SSRG International Journal of Electronics and Communication Engineering
ISSN: 2348-8549/ https://doi.org/10.14445/23488549/IJECE-V11I8P126

Volume 11 Issue 8, 269-281, August 2024
© 2024 Seventh Sense Research Group®

Original Article

Optimized Schizophrenia Detection via EEG: A CNN-LSTM and CNN-GRU Ensemble with SVM

N. Shan[1], V. Amsaveni[2]

[1]*Department of Electronics and Communication Engineering, Noorul Islam Centre for Higher Education, Tamil Nadu, India*
[2]*Department of Electronics and Instrumentation Engineering, Noorul Islam Centre for Higher Education, Tamil Nadu, India.*

[1]*Corresponding Author : nshan24@outlook.com*

Received: 20 June 2024 Revised: 31 July 2024 Accepted: 15 August 2024 Published: 31 August 2024

Abstract *- Human illnesses that impact a person's ability to think, communicate, and behave are primarily classified as psychological and neurological disorders. Currently, neurological illnesses affect almost 12% of the world's population. About 1% of people worldwide have schizophrenia, a chronic psychological illness. The proposed ensemble model combines two hybrid Deep Learning (DL) approaches with the Support Vector Machine (SVM) as the meta learner. Convolutional Neural Network-Gated Recurrent Unit (CNN-GRU) and Convolutional Neural Network-Long Short-Term Memory (CNN-LSTM) are the DL approaches utilized in the ensemble approach for Schizophrenia detection. The EEG signals utilized in this study are sourced from the Kaggle repository, an extensive online database known for its diverse collection of high-quality medical data. This novel DL model for schizophrenia classification aims to enhance diagnostic accuracy and reduce reliance on expert interpretation. The first hybrid model, CNN-LSTM, attains an accuracy of 92.93% and CNN-GRU with an accuracy of 93.10%. Thus, the staking ensemble model achieves an overall accuracy of 94.85%, indicating strong overall correctness in classifying cases. Comparative simulations of the suggested approach against several current solutions in schizophrenia diagnosis reveal that the suggested approach attains superior accuracy when juxtaposed with other existing methods. These findings bolster the advantages of DL for neuroscientific research in general and psychiatric classification in particular.*

Keywords *- Schizophrenia, Gated recurrent unit, Psychological disease, Long short-term memory, Convolutional neural network, Support vector machine.*

1. Introduction

A mental illness known as Schizophrenia (SZ) causes an imbalance in certain brain regions, which impairs the ability to coordinate emotions, behaviors, and thoughts. It is a serious mental illness that is chronic and has a major impact on sufferers' day-to-day lives [1]. Significant deficits in reality perception and behavioural abnormalities associated with persistent delusions, hallucinations, influence experiences, disorganized thinking, severely chaotic conduct, excessive agitation, or slowing of movements are characteristics of schizophrenia [2, 3]. The symptoms and signs of schizophrenia are depicted in Figure 1. Patients' social activity and brain development are significantly impacted by the illness, which frequently starts in late adolescence.

In clinical examinations, negative symptoms of schizophrenia are typically misinterpreted [4]. Clinicians' experience also plays a role in diagnosing. Since the emergence of neuroimaging techniques, structural brain changes associated with schizophrenia have been intensively explored. Many studies have reported diminished volume of the bilateral medial temporal areas, a left superior temporal region deficit, and overall gray matter loss, while there are still some disagreements over the duration of the illness and the use of antipsychotics are the causes of this disease [4,6]. Numerous research has shown how conventional Machine Learning (ML) methods may identify structural and functional abnormalities in the brain associated with schizophrenia. The development of a reliable technique for diagnosing schizophrenia with negative symptoms is urgent. Recent research has demonstrated that speech impairment may serve as a sign of the diagnosis of schizophrenia. Generally speaking, the majority of current approaches use feature engineering techniques, intensity-related features, spectrum-related features, and other ways to evaluate schizophrenia speech. According to these studies, speech can be used as an automated biomarker to diagnose schizophrenia. However, it is still challenging to suggest a strong model because of the limitations in the amount of data and the challenges associated with efficient feature extraction. This article proposes a staking ensample model to accomplish end-to-end detection of schizophrenia.

This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/)

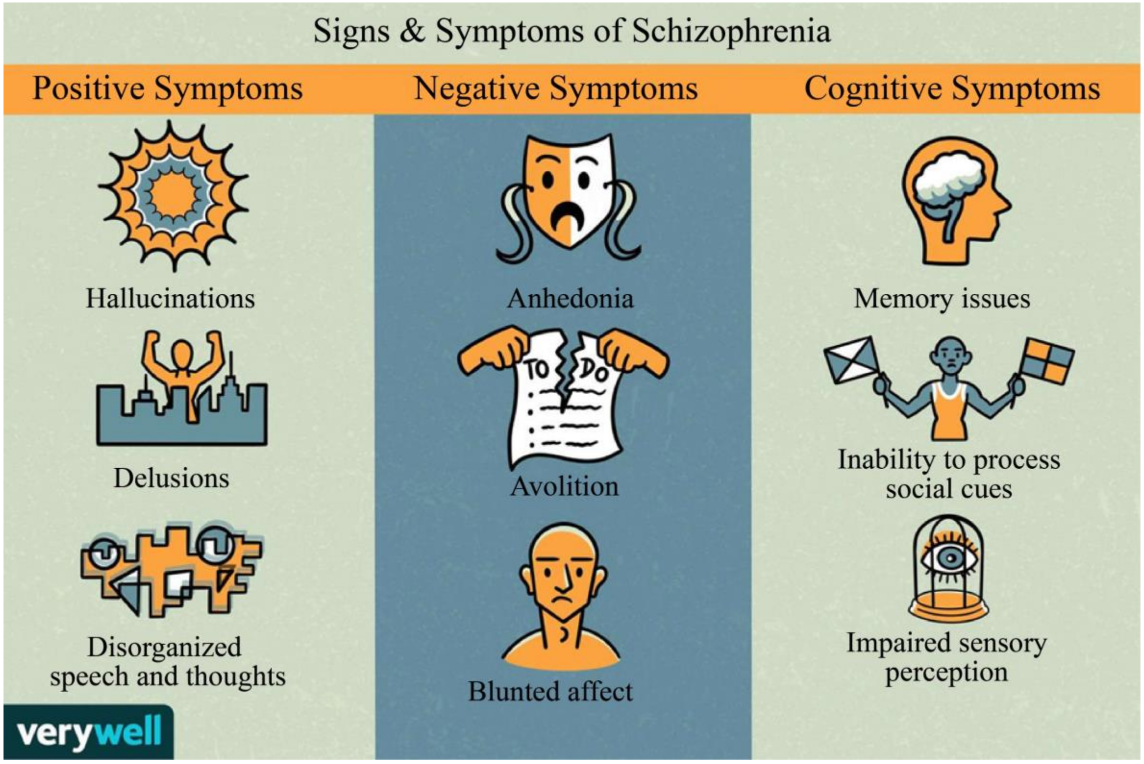

Fig. 1 Symptoms of schizophrenia

The feature extraction issues can be avoided with the suggested method. The following succinctly describes the contributions of the proposed work:

- Proposed an efficient DL model for SZ detection from EEG data.
- Implement an ensemble approach to increase the detection performance.
- Reduce the error rates and false positive rate

2. Literature Review

Over the past few decades, research has focused on the identification of abnormal speech in cases of schizophrenia. The majority of earlier research relied on feature engineering.

Aslan et al. [7] extracted important features from scalogram images and trained the network for the identification of schizophrenia using the Visual Geometry Group-16 (VGG16) DL network architecture, an advanced CNN architecture. The study used two separate datasets with participants from various age groups and demonstrated a high degree of success in categorizing SZ patients and healthy persons. The difficulty of choosing suitable parameter values was one of the study's shortcomings. In addition, the quality and diversity of the training data have a big impact on the effectiveness of the model.

An automated method for detecting schizophrenia was presented by Wawer et al. [8]. The authors employed three different forms of text representation, namely utterance embedding vectors, dictionary vectors, and bag of words. The study's shortcomings were the omitted chances to comprehend the disorder's larger context and connections to other illnesses. The method might not fully account for the diversity and complexity of psychiatric disorders. It has also made the data more complex and presented more processing difficulties.

During auditory processing, the potential of aberrant patterns in electrical brain activity to distinguish SZ from healthy people was investigated by Barros et al. [9]. The authors suggested an architecture to do the categorization based on multichannel EEG obtained during a passive listening task using deep CNN. According to the results, the algorithm distinguished between individuals with schizophrenia and healthy patients with 78% accuracy. One of the study's shortcomings is the low signal-to-noise ratio caused by the overlay of EEG recordings with brain processes unrelated to the job at hand. The method could also result in false positives or false negatives.

Hu et al. [10] integrated 3D structural and Magnetic Resonance Imaging (MRI) data to create a deep feature strategy based on pre-trained 2D CNN for the categorization of schizophrenia. This study employed two separate MRI datasets of controls and schizophrenia that were similar in terms of age and gender for both groups. The authors created multimodal three-dimensional CNN models and employed a deep feature technique based on two-dimensional pretrained CNN, which showed an accuracy of 81.02%. One of the study's disadvantages was the significant computational cost that arises from the high dimensionality of the input data during training. The sample size is quite small, particularly when it comes to 3D CNN network training.

A lightweight three-dimensional CNN based system for SZ diagnosis utilizing MRI images was proposed by Patro et al. [11]. The model used a bagging classifier with ensemble learning for classification after the features were extracted from MRI scans. The MCICShare, COBRE, and fBRINPhase-II benchmark databases were the three datasets from which the algorithm was tested. In comparison to the existing approaches, the model attained the highest accuracy of 92.22% and sensitivity of 94.44%. The limitations of this study included the increased computational cost associated with training 3D models. The DL algorithm was trained on binary labels seen in MRI data, which oversimplifies the fact that individuals may have many diseases at the same time and that many psychiatric problems progress along with it. The model was also susceptible to tweaking its hyperparameters.

DL algorithms created by Oh et al. [12] were able to identify SZ fairly well on MRI data sets obtained by a single source, even if the individuals' clinical features varied slightly. The DL method was trained on MRI data that was all binary labeled. Although this binary categorization is frequently employed in AI research, it may provide challenges when implementing this approach in clinical settings. A deep CNN was trained using 5 public structural MRI data sets that included both normal and schizophrenic patients. In all, 873 structural MRI data sets were employed for this training. In order to get over the drawbacks of feature extraction-based techniques, Khare et al. [13] suggested an automated identification of SZ utilizing a mix of time-frequency analysis and CNN. The EEG signals undergo three different processing methods, namely smoothed pseudo-Wigner–Ville distribution, short-time Fourier transform and continuous wavelet transform. The resulting Time-Frequency Representation (TFR) plots were scalogram, spectrogram, and SPWVD-based, respectively. The SPWVD-based TFR and CNN model have yielded an accuracy of 93.36% for the approach. One limitation of the method is the reliance on empirical selection of parameters and additional memory requirements.

Zheng et al. [14] used functional MRI data for SZ to retrieve effective time series from preprocessed fMRI data. They then use TL and VGG16 net to perform correlation analysis on regions of interest, classifying the functional relationship between SZ and healthy control. Based on VGG16, experimental results indicate that fMRI classified data with up to 84.3% accuracy.

Current methods often rely on empirical parameter selection, leading to increased memory requirements and computational costs, particularly when training 3D CNN models. The variability in signal-to-noise ratio across subjects, influenced by differences in EEG data acquisition parameters, can further affect signal quality. Existing nonlinear techniques for measuring the brain's chaotic behaviors, such as fractal dimension and Lyapunov exponent, is computationally intensive and unsuitable for real-time applications, leaving a gap for real-time, visually interpretable methods for neurologists. Additionally, most studies utilize all EEG channels without exploring the impact of a specific number of channels on classification performance. The modest sample sizes, especially for 3D CNN training, limit the efficiency of feature extraction and generalizability of the models. Moreover, schizophrenia's high heterogeneity, with various subtypes and symptom presentations, poses a significant challenge for machine learning models to capture effectively. Addressing these gaps requires developing more robust, interpretable, and computationally efficient models capable of handling the nuanced and heterogeneous nature of schizophrenia.

3. Materials and Methods

The proposed methodology for schizophrenia detection from EEG data begins with data collection from the Kaggle repository, followed by essential preprocessing and data augmentation techniques. The core of the methodology is built around two hybrid DL approaches: CNN-LSTM and CNN-GRU, as depicted in Figure 2. These architectures are designed to capture both temporal patterns and spatial features in the EEG data. In the CNN-LSTM model, the CNN component is responsible for extracting local patterns and features from the input EEG data, generating a set of learned features that encode spatial information. These features are then passed to the LSTM component, which processes them sequentially to capture dependencies and temporal dynamics in the data. The CNN-GRU model follows a similar approach, with the GRU component excelling at capturing temporal dependencies. The final step in the methodology involves a stacked ensemble method [15] to improve the classification performance. Predictions from the CNN-LSTM and CNN-GRU models are gathered and stacked together to create a new feature set. This new set of features, capturing complementary information from both models, is then used as input for an SVM classifier. The SVM, serving as the meta learner, effectively combines these diverse predictions to produce a final output. This ensemble approach leverages the strengths of each base model, enhancing the overall robustness and accuracy of the schizophrenia detection system.

3.1. Dataset Description

The EEG data for schizophrenia used in this study has been sourced from Kaggle. This dataset consists of 19 electrode values and one label column, as shown in Figure 3. The electrode names listed in the left column include Pz, Fp2, F4, F7, C3, C4, P4, F3, O2, F8, T3, T4, T5, P3, T6, Fz, O1 and Cz. These labels correspond to specific scalp locations according to the international system for EEG electrode placement.

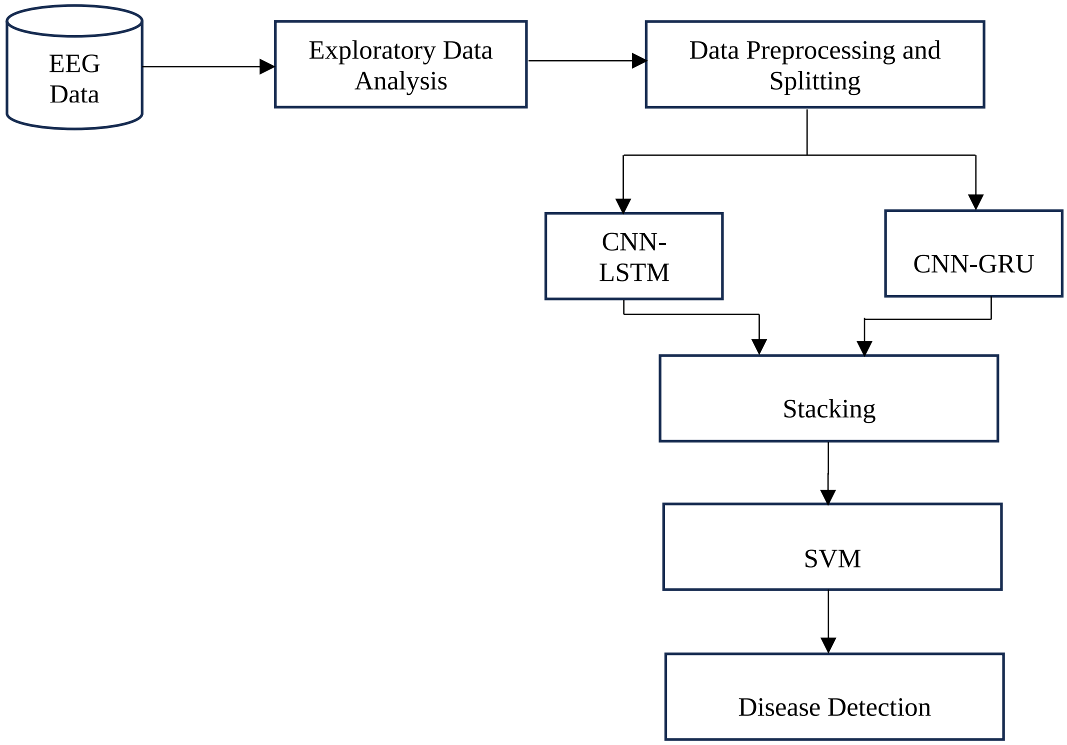

Fig. 2 Block diagram of proposed methodology

	Fp2	F3	F4	C3	C4	P3	P4	O1	O2	F7	F8	T3	T4	T5	T6	Fz	Cz	Pz	Fp1	label
0	0.000010	0.000012	-0.000008	0.000010	-0.000006	0.000005	-3.971580e-06	-0.000003	-0.000008	0.000012	-0.000005	4.438200e-06	-2.289624e-06	0.000004	-0.000005	-0.000001	3.214959e-06	3.952544e-09	-5.658102e-10	0
1	0.000008	0.000010	-0.000008	0.000009	-0.000006	0.000002	-4.277390e-06	-0.000005	-0.000008	0.000003	0.000002	-1.372193e-06	3.097627e-07	-0.000002	-0.000005	-0.000003	2.297529e-06	-2.442529e-06	-5.658102e-10	0
2	0.000002	0.000008	-0.000007	0.000008	-0.000004	0.000002	-3.512864e-06	-0.000006	-0.000008	0.000003	-0.000002	-2.289624e-06	-2.289624e-06	-0.000002	-0.000006	-0.000003	1.533003e-06	-1.983814e-06	-5.658102e-10	0
3	0.000002	0.000008	-0.000008	0.000007	-0.000005	0.000001	-2.748339e-06	-0.000006	-0.000009	0.000002	-0.000010	-3.054149e-06	-3.818675e-06	-0.000004	-0.000006	-0.000004	9.213831e-07	-1.830909e-06	-5.658102e-10	0
4	0.000010	0.000008	-0.000003	0.000005	-0.000002	-0.000001	-9.134780e-07	-0.000008	-0.000011	-0.000001	-0.000001	-6.076678e-07	-2.136719e-06	-0.000007	-0.000005	-0.000002	9.213831e-07	-2.595434e-06	-5.658102e-10	0

Fig. 3 Dataset sample

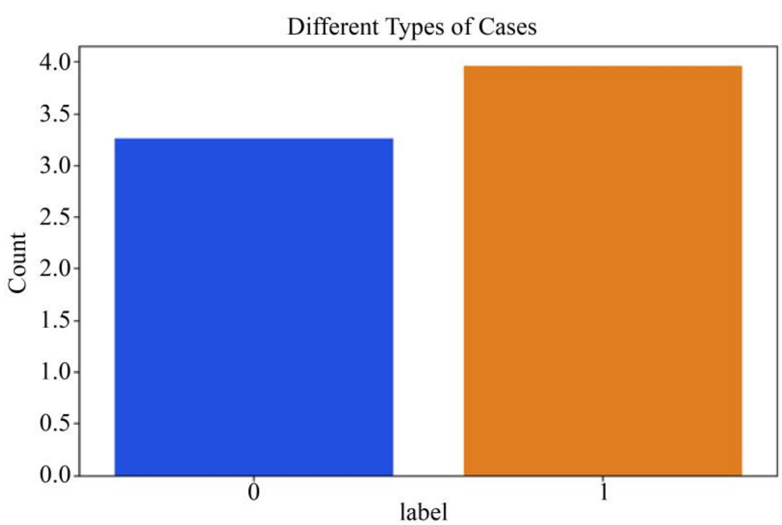

Fig. 4 Visualization of dataset

The locations represent different brain regions, with prefixes indicating Parietal (P), Frontal (F), Occipital (O), Central (C), and Temporal (T) regions. The suffixes denote whether the electrode is placed on the left (odd numbers) or right (even numbers) hemisphere. The numerical values following the electrode labels represent the voltage measurements recorded by each electrode at a given time point. The final column, labelled "label," likely contains the class labels associated with the EEG data, identifying different conditions or states relevant to schizophrenia. The data split for the method is 80:20. The visualization of the dataset in the case of the number of classes is depicted in Figure 4.

A heatmap is a data visualization approach that uses a colour-coded matrix or grid to display data values. With each cell in the matrix represents a data point or a combination of variables, and its color denotes the magnitude or intensity of the data, as shown in Figure 5. By emphasizing regions with high or low values, heatmap visualization aims to reveal patterns, trends, and relationships within the data [16]. Heatmaps make it easy and quick for viewers to comprehend data by using colors to represent distinct data ranges. This makes it possible to spot anomalies, gradients, and clusters.

Spectrograms of EEG data provide a visual indication of the signal's frequency content over time, capturing both temporal and spectral characteristics [5]. These visualizations can highlight distinct patterns and anomalies in brain activity, aiding in the identification of neurological conditions like schizophrenia. The Spectrogram representation of nineteen electrodes is represented in Figure 6. By transforming raw EEG signals into spectrograms, DL models can more effectively learn and classify complex temporal patterns in the data.

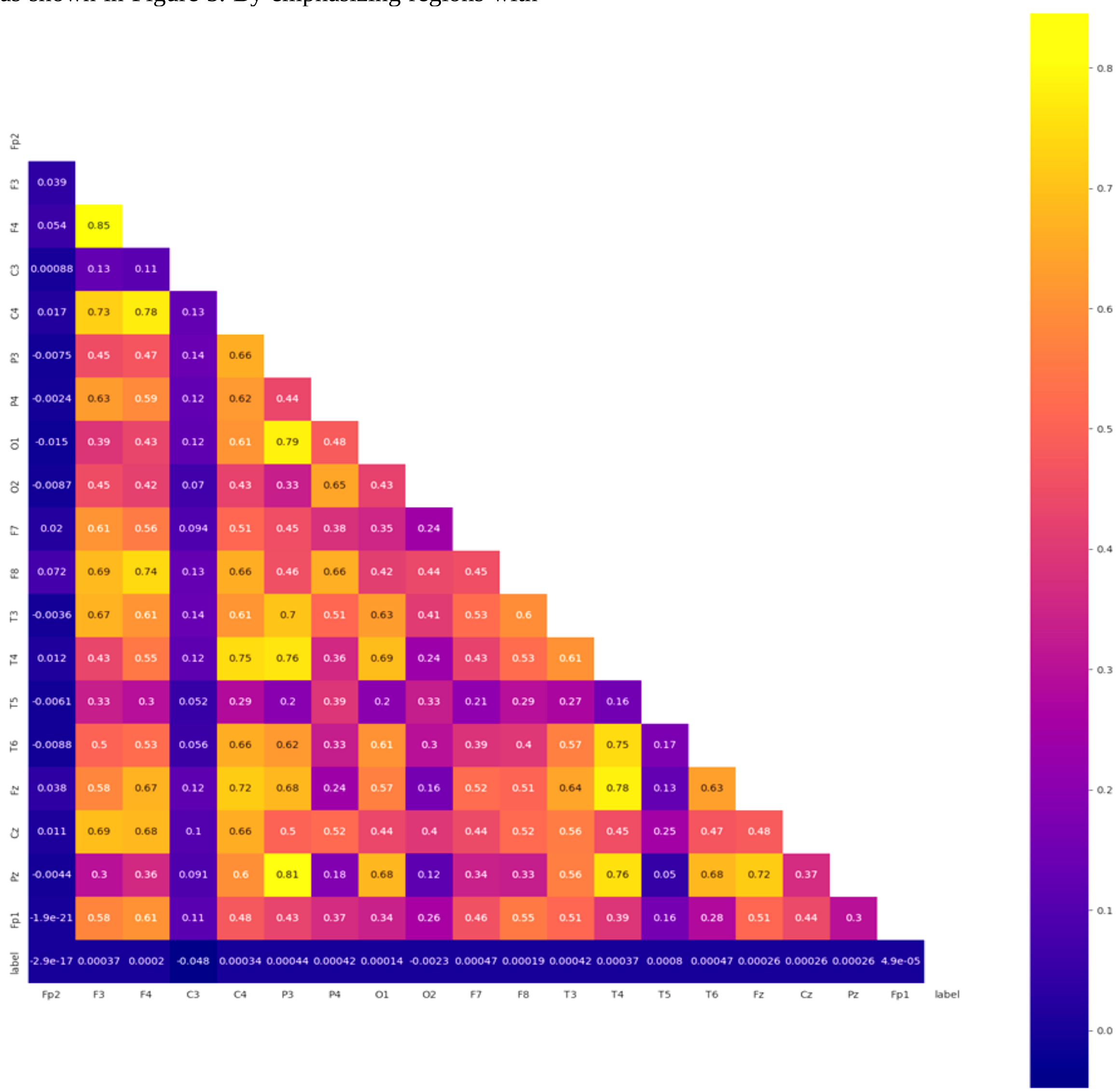

Fig. 5 Heatmap visualization

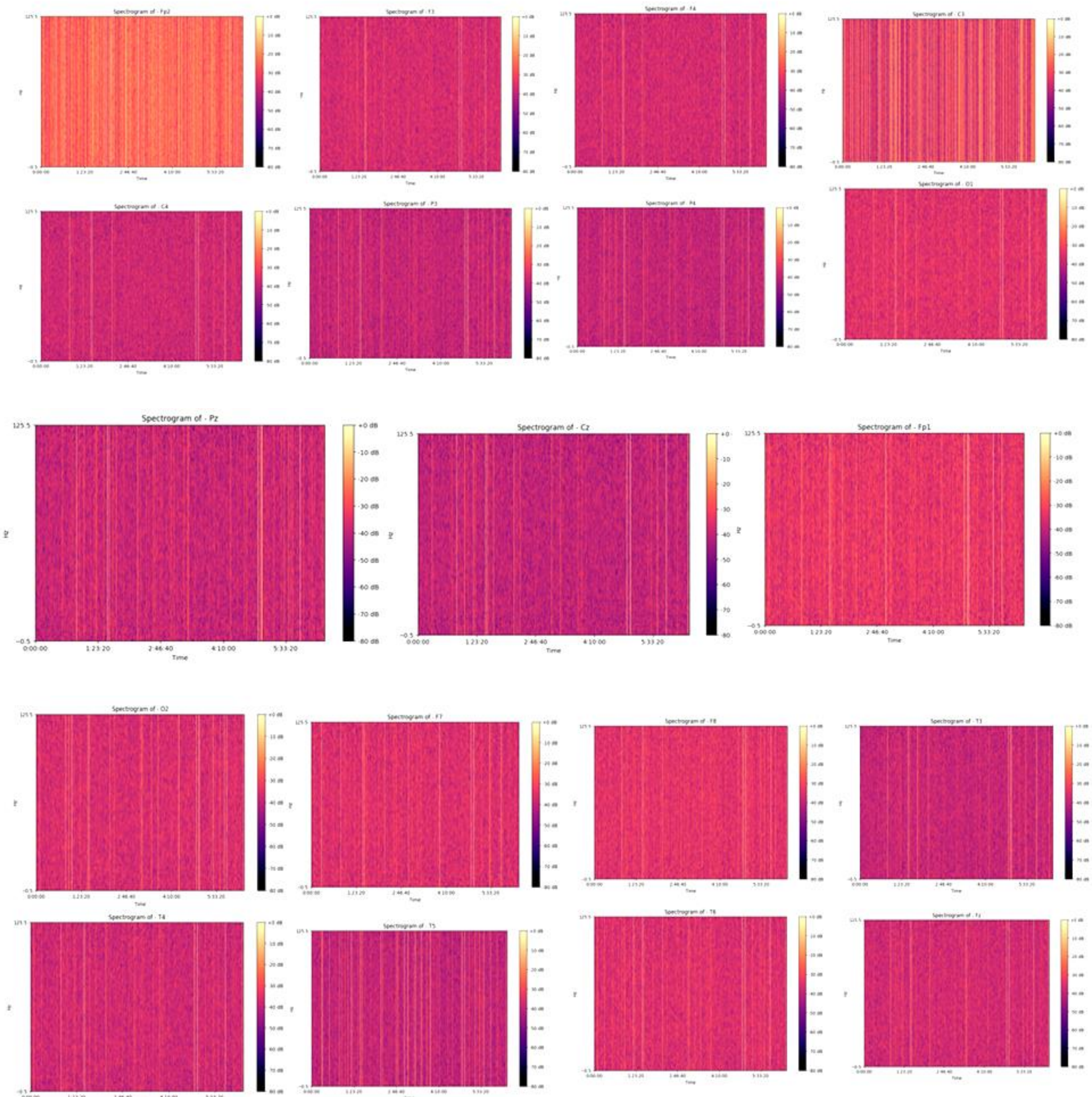

Fig. 6 Spectrograms of electrodes

3.2. Convolutional Neural Network

CNNs are a kind of Deep Neural Network that are frequently applied to the processing of visual images. They are capable of identifying and categorizing specific features from images. In the CNN model, Convolution operations are carried out by the Convolutional Layer, which forms the basis of CNN. This layer's Kernel is the part that handles the convolution (matrix) process. The stride rate determines how the kernel adjusts both horizontally and vertically until the entire image is scanned. The convolutional operation is represented by the following equations.

$$(I * K)(p,q) = \sum_m \sum_n I(m,n) \cdot K(p-m, q-n) \quad (1)$$

Where I is the input image, K is the convolution kernel, and (p,q) denotes the spatial position in the output feature map. The non-linear activation function is an additional crucial component of convolutional layers in addition to convolution. The activation function Relu is represented by the following equations.

$$f(p) = max\ (0, p) \quad (2)$$

By replacing negative values with zero, it introduces non-linearity to the network. The pooling layer helps to lower the amount of processing power needed to handle the data. The pooling operation is represented by the following equations.

$$MaxPooling\ (p,q) = max(I_{p,q}) \quad (3)$$

It reduces the spatial dimension of the feature map by retaining the maximum values within each pooling window. Every neuron in the Fully Connected layer (FC) is interconnected to every other neuron due to the flattened input. The flattened vector is then transmitted via a few more FC layers, which are typically where the functional operations related to mathematics are carried out. This is the point where the classifying process begins.

$$q = Wp + b \quad (4)$$

Where the bias vector is denoted by b, W is the weight matrix, the input vector is p, and q is the output vector. In the multiclass classification problem, the softmax function is an activation function that normalizes output real values from the last fully connected layer to target class probability.

$$Softmax(xi) = \frac{e^{pm}}{\sum_n e^{pn}} \quad (5)$$

It converts the raw scores (logits) into probabilities, ensuring that the sum of the probabilities across all classes equals one. A mask known as the Dropout layer eliminates certain neurons' contributions to the subsequent layer while maintaining the integrity of all other neurons. Because they keep the training data from overfitting, dropout layers are essential to CNN training.

3.3. Long Short-Term Memory

A Recurrent Neural Network (RNN) architecture, LSTM is characterized by a more sophisticated structure that includes extra memory cells and gates, which enable the model to recall or forget information from past time steps selectively. The basic architecture of an LSTM is visualised in Figure 7, consisting of input, hidden state, and output. An LSTM cell is made up of various parts. The information that enters the memory cell from the current input and the previously hidden state is managed by the input gate and is represented as:

$$i_t = \sigma(W_i.[[h_{t-1}, x_t] + b_i \sim C_t = tanh(W_c \cdot [h_{t-1}, x_t] + b_c\ ) \quad (6)$$

Information transfer from the previous memory cell to the current memory cell is managed by the forget gate. It enables the LSTM to choose to retain or forget data from earlier time steps. The sigmoid function is represented by σ, the weight matrices for the input gate are represented by W_i and W_c, the concatenation of the previous state and present input is represented by $[h_{t-1}, x_t]$, the bias vectors are represented by b_i and b_c.

$$F_t = \sigma\big(W_f.[h_{t-1}, x_t\big] + b_f) \quad (7)$$

b_f is the bias vector for the forget gate, the weight matrix W_f is for the forget gate, and the vector of forget gate values F_t for the current time step. The LSTM's internal state is called the memory cell. It retains data that the input and forget gates can modify selectively. The LSTM uses three different kinds of gates to regulate information flow inside the network. The information flows from the memory cell to the output, and the output gate manages the current hidden state.

$$O_t = \sigma(w_o.[h_{t-1}, x_t] + b_o)h_t = O_t * \tanh\,(c_t) \quad (8)$$

Where $w_o$is the weight matrix for the output gate. The LSTM receives a series of inputs during the forward pass, updating its hidden state and memory cell at each time step. The output gate creates the current hidden state and output by combining a sigmoid function with a tanh function. The LSTM is an excellent choice for jobs requiring the modeling of long-term dependencies because of its capacity to selectively recall or forget information from earlier time steps.

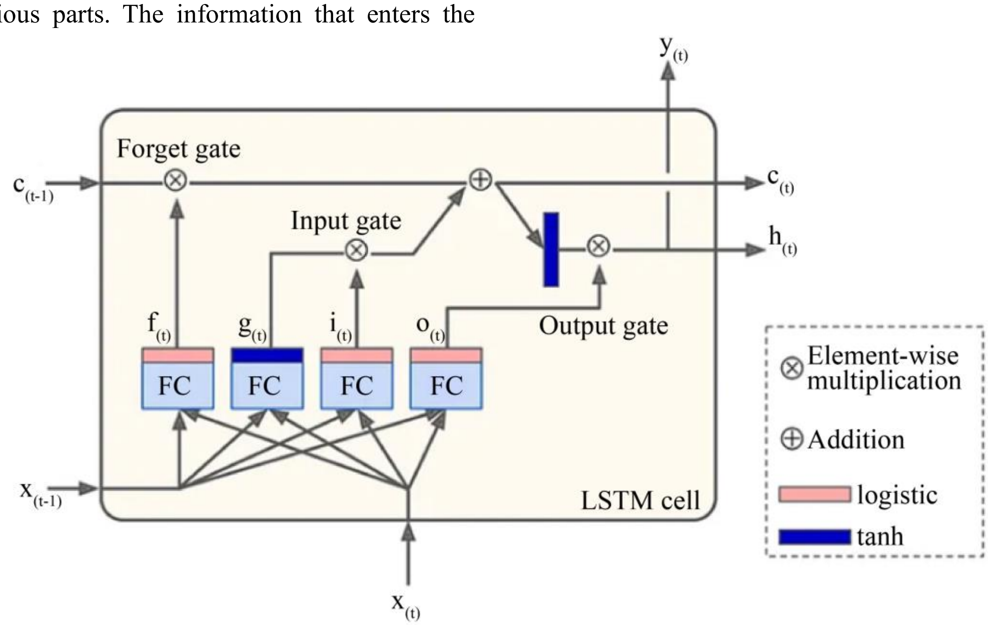

Fig. 7 Basic LSTM architecture

3.4. Gated Recurrent Unit

GRU describes sequential data by applying the concept of selective memory retention and forgetting, just like LSTM. However, GRU is simpler, more computationally efficient, and has fewer parameters than LSTM, which makes it easier to train. An input gate, an output gate, and a forget gate compose the basic architecture of the GRU, which is shown in Figure 8. To create a new hidden state, the update gate has to decide how much of the candidate activation vector and how much of the previous hidden state to include.

Using the prior hidden state h_{t-1} and the current input x, the reset gate r and update gate z are calculated.

$$r_t = Sigmoid(w_r * [h_{t-1}, x_t] \quad (9)$$

$$Z_t=\text{Sigmoid}\ (w_z * [h_{t-1}, x_t] \quad (10)$$

Where the weight matrices w_r and w_z are acquired during training. A "candidate activation vector" is calculated by the GRU at each time step by fusing data from the previous hidden state and present input. The concealed state is subsequently updated for the subsequent time step using this candidate vector. The candidate activation vector $h_{t\sim}$ is calculated using the current input x.

$$h_{t\sim} = \tanh\ (w_h * [r_t * h_{t-1}, x_t]) \quad (11)$$

w_h represents an additional weight matrix. the new hidden state h_t is measured as

$$h_t = (1 - z_t) * h_{t-1} + z_t * h_{t\sim} \quad (12)$$

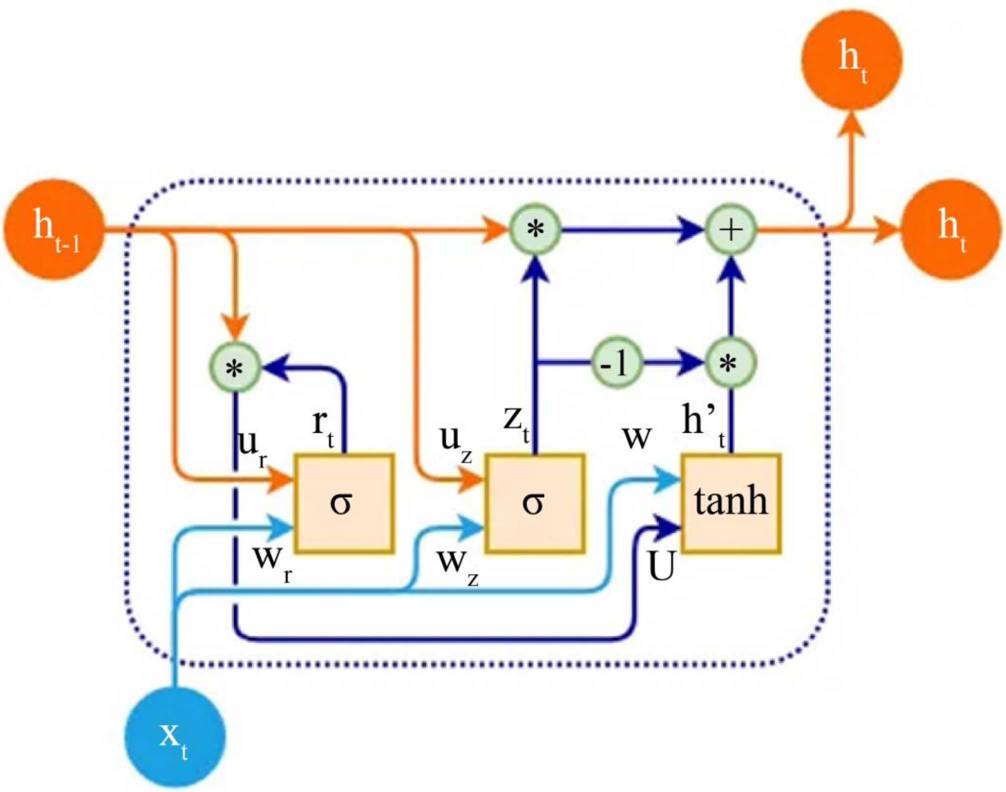

Fig. 8 Basic architecture of GRU

3.5. Proposed CNN- LSTM Architecture

The proposed architecture integrates CNN with LSTM networks to leverage the strengths of both components. The model framework is depicted in Figure 9. The model begins with a 1-D convolutional layer with a kernel size of 3 and 64 filters. This layer is responsible for learning spatial features from the EEG data, focusing on extracting local patterns from the input sequences. The convolutional layer's ability to capture intricate details at different spatial hierarchies is crucial for identifying subtle local patterns that may be indicative of schizophrenia. Following the convolutional layer, a max-pooling layer is applied to downsample the features. This reduces the dimensionality of the feature map, thereby focusing on the most salient features and improving the computational efficiency of the model. The downsample output from the max-pooling layer is then fed into the LSTM component of the hybrid model.

The LSTM layer in this architecture consists of 32 units and is designed to capture long-term dependencies and temporal patterns within the EEG signals. By processing the sequence of features extracted by the CNN, the LSTM layer retains the memory of past states and updates this memory based on the input sequence. This capability allows the model to detect subtle temporal changes and complex temporal dynamics in the EEG data, which are critical for distinguishing schizophrenia-related patterns. Finally, the output from the LSTM layer is passed through a flattened layer, which transforms the sequential output into a one-dimensional vector. This flattened representation is then fed into a dense layer with a sigmoid activation function, which produces the final binary classification output. The sigmoid function maps the output to a probability value between 0 and 1, indicating the likelihood of the presence or absence of schizophrenia.

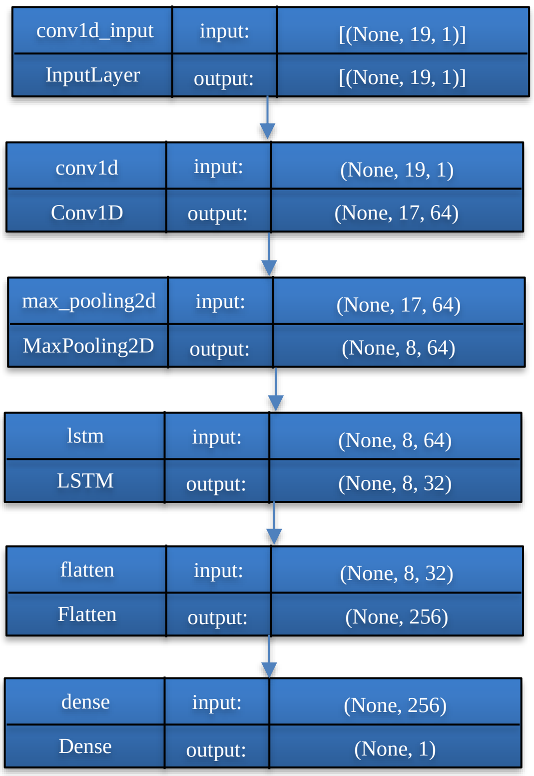

Fig. 9 Proposed CNN- LSTM model architecture

3.6. Proposed CNN- GRU Model

The proposed CNN-GRU hybrid model is a sophisticated deep learning architecture designed to detect schizophrenia from EEG (Electroencephalography) data by leveraging the strengths of both CNN and GRU. The model architecture is shown in Figure 10. The model architecture begins with a one-dimensional convolutional layer consisting of 64 filters with a kernel size of 3. The CNN component focuses on extracting local spatial patterns and features from the input sequences, capturing important nuances in the EEG signals. Following the convolutional layer, a MaxPooling one-dimensional layer with a pool size of 2. It reduces the dimensionality of the feature map by downsampling the extracted features. This process helps reduce the computational load and focus on the most relevant features, enhancing the model's ability to generalize new data well.

The output of the max-pooling layer is then fed into a GRU layer. The GRU layer in this model consists of 32 units and utilizes the ReLU activation function. This GRU layer is crucial for capturing temporal dependencies within the EEG data learning sequential patterns that may indicate schizophrenia over time. GRUs, known for their ability to retain important information over long sequences and their simpler structure compared to traditional LSTMs, make the model more computationally efficient while still effectively capturing temporal dynamics. After the GRU layer, the output is passed through a flattened layer. This layer transforms the multi-dimensional output from the GRU into a one-dimensional vector, making it suitable for the subsequent dense layer. The final layer in the model is the dense layer, which contains a single neuron with a sigmoid activation function. This activation function maps the output to a probability value between 0 and 1, indicating the likelihood of schizophrenia.

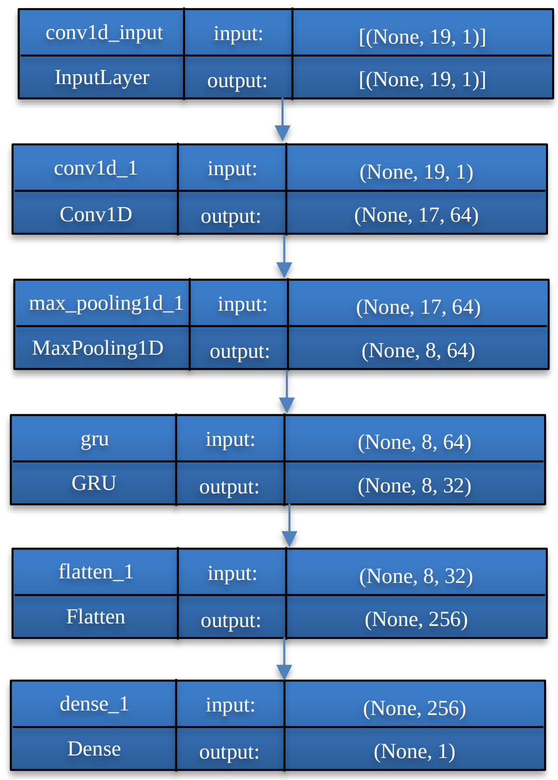

Fig. 10 Proposed CNN- GRU model architecture

3.7. Support Vector Machine

SVM is an ML technique used for linear or nonlinear classification, regression, and outlier detection. The primary goal of the SVM method is to identify the best hyperplane in an N-D space for dividing data points into various feature space classes. The goal of the hyperplane is to maintain the largest possible buffer between the nearest points of various classes. The number of features determines the dimension of the hyperplane. The hyperplane can be viewed as a line when the number of input features is two. If there are three input features, the hyperplane transforms into a 2-D plane. If there are more than three features, it becomes more difficult to imagine.

The hyperplane that shows the greatest margin or separation between the two classes is a plausible candidate for the best hyperplane. The hyperplane is selected with the maximum distance between it and the closest data point on each side. A hyperplane that satisfies this condition is referred to as the maximum-margin hyperplane or hard margin. The linear hyperplane equation is expressed as follows:

$$X^u y + c = 0 \tag{13}$$

The direction perpendicular to the hyperplane, or the normal vector, is represented by the vector X. The offset or distance of the hyperplane from the origin along the normal vector X is represented by the parameter c in the equation. To find the distance between the data point and the decision boundary:

$$D_m = \frac{X^u * y_i + c}{\|X\|} \tag{14}$$

Where the weight vector X's Euclidean norm is denoted by ||X||. For Linear SVM classifier:

$$\hat{Z} = \begin{cases} 1: X^u y + c \geq 0 \\ 0: X^u y + c < 0 \end{cases} \tag{15}$$

3.8. Proposed Stacking Ensemble Model

The proposed stacked ensemble method in this work combines predictions from multiple base models, specifically a CNN-LSTM and a CNN-GRU model. These base models generate predictions that capture different aspects of the EEG data, and these predictions are then stacked together horizontally to create a new feature set for each sample. This stacking process aggregates diverse information, potentially capturing complementary patterns, and the resulting feature set is used as input for an SVM classifier. The SVM, a robust and versatile classification algorithm, learns to combine these predictions effectively and finds the optimal decision boundary for the final classification, enhancing the accuracy of schizophrenia detection.

Table 1. Hyperparameters

Optimizer	Adam
Loss	Binary Crossentropy
Batch Size	128
Number of Epochs	10

3.9. Hardware and Software Setup

TensorFlow and Python were used during the creation and training of the model on Google Collaboratory. High-performance CPU and GPU were the main components of the hardware configuration, which allowed it to carry out the computational activities required for training effectively and assessing the deep learning models. A sophisticated processor, such as an AMD Ryzen or an Intel Core i9, was used to manage the computational load efficiently. In order to speed up the training of deep neural networks, which usually require labor-intensive matrix operations—a potent GPU, NVIDIA GeForce RTX, was employed. Adam is the optimizer used in the training process, while categorical crossentropy is the loss function. For training, a batch size of 128 samples in each iteration is used, and the process is carried out over 10 epochs. Python would have been the main programming language in the software stack because of its wide usage and abundance of libraries for deep learning and machine learning applications. TensorFlow on Google Collaboratory enabled web-based collaborative coding and deep learning model experimentation. TensorFlow is a cloud-based platform that provides free access to GPU resources.

4. Results and Discussion

The precision, recall, accuracy, and f1-score of the model are the metrics used to assess its performance. These parameters shed light on how well the model can handle imbalances between classes and categorize examples.

$$Accuracy = \frac{T_{pos}+T_{neg}}{T_{pos}+T_{neg}+F_{pos}+F_{neg}} \tag{16}$$

$$Precision = \frac{T_{pos}}{T_{pos}+F_{pos}} \tag{17}$$

$$Recall = \frac{T_{pos}}{T_{pos}+F_{neg}} \tag{18}$$

$$F1 - Score = 2\left(\frac{Precision * Recall}{Precision + Recall}\right) \tag{19}$$

In the hybrid model of CNN and LSTM, the accuracy improved from 97.18% to 97.53%, while the loss decreased, indicating effective learning. The accuracy and loss plot of the CNN-LSTM system is depicted in Figure 11. Initial accuracy was relatively high due to the model's inherent ability to capture data patterns. Accuracy generally increased with each epoch, though minor fluctuations occurred, which are normal due to factors like learning rate adjustments, data complexity, and the stochastic nature of gradient descent. These fluctuations were minor, showing a consistently upward trend in accuracy and suggesting that the model effectively learned and enhanced its performance over time.

At the start of training (Epoch 1), the model exhibits a loss of 3.431, indicating that the model's initial predictions are moderately accurate despite randomly initialized weights. By the final epoch (Epoch 10), the loss decreases to 0.214, showing successful learning and enhanced predictive performance.

The accuracy and loss plot of the CNN-GRU model is depicted in Figure 12. The CNN-GRU model starts with a training accuracy of 73.44% in Epoch 1 and improves to 98.24% by Epoch 10. This steady improvement indicates that the model effectively learns from the training data over time, with its predictions becoming more accurate. The training process proves effective, enabling the model to capture data patterns better and improve its prediction capabilities incrementally. At the start of training, the system's loss is 0.6106, which represents the initial error in the model's predictions. By the final epoch, the loss decreases to 0.129. This reduction in loss indicates that the model has become more accurate over time, as it has learned to make predictions that are closer to the actual target values. The consistent decrease in loss across the epochs reflects the model's improving performance and effective learning process.

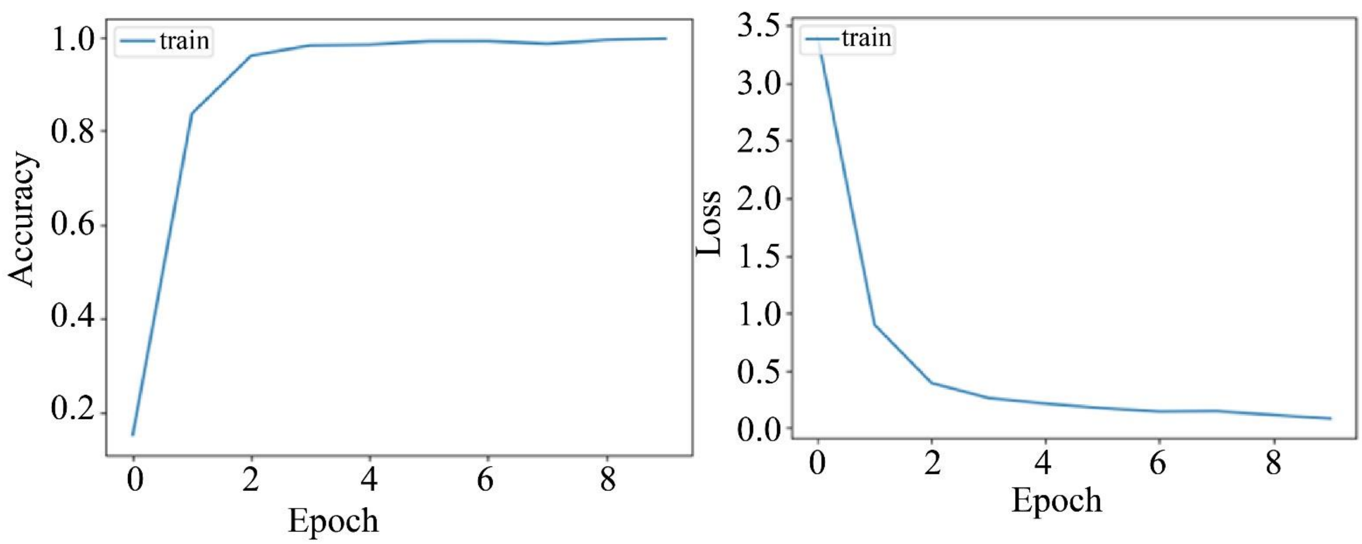

Fig. 11 Accuracy plot and loss plot of CNN- LSTM model

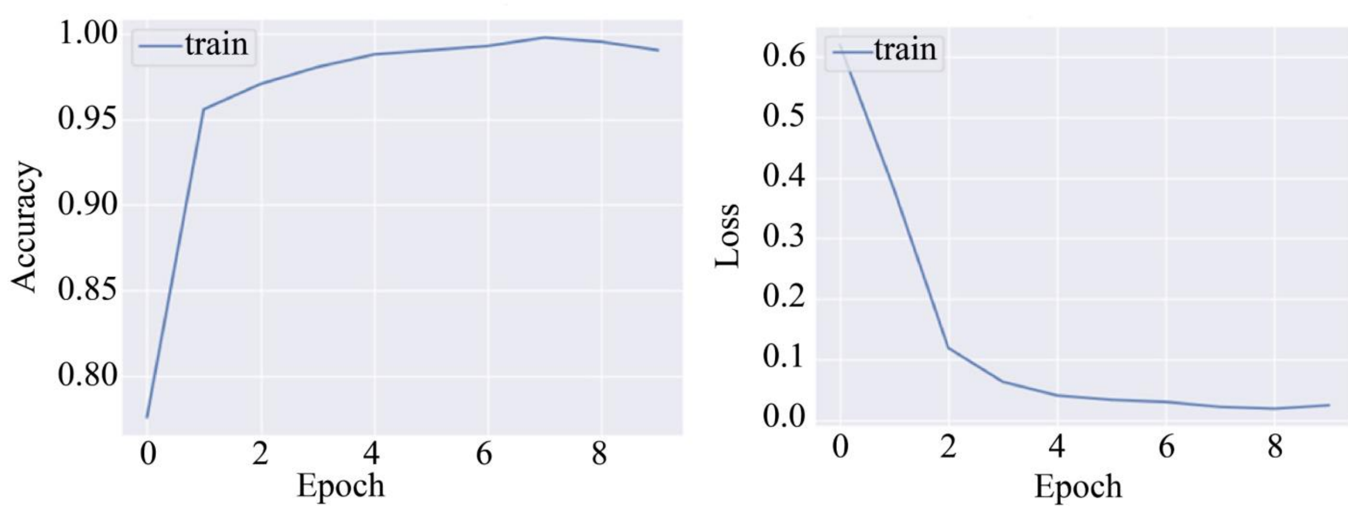

Fig. 12 Accuracy plot and loss plot of CNN- GRU model

Table 2. Classification report of proposed methodology

Performance Metrics	CNN-LSTM Model	CNN-GRU Model	Ensemble Model
Accuracy	0.9293	0.9310	0.9485
Recall	0.9234	0.9215	0.9492
Precision	0.9128	0.9226	0.9510
F1-Score	0.9197	0.9216	0.9500

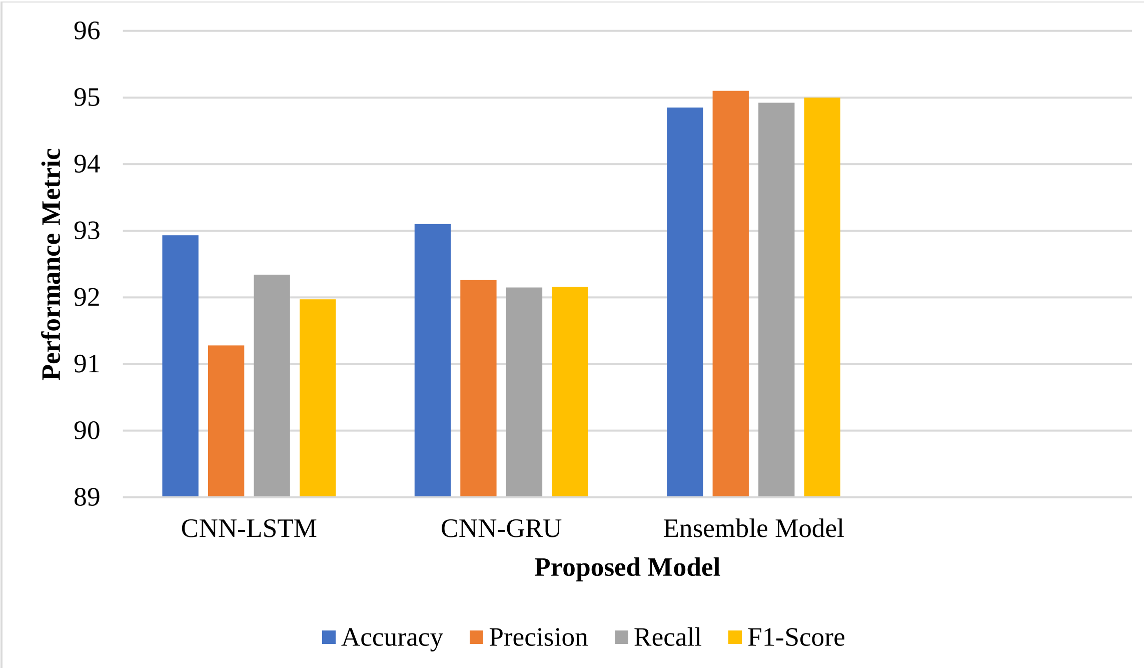

Fig. 13 Performance comparison of the proposed methodology

Table 3. Performance comparison with existing works

Author	Methodology	Accuracy
Barros et al.	CNN	78%
Hu et al.	Deep learning approach with two different datasets	81.02%
Patro et al.	Deep ensemble model	92.22%
Khare et al.	CNN	93.36%
Proposed Model	Ensemble model	94.85%

As shown in Figure 13, the classification report compares the performance metrics of three models: CNN-LSTM, CNN-GRU, and an Ensemble Model. The Ensemble Model, which combines the predictions of the CNN-LSTM and CNN-GRU models, demonstrates superior performance across all metrics. The accuracy of the Ensemble Model is 0.9485, higher than both the CNN-LSTM (0.9293) and CNN-GRU (0.9310) models. Similarly, the precision of the Ensemble Model is 0.9510, indicating that it is more effective at correctly identifying positive instances compared to the CNN-LSTM (0.9128) and CNN-GRU (0.9226) models. The recall of the Ensemble Model is 0.9492, showing that it has a higher sensitivity in detecting true positives than the CNN-LSTM (0.9234) and CNN-GRU (0.9215) models. The F1-Score, which balances precision and recall, is also highest for the Ensemble Model at 0.9500, compared to 0.9197 for CNN-LSTM and 0.9216 for CNN-GRU. This indicates that the Ensemble Model effectively leverages the strengths of both base models to achieve the best overall performance in schizophrenia detection.

5. Conclusion

Schizophrenia (SZ) is often diagnosed by a qualified psychiatrist through patient interviews. This procedure is laborious, time-consuming, and prone to mistakes, and prejudice. This paper introduces a staking ensample model for the identification of SZ patients using EEG signals. The proposed ensemble model, combining CNN-LSTM and CNN-GRU deep learning architectures with an SVM meta learner, demonstrates significant advancements in the detection of schizophrenia from EEG signals.

By leveraging high-quality data from the Kaggle repository, the ensemble model achieves an impressive accuracy of 94.85%, surpassing the individual performances of the CNN-LSTM (92.93%) and CNN-GRU (93.10%) models. This superior accuracy highlights the efficacy of the ensemble approach in enhancing diagnostic precision and reducing dependence on expert interpretation. Comparative analyses underscore the model's robustness and its potential to outperform current state-of-the-art methods, thus underscoring the promising role of deep learning in advancing psychiatric classification and contributing to the broader field of neuroscientific research.

Acknowledgments

I would like to express my sincere gratitude to all those who contributed to the completion of this research paper. I extend my heartfelt thanks to my supervisor, my family, my colleagues and fellow researchers for their encouragement and understanding during the demanding phases of this work.

References

[1] Anne Berghöfer et al., “Quality of Life in Patients with Severe Mental Illness: A Cross-Sectional Survey in an Integrated Outpatient Health Care Model,” *Quality of Life Research*, vol. 29, pp. 2073-2087, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[2] Vanteemar S. Sreeraj et al., “Effect of Add-On Transcranial Alternating Current Stimulation (tACS) on Persistent Delusions in Schizophrenia,” *Psychiatry Research*, vol. 290, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[3] Sher Singh, Deepa Khanna, and Sanjeev Kalra, “Role of Neurochemicals in Schizophrenia,” *Current Psychopharmacology*, vol. 9, no. 2, pp. 144-161, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[4] Sergey N. Mosolov, and Polina A. Yaltonskaya, “Primary and Secondary Negative Symptoms in Schizophrenia,” *Frontiers in Psychiatry*, vol. 12, pp. 1-12, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[5] Ayşe Gül Eker, Nevcihan Duru, and Kadir Eker, “EEG Signals and Spectrogram with Deep Learning Approaches Emotion Analysis with Images,” *2022 7th International Conference on Computer Science and Engineering (UBMK)*, Diyarbakir, Turkey, pp. 1-5, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[6] Robert A. McCutcheon, John H. Krystal, and Oliver D. Howes, “Dopamine and Glutamate in Schizophrenia: Biology, Symptoms and Treatment,” *World Psychiatry*, vol. 19, no. 1, pp. 15-33, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[7] Zülfikar Aslan, and Mehmet Akin, “A Deep Learning Approach in Automated Detection of Schizophrenia Using Scalogram Images of EEG Signals,” *Physical and Engineering Sciences in Medicine*, vol. 45, no. 1, pp. 83-96, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[8] Aleksander Wawer et al., “Single and Cross-Disorder Detection for Autism and Schizophrenia,” *Cognitive Computation*, vol. 14, pp. 461-473, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[9] Carla Barros et al., “From Sound Perception to Automatic Detection of Schizophrenia: An EEG-Based Deep Learning Approach,” *Frontiers in Psychiatry*, vol. 12, pp. 1-17, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[10] Mengjiao Hu et al., “Structural and Diffusion MRI based Schizophrenia Classification using 2D Pretrained and 3D Naive Convolutional Neural Networks,” *Schizophrenia Research*, vol. 243, pp. 330-341, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[11] P. Supriya Patro et al., “Lightweight 3D Convolutional Neural Network for Schizophrenia Diagnosis Using MRI Images and Ensemble Bagging Classifier,” *arXiv*, pp. 1-25, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[12] Jihoon Oh et al., “Identifying Schizophrenia using Structural MRI with a Deep Learning Algorithm,” *Frontiers in Psychiatry*, vol. 11, pp. 1-11, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[13] Smith K. Khare, Varun Bajaj, and U. Rajendra Acharya, “SPWVD-CNN for Automated Detection of Schizophrenia Patients Using EEG Signals,” *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1-9, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[14] JinChi Zheng et al., “Diagnosis of Schizophrenia Based on Deep Learning Using fMRI,” *Computational and Mathematical Methods in Medicine*, vol. 2021, no. 1, pp. 1-7, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[15] Yesh Khandelwal, Ensemble Stacking for Machine Learning and Deep Learning, Analytics Vidhya, 2021. [Online]. Available: https://www.analyticsvidhya.com/blog/2021/08/ensemble-stacking-for-machine-learning-and-deep-learning/

[16] Yan Holtz, Heatmap, Data-to-Viz.com. [Online]. Available: https://www.data-to-viz.com/graph/heatmap.html

[17] *Chapter 1: Introduction to EEG Spectrograms*, Atlas of Intensive Care Quantitative EEG, Springer Publishing Company, 2023. [CrossRef] [Publisher Link]