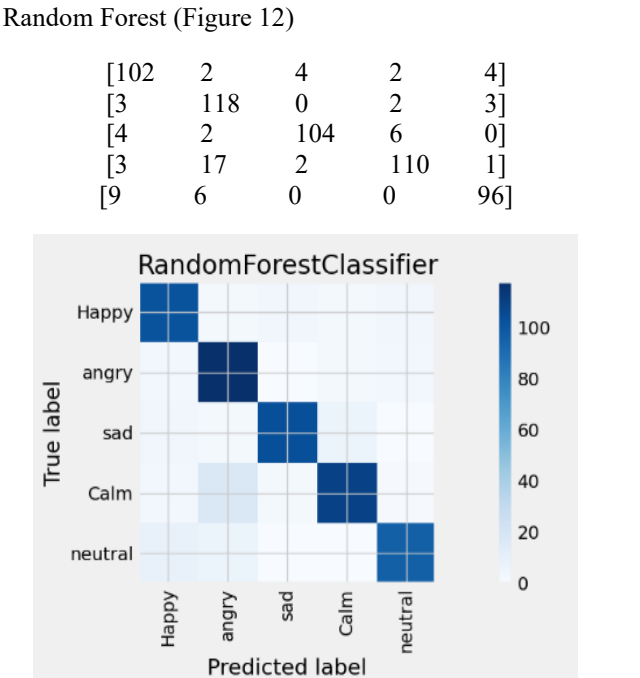


Figure 12. RFC convolution matrix

Finally, a classification report was printed for the deep learning model (GRU & CNN) that was previously trained on the same dataset and compares the results of all classifiers (Table 3).

The classification report for our Brain Waves GRU & CNN model provides a comprehensive evaluation of its performance in emotion recognition. It reveals precision values ranging from 0.77 to 0.87 across different emotion categories, highlighting the model's ability to make accurate predictions. Moreover, the recall scores, ranging from 0.79 to 0.85, underscore the model's capacity to effectively capture true positive instances. The F1-score, a harmonic mean of precision and recall, demonstrates strong values, ranging from 0.78 to 0.85, indicating a balanced trade-off between precision and recall. Overall, the model achieves an impressive accuracy of 81% on a dataset of 600 samples, reflecting its robustness in classifying emotions. The macro-average and weighted-average metrics further support the model's consistency, reliability and superiority over other algorithms, including LSTM, with macro-average values of 0.81, highlighting a balanced performance across emotion categories. These results affirm the model's potential for applications requiring precise emotion recognition.

Table 3. Classification report of brain waves GRU & CNN

	Precision	Recall	F1-Score	Support
0	0.77	0.79	0.78	114
1	0.83	0.85	0.84	126
2	0.87	0.83	0.85	116
3	0.80	0.79	0.80	133
4	0.78	0.79	0.79	111
Accuracy			0.81	600
Macro avg	0.81	0.81	0.81	600
Weighted avg	0.81	0.81	0.81	600

The results of our experiment were promising, with the custom neural network model achieving an accuracy of 81% on the training dataset as figure 13 showed. While. However, it is essential to assess the model's performance on unseen data. On the testing dataset, our model reported a loss of 7.2959 and

an accuracy of 81%, demonstrating its ability to generalize well to new, previously unseen data. It is superior to other classifiers, including LSTM, the accuracy results were 67% Figure 14 shows this.

Then the trained GRU and CNN model was saved to a file so that new predictions are performed on the new data in real-time. Doing that, the classification process has been completed for the dataset after collecting, labeling, and processing it.

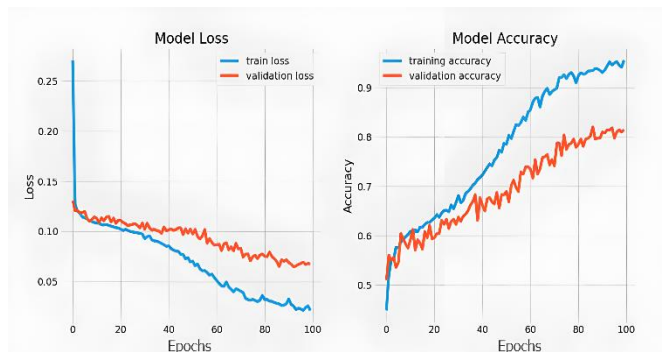


Figure 13. CNN & GRU results

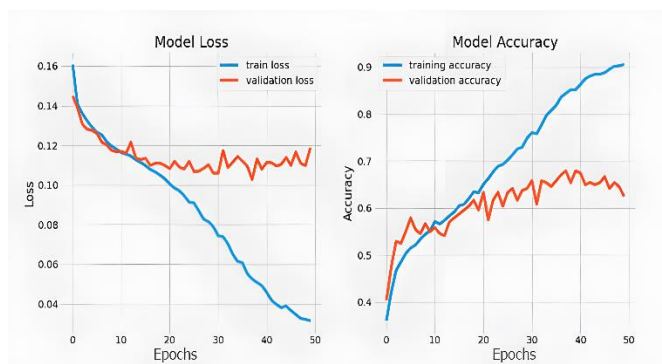


Figure 14. LSTM results

5. CONCLUSIONS

This study illuminates the proficiency of advanced neural networks in emotion recognition tasks, discussing the intricate process of recognizing emotions using MindWave (EEG) signals and deep neural networks. The process encompasses several stages including data collection, processing and labeling, preprocessing, feature extraction, feature selection, classification, model evaluation, and finally prediction of new data. Furthermore, several algorithms, namely SVM, CNN, RNN, LSTM, and GRU, were compared. SVM demonstrated its ability to manage high-dimensional data with limited samples, although its capacity to capture temporal dynamics may be constrained. CNN can autonomously learn relevant features from raw data but necessitates substantial amounts of labeled data and can be computationally challenging. RNN captures short-term dependencies in sequential data but may be hindered by the vanishing gradient problem and limited in capturing long-term dependencies. LSTM can address the vanishing gradient problem and capture long-term dependencies, but it can also be computationally burdensome. GRU, similar to LSTM but with fewer parameters, can be quicker to train, making it beneficial in scenarios where labeled data is limited or sequence-to-sequence learning is required.

These algorithms were tested alongside some traditional machine learning algorithms. Each classifier model was trained on (X and Y) data, predictions for the test data were made, and the classification was printed. This included Accuracy, Recall, F1 score, and support for each, separately. Additionally, the confusion matrix was calculated and depicted using the confusion matrix. These were compared with the proposed model specifically designed for this study, which encompasses the integration of two algorithms (Multi 1D CNN & GRU).

Our custom-constructed model, with its multi-layered architecture, attains noteworthy accuracy levels on both training and testing datasets. When integrating CNN and GRU and training the model, results of 81% accuracy were obtained, exceeding the performance of other classifiers. While the model was also trained on LSTM, less efficient results were achieved, with an accuracy of 67%.

The results suggest that our custom neural network model, which integrates convolutional and recurrent layers, exhibits competitive accuracy levels in emotion recognition. Moreover, it surpasses traditional machine learning classifiers across multiple emotion categories. This underlines the potential of advanced neural network architectures in augmenting emotion recognition capabilities, which could have significant implications for applications such as human-computer interaction and mental health monitoring.

As the field of emotion recognition advances, the incorporation of advanced neural network models harbors substantial promise for real-world applications, ultimately contributing to the development of more emotionally intelligent AI systems.

6. FUTURE WORK

In this research, we proposed an innovative architecture for emotion recognition in EEG signals by synergizing multiple 1D CNN layers with a GRU layer. This architecture aims to exploit the strengths of both CNNs and GRUs to effectively capture and model the complex temporal dynamics inherent in EEG data.

Future works can explore and incorporate alternative structures. For instance, a combination of LSTM and CNN or RNN could be utilized.

The dataset employed in this study was captured, processed, and input via a MindWave portable device, lending strength to the study by relying on a self-constructed dataset rather than a pre-existing, published one. However, this feature also introduces drawbacks, including the cost of EEG signal capture devices and some devices' limitations in capturing EEG signals. These limitations include a restricted number of columns (samples) such as 'Attention', 'Meditation', 'delta', 'theta', 'low-alpha', 'high-alpha', 'low-beta', 'high-beta', 'low-gamma', and 'mid-gamma'. This could negatively impact classification results and accuracy. Pre-existing datasets, with their larger sample sizes, might provide superior model training results when used.

Furthermore, the enhancement of results is influenced by several factors, including the number of filters for feature extraction, the number of layers, and the number of time periods. The accuracy of results could also be improved by reducing the number of ratings or emotional features. For instance, in our study, emotions were divided into five traits (sad, angry, happy, calm, and neutral). These could be

replaced with fewer attributes such as negative condition, normal condition, and positive condition. This would increase the relative difference between the values of the data captured through the supervised technique, potentially enhancing model performance.

REFERENCES

- [1] Shahnaz, C., Hasan, S.S. (2016). Emotion recognition based on wavelet analysis of Empirical Mode Decomposed EEG signals responsive to music videos. In 2016 IEEE Region 10 Conference (TENCON), Singapore, pp. 424-427. <https://doi.org/10.1109/TENCON.2016.7848034>
- [2] Picard, R.W. (2003). Affective computing: challenges. *International Journal of Human-Computer Studies*, 59(1-2): 55-64. [http://doi.org/10.1016/S1071-5819\(03\)00052-1](http://doi.org/10.1016/S1071-5819(03)00052-1)
- [3] Sarıkaya, M.A., İnce, G. (2017). Emotion recognition from EEG signals through one electrode device. In 2017 25th Signal Processing and Communications Applications Conference (SIU), Antalya, Turkey, pp. 1-4. <https://doi.org/10.1109/SIU.2017.7960390>
- [4] Gupta, A., Singh, P., Karlekar, M. (2018). A novel signal modeling approach for classification of seizure and seizure-free EEG signals. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 26(5): 925-935. <https://doi.org/10.1109/tnsre.2018.2818123>
- [5] Zheng, W.L., Zhu, J.Y., Lu, B.L. (2017). Identifying stable patterns over time for emotion recognition from EEG. *IEEE Transactions on Affective Computing*, 10(3): 417-429. <http://doi.org/10.1109/TAFFC.2017.2712143>
- [6] Neurosky. (2015). MindWave Mobile: User Guide. http://download.neurosky.com/support_page_files/MindWaveMobile/docs/mindwave_mobile_user_guide.pdf.
- [7] Tomarken, A.J., Davidson, R.J., Henriques, J.B. (1990). Resting frontal brain asymmetry predicts affective responses to films. *Journal of Personality and Social Psychology*, 59(4): 791. <https://doi.org/10.1016/j.biopsycho.2004.03.008>
- [8] Davidson, R.J. (2004). What does the prefrontal cortex “do” in affect: Perspectives on frontal EEG asymmetry research. *Biological psychology*, 67(1-2): 219-234. <https://doi.org/10.1016/j.biopsycho.2004.03.008>
- [9] Lee, Y.Y., Hsieh, S. (2014). Classifying different emotional states by means of EEG-based functional connectivity patterns. *PloS one*, 9(4): e95415. <https://doi.org/10.1371/journal.pone.0095415>
- [10] Wang, X.W., Nie, D., Lu, B.L. (2014). Emotional state classification from EEG data using machine learning approach. *Neurocomputing*, 129: 94-106. <https://doi.org/10.1016/j.neucom.2013.06.046>
- [11] Murugappan, M., Ramachandran, N., Sazali, Y. (2010). Classification of human emotion from EEG using discrete wavelet transform. *Journal of Biomedical Science and Engineering*, 3(4): 390-396. <http://doi.org/10.4236/jbise.2010.34054>
- [12] Peining, P., Tan, G., Phyto Wai, A.A. (2017). Evaluation of consumer-grade EEG headsets for BCI drone control. In IRC Conference on Science, Engineering, and Technology. http://ircset.org/anand/2017papers/IRC-SET_2017_paper_S6-4.pdf.
- [13] Zhang, Y.Q., Chen, J.L., Tan, J.H., Chen, Y.X., Chen, Y.Y., Li, D.H., Yang, L., Su, J., Huang, X. (2020). An investigation of deep learning models for EEG-based emotion recognition. *Frontiers in Neuroscience*, 14. <https://doi.org/10.3389/fnins.2020.622759>
- [14] Liu, S., Wang, L., Ding, X.T. (2020). Emotional EEG recognition based on Bi-LSTM. *Journal of Shandong University (Engineering Science)*, 50(4): 35-39. <http://10.0.23.152/j.issn.1672-3961.0.2019.679>
- [15] Yang, H., Han, J., Min, K. (2019). A multi-column CNN model for emotion recognition from EEG signals. *Sensors*, 19 (21): 4736. <https://doi.org/10.3390/s19214736>
- [16] Phan, T.D.T., Kim, S.H., Yang, H.J., Sang, G. (2021). LeeEEG-based emotion recognition by convolutional neural network with multi-scale kernels. *Sensors*, 21(15): 5092. <https://doi.org/10.3390/s21155092>
- [17] Zhu, J. H., Munjal, R., Sivaram, A., Paul, S. R., Tian, J., Jolivet, G. (2022). Flow regime detection using gamma-ray-based multiphase flowmeter: A machine learning approach. *International Journal of Computational Methods and Experimental Measurements*, 10(1): 26-37. <https://doi.org/10.2495/CMEM-V10-N1-26-37>
- [18] Yao, J., Luo, Z., Automation, S.O. (2015). Application of dual-tree complex wavelet transform in EEG denoising. *Journal of Huazhong University of Science and Technology*.
- [19] Chen, J. X., Zhang, P.W., Mao, Z.J., Huang, Y.F., Jiang, D.M., Zhang, Y.N. (2019). Accurate EEG-based emotion recognition on combined features using deep convolutional neural networks. *IEEE Access*, 7: 44317-44328. <https://doi.org/10.1109/ACCESS.2019.2908285>
- [20] Chang, C.Y., Tsai, J.S., Wang, C.J., Chung, P.C. (2009). Emotion recognition with consideration of facial expression and physiological signals. In 2009 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology, Nashville, TN, USA, pp. 278-283. <https://doi.org/10.1109/CIBCB.2009.4925739>
- [21] Guo, K., Yu, H., Chai, R., Nguyen, H., Su, S.W. (2019). A hybrid physiological approach of emotional reaction detection using combined FCM and SVM classifier. In 2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Berlin, Germany, pp. 7088-7091. <https://doi.org/10.1109/embc.2019.8857698>
- [22] Kumar, D.K., Nataraj, J.L. (2019). Analysis of EEG based emotion detection of DEAP and SEED-IV databases using SVM. In Proceedings of the Second International Conference on Emerging Trends in Science & Technologies for Engineering Systems, Karnataka, India. <https://doi.org/10.2139/ssrn.3509130>
- [23] Ma, M., Guo, L., Su, K., Liang, D. (2017). Classification of motor imagery EEG signals based on wavelet transform and sample entropy. In 2017 IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), Chongqing, China, pp. 905-910. <https://doi.org/10.1109/IAEAC.2017.8054145>
- [24] Li, Y., Wu, L., Wang, T., Gao, N., Wang, Q. (2019). EEG signal processing based on genetic algorithm for extracting mixed features. *International Journal of Pattern Recognition and Artificial Intelligence*, 33(6): 1958008. <https://doi.org/10.1142/S0218001419580084>
- [25] Hu, Y., Wang, L., Fu, W. (2018). EEG feature extraction

- of motor imagery based on WT and STFT. In 2018 IEEE International Conference on Information and Automation (ICIA), Wuyishan, China, pp. 83-88. <https://doi.org/10.1109/ICInfA.2018.8812377>
- [26] LeCun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W., Jackel, L.D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4): 541-551. <https://doi.org/10.1162/neco.1989.1.4.541>
- [27] Tripathi, S., Acharya, S., Sharma, R., Mittal, S., Bhattacharya, S. (2017). Using deep and convolutional neural networks for accurate emotion classification on DEAP data. *Proceedings of the AAAI Conference on Artificial Intelligence*, 31(2): 4746-4752. <https://doi.org/10.1609/aaai.v31i2.19105>
- [28] Li, J., Zhang, Z., He, H. (2018). Hierarchical convolutional neural networks for EEG-based emotion recognition. *Cognitive Computation*, 10: 368-380. <https://doi.org/10.1007/s12559-017-9533-x>
- [29] Yu, C., Wang, M. (2022). Survey of emotion recognition methods using EEG information. *Cognitive Robotics*, 2: 132-146. <https://doi.org/10.1016/j.cogr.2022.06.001>
- [30] Djamal, E.C., Putra, R.D. (2020). Brain-computer interface of focus and motor imagery using wavelet and recurrent neural networks. *Telkomnika (Telecommunication Computing Electronics and Control)*, 18(5): 2748-2756. <http://doi.org/10.12928/telkomnika.v18i5.14899>
- [31] Tao, W., Li, C., Song, R., Cheng, J., Liu, Y., Wan, F., Chen, X. (2020). EEG-based emotion recognition via channel-wise attention and self attention. *IEEE Transactions on Affective Computing*, 14(1): 382-393. <https://doi.org/10.1109/TAFFC.2020.3025777>
- [32] Yin, Z., Zhao, M., Wang, Y., Yang, J., Zhang, J. (2017). Recognition of emotions using multimodal physiological signals and an ensemble deep learning model. *Computer Methods and Programs in Biomedicine*, 140: 93-110. <https://doi.org/10.1016/j.cmpb.2016.12.005>
- [33] Hochreiter, S., Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8): 1735-1780. <https://doi.org/10.1162/neco.1997.9.8.1735>
- [34] Li, Y., Dong, H. (2018). Text sentiment analysis based on feature fusion of convolution neural network and bidirectional long short-term memory network. *Journal of Computer Applications*, 38(11): 3075. <https://doi.org/10.11772/j.issn.1001-9081.2018041289>
- [35] Han, X., Kheir, N., Balzarotti, D. (2016). Phisheye: Live monitoring of sandboxed phishing kits. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, Vienna, Austria, pp. 1402-1413. <https://doi.org/10.1145/2976749.2978330>
- [36] Liu, S., Wang, L., Ding, X. (2020). Emotional EEG recognition based on Bi-LSTM. *Journal of Shandong University*, 50(4): 35-39. <https://doi.org/10.6040/j.issn.1672-3961.0.2019.679>
- [37] Lu, G.M., Cong, W.K., Wei, J.S., Yan, J.J. (2021). EEG based emotion recognition using CNN and LSTM, *Nanjing University of Posts and Telecommunications*. 41(1): 58-64. <https://doi.org/10.14132/j.cnki.1673-5439.2021.01.008>
- [38] Chung, J., Gulcehre, C., Cho, K., Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*. <https://doi.org/10.48550/arXiv.1412.3555>