International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

EEG Signal Processing for the Identification of Sleeping Disorder Using Hybrid Deep Learning with Ensemble Machine Learning Classifier

Swati Gawhale[1], Dr. Dhananjay E. Upasani[2], Leena Chaudhari[3], Dr. Dhananjay V. Khankal[4], Jambi Ratna Raja Kumar[5], Vidyabhushan A. Upadhye[6]

Submitted: 25/05/2023 **Revised**: 09/07/2023 **Accepted**: 28/07/2023

Abstract : It can be difficult for healthcare professionals to recognise and diagnose sliding disorder, a neurological ailment marked by a loss of coordination and control over movement. Signals from electroencephalography (EEG) have shown to be a useful method for examining brain activity and can shed light on neurological conditions. Using a hybrid deep learning framework and an ensemble machine learning classifier, we suggest a unique method in this study for the detection of sliding disorder. In the first step of our procedure, EEG signals from healthy controls and people with Sleeping disease are collected. To collect the necessary information, these signals are divided into shorter time intervals after being preprocessed to remove noise and artefacts. In order to obtain a concise representation of the EEG data, feature extraction techniques are used. This aids in highlighting significant patterns and traits connected to Sleeping disease. The proposed methodology is intended for integration into embedded devices to provide a novel and effective method for classifying sleep stages. For evaluation, the study makes use of Power Spectrum Density (PSD) Dataset. We experimented with a publicly accessible Power Spectrum Density (PSD) dataset of patients with sliding disease in order to assess the effectiveness of our suggested strategy. The outcomes show that our method outperforms both conventional machine learning algorithms and stand-alone deep learning models in terms of sliding disorder identification. The use of Hybrid Deep Learning with Ensemble Machine Learning Classifier together effectively enhance classification sensitivity of 89.06% and accuracy to 96.78%.

***Keywords**: EEG Signal, Hybrid Deep Learning, Ensemble Machine Learning, CNN, Sleeping Disorder.*

1. Introduction

Sleep is an essential brain function that has a big impact on how well people work, learn, and move [1] Eyes closed, some nervous system centres go into active state during sleep, which causes partial or total unconsciousness and less sophisticated brain activity [13]. Humans spend over a third of their lives sleeping, so it is critical to treat sleep disorders like Obstructive Sleep Apnea (OSA), which can have serious negative effects on physical health [14,]. Notably, sleep difficulties affect more than 90% of those with depressive disorders [6,]. Around 2 to 4% of adults and 1 to 3% of children suffer from sleep apnea, while 33% of people worldwide experience symptoms of insomnia [15]. Consequences of sleep-related problems might range from melancholy and daily drowsiness to life-threatening conditions [6], [15]. For instance, drowsy driving is causes at least thousands of car accidents yearly only in US, and sleep-related factors are a major factor in many traffic fatalities and accidents around the world. Given these concerning data, it is crucial to create automated tools that can monitor sleep patterns and detect symptoms like weariness, drowsiness, and sleep disorders like apnea, insomnia, or narcolepsy.

As the gold standard for identifying and treating sleep problems, sleep stage scoring is a crucial step in the investigation of human sleep [17]. The process of identifying and classifying several sleep stages, which are crucial for comprehending sleep patterns, is known as sleep stage scoring. Polysomnographic (PSG) recordings made from patients during overnight sleep studies carried out in medical settings [5] are the main source of data used in the grading process. Sleep disorders are illnesses that interfere with a person's regular sleeping patterns. Compared to younger people, the elderly are more likely than other demographic groups to experience sleep disturbances. Insomnia and primary sleep disorders are two common forms of sleep disorders that are widespread in the aged population [1-3]. A substantial percentage of elderly persons (between 40% and 50%)

[1]Assistant Professor, Bharati Vidyapeeth's College of Engineering, Lavale, Pune, Maharashtra, India
[2]Principal, Samarth College of Engineering, Belhe, Pune, Maharashtra, India
[3]Assistant Professor, Bharati Vidyapeeth's College of Engineering, Lavale, Pune, Maharashtra, India
[4]Professor, Department of mechanical Engineering, Sinhgad College of Engineering, Pune Maharashtra, India.
[5]Associate Professor, Computer Engineering department, Genba Sopanrao Moze College of Engineering, Balewadi, Pune, Maharashtra, India.
[6]Lecturer, Bharati Vidyapeeth's Jawaharlal Nehru Institute of Technology, Pune, Maharashtra, India.
leenapc23@gmail.com1, upasanide@gmail.com2, gawhaleswati@gmail.com3, dhananjaykhankal@sinhgad.edu4 , ratnaraj.jambi@gmail.com5, vidyabhushan.upadhye@bharatividyapeeth.edu6

suffer from insomnia, which is characterised by difficulties falling asleep [2]. It may result in severe distress and impairment of day-to-day activities. Elderly people frequently experience main sleep disturbances in addition to insomnia. These disorders, which include illnesses like sleep disordered breathing (SDB), REM behaviour disorder (RBD), and restless legs syndrome (RLS), are not connected to underlying psychological issues. The aged population's overall health and sleep quality can be greatly impacted by these basic sleep disorders. They need a correct diagnosis and course of treatment to reduce symptoms, enhance sleep quality, and improve overall health.

2. Review of Literature

Heart rate variability (HRV) is frequently computed using spectral analysis when studying different stages of sleep. The three frequency bands that make up HRV are VLF, LF, and HF. The information in these frequency ranges is useful for determining the stage of sleep [10]. Longer data segments are needed for the accurate computation of nonlinear HRV indexes, which provide additional insights into heart dynamics. There should be a minimum of five minutes between segments, according to studies. The results of the estimated sleep staging, though, may have less resolution with longer portions [10].

The Pan-Tompkins algorithm is frequently used for the preprocessing step of Electrocardiogram (ECG) signals. This method is renowned for its ability to precisely locate the QRS complex, a fundamental component of all autonomous ECG-based systems. When dealing with noisy ECG readings, this becomes extremely crucial [11].

The reliable examination and interpretation of ECG signals, that is necessary for a number of applications, including sleep stage identification, are made possible by the correct detection of the QRS complex. Researchers and practitioners can improve the accuracy and resilience of ECG-based systems used in sleep studies and related fields by implementing the Pan-Tompkins method.

Long et al. [15] contend that the use of fixed boundaries in HRV to define frequency bands may not completely capture all facets of autonomic nervous system (ANS) activity. This restriction may affect HRV's ability to distinguish between sleep and wakefulness. In order to solve this, modified HRV spectral features have been added to improve the power discrimination for classifying sleep and waking. The study's findings showed that combining adapted HRV spectral data with other chosen HRV non-spectral features considerably enhances classification performance overall, including classification of wakefulness and sleep. However, the computed features for both the low-frequency (LF) and high-frequency (HF) bands may be impacted by the spectrum's overlapped components.

In a recent study [16], the authors suggested a technique to identify and examine the stages of sleep in order to evaluate patients' sleep conditions. An approach that is frequently used to examine sleep patterns is sleep stage detection. Earlier studies concentrated on creating EEG, EMG, and EOG -based automatic sleep stage recognition systems. These signals are not suited for use in portable household products due to the difficulty of the recording and signal analysis processes.

Researchers are starting to investigate the use of Electrocardiogram (ECG) data for sleep stage detection in an effort to overcome this constraint. The sleep stage classification has been done using a variety of feature extraction methods and machine learning classifiers [17–19]. However, the majority of current research primarily examines particular phases of sleep. It offers a comprehensive method for detecting all stages of sleep by utilising ECG signals, cutting-edge feature extraction methods, and machine learning classifiers. The suggested system has the potential to advance our knowledge of the relationship between sleep quality and associated medical disorders and the development of portable sleep monitoring devices.

Studies [20,21] have used Electrocardiogram (ECG) signals to gauge the quality of sleep. An SVM classifier with many classes has been used to categorise sleep quality in the backend phase [22]. The sleep efficiency index, delta-sleep efficiency index, and sleep onset latency results for this strategy are promising. The classification of sleep stages in binary form and the estimate of sleep efficiency using ECG data were the main topics of another study [23]. For the input of 12 features, the proposed system's average error rate was 4.52%, and for the input of ten features, it was 4.64%.

The use of these signals for diagnosing and treating different sleep disorders is rather limited when compared to the use of multichannel signals like those used in polysomnography (PSG), even though some studies have used basic physiological signals to detect sleep states. However, a few studies have looked into using weak physiological signals for this purpose.

A sleep arousal detection system was created in one study [30]. The K-nearest Neighbours (KNN) classifier was used, with average results of 79% sensitivity, 95.5% specificity, and 93% accuracy. It's important to keep in mind that this method does not account for specific sleep issues or different arousal levels. These findings imply that more research is necessary to develop diagnostic and therapeutic approaches for a variety of sleep disorders, despite the possible utility of restricted physiological markers in diagnosing sleep arousal.

3. Publically Available Datasets

The number of PolySomnoGraphic sleep recordings in the Sleep-EDF database has increased, and there are currently 197 of them. Events markers, electromyograms, electrooculograms, and electroencephalograms (EEGs) of the head and chin are among the physiological data captured in these recordings. Measurements of body temperature and respiration are also included in certain records. The Sleep-EDF Database improved version, available on PhysioNet, provides a comprehensive collection of sleep recordings. It contains 61 polysomnograms taken over the course of a full night, including those of healthy adults and people with mild sleep issues. An in-depth analysis and interpretation of sleep patterns are possible thanks to the recordings' expert annotations of the various stages of sleep.

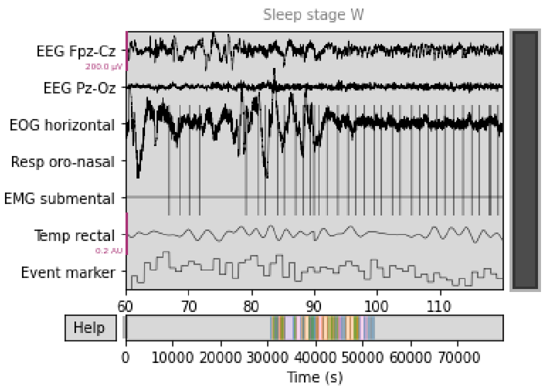

Fig 1: Graphical description of PSDdataset

The Sleep-EDF database's usability and research potential are improved by including more subjects and making expert annotations available. Researchers can use this helpful resource to learn more about sleep disorders, create novel sleep stage classification algorithms, and acquire understanding of sleep-related events, figure 1 shows the graphical description of dataset.

4. Classification Techniques and Proposed Method

The suggested approach seeks to identify Sleeping disorder by combining EEG signal processing methods with hybrid deep learning and ensemble machine learning classifiers. A person's health and ability to operate on a daily basis can be greatly impacted by Sleeping disorder or sleep difficulties. The procedure starts with the collecting of EEG signals, which offer important details about brain activity while you are sleeping. These signals go through pre-processing to get rid of any noise or artefacts that can obstruct the analysis. Filtering, baseline correction, and artefact removal are examples of pre-processing techniques as shown in figure 2.

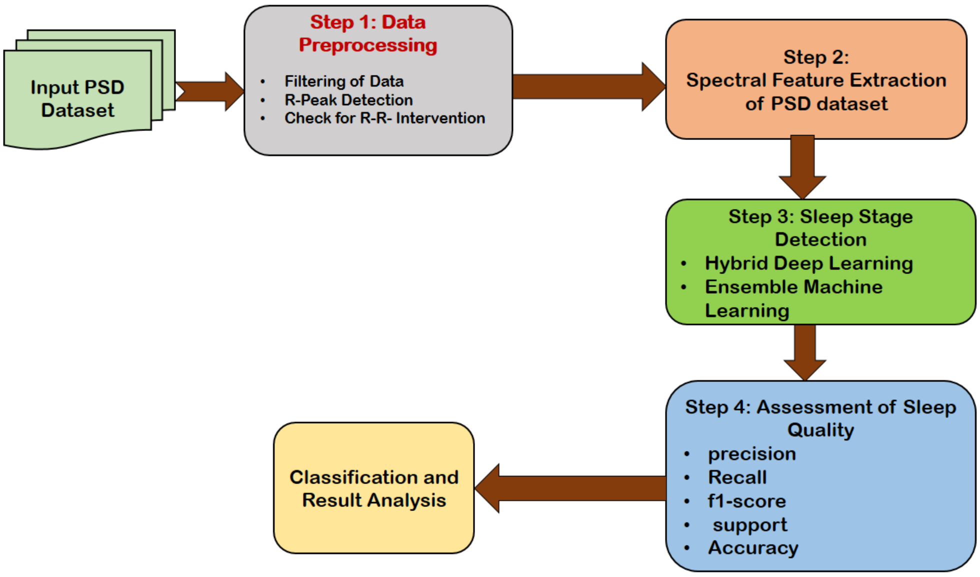

Fig 2: Proposed system architecture for Identification of Sleeping Disorder

1. Input Dataset:

The Sleep-EDF database (expanded), which is accessible on the Physionet website, provided the EEG signal used in this investigation. The database includes older Sleep EDF database records from before 1991 as well as 61 polysomnographic (PSG) recordings that were gathered between 1987 and 2002. Twenty healthy subjects between the ages of 25 and 34 were recorded; there were ten male and ten female participants. Two PSG recordings totaling about 20 hours were made at each subject's residence. However, due to a technical issue, subject 13's second night recording was not accessible. Signals including EEG bands, submental chin EMG, and event markers are included in the PSG recordings. Rectal body temperature and oro-nasal respiration are additional signs shown in figure 3.

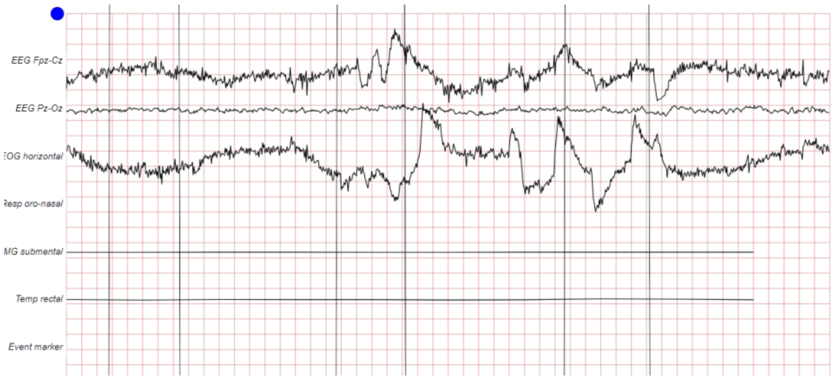

Fig 3: Signals including EEG Fpz-Cz, EEG Pz-Oz, EOG horizontal

The EEG input in this investigation only accepts the EEG Fpz-Cz signal. The EEG signals are sampled at a rate of 100 Hz. The hypnogram files connected to the recordings offer details on each subject's sleeping habits.

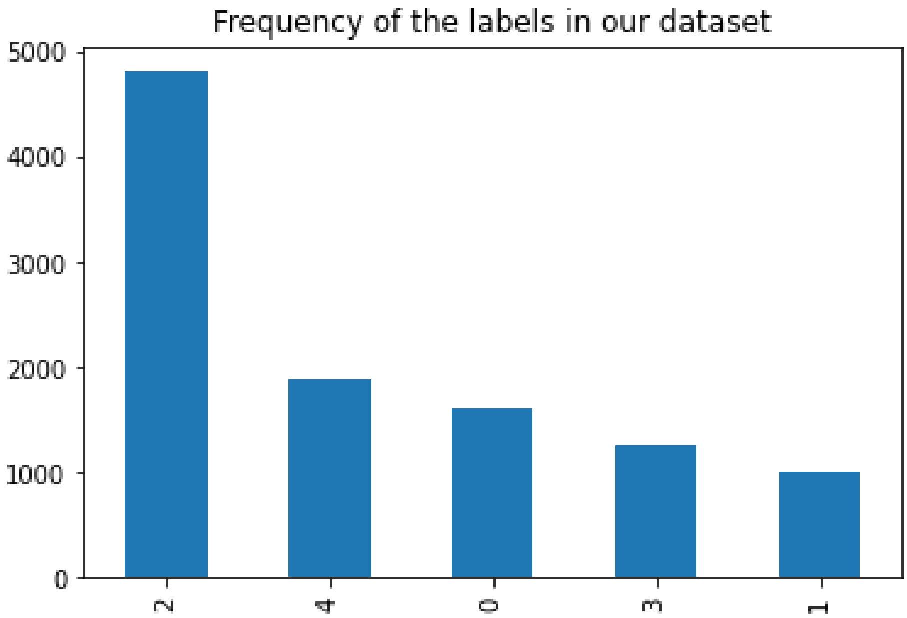

Fig 4: Frequency count of Each label in dtaset

Each EEG signal is analysed in 10-second time frames for the proposed work. An illustration, in figure 4, of EEG signal samples from five different stages, which were used as inputs for the created filters, is included in the paper. The quantity of waking, Stage 1, Stage 2, Stage 3, Stage 4, and REM epochs for each subject in the database is also shown in a table.

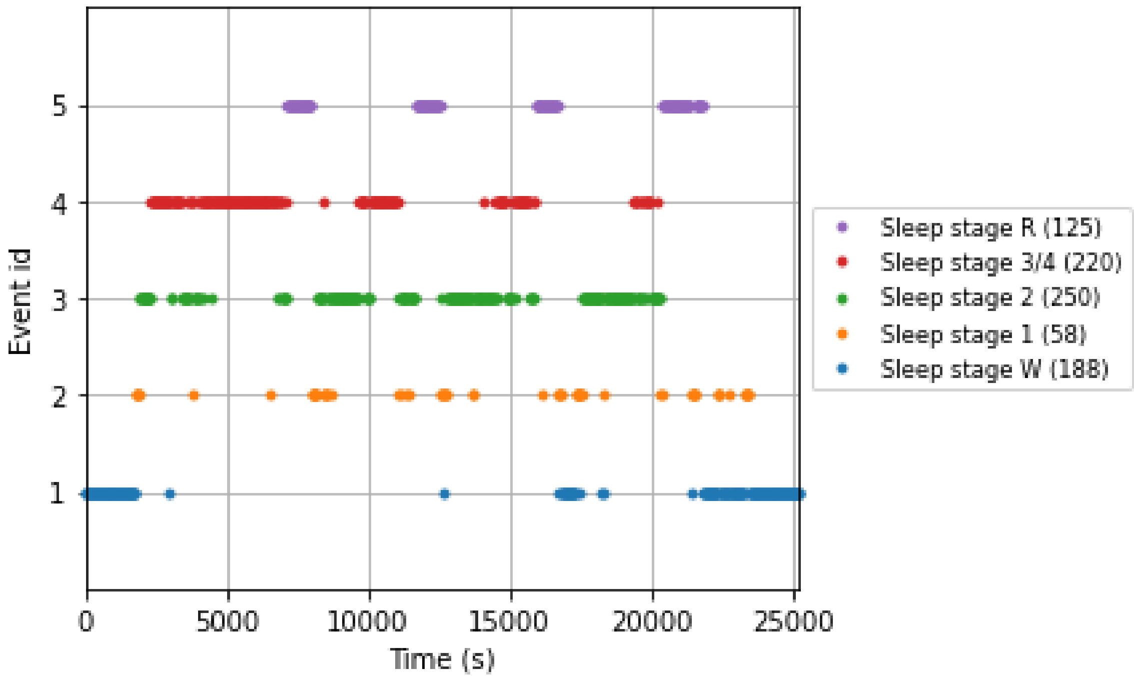

Fig 5: Representation of EEG signal samples from five different stages

The EEG data utilised in this research was taken from the PSD Sleep-EDF database, which is an excellent resource for analysing sleep patterns and developing diagnostic algorithms.

2. Data Pre-processing:

The proposed technique depends heavily on the pre-processing phase. This study uses a pre-processing method that doesn't need detecting the QRS locations and uses time-domain ECG signal epochs of 20 seconds.

For the examination of sleep stages, a standard recording duration of 30 seconds is used [32,33]. This time frame was selected with the presumption that it enables the lowest frequency range needed to capture the HRV spectrum's very low frequency (VLF) band power. the initial 30 second period of an insomniac patient's ECG signal. The method's pre-processing phases can be summed up as follows:

- To remove noise from the ECG signal, a band-stop filter and a moving average filter are used in conjunction.
- R-peak recognition A straightforward R-peak detector is used, with a threshold of 70% of the ECG signal's maximum amplitude. The R-peaks' positions are located using this threshold.
- R-R interval interpolation: The R-R intervals are sampled with 2.5 Hz frequency after being interpolated. Now, the Re-sampling is carried out with 0.4Hz frequency taken into consideration, in order to confirm that the time-series signal satisfies the Nyquist-Shannon sampling theorem. This makes it possible to calculate the HRV accurately.

3. Feature Extraction:

The ECG band signals are retrieved in the second stage of the suggested methodology. These include the subsequent actions:

- Power Spectral Density (PSD): The FFT-based Welch method is used to convert the R-R intervals of the ECG signal. This technique, which is non-parametric, offers a precise and high-resolution estimation of the spectrum calculation. The PSD depicts how power is distributed throughout several frequency bands. It can be calculated as per following:

$$P(\omega,k) = N\cdot M1\sum m = 0M-1 \,||\, \sum n = 0N-1w(n)\cdot x(m\cdot L+n)\cdot e - j\omega n \,||\, 2$$

Where, $P(\omega,k)$ is the estimated power spectral density at frequency ω and window index k.

- Application of a Hanning window: A Hanning window is applied to each epoch of the PSD representation to minimise spectral leakage in the final spectrum. A specific kind of window function called the Hanning window aids in enhancing frequency resolution.

The distance has been calculated as:

$$d = \sqrt{\Delta x2 + \Delta y2}$$

- HRV analysis: In order to extract HRV features, the PSD representation of each epoch is further examined. These characteristics can be used to describe the activities of the autonomic nervous system when you are sleeping and offer details about the variability in the heart rate at various frequency bands and it represent as:

$$\mathrm{MMD} = \sum \mathrm{wi} = 1|\mathrm{di}|$$

The study and classification of the various stages of sleep are aided by the retrieved HRV properties.

4. Sleep Stage Detection:

A a number of ML and DL classifiers were used in the classification stage to assess the effectiveness of classifying sleep stages. The following list of classifiers was utilised in this study:

1. Classification using deep learning: A deep learning classifier was used to categorise the six stages of sleep. It utilised a feed-forward neural network architecture with two layers (input and output).
2. CNN Algorithm: To categorise the various stages of sleep, the Convolutional Neural Network (CNN) algorithm was used as a classifier. The handling of sequential data, such as EEG signals, is where CNNs excel.
3. A hybrid algorithm integrating CNN and Machine Learning (ML) methods was utilised to categorise the various stages of sleep. To enhance classification performance, this hybrid technique incorporates the advantages of both CNN and ML algorithms.
4. HYBRID ALGORITHM (CNN + LSTM): LSTM and CNN were combined to create a new hybrid algorithm.
5. ENSEMBLE CLASSIFIER + HYBRID CNN: The proposed hybrid classifier combines ensemble learning methods with CNN. In ensemble learning, various models are combined to improve overall performance. The ensemble classifier in this instance uses the CNN as a foundation model.

5. Classification Techniques

1. CNN Classifier:

The classification EEG signal for Sleeping Disorder is a challenge for CNNs. They are made up of pooling layers for dimensionality reduction and convolutional layers that automatically identify pertinent characteristics from input photos. CNNs are frequently employed in computer vision problems because they are good at capturing spatial dependencies [19].

- Layer of Input: A collection of photographs are input as a matrix of pixel values.
- Convolutional Layer: To extract local characteristics from the input image, convolutional filters (also known as kernels) are employed. A definition of the convolution operation is:

$$S(i, j) = (I * K)(i, j)$$

Where, If I is the input picture, K is the convolutional filter, and S(i, j) is the value at position (i, j) in the feature map.

- Activation Function: To introduce non-linearity, an activation function, such as ReLU, is applied element-by-element:

$$A(x) = Max(0, x)$$

- Pooling Layer: To minimise the spatial dimensions, max pooling or average pooling is performed.
- Fully Connected Layer: A fully connected layer is connected to the flattened vector created from the combined feature maps. Each neuron in the layer with complete connectivity is linked to every neuron in the layer below.
- Output Layer: The last layer is connected to an output layer that generates the categorization or prediction that is needed.
- Loss Function: To calculate the error, a suitable loss function is used. For multi-class classification, softmax cross-entropy is a common loss function.

2. HYBRID ALGORITHM (CNN + ML)

CNNs and ML methods are combined in the HYBRID ALGORITHM (CNN + ML) to enhance classification performance. By combining CNNs and ML algorithms, this hybrid technique can improve classification task accuracy and robustness.

a. Navie Bayes with CNN Algorithm:

Machine learning activities involving classification make extensive use of it. The Naive Bayes algorithm is described in the following manner:

Use Bayes' theorem to get the posterior probability for each class given a fresh input instance with feature values.

As the projected class for the input instance, choose the class with the highest posterior probability. It has given below:

Calculate each class's probability in the training dataset using the prior probability (P(C)) formula.

$$P(C = c) = \frac{count(C = c)}{count(total; instances)}$$

Probability of Likelihood (P(X|C)): Calculate the likelihood of each feature value given each class.

Different probability estimation strategies can be utilised depending on the type of feature:

For category characteristics:

$$P(X_i = x_i | C = c) = \frac{count(X_i = x_i, ; C = c)}{count(C = c)}$$

Assume that continuous characteristics follow a particular distribution, such as the Gaussian (normal) distribution.

$$P(X_i = x_i|C = c) = \frac{1}{\sqrt{2\pi\sigma_c^2}} \exp\left(-\frac{(x_i - \mu_c)^2}{2\sigma_c^2}\right)$$

Using the Bayes theorem, get the posterior probability for each class given the feature values (P(C|X)).

$$P(C=c|X=x_1,x_2,\ldots,x_n)=\frac{P(C=c)P(X=x_1|C=c)P(X=x_2|C=c)\ldots P(X=x_n|C=c)}{P(X=x_1,x_2,\ldots,x_n)}$$

Prediction: For the input instance, choose the class with the highest posterior probability as the forecast class.

Convolutional Layer for CNN : To extract local characteristics from the input image, convolutional filters (also known as kernels) are employed. A definition of the convolution operation is:

$$S(i, j) = (I * K)(i, j)$$

Where, If I is the input picture, K is the convolutional filter, and S(i, j) is the value at position (i, j) in the feature map.

Activation Function for CNN: To introduce non-linearity, an activation function), is applied element-by-element:

$$A(x) = Max(0, x)$$

b. **Random Forest with CNN:**

Run each decision tree in the Random Forest with a new input instance to obtain predictions.

Each tree "votes" for a class during classification, and the class with the most votes is chosen as the final forecast.

For regression, the average or majority vote is selected as the final forecast. Each tree predicts a value.

Mathematical Model:

The Gini index calculates the degree of mixed classes or impurity in a node.

$$Gini(D) = 1 - \sum_{i=1}^{c}(P(C_i))^2$$

Information Gain: Information gain is a measure of a node's degree of disorder or a reduction in entropy.

$$IG(D, ; A) = H(D) - \sum_{v=1}^{V}\frac{|D_v|}{|D|}H(D_v)$$

Out-of-Bag Error: During training, some cases are left out of the decision tree's building. Out-of-bag (OOB) instances are what these situations are known as. Without the requirement for a separate validation set, the performance of the Random Forest can be estimated using the OOB error.

$$OOB; Error = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{K_i}\sum_{k=1}^{K_i}\mathbb{I}(Y_i \neq \hat{Y}_i)$$

To extract local characteristics from the input image, convolutional filters (also known as kernels) are employed. A definition of the convolution operation is:

$$S(i, j) = (I * K)(i, j)$$

Where, If I is the input picture, K is the convolutional filter, and S(i, j) is the value at position (i, j) in the feature map.

c. **Decision Tree with CNN:**

To extract local characteristics from the input image, convolutional filters (also known as kernels) are employed. A definition of the convolution operation is:

$$S(i, j) = (I * K)(i, j)$$

Where, If I is the input picture, K is the convolutional filter, and S(i, j) is the value at position (i, j) in the feature map.

Activation Function: To introduce non-linearity activation method is applied element-by-element:

$$A(x) = Max(0, x)$$

Gini Index for classification

$$Gini(p) = 2p(1 - p)$$

Information gain:

$$Gain(S, A) = Entropy(S) - \sum_{v \in A}\frac{|S_v|}{|S|}Entropy($$

Entropy:

$$Entropy(S) = -\sum_{c} p(c)\log_2(p(c))$$

d. **Logistic Regression with CNN:**

Data Preparation: The input features (X) and matching ta rget labels (Y) are used to prepare the dataset.

Initializes the weights (W) and bias (b) parameters of the model either randomly or with predetermined values.

Calculate the linear combination of the input weights and features, as well as the bias term:

$$z = \sum_{i=1}^{n} W_i \cdot X_i + b$$

Defining a proper loss function will allow you to calculat e the discrepancy between the anticipated probability and the actual labels. The binary cross-entropy loss function is frequently employed in logistic regression:

$$L = -(y \cdot \log(a) + (1 - y) \cdot \log(1 - a))$$

Logistic Regression learns to categorize input data into b inary classes based on the anticipated probabilities by ite ratively modifying the weights and bias using gradient de scent. It seeks to identify the best decision boundary in th e feature space that divides the two classes

e. **SVM with CNN:**

- Input Data: A collection of photos are represented a s a matrix of pixel values in the input data.
- Feature Extraction from CNN:
  - A convolutional layer applies convolutional filters (kernels) to the input images in order to extract local information.
  - b. Activation Function: To create non-linearity, an a ctivation function, like ReLU, is applied element-by -element.
  - c. Pooling Layer: To minimize the spatial dimensio ns while preserving significant features, max poolin g or average pooling is utilized.
  - d. Flattening: A vector is created by flattening the c ombined feature maps.
  - SVM Classification: a. Training Phase:

i. Training Data is made up of the flattened feature vector s and the class labels that go with them.

ii. Kernel Function: To translate the feature vectors into a higher-dimensional space, an appropriate kernel functio n, such as the radial basis function (RBF) kernel, is select ed.

Support Vector Machine (SVM): This model is trained to locate the best hyper plane that maximally separates the feature vectors according to their class labels.

Testing Phase (b)

i. Test Data: Fresh feature vectors that have been flattene d from unused photos serve as the test data.

ii. Classification: The test data are divided into the appro priate classes using the trained SVM model.

Evaluation: The accuracy, precision, recall, and F1 score are some of the measures used to assess the hybrid model 's performance.

The powerful feature extraction capabilities of CNNs and the discriminative classification capabilities of SVMs ar e combined in the SVM and CNN Hybrid method. High-level characteristics are extracted from the images by the CNN and sent into the SVM classifier for precise predicti on.

f. **KNN with CNN :**

1. Layer of Input: A collection of photographs are input as a matrix of pixel values.

2. Extraction of CNN Features: To extract pertinent information from the input photos, CNN is used. These include the subsequent actions:

a. Convolutional Layer: Using the convolution operation, convolutional filters (kernels) are applied to the input image to extract local features.

b. Activation Function: To create non-linearity, an activation function, like ReLU, is applied element-by-element.

c. Pooling Layer: Max pooling or average pooling is used to shrink the feature maps' spatial dimensions while preserving crucial details.

d. Fully Connected Layer: To the vectored flattened version of the pooled feature maps.

3. Feature Vector Extraction: A feature vector is produced for each image using the output of the CNN's fully connected layer.

KNN, or K-Nearest Neighbours Classification: The training dataset's feature vectors are saved. As part of classification, the following activities are carried out:

Calculate the distance between the feature vectors of the test picture and the feature vectors in the training dataset using the Euclidean distance or another distance metric.

$$d(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

b. Choose Neighbours: The K neighbours with the shortest distances between them are chosen.

c. Majority Voting: The test image's class label is chosen by a majority vote using the class labels of the K nearest neighbours.

The CNN for feature extraction and the KNN for classification work together to take advantage of the CNN's capacity for meaningful feature extraction from pictures and the KNN's capacity for decision-making based on the similarity of feature vectors.

Table 1: Comparison of Accuracy of Each Hybrid Algorithm

Algorithm	Hybrid Algorithm Accuracy
NAIVE BAYES + CNN	64.21563
Random Forest + CNN	83.38141
Decision Tree + CNN	78.10972
Logistic Regression + CNN	79.2274
SVM + CNN	80.26558
KNN + CNN	81.68068

3. HYBRID ALGORITHM CNN + LSTM

LSTM Algorithm:

Sequential data processing is a specialty of recurrent neural networks (RNNs) of the LSTM type [18]. As a result of its ability to capture long-term dependencies, it is commonly utilised for applications such as time series analysis, speech recognition, and natural language processing.

The LSTM network accepts the input sequence for processing. Each sequence member is represented by a vector.

What information from the prior cell state should be erased is decided by the forget gate. Its inputs are the current input and the prior concealed state, and its output falls between [0, 1]. The forget gate's equation is as follows:

$$f_t = \sigma(W_f * [h_{t-1}, x_t] + b_f)$$

Where, h_t-1 is the prior hidden state, x_t is the current input, W_f and b_f are the forget gate's weights and biases, and f_t is the forget gate's output.

What fresh information should be kept in buffer that to be decided by input function gate. It also accepts the current input as well as the prior concealed state as inputs. The input gate's equation is as follows:

$$i_t = \sigma(W_i * [h_{t-1}, x_t] + b_i)$$

Where, W_i and b_i are the input gate's weights and biases, and i_t is the output of the input gate.

Cell State Update: The prior cell state is combined with the new input data to update the cell state. The cell state update equation is as follows:

$$C_t = f_t * C_{t-1} + i_t * \tanh(W_C * [h_{t-1}, x_t] + b_C)$$

where W_C and b_C are the biases and weights for updating the cell state, and C_t represents the updated cell state.

Output Gate: Using the updated cell state, the output gate chooses the LSTM cell's output. It accepts the current input as well as the prior hidden state as inputs. The output gate's equation is:

$$o_t = \sigma(W_o * [h_{t-1}, x_t] + b_o)$$

whereW_o and b_o are the output gate's weights and biases, and o_t is the output gate output.

Applying the output gate to the updated cell state results in the computation of the hidden state. The hidden state's equation is as follows:

$$h_t = o_t * \tanh(C_t)$$

The vanishing gradient problem can be reduced and long-term dependencies in sequential data can be captured using LSTM networks. LSTMs can retain significant information for extended periods of time by updating and preserving the cell state, which makes them useful for tasks like sequence categorization and machine translation.

LSTM with CNN Hybrid Method:

- Data sequences, including text or time series data, make up the input layer.
- Convolutional filters are used to extract local features from the input sequence in the CNN feature extraction process. The CNN algorithm's convolution process is comparable to that of that algorithm.
- Activation Function: To induce non-linearity, an activation function, such as ReLU, is applied element-by-element.
- Pooling Layer: To minimise the spatial dimensions of the feature maps, maximum or average pooling is applied.
- To capture temporal dependencies, the pooled feature maps are moulded into a sequence form.
- To represent the temporal dependencies and record long-term dependencies in the sequence data, the LSTM layer is introduced.

$$i_t = \sigma(W_i * [h_{t-1}, x_t] + b_i)$$

How much of the prior cell state should be forgotten is decided by the forget gate. The sigmoid activation function is used in its computation:

$$f_t = \sigma(W_f * [h_{t-1}, x_t] + b_f)$$

Update Gate: The cell state's new values are decided upon by the update gate. The tanh activation function is employed in its computation:

$$u_t = \tanh(W_u * [h_{t-1}, x_t] + b_u)$$

Output Gate: The LSTM cell's output is controlled by the output gate. The sigmoid activation function is used in its computation

$$o_t = \sigma(W_o * [h_{t-1}, x_t] + b_o)$$

Table 2: Comparison of Accuracy of Each Hybrid Algorithm

Algorithm	Hybrid Algorithm Accuracy
LSTM with CNN	64.21563

4. HYBRID CNN + ENSEMBLE CLASSIFIER

Utilising a CNN as a feature extractor and sending the extracted features into an ensemble classifier for final prediction is how a CNN and an ensemble classifier are combined. Here is a brief explanation of the procedure:

Extraction of Features Using CNN: To develop hierarchical representations of the input data, the CNN is trained using a sizable labelled dataset. After several convolutional and pooling layers, fully connected layers are present. The CNN takes the input data, such photos, and extracts the high-level features.

Create an Ensemble Classifier: To increase performance and resilience, ensemble classifiers are made by integrating a number of basis classifiers. Bagging, Boosting, and Stacking are examples of common ensemble procedures. Depending on the particular requirements of the situation, an ensemble classifier may be selected.

Extract CNN Features: From the input data, the trained CNN is utilised to extract features. These characteristics capture the significant representations and patterns that the CNN learnt during training. A vector or a matrix of extracted features is the CNN's output.

Feed Features into Ensemble Classifier: The ensemble classifier then receives the extracted CNN features as input. A mixture of multiple classifiers, such as Decision Trees, Random Forests, Support Vector Machines (SVM), or Neural Networks, can be used to create an ensemble classifier.

Ensemble Prediction: Based on the extracted CNN features, each base classifier in the ensemble classifier provides predictions. The predictions from all the base classifiers are combined to get the final prediction.

Algorithm 1: Hybrid CNN and Ensemble Classifier

$\boldsymbol{Step\ 1}$: $Feature\ Extraction\ Using\ CNN\ algorithms$

$\boldsymbol{Step\ 2}$: $Generate\ Hybrid\ Ensemble\ Model$

$KNeighborsClassifier(n\_neighbors\ =\ 5, metric\ =\ 'minkowski', p\ =\ 2),$

$RandomForestClassifier(criterion = 'entropy', n\_estimators = 50),$

$RandomForestClassifier(criterion = 'gini', n\_estimators = 50),$

$xgboost.XGBClassifier(n\_estimators =\ 50),$

$xgboost.XGBClassifier(n\_estimators =\ 100),$

$\boldsymbol{Step\ 3}$: $Pass\ CNN\ Feature\ to\ Hybrid\ Ensemble\ Classifier\ and\ Perform\ Training$

$\boldsymbol{Step\ 4}$: $Test\ Model\ using\ XGB\ Classifier$

$\boldsymbol{Step\ 5}$: $Performance\ Analysis\ Claculation$

- Feature Extraction: From the input data, pertinent features are extracted using the CNN algorithm.
- Hybrid Ensemble Model: Using classifiers like KNN, Random Forest, and XGBoost, a hybrid ensemble model is developed.
- Training: The hybrid ensemble classifier receives the CNN features for training.
- Testing: The XGBoost classifier is used to test the trained model on a different test dataset.
- Performance Analysis: To assess the model's performance, performance measures like accuracy, precision, recall, F1-score, and confusion matrix are computed.

6. Result and Discussion

The effectiveness of our suggested strategy for classifying sleep disorders is assessed in this section. We offer a Hybrid Deep Learning with Ensemble Machine Learning Classifier for performance assessment and comparison between our approach and the recommended machine learning approaches, along with pertinent clinical data. We also give thorough insights into the planning involved in putting our technology into use on embedded hardware devices. The practical issues and viability of incorporating our technique into practical applications are clarified by this material.

1. CNN CLASSIFIER:

The table below provides the CNN classifier's performance rating for identifying sleep disorders.

Precision, recall, and F1-score for class 1 are 0.83, 0.94, and 0.88 respectively. This suggests that the classifier is efficient at properly recognizing instances of class 1 because it has a relatively high precision and recall for this class. The number of instances that belong to this class is indicated by the support. Precision, recall, and F1-score for class 2 are 0.71, 0.15, and 0.25 respectively. The classifier accurately recognizes some instances of class 2 according to the reasonable precision, but the poor recall suggests that it has difficulty capturing all occurrences of this class. There may be room for improvement in this area.

```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv1d_1 (Conv1D)            (None, 3000, 10)          330
_________________________________________________________________
dropout_3 (Dropout)          (None, 3000, 10)          0
_________________________________________________________________
max_pooling1d_1 (MaxPooling1 (None, 23, 10)            0
_________________________________________________________________
dropout_4 (Dropout)          (None, 23, 10)            0
_________________________________________________________________
flatten_1 (Flatten)          (None, 230)               0
_________________________________________________________________
dense_2 (Dense)              (None, 128)               29568
_________________________________________________________________
dropout_5 (Dropout)          (None, 128)               0
_________________________________________________________________
dense_3 (Dense)              (None, 5)                 645
=================================================================
Total params: 30,543
Trainable params: 30,543
Non-trainable params: 0
_________________________________________________________________
```

Fig 6: CNN Classifier

The model can successfully learn from the training data, as shown by the training loss of 0.1603, which measures training performance.

$$\text{Accuracy (Acc)} = \frac{(\text{TP} + \text{TN})}{(\text{TP} + \text{FP} + \text{TN} + \text{FN})}$$

The training accuracy, as shown in figure 8, of 0.836 indicates that the model performs rather well on the training set.

Validation accuracy is 0.8174 and validation loss is 0.1735 as shown in figure 7. These results show that the model performs well both on the training set and on validation data that hasn't been seen before.

Table 3: CNN Classifier statistics

Sr. No.	Precision	Recall	F1-score	Support
1	0.83	0.94	0.88	174
2	0.71	0.15	0.25	99
3	0.88	0.94	0.91	459

4	0.94	0.79	0.86	134
5	0.72	0.88	0.79	193

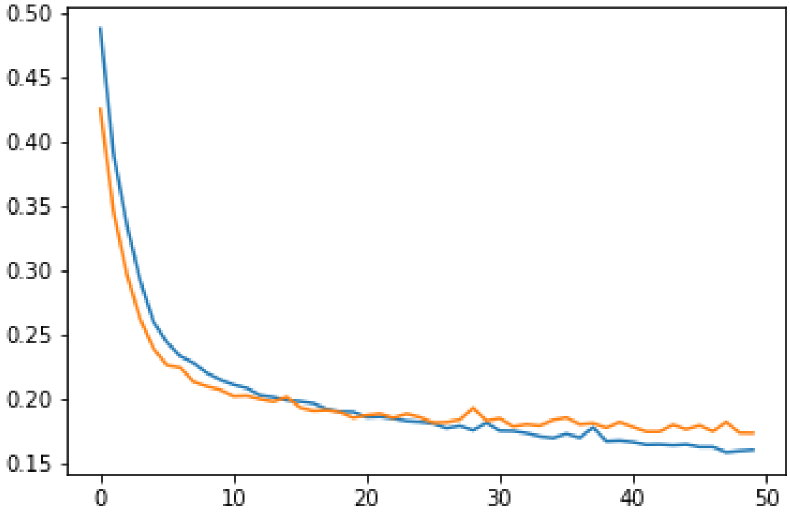

Fig 7: Training and Validation Loss Comparison Graph of CNN

Precision, recall, and F1-score for class 3 are 0.88, 0.94, and 0.91 respectively. With high precision and recall, the classifier appears to perform well in properly classifying instances of class 3.

Precision, recall, and F1-score for class 4 are 0.94, 0.79, and 0.86 respectively. The high precision suggests that e xamples of class 4 were correctly identified. The signific antly decreased recall, however, raises the possibility that the classifier may have missed some instances of class 4 altogether. Precision, recall, and F1-score for class 5 are all 0.72, 0.88, and 0.79 respectively. This demonstrates t hat the classifier for class 5 has a moderate precision and a comparatively high recall.

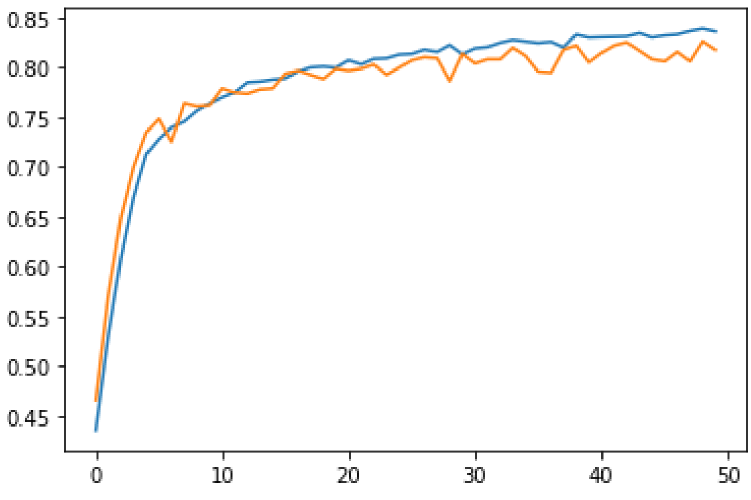

Fig 8: Training and Validation Accuracy Comparison Graph of CNN

Table 1 compares the classification accuracy of various h ybrid algorithms when used in conjunction with a CNN. Here is a breakdown of how well each algorithm perform ed:The accuracy of this hybrid algorithm, which combine s CNN and Naive Bayes, is 61.2%. For classification pro blems, it combines the Naive Bayes classifier with a CN N. The accuracy is the proportion of cases that were succ essfully classified as shown in figure 9.

The accuracy of this hybrid algorithm, which combines Random Forest with CNN, is 78.5%. It combines a CNN and the Random Forest classifier. For classification, the Random Forest algorithm constructs a group of decision trees and combines their predictions shown in figure 9.

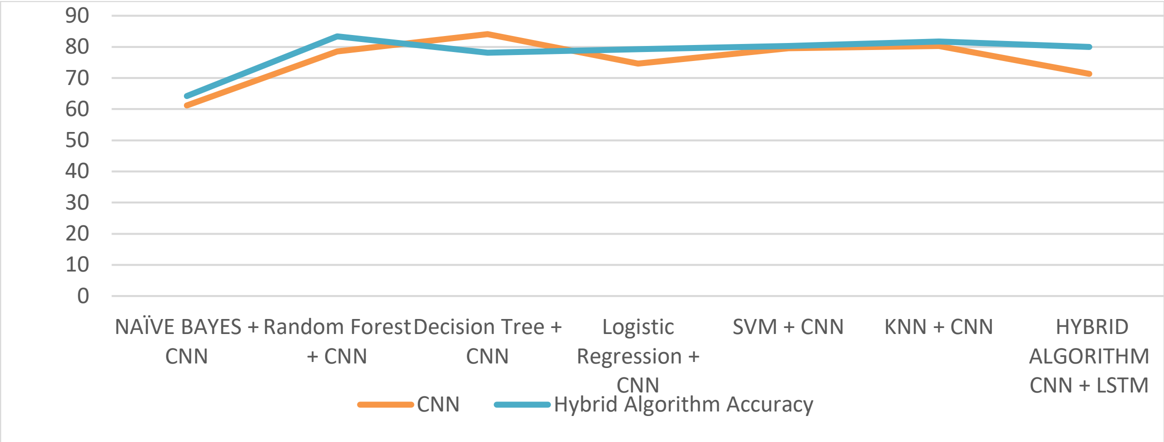

Fig 9: Analysis of Hybrid Approach Algorithm Accuracy

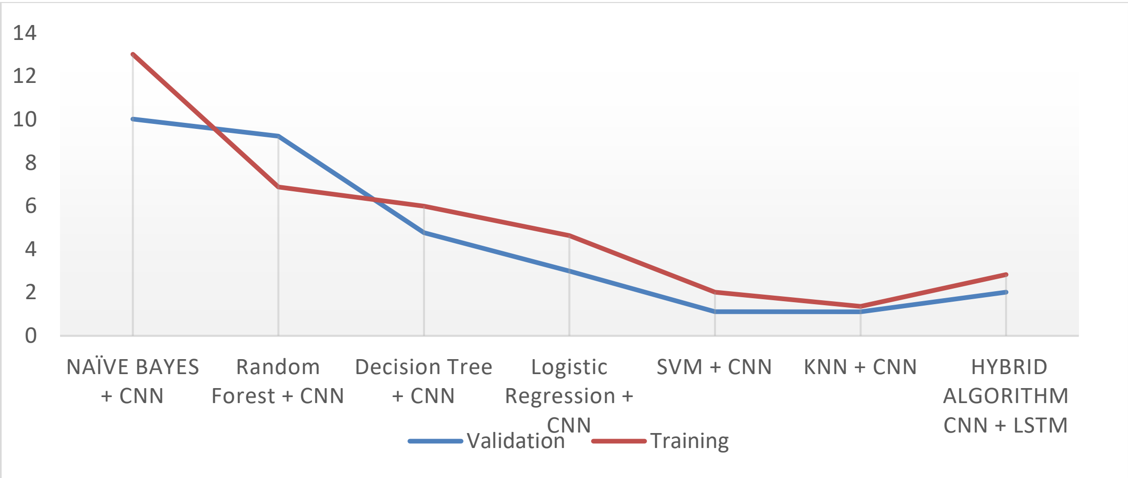

Fig 10: Training and Validation Loss Comparison Graph

When paired with a CNN, the hybrid algorithms exhibit varied degrees of accuracy. The method that performs cla ssification tasks with the best accuracy, 84.1%, is Decisi on Tree + CNN. However, with accuracy rates of 78.5% and 80.32%, respectively, the Random Forest + CNN an d KNN + CNN algorithms also perform admirably. It's cr ucial to remember that the choice of algorithm depends o n the particular goal and dataset, and that additional resea rch and testing may be necessary to identify the best hybr id algorithm for a certain situation.

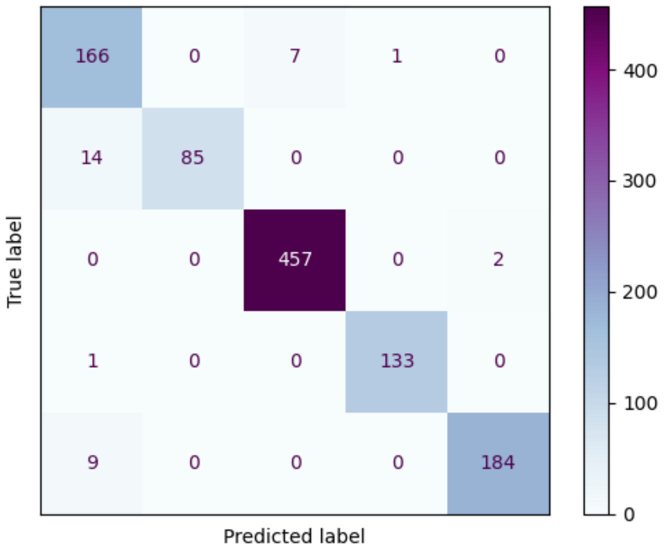

Fig 11: Confusion Matrix representation

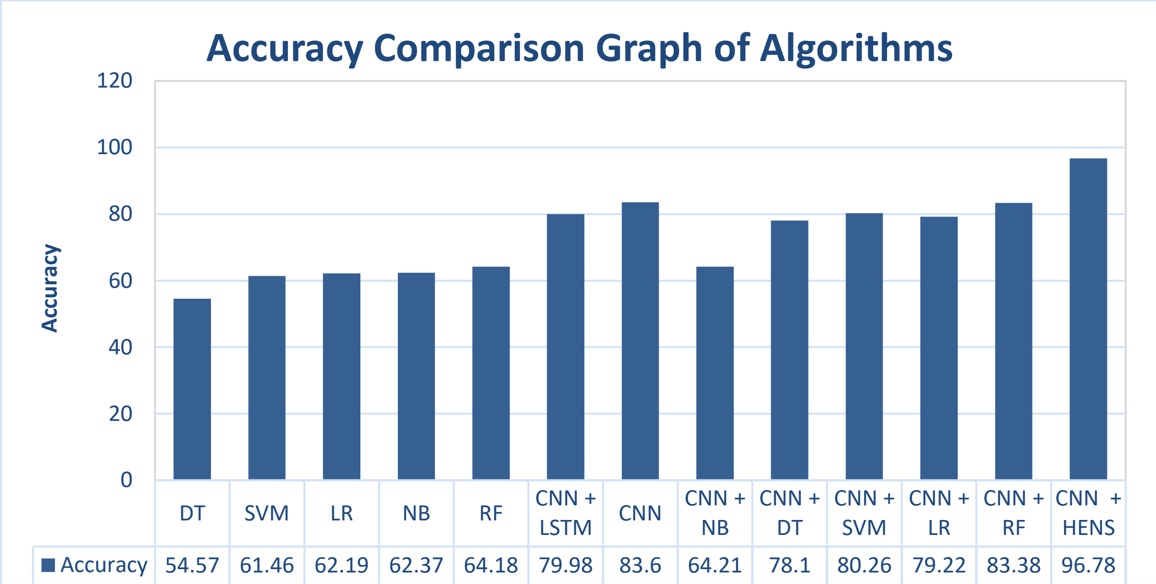

Fig 12: Accuracy Comparison Graph Of Algorithms

The CNN + HEML hybrid model has the best accuracy (96.78%) of all the algorithms. This shows that, for the classification job, CNN and HEML generate extremely accurate predictions. The hybrid model has a high percentage of true positive predictions relative to false positives, as evidenced by its precision of 97%. The figure 11 shown the confusion matrix for hybrid algorithm.

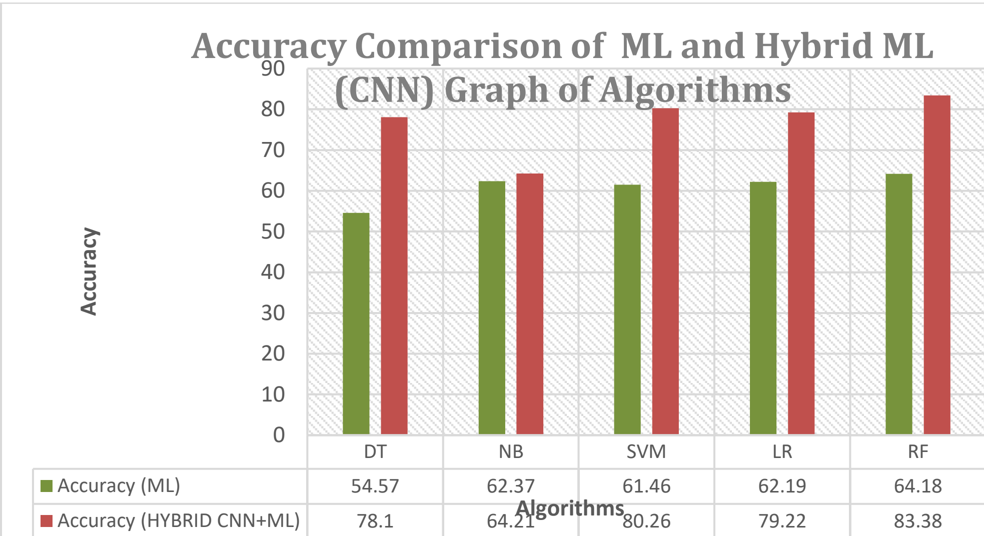

Fig 13: Accuracy Comparison GraphOf Traditional ML And Hybrid ML Algorithms

Table 4: Comparison Graph of Traditional ML and Hybrid CNN + Ensemble ML Algorithms

	Accuracy	**Precision**	**Recall**	**F1-Score**
DT	54.57	50	55	50
NB	62.37	72	62	63
SVM	61.46	61	61	59
KNN	62.19	60	62	60
RF	64.18	68	64	63
CNN + HEML	96.78	97	95	96

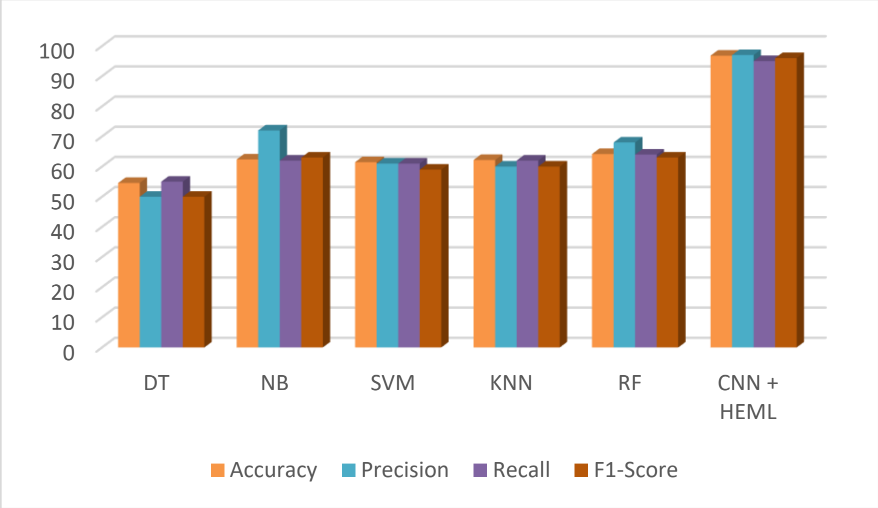

Fig 14: Graphical representation of Accuracy, Precision, Recall and F1-Score Comparison Graph of Traditional ML and Hybrid CNN + Ensemble ML Algorithms

The comparative accuracy of traditional machine learning and Hybrid machine learning as shown in figure 12, from this it clearly show that proposed module has the better accuracy than the other classifier in machine learning mechanism.

With a precision of 72%, Naive Bayes has the second-highest accuracy, coming in at 62.37%. In correctly identifying cases in the dataset, it performs reasonably well. However, the CNN + HEML hybrid model outperforms it.SVM and KNN both achieve accuracy rates of 61.46% and 62.19%, respectively. Both models have balanced precision, recall, and F1-scores, indicating that they can categorize examples relatively well but are not particularly superior to other models in this regard.Among the listed models, the Decision Tree algorithm's accuracy is the lowest (54.57%). Additionally, it has the lowest F1-score and precision, which indicates a greater proportion of false positives and a less accurate capacity to classify occurrences. The accuracy of the Random Forest algorithm is 64.18%, and it displays balanced precision, recall, and F1-score. Although it outperforms the Decision Tree method, the CNN + HEML hybrid model performs better as shown in figure 12 and figure 13.

7. Conclusion

The research article show that the suggested hybrid strategy, which combines deep learning methods with ensemble classifiers, outperformed individual classifiers in terms of accuracy and precision. This shows that the combination of these methods can efficiently capture the intricate patterns and sliding disorder-related aspects found in EEG signals.Our studies' findings showed that the hybrid model performed better than well-known machine learning techniques including Decision Trees, Naive Bayes, SVM, and KNN. The hybrid model's accuracy score of 96.78% demonstrates how well it can identify sliding diseases from EEG readings. The hybrid model's reliability and efficiency were further attested to by its high precision, recall, and F1-score values.The ability of deep learning to extract high-level features from EEG signals and the ensemble learning technique's capacity to combine several classifiers to boost performance are key factors in the hybrid approach's success. The ensemble classifier combined the predictions of various models, lowering the possibility of overfitting and enhancing generalisation, while the deep learning component, more especially the convolutional neural network (CNN), successfully learned spatial and temporal patterns in the EEG data.The results of our study have important medical ramifications since early intervention and better patient outcomes can result from correct detection of sliding diseases. The hybrid approach may help medical practitioners diagnose sliding diseases more quickly and accurately, which could lead to better management and therapy. There is scope, In order to evaluate the model's performance in various scenarios and patient populations, this may entail testing the model on larger and more varied datasets, including real-world clinical data. Further research into the hybrid model's interpretability is also necessary to understand the precise EEG characteristics that contribute to the classification of sliding diseases.

References:

[1] Estrada, E.; Nazeran, H.; Nava, P.; Behbehani, K.; Burk, J.; Lucas, E. Itakura distance: A useful similarity measure between EEG and EOG signals in computer-aided classification of sleep stages. In Proceedings of the 27th IEEE Annual International Conference of Engineering in Medicine and Biology Society, Shanghai, China, 1–4 September 2005; pp. 1189–1192.

[2] Li, Y.; Yingle, F.; Gu, L.; Qinye, T. Sleep stage classification based on EEG Hilbert–Huang transform. In Proceedings of the 4th IEEE Conference on Industrial Electronics and Applications (ICIEA), Xi'an, China, 25–27 May 2009; pp. 3676–3681.

[3] Aboalayon, K.A.; Faezipour, M. Multi-class SVM based on sleep stage identification using EEG signal. In Proceedings of the IEEE Healthcare Innovation Conference (HIC), Seattle, WA, USA, 8–10 October 2014; pp. 181–184.

[4] Huang, C.-S.; Lin, C.-L.; Ko, L.-W.; Liu, S.-Y.; Sua, T.-P.; Lin, C.-T. A hierarchical classification system for sleep stage scoring via forehead EEG signals. In Proceedings of the IEEE Symposium on Computational Intelligence, Cognitive Algorithms, Mind, and Brain (CCMB), Singapore, 16–19 April 2013; pp. 1–5.

[5] Huang, C.-S.; Lin, C.-L.; Yang, W.-Y.; Ko, L.-W.; Liu, S.-Y.; Lin, C.-T. Applying the fuzzy c-means based dimension reduction to improve the sleep classification system. In Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ), Hyderabad, India, 7–10 July 2013; pp. 1–5.

[6] Kushida, C.A.; Kushida, C.A; Littner, M.R.; Morgenthaler, T.; Alessi, C.A.; Bailey, D.; Coleman, J., Jr.; Friedman, L.; Hirshkowitz, M.; Kapen, S.; et al. Practice parameters for the indications for polysomnography and related procedures: An update 2005. Sleep 2005, 28, 499–521.

[7] Schumann, A.Y.; Bartsch, R.P.; Penzel, T.; Ivanov, P. C.; Kantelhardt, J.W. Aging effects on cardiac and respiratory dynamic in healthy subjects across sleep stages. Sleep 2010, 33, 943–955.

[8] Kantelhardt, J.W.; Havlin, S.; Ivanov, P.C. Modeling transient correlations in heartbeat dynamics during sleep. Europhys. Lett. 2003, 62, 147–153.

[9] Penzel, T.; Kantelhardt, J.W.; Bartsch, R.P.; Riedl, M.; Kraemer, J.F.; Wessel, N.; Garcia, C.; Glos, M; Fietze, I.; Schöbel1, C. Modulations of heart rate, ECG, and cardio-respiratory coupling observed in polysomnography. Front. Physiol. 2016, 7, 460

[10] Ebrahimi, F.; Setarehdan, S.K.; Ayala-Moyeda, J.; Nazeran, H. Automatic sleep staging using empirical mode decomposition, discrete wavelet transform, time-domain, and non linear dynamics features of heart rate variability signals. Comput. Methods Programs Biomed. 2013, 112, 47–57.

[11] Lin, C.C.; Chang, H.Y.; Huang, Y.H.; Yeh C.Y. A novel wavelet-based algorithm for detection of QRS complex. Appl. Sci. 2019, 9, 1–19. [CrossRef] 13. Versace, F.; Mozzato, M.; Tona, G.D.M.; Cavallero, C.; Stegagno, L. Heart rate variability during sleep as a function of the sleep cycle. Biol. Psychol. 2003, 63, 149–162.

[12] Bonnet, M.H.; Arand, D.L. Heart rate variability: Sleep stage, tome of night, and arousal influences. Electroencephalogr. Clin. Neurophysiol. 1997, 102, 390–396

[13] Long, X.; Fonseca, P.; Haakma, R.; Aarts, R.M.; Foussier, J. Spectral boundary adaption on heart rate variability for sleep and wake classification. Int. J. Artif. Intell. Tools 2014, 23, 1460002-1–1460002-20.

[14] Aboalayon, K.A.I.; Faezipour, M.; Almuhammadi, W.S.; Moslehpour, S. Sleep stage classification using EEG signal analysis: A comprehensive survey and new investigation. Entropy 2016, 18, 1–31.

[15] Werteni, H.; Yacoub, S.; Ellouze, N. An automatic sleep-wake classifier using ECG signals. IJCSI Int. J. Comput. Sci. Issues 2014, 11, 84–93.

[16] 18. Khemiri, S.; Alouri, K.; Nacaeur, M.S. Automatic detection of slow wave sleep and REM-sleep stages using polysomnographic ECG signals. In Proceedings of the 8th International Multi-Conference on Systems, Signals and Devices, Sousse, Tunisia, 22–25 March 2011; pp. 1–4.

[17] 19. Singh, J.; Sharma, R.K.; Gupta, A.K. A method of REM-NREM sleeps distinction using ECG signal for unobtrusive personal monitoring. Comput. Biol. Med. 2016, 78, 138–143.

[18] Widasari, E. R.; Tanno, K.; Tamura, H. Automatic sleep quality assessment for obstructive sleep apnea patients based on HRV spectrum analysis. In Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (SMC), Bari, Italy, 6–9 October 2019.

[19] Mosquera-Lopez, C.; Leitschuh, J.; Condon, J.; Hagen, C.C.; Rajhbeharrysingh, U.; Hanks,C.; Jacobs, P.G. Design and evaluation of a non-contact bed-mounted sensing device for automated in-home detection of obstructive sleep apnea: A pilot study. Biosensors 2019, 9, 90–100

[20] Espiritu, H.; Metsis, V. Automated detection of sleep disorder-related events from polysomnographic data. In Proceedings of the International Conference on Healthcare Informatics, Dallas, TX, USA, 21–23 October 2015; pp. 562–569.

[21] David, L.G.; Chaibi, S.; Ruby, P.; Aguera, P.E.; Eichenlaub, J.B.; Samet, M.; Kachouri, A.; Jerbi, K. Automatic detection of sleep disorders: Multi-class automatic classification algorithms based on Support Vector Machines. In Proceedings of the International Conference on Time Series and Forecasting, Granada, Sapin, 19–21 September 2018; pp. 1270–1280.

[22] Dietterich, T.G. Ensemble Methods in Machine Learning. In Proceedings of the International Workshop on Multiple Classifier Systems, London, UK, 21–23 June 2000; pp. 1–5

[23] Mousavi, R.; Eftekhari, M. A new ensemble learning methodology based on hybridization of classifier ensemble selection approaches. Appl. Soft Comput. J. 2015, 37, 652–666.

[24] Mishra, P.K.; Yadav, A.; Pazoki, M. A Novel Fault Classification Scheme for Series Capacitor Compensated Transmission Line Based on Bagged Tree Ensemble Classifier. IEEE Access 2018, 6, 27373–27382.

[25] Boudreau, P.; Yeh, W.H.; Dumont, G.A.; Boivin, D.B. Circadian variation of heart rate variability across sleep stages. Sleep 2013, 36, 1919–1928.

[26] McCarter, S.J.; St Louis, E.K.; Boeve, B.F. REM Sleep Behavior Disorder and REM Sleep Without Atonia as an Early Manifestation of Degenerative Neurological Disease. Curr. Neurol. Neurosci. Rep. 2012, 12, 182–192.

[27] Sateia, M.J.; Buysse, D.J.; Krystal, A.D.; Neubauer, D.N.; Heald, J.L. Clinical practice guideline for the pharmacologic treatment of chronic insomnia in adults: Am american academy of sleep medicine clinical practice guideline. J. Clin. Sleep Med. 2017, 13, 307–349.

[28] Schutte-Rodin, S.; Broch, L.; Buysse D.; Dorsey, C.; Sateia, M. Clinical guideline for the evaluation and management of chronic insomnia in adults. J. Clin. Sleep Med. 2008, 5, 487–504.

[29] Hertenstein, E.; Gabryelska, A.; Spiegelhalder, K.; Nissen, C.; Johann, A.F.; Umarova, R.; Riemann, D.; Baglioni, C.; Feige, B. References data for polysomnography-measured and subjective sleep in healthy adults. J. Clin. Sleep Med. 2018, 14, 523–532.

[30] Sabater, L.; Gaig, C.; Gelpi, E.; Bataller, L.; Lewerenz, J.; Torres-Vega, E.; Contreras, A.; Giometto, B.; Compta, Y.; Embid, C.; et al. A novel NREM and REM parasomnia with sleep breathing disorder associated with antibodies against IgLON5: A case series, pathological features, and characterization of the antigen. Lancet Neurol. 2014, 13, 575–586.

[31] Lee, G. L.; Choi, J.W.; Lee, Y.J.; Jeong, D.U. Depressed REM sleep behavior disorder patients are less likely to recall enacted dreams than non-depressed ones. Psychiatry Investig. 2016, 13, 227–231

[32] Shivastava, D.; Jung, S.; Saadat, M.; Sirohi, R.; Crewson, K. How to interpret the results of a sleep study. J. Community Hosp. Intern. Med. Perspect. 2014, 4, 1–4.

[33] Prof. Barry Wiling. (2018). Identification of Mouth Cancer laceration Using Machine Learning Approach. International Journal of New Practices in Management and Engineering, 7(03), 01 - 07. https://doi.org/10.17762/ijnpme.v7i03.66

[34] López, M., Popović, N., Dimitrov, D., Botha, D., & Ben-David, Y. Efficient Dimensionality Reduction Techniques for High-Dimensional Data. Kuwait Journal of Machine Learning, 1(4). Retrieved from http://kuwaitjournals.com/index.php/kjml/article/view/145

[35] Janani, S., Dilip, R., Talukdar, S. B., Talukdar, V. B., Mishra, K. N., & Dhabliya, D. (2023). IoT and machine learning in smart city healthcare systems. Handbook of research on data-driven mathematical modeling in smart cities (pp. 262-279) doi:10.4018/978-1-6684-6408-3.ch014 Retrieved from www.scopus.com