Biomedical Signal Processing and Control 83 (2023) 104730

Contents lists available at ScienceDirect

Biomedical Signal Processing and Control

journal homepage: www.elsevier.com/locate/bspc

An attention-based multi-resolution deep learning model for automatic A-phase detection of cyclic alternating pattern in sleep using single-channel EEG

Barproda Halder *,1, Tanvir Anjum *,1, Mohammed Imamul Hassan Bhuiyan *

Department of Electrical and Electronic Engineering, Bangladesh University of Engineering and Technology, Dhaka, 1205, Bangladesh

ARTICLE INFO

Dataset link: https://physionet.org/content/capslpdb/1.0.0/

Keywords:
Cyclic alternating pattern
Sleep EEG
Deep learning
Attention model

ABSTRACT

Sleep is a crucial part of human well-being. Many people suffer from various sleep disorders and insufficient sleep. The detection of the cyclic alternating pattern (CAP) of electroencephalogram (EEG) activity during sleep is essential for identifying and monitoring these problems. In this paper, we present a multi-resolution deep neural network model with temporal and channel attention for detecting A-phase and its subtypes. A multi-branch one-dimensional convolutional neural network (1D-CNN) is employed where each branch has different kernel sizes to extract features of different frequency resolutions automatically. An attention-based transformer network exploits the dynamic and temporal relationship between CAP event features extracted from the single-channel EEG data. Our model achieves 90.31% accuracy, 95.30% specificity, and 65.73% F1-Score in A-phase detection and 86.72% accuracy, 89.53% specificity, and 59.59% F1-Score in the detection of its subtypes, superior performance as compared to those of the recent approaches.

1. Introduction

Sleep is one of the basic needs of human beings. It is essential for good mental and physical health. Sleep deficiency may cause frustration, emotional instability, trouble in learning, focusing, decision-making, and social functioning [1]. Moreover, various sleep disorders are linked to chronic health problems. For instance, obstructive sleep apnea (OSA) is associated with increased stroke risk [2], and insomnia is linked to the development of diabetes and increased risk of cardiovascular diseases [3]. Therefore, to live a healthy life, it is necessary to assess sleep quality and diagnose sleep problems.

To better understand human sleep, Rechtschaffen and Kales (R&K) [4] divided sleep into seven stages: wakefulness, stage-1 (S1), stage-2 (S2), stage-3 (S3), stage-4 (S4), rapid eye movement (REM), and movement time. S1–S4 constitute the non-rapid eye movement (NREM) part of sleep. Later, the American Academy of Sleep Medicine (AASM) developed a new guideline [5] for sleep staging. According to AASM, there are 5 stages in sleep: Wakefulness (W), REM, N1, N2, and N3. Here, N3 is the slow wave sleep consisting of S3 and S4 stages of R&K rule [6]. According to the AASM classification, NREM sleep is comprised of N1–N3 stages. These stages can be identified in an electroencephalogram (EEG). However, sleep staging only depicts the macro-structure of sleep, leaving out the micro-structure, which only lasts a few seconds during slow wave sleep (SWS). This micro-structure includes cyclic alternating pattern (CAP), which is a periodic EEG activity of non-REM sleep in which A-phase and B-phase can range between 2 to 60 s [7]. A-phase followed by periods of deactivation (B-phase) comprises a CAP cycle. At least, two consecutive CAP cycles form a CAP sequence. A-phase can be divided into three subtypes. These subtypes are described as follows:

- Subtype A1: Low-frequency waveforms with large amplitudes make up this subtype. Its amplitude is larger than the background activities (B-phase). The waveforms are mainly bursts and K-complex sequences [8]. In Fig. 1, the red shaded part represents this subtype.
- Subtype A2: This subtype is a mixture of fast and slow rhythms. The waveforms are mainly polyphasic bursts [8]. In Fig. 1, the yellow shaded part indicates this subtype.
- Subtype A3: This subtype consists of voltages with a high frequency but a low amplitude. Its frequency is higher than the background activity. The waveforms include K-alpha, EEG arousals, and polyphasic bursts [8]. In Fig. 1, the green shaded part represents this subtype.

* Corresponding authors.
E-mail addresses: barprodahalder@gmail.com (B. Halder), tanvir1167052@gmail.com (T. Anjum), imamul@eee.buet.ac.bd (M.I.H. Bhuiyan).

[1] Shared first authorship.

https://doi.org/10.1016/j.bspc.2023.104730
Received 30 June 2022; Received in revised form 3 February 2023; Accepted 14 February 2023
Available online 22 February 2023

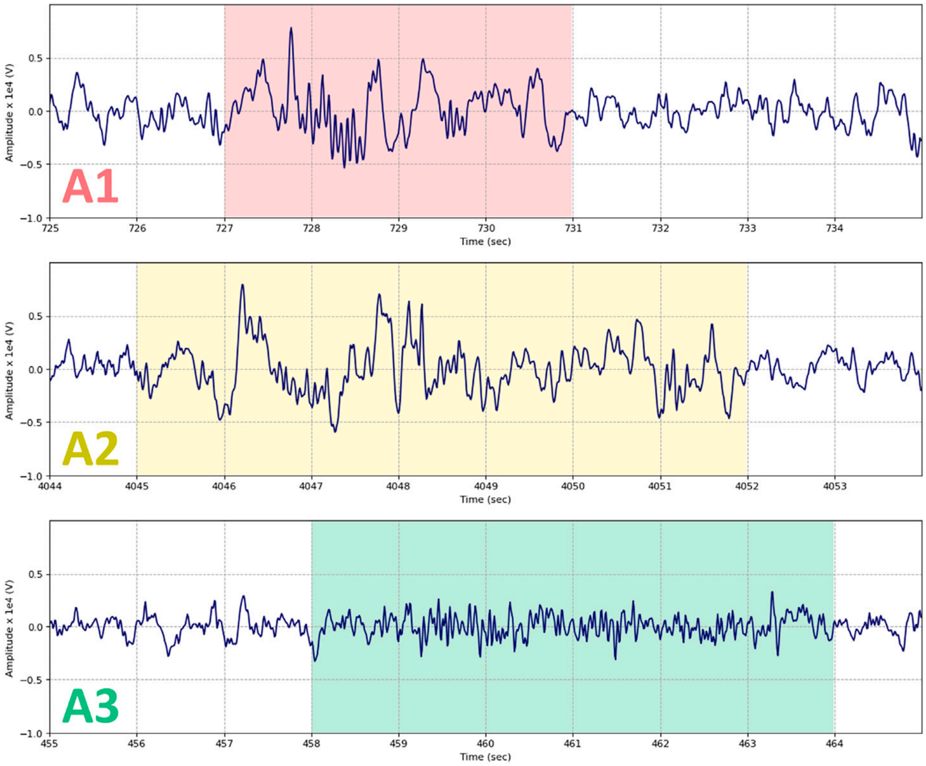

Fig. 1. A single EEG channel with A1, A2 and A3 subtypes of A-phase.

The cyclic alternating pattern can indicate sleep instability, disruption, or both. It can give better clues regarding sleep disorders [9]. Study shows that the severity of obstructive sleep apnea (OSA) has an inverse relationship with subtype A1 and a direct correlation with subtype A3 [10]. OSA patients with daytime sleepiness (ES) have more CAP measures than those without ES [11]. The study also shows that in narcoleptic patients, CAP time, CAP rate, number of CAP cycles, and number of A-phase subtypes (in particular subtype A1) are significantly reduced [12] and patients with primary insomnia have increased CAP rate [13]. The subtype A3 of CAP is more observed in patients with vanishing periodic limb movements in sleep (PLMS) than those with newly emergent PLMS. The newly emergent PLMS are more related to the A1-subtype than the A3-subtype [14]. The presence of cyclic alternating pattern (CAP) can also be an indicator of coma patients' survival, and recovery [15]. As a result, detecting and classifying different phases of the cyclic alternating pattern in sleep is required. Human experts have traditionally performed CAP phase detection, which is time-consuming and exhausting. It may also be error-prone due to its subjective nature. For clinical and scientific reasons, an automatic and reliable A-phase detector is needed.

We present for the first time, an attention-based multi-resolution deep neural network architecture for the detection of A-phase and its subtypes. The major contributions can be summarized as follows:

- The proposed model introduces a multi-resolution feature extractor (MRFE) to the task of detection of A-phases in the cyclic alternating pattern in sleep EEG. To the best of our knowledge, using multi-resolution deep learning architectures as feature extractor is a novel technique in the field of A-phase and its subtypes detection.
- In human perception, the behavior of concentrating on some distinct portions of information while ignoring the rest is known as attention. The attention mechanism in deep learning tries to mimic this attention from human perception by automatically training the model to detect the relevant features. Our model includes two types of attention mechanisms: channel attention module (CAM) [16] and multi-head attention (MHA) [17]. CAM helps MRFE to extract better features and MHA exploits the temporal information between the extracted features. It is expected that the incorporation of the attention mechanism will further improve the detection of the A-phase and its subtypes.

2. Related works

Several methods have been presented to automate the detection of A-phase in EEG signals. These methods can be divided into two main approaches. Some researchers utilized hand-crafted features as input to a machine learning or deep learning classifier [8,18–24]. In other research, a deep learning network was directly applied to the raw EEG data [25,26]. Machado et al. [8] extracted 55 features from the EEG recordings. Then the classification was performed by support vector machine (SVM), k-nearest neighbors (k-NN), and discriminant analysis(DA). Hartmann et al. in their works [22,27] used five different features such as Hjorth activity, Shannon entropy, Teager energy operator, band power descriptor, and differential EEG variance with either linear discriminant analysis (LDA), k-NN, shallow neural network (NN), long short-term memory (LSTM) as a classifier in [27] and either of NN and LSTM in their later work [22]. Mendonça et al. chose features generated by a sequential feature selection algorithm and principal component analysis (PCA). And they used a feed-forward neural network to classify A-phase and B-phase [28]. Raw EEG signal was passed through an LSTM classifier in their subsequent work [25]. In another study by Mendonça et al. the Gaussian mixture model (GMM) and self-organizing map (SOM) were used to cluster the data to produce higher time resolution. Hidden Markov model (HMM) was then used to detect A-phase [29]. In each of their works, a one-second window was used. Arce-Santana et al. [23] used a spectrogram of 4s EEG segment which was treated as a two-dimensional (2D) input feature and the classification was done by a deep 2D convolutional neural network (2D-CNN). Dhok et al. used Wigner–Ville distribution (WVD) and Rényi entropy (RE) as features [24]. The medium Gaussian support vector

Table 1
Summary of the total numbers of the samples used for our work.

Subject	B-phase	A1	A2	A3	A-phase
n1	30311	2217	747	1135	4099
n2	29765	1188	688	1239	3115
n3	27701	656	631	1043	2330
n4	31795	986	356	893	2235
n5	26266	2863	328	784	3975
n6	27034	1871	976	1414	4261
n7	26879	1616	565	480	2661
n8	26750	949	465	1876	3290
n9	29790	1036	377	678	2091
n10	23053	1489	336	922	2747
n11	28487	1724	583	796	3103
n12	27910	1064	153	573	1790
n13	25448	1628	1040	1041	3709
n14	25713	1035	1234	1209	3478
n15	25671	1449	1046	1244	3739
n16	26489	2252	1138	891	4281
Total	439062	24023	10663	16218	50904

machine (SVM) algorithm was then applied for A-phase detection. In their other work [26], deep 1D-CNN was applied to raw EEG data for A-phase detection. A similar approach was followed by Murarka et al. [30]. They deployed a 1D-CNN method for A-phase detection where data from 75 subjects, both healthy and disordered, were used.

All the works stated above suffer from some drawbacks. Most of the feature-based methods are computationally complex to produce [22–24]. Using raw signal and LSTM classifier [25] requires high training and inference time and delivers relatively poor performance (69.7±5.9% accuracy). The 1D-CNN method implemented by Loh et al. [26] and Murarka et al. [30] is faster and computationally simpler but provide modest accuracy (52.99% and 60.59% accuracy for unbalanced test data, respectively).

Although deep neural networks have been employed for many sleep-related tasks such as automatic sleep staging [31–33] and the diagnosis of various sleep disorders [34], their application in A-phase and its subtypes detection is rather limited. Thus, there is ample scope for developing new deep neural network models for effective and improved detection of A-phase and its subtypes.

3. Materials and methodology

This Section consists of five subsections: database, pre-processing, model architecture, performance metrics, and experimental setup. The first subsection describes the data used for the detection task, and the following three subsections explain the overall workflow and the metrics used to evaluate the model. The last subsection includes the resources and the hyper-parameters used for the experiment.

3.1. Database

We use the Physionet's CAP sleep database, which is available to the public [35]. The database consists of 108 polysomnographic (PSG) recordings registered at the Sleep Disorders Center of the Ospedale Maggiore of Parma, Italy. 16 healthy subjects with no neurological problems or medicines affecting the central nervous system and 92 pathological subjects were included in 108 PSG recordings. The PSG recordings include at least 3 EEG channels, 2 EOG channels, 2 EMG channels, respiration signals (airflow, abdominal and thoracic effort, and SaO2), and EKG. The CAP events were labeled in agreement with Terzano's reference atlas [7]. In our work, we use a subset of the database of 16 healthy subjects (n1–n16) for the detection task. In Table 1, the summary of our used dataset is noted. The database contains 439,062 s of non-CAP (B-phase) and 50,904 s of A-phase events which is a highly unbalanced distribution.

3.2. Pre-processing

We used single-channel EEG data (either C4-A1 or C3-A2) from each subject which was resampled, filtered, and windowed. Channel C4-A1 or C3-A2 was selected for the experiment because they are the most commonly used channels in the related literature [8,22,24–26,28–30]. The EEG signals belonging to different subjects had different sampling frequencies. Therefore, all the EEG data were resampled at 100 Hz. Our dataset has no labeling for CAP events during REM and wake stages. A-phases that occur during REM or wakefulness are not a part of any CAP sequence by definition [7]. Hence, only the non-REM stage of the signal was kept. It has been reported that the 0.5 Hz to 30 Hz frequency range contains most of the relevant data for CAP analysis [7]. Thus, input signals were filtered using a bandpass filter with bandwidth from 0.5 Hz to 30 Hz. Then, the five EEG rhythms: Delta (0.5–4 Hz), Theta (4–8 Hz), Alpha (8–12 Hz), Sigma (12–16 Hz), and Beta (16–30 Hz) were extracted using bandpass filters of respective frequency ranges from each single-channel data and concatenated to create the multi-band data which was used as input to our model. This multi-band data was then divided into 30 s windows and each second of each window was labeled as either A1, A2, A3, or B-phase (NCAP) according to the annotation provided in the dataset. Annotation of each second in a 30 s window is illustrated in Fig. 2. Each instance of this multi-band data has a shape of (5, 3000) since five instances (delta, theta, alpha, sigma, and beta) of 30 s data sampled at 100 Hz are stacked together. For single-class detection, the data were annotated as A-phase or B-phase. Since A-phase can only be determined in relation to its background, the larger non-overlapping 30 s window was utilized for training the model to understand the context better.

3.3. Model architecture

In this Section, our proposed deep learning architecture is explained in detail.

3.3.1. Overview

Our model works in three stages: at first, a three-branch one-dimensional convolutional neural network (1D-CNN) extracts the most essential features. Then, a multi-head attention-based temporal feature decoder captures the relationship between the extracted features. Finally, a simple fully connected network performs as the final prediction layer. 30 s window of pre-processed single-channel EEG data is fed into the model, and the model produces 30 predictions corresponding to each second of the window at once. The following sections describe each component of the model in detail.

3.3.2. Multi-Resolution Feature Extractor (MRFE)

Conv1D: Deep convolutional neural networks have the ability to recognize complex patterns and features in extensive data with relatively low computational complexity. 2D convolutional networks have become the standard for various image-related operations [36]. But, for one-dimensional data, 2D-CNNs cannot be directly applied. Therefore, 2D features such as spectrograms [37] or scalograms [38] are extracted first, and then 2D-CNN is applied to the extracted 2D features. These models need high computation time and thus require powerful hardware to work in real-time situations, which is unsuitable for mobile and low-power devices. To counter this, 1D-CNN was proposed [39]. Here, we implemented a multi-branch 1D-CNN to extract the relevant features. The features that constitute an A-phase vary in a relatively wide frequency range. We used 3 branches of one-dimensional convolutional neural networks with different kernel sizes to extract low, medium, and high-frequency resolution features. Using different kernel sizes to collect multiple resolutions was previously employed by [40–42]. In our model, we selected the kernel sizes of the initial convolutional layers such that the kernels correspond to 0.5 s, 2 s, and 4 s of the data, respectively. The actual kernel sizes are dependent on the sampling size

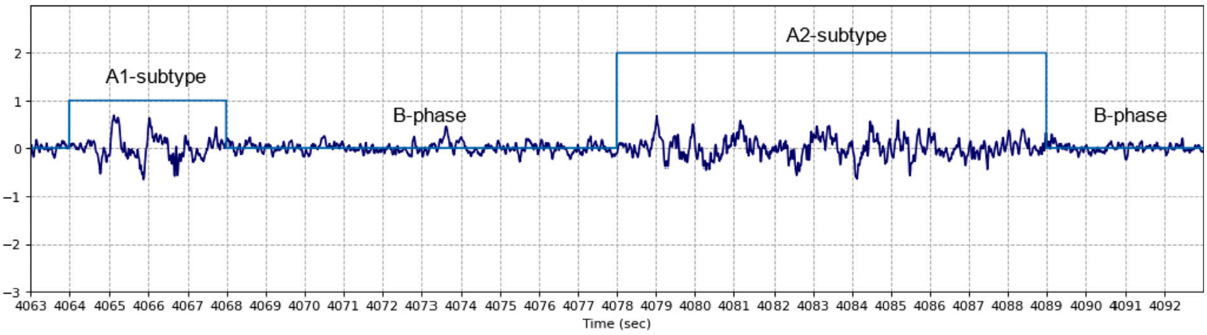

Fig. 2. 30 s epoch with labeling.

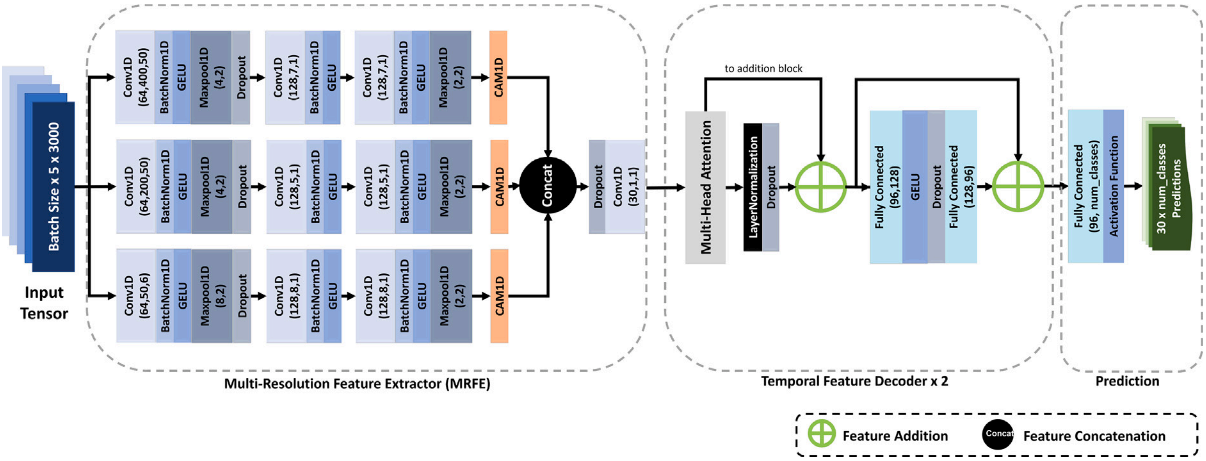

Fig. 3. Architecture of the proposed model.

of the signal. In our case, we used a sampling rate of 100 Hz. Therefore, the kernel sizes were selected to be 50, 200, and 400 for 0.5 s, 2 s, and 4 s windows. Subsequent convolutional layers are employed to extract features in various frequency ranges. The parameters in the Conv1D block, as shown in Fig. 3, represent the number of channels, kernel size, and stride, respectively. Therefore, Conv1D (64, 400, 50) represents a convolutional layer with an output channel size of 64, kernel size of 400, and stride of 50.

BatchNormalization: While training a model, the distribution of the input tensor changes with each iteration and after each convolution operation. The network training converges faster if, for each iteration and each convolutional layer, the input distribution remains the same [43]. Therefore, a batch normalization [44] layer was used after each convolutional layer. For each tensor with d channels, we normalize each channel as:

$$\hat{x}^{(k)} = \frac{x^{(k)} - E[x^{(k)}]}{\sqrt{Var[x^{(k)}] + \epsilon}}, k \in (1, d) \tag{1}$$

where $x^{(k)}$ and $\hat{x}^{(k)}$ denote *k-th* channel data for the whole batch before and after batch-normalization, respectively.

Gaussian Error Linear Unit (GELU): To introduce non-linearity in our network, Gaussian Error Linear Unit (GELU) was utilized as an activation function. For a Gaussian distribution CDF with zero mean and unit variance P(x), the GELU function is defined as $GELU(x) = xP(x)$ [45], which is a deterministic function based on stochastic characteristics. The GELU function can be closely approximated as,

$$GELU(x) = 0.5x(1 + tanh[\sqrt{2/\pi}(x + 0.044715x^3)]) \tag{2}$$

As seen in Fig. 4, the function is almost linear for $x > 0$ and tapered downward for $-1 < x < 0$. Since it has a linear slope on the positive side and a finite non-linear area on the negative side, it successfully solves the exploding and vanishing gradient problems. Since the negative side of the curve is not a constant, it avoids the 'dead-ReLU' problem [46]. In Table 2, GELU is compared with the rectified linear unit (ReLU) as the activation function for our model. Better accuracy and F1-Score are achieved using GELU. And henceforth, GELU has been utilized in the proposed model.

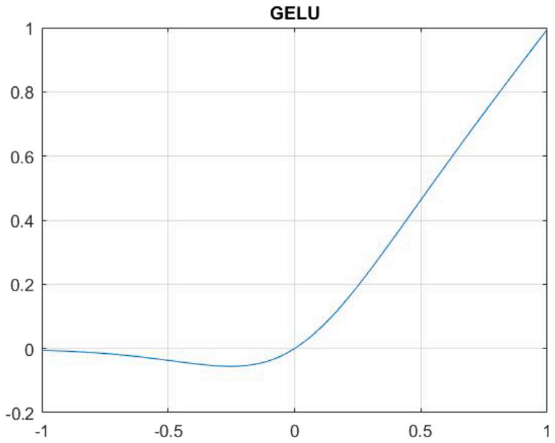

Fig. 4. Gaussian Error Linear Units (GELU).

MaxPool: After each convolution operation, a maxpooling operation was performed to reduce the resolution for the following convolution process.

Table 2
Comparison between ReLU and GELU activation functions.

	GELU			ReLU		
Learning rate	1e−4	5e−4	1e−3	1e−4	5e−4	1e−3
Accuracy (%)	89.45	89.84	90.07	85.01	89.62	89.20
F1-Score (%)	44.93	74..34	74.01	44.15	71.62	72.79

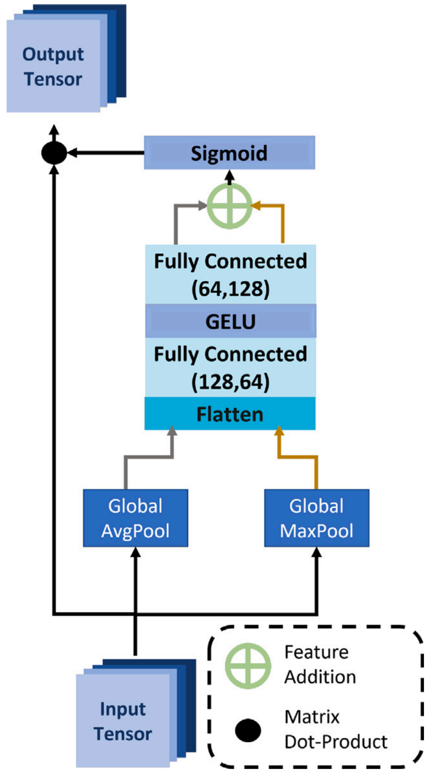

Fig. 5. Channel Attention Module (CAM).

Channel Attention Module (CAM): Since our features are stacked in channel dimension, we used a channel attention module [16] to capture the inter-channel relationships of the features. The channel attention module is illustrated in Fig. 5.

At first, Max pooling and Average pooling techniques are applied on each channel of the input tensor to produce two single values representing each channel. These values from each channel are then passed through a single feed-forward network. The results from the two branches are then added together. Then, *sigmoid* function is applied on the result of the addition. Finally, the result from the *sigmoid* operation is multiplied by each value of the corresponding channels of the input tensor.

The output tensors from the three branches of the MRFE have shapes of (batch size, 128, 64), (batch size, 128, 16), and (batch size, 128, 16), respectively. These tensors are then appended in the third dimension of the respective shapes to form a tensor of shape (batch size, 128, 96). A 1 × 1 convolutional layer is applied to this tensor to reduce the number of channels from 128 to 30.

3.3.3 Temporal Feature Decoder (TFD)

For capturing the temporal dependency of the extracted features, we employed a multi-head attention-based decoder. The decoder is based on the temporal context encoder (TCE) block adopted in [42] which is a modified version of the decoder part of a transformer network [17]. This decoder has been widely used in various applications involving time-series data such as speech, natural language, and EEG. However, this type of decoder has not been utilized in A-phase detection problems. Since CAP events are detected from sleep EEG which is a time-series data, we expected that this block could improve the performance of the model. The different parts of this module are illustrated in Fig. 3.

Table 3
Results for A-phase detection.

Subject	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	Specificity (%)
n1	91.18	71.69	72.53	72.11	94.65
n2	86.57	53.51	64.67	58.57	90.34
n3	92.65	70.73	60.06	64.96	96.82
n4	91.90	66.07	75.41	70.43	94.32
n5	89.64	73.93	61.95	67.42	95.43
n6	89.80	74.59	72.93	73.75	93.93
n7	91.92	65.43	80.11	72.03	93.68
n8	91.84	80.06	63.13	70.59	97.11
n9	93.54	74.44	54.44	62.89	97.91
n10	85.72	56.35	32.09	40.89	95.48
n11	87.22	66.74	47.34	55.39	95.25
n12	94.19	70.71	60.84	65.41	97.50
n13	89.84	77.30	71.59	74.34	94.56
n14	90.74	78.17	66.34	71.77	96.00
n15	92.17	79.06	72.80	75.80	96.09
n16	86.01	71.90	44.97	55.34	95.80
Average	90.31	70.67	62.58	65.73	95.30

Multi-Head Attention (MHA): The concept of multi-head attention is based on the theory of self-attention. Applying self-attention to a feature map trains it to learn which parameters it should pay more importance or 'attention' [47] to. It achieves this by generating a set of keys, values, and queries from the feature map, generally using a shallow feed-forward network. Considering a feature map, X containing n features; at first key, query, and value (q, k, v) for each feature are generated by the following equations:

$$q^{(k)} = F_1(X^{(k)}) \tag{3}$$

$$k^{(k)} = F_2(X^{(k)}) \tag{4}$$

$$v^{(k)} = F_3(X^{(k)}) \tag{5}$$

Here, F_1, F_2 and F_3 are shallow networks. The keys of every feature are then multiplied with the query of a particular feature. Results of each case are added together and after that, *softmax* function is applied to the result of the addition for generating a score for the feature. This operation is carried out as:

$$score^{(k)} = Softmax(\sum_{i=1}^{n} q^{(k)} \times k^{(i)}) \tag{6}$$

The resultant score is multiplied by the value of the feature to produce the output feature:

$$\hat{X}^{(k)} = score^{(k)} \times v^{(k)} \tag{7}$$

The multi-head attention mechanism works on the same principle as self-attention, but instead of multiplying queries with keys of all of the features, the features are divided into h heads, and self-attention is applied to each of them. The results from each head are then concatenated. In our model, self-attention is implemented as illustrated in Fig. 6. In our version of self-attention, we used 1D convolutional network to generate the keys, values, and queries.

3.3.4 Prediction

The final layer of our model is a fully connected layer followed by an activation function. As the activation function, *softmax* was used for multi-class detection and *sigmoid* function was used for the binary detection. The fully connected layer outputs 30 predictions as 30 × 1 tensor for A-phase detection and 30 × 4 tensor for detection of subtypes of A-phase. These predictions correspond to the outputs for the 30 s of input at once.

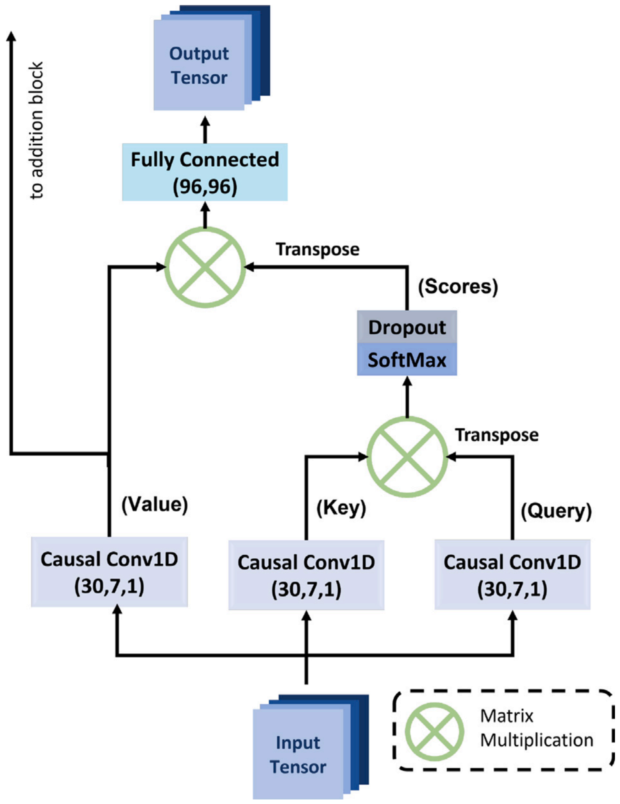

Fig. 6. Multi-Head Attention (MHA).

3.4 Performance metrics

We evaluated our model using different performance measures: Accuracy, Precision, Recall, F1-Score, and Specificity. These measures were calculated for both binary detection and multi-class detection task. In binary detection, A-phase was detected and the measures were calculated as [48]:

$$Accuracy = \frac{t_p + t_n}{t_p + f_p + t_n + f_n} \tag{8}$$

$$Precision = \frac{t_p}{t_p + f_p} \qquad Recall = \frac{t_p}{t_p + f_n} \tag{9}$$

$$F_1 \ \ Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{10}$$

$$Specificity = \frac{t_n}{f_p + t_n} \tag{11}$$

Here, t_p = True positive, t_n = True negative, f_p = False positive, f_n = False negative.

For multi-class detection where the subtypes of A-phase were detected, the performance measures were calculated as follows [48]:

$$Accuracy = \frac{\sum_{i=1}^{l}(tp_i + tn_i)}{\sum_{i=1}^{l}(tp_i + fp_i + tn_i + fn_i)} \tag{12}$$

$$Precision_M = \frac{\sum_{i=1}^{l} \frac{tp_i}{tp_i + fp_i}}{l} \tag{13}$$

$$Recall_M = \frac{\sum_{i=1}^{l} \frac{tp_i}{tp_i + fn_i}}{l} \tag{14}$$

$$F_1 \ \ Score_M = \frac{2 \times Precision_M \times Recall_M}{Precision_M + Recall_M} \tag{15}$$

$$Specificity_M = \frac{\sum_{i=1}^{l} \frac{tn_i}{fn_i + tn_i}}{l} \tag{16}$$

Here, l represents the number of classes and M represents macro averaging.

3.5 Experimental setup

All of the experimentations were implemented in Google Colaboratory™ Environment using NVIDIA Tesla T4 GPU, Intel(R) Xeon(R) CPU @ 2.20 GHz with 12.68 GB RAM, which is available for free. GPU specification changed sporadically due to Google Colaboratory™'s GPU allocation policy. For A-phase detection, a binary cross-entropy loss function was utilized, while a categorical cross-entropy loss function was used for subtypes detection. The model was trained for 100 epochs with a learning rate of 5×10^{-4}.

4 Results and discussion

In this paper, we used the leave-one-out (LOO) method as the cross-validation approach. This method was also used in [22]. In this method, data from one of the subjects is kept as the test set, and data from the remaining 15 subjects are used as the training set. The measures were computed for each test set, and the final metrics were calculated by averaging the results acquired from assessing the model with all the test subjects.

Table 3 lists the values of the A-phase detection performance measures. Measures for each of the 16 normal subjects and the average measures are displayed. For A-phase detection, we get 90.31% accuracy, 70.67% precision, 62.58% recall, 65.73% F1-Score, and 95.30% specificity on average. Furthermore, we have determined the performance metrics for each subtype and B-phase. Macro averaging is used to give equal importance to each subtype and B-phase [48] to calculate the values of precision, recall, specificity, and F1-Score using (13)–(16). Table 4 includes individual subject measures, measures for each subtype and non-CAP (B-phase), macro average, and the average of measures from all subjects. For subtypes detection, our model achieves an average of 86.72% accuracy, 59.96% precision, 60.65% recall, 59.59% F1-Score, and 89.53% specificity. It is also seen that for subtype A2 the value of F1-Score is relatively lower, 31.77%. The model might have misclassified A2 for the other subtypes because it has characteristics shared by both A1 and A3 subtypes. It may be noticed that the accuracy obtained for classifying subtypes is lower than that of A-phase detection. This can happen because (i) the highly unbalanced nature of the data where most of the EEG segments are non-CAP, (ii) during the subtype classification, the samples from the A-phase are further divided into A1, A2, and A3 subtypes, making the distribution more unbalanced. The availability of a larger dataset with a more balanced combination of the subtypes of A-phase may improve the result.

In Table 5, an analysis is shown for three model variants. The first and second ones are without the TFD block. It is obvious that without this block, F1-Score drops significantly. If the multi-resolution block includes channel attention module (CAM), for subject n5, accuracy does not change but F1-Score increases by 1.11% from the first variant, and for subject n13, accuracy increases by 1.23% and F1-Score improves by 3.79% from the first variant. The third variant shows that without CAM, accuracy decreases by 0.54%, and F1-Score decreases by 2.23% for the subject n5, and for the subject n13, accuracy decreases by 0.63%, and F1-Score decreases by 2.88% from the proposed model. These results imply that the channel attention module (CAM) improves the F1-Score, but the measures do not deteriorate largely without it. However, without the TFD block, the metrics drop significantly.

Table 6 compares our work with that of others for A-phase detection. Our model achieves better accuracy, specificity, and F1-Score for the A-phase detection task with some recall decrease. However, the previous works [26,30] sacrifice precision to achieve high recall, while we achieve a good balance between precision (70.67%) and recall (62.58%). Loh et al. and Dhok et al. both use data from 6 normal subjects (n1, n2, n3, n5, n10, and n11) as the dataset in [26] and

Table 4
Result for subtypes detection.

Subject	Accuracy (%)	Precision (%)					Recall (%)				
		B-Phase	A1	A2	A3	Macro Avg.	B-Phase	A1	A2	A3	Macro Avg.
n1	87.95	94.92	49.75	46.63	73.22	66.13	92.67	63.42	35.21	79.24	67.63
n2	80.21	94.15	40.23	23.01	34.71	48.02	85.25	38.13	43.17	67.61	58.54
n3	90.55	95.01	32.95	50.81	65.97	61.18	95.76	30.64	39.94	68.44	58.70
n4	88.80	95.81	45.20	23.88	68.61	58.38	92.89	56.80	30.06	77.78	64.38
n5	87.39	92.80	63.92	22.14	65.06	60.98	93.99	52.53	28.35	79.44	63.58
n6	85.36	94.94	50.08	38.24	67.49	62.69	90.81	64.62	36.70	79.54	67.92
n7	88.10	97.12	47.98	32.56	57.26	58.73	91.71	68.44	40.65	76.52	69.33
n8	87.06	94.88	37.70	25.30	77.20	58.77	92.17	58.86	36.77	65.59	63.35
n9	92.66	94.95	63.84	26.58	78.36	65.93	98.06	44.31	16.71	60.53	54.90
n10	84.27	89.66	41.96	17.19	54.88	50.92	93.87	35.39	11.66	32.14	43.27
n11	80.63	92.52	38.30	16.29	68.34	53.86	86.69	52.38	27.27	63.44	57.45
n12	93.17	96.05	54.02	31.36	78.65	65.02	97.29	50.47	24.18	61.08	58.26
n13	84.70	93.99	57.92	41.49	61.59	63.75	91.48	63.08	55.26	54.34	66.04
n14	87.35	93.81	41.64	51.30	73.44	65.05	95.10	44.06	46.35	63.31	62.21
n15	86.20	95.84	42.02	33.14	71.43	60.61	91.95	70.19	27.63	68.73	64.62
n16	83.07	88.60	46.57	44.07	58.04	59.32	94.52	36.77	22.85	46.58	50.18
Average	86.72	94.06	47.13	32.75	65.89	59.96	92.76	51.88	32.67	65.27	60.65

Subject	F1-Score (%)					Specificity (%)				
	B-Phase	A1	A2	A3	Macro Avg.	B-Phase	A1	A2	A3	Macro Avg.
n1	93.78	55.76	40.12	76.11	66.44	73.41	94.04	98.81	98.68	91.24
n2	89.48	39.15	30.02	45.87	51.13	69.21	96.63	95.15	92.14	88.28
n3	95.38	31.75	44.72	67.18	59.76	60.67	97.92	98.76	98.14	88.87
n4	94.32	50.34	26.62	72.91	61.05	72.28	95.88	98.01	98.09	91.06
n5	93.39	57.67	24.87	71.54	61.86	65.10	95.78	98.56	98.50	89.48
n6	92.83	56.43	37.45	73.02	64.93	80.18	93.85	97.20	97.37	92.15
n7	94.34	56.41	36.16	65.51	63.10	81.80	93.51	97.63	98.71	92.91
n8	93.50	45.96	29.97	70.93	60.09	72.90	95.19	97.43	98.20	90.93
n9	96.48	52.31	20.52	68.30	59.40	53.28	98.66	99.13	99.46	87.63
n10	91.72	38.40	13.89	40.54	46.14	40.49	95.37	98.92	98.65	83.36
n11	89.51	44.24	20.40	65.80	54.99	65.19	91.33	95.44	98.68	87.66
n12	96.67	52.19	27.31	68.76	61.23	59.66	97.56	99.59	99.51	89.08
n13	92.72	60.39	47.39	57.74	64.56	77.40	95.35	95.14	98.04	91.48
n14	94.45	42.82	48.70	68.00	63.49	70.92	96.50	96.99	98.54	90.74
n15	93.86	52.57	30.14	70.05	61.65	80.32	93.24	97.24	98.37	92.29
n16	91.46	41.09	30.09	51.68	53.58	49.04	95.24	98.43	98.59	85.33
Average	93.37	48.59	31.77	64.62	59.59	66.99	95.38	97.65	98.10	89.53

Table 5
A-phase detection measures for different model variants.

Subject, n5	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
MRFE without CAM	88.29	72.93	51.31	60.24
MRFE with CAM	88.29	71.46	53.75	61.35
MRFE without CAM + TFD	89.1	72.79	59.03	65.19
MRFE with CAM + TFD (Proposed Model)	89.64	73.93	61.95	67.42
Subject, n13	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
MRFE without CAM	87.16	72.21	61.02	66.15
MRFE with CAM	88.39	74.80	65.68	69.94
MRFE without CAM + TFD	89.21	78.30	65.73	71.46
MRFE with CAM + TFD (Proposed Model)	89.84	77.30	71.59	74.34

Table 6
Comparison of performance parameters for A-phase detection.

Author	Subjects	Accuracy (%)	Precision (%)	Recall (%)	Specificity (%)	F1-Score (%)
Hui Wen Loh et al. [26]	6	52.99	20.13	92.06	47.36	33.04
Murarka et al. [30]	6	60.59	19.76	69.52	59.29	30.77
Dhok et al. [24]	6	87.45	–	87.75	52.09	–
Arce-Santana et al. (50% training) [23]	9	88.1	–	–	–	–
Arce-Santana et al. (50% re-training) [23]	9	81.71	–	–	–	–
Mendonça et al. (multi-channel model) [29]	8	76	–	61	85	–
Mendonça et al. (single-channel model) [29]	15	72	–	66	75	–
Hartmann et al. [22]	16	86.43	–	76.1	88.49	63.46
This work	16	90.31	70.67	62.58	95.30	65.73

Table 7
Comparison of performance parameters for subtypes detection.

Author	Accuracy (%)	B-Phase		A1		A2		A3	
		Recall (%)	F1-Score (%)	Recall (%)	F1-Score (%)	Recall (%)	F1-Score (%)	Recall (%)	F1-Score (%)
Arce-Santana et al. (50% training) [23]	77.3	-	–	-	–	-	–	-	–
Arce-Santana et al. (50% re-training) [23]	88.32	-	–	-	–	-	–	-	–
Machado et al. [8]	71	-	–	58	–	24	–	44	–
Hartmann et al. [22]	81.89	85.51	90.23	63.14	46.61	42.31	32.96	70.62	60.32
This work	86.72	92.76	93.37	51.88	48.59	32.67	31.77	65.27	64.62

[24] respectively. In [26], Loh et al. use 70% of the total dataset for training, 15% for validation, and 15% for evaluating the model. On the other hand, in [24], Dhok et al. use 10-fold cross-validation. Our model achieves an improvement of 37.32%, 47.94%, and 32.69% in accuracy, specificity, and F1-Score respectively compared to [26] and an increase of 2.86% and 43.21% in accuracy and specificity compared to that of [24]. Murarka et al. [30] use data from 75 patients, including both healthy (n1, n2, n3, n5, n10, and n11) and sleep-disordered (narcolepsy, REM behavior disorder (RBD), PLM, NFLE, and insomnia) subjects. Compared to the results for the healthy subjects reported in [30], our model achieves an improvement of 29.72%, 36.01%, and 34.96% in accuracy, specificity, and F1-Score. Arce-Santana et al. mention the accuracy of the model for two cases- training and re-training while using 9 healthy subjects in [23]. Our model outperforms [23] by 2.21% and 8.6% for 50% training and re-training respectively in accuracy. Mendonça et al. [29] use 15 subjects for the single-channel model and 8 subjects for the multi-channel model. Our model achieves 14.31%, and 18.31% increase in accuracy compared to the multi-band channel model and the single-band channel model, respectively. In [22], Hartmann et al. use 16 normal subjects and the LOO method for cross-validation same as our method. Our model achieves an improvement of 3.88%, 6.81%, and 2.27% in accuracy, specificity, and F1-Score compared to that of [22].

The comparison with other approaches for the detection of subtypes of A-phase and non-CAP are shown in Table 7. Compared to the accuracy for the 50% training approach used by Arce-Santana et al. [23], our model achieves an improvement of 9.42%. Accuracy for the 50% re-training approach is better than ours. It should be mentioned that Arce-Santana et al. use each subject for both training and testing purposes; thus, there may be a chance to have a biased result. Our model achieves a 15.72% improvement in accuracy compared to that of Machado et al. [8]. Also, our model obtains mostly higher recall for A1, A2, and A3 subtypes than in [8]. On average, our model achieves a 4.83% increase in accuracy compared to [22]. Our model achieves an improvement in F1-Score by 3.14%, 1.98%, and 4.3% for B-phase, subtype A1, and subtype A3 respectively.

The results shown in Tables 6 and 7 were obtained by evaluating the model with an unbalanced test dataset. Results using an unbalanced test dataset were kept since in practical scenarios, test patients will be unknown to the system, and the annotation will be unavailable. Therefore, making the samples of CAP and non-CAP events equal is not possible in a real-world scenario.

Training for the full overnight recordings of 15 subjects took around 12 min. For each 30 s test epoch, it took around 281.25 microseconds, in general, to provide the inference using our model. The entire set of model parameters took up 3.4 MB of disk space. As a result, our approach is suitable for low-power, lightweight real-time applications.

5 Conclusions

In this paper, we have introduced an attention-based multi-resolution deep learning network for detecting A-phase and subtypes of A-phase. Our model consists of three basic blocks: MRFE, TFD, and prediction. Including the channel attention module (CAM) in the multi-branch 1D convolutional network has improved our F1-Score. The TFD blocks have improved both the accuracy and F1-Score. Overall, we have achieved satisfactory results in detecting A-phase, and its subtypes. Moreover, our model is lightweight, and the detection of CAP phases is faster with a potential for real-time usage.

However, there are some scopes of improvement in the performance of our model. One of the subjects from our dataset, labeled 'n10' proved to be more challenging for A-phase detection than other subjects. We believe that better pre-processing can improve the result for this subject. We only trained and evaluated our model on data from healthy subjects. In the future, we would like to work with a more diverse and larger dataset by incorporating subjects with various clinical conditions. Another area of future exploration is the detection of the CAP cycle and other CAP-related parameters.

CRediT authorship contribution statement

Barproda Halder: Conceptualization, Methodology, Software, Investigation, Writing – original draft, Writing – review & editing, Visualization. **Tanvir Anjum:** Conceptualization, Methodology, Software, Investigation, Writing – original draft, Writing – review & editing, Visualization. **Mohammed Imamul Hassan Bhuiyan:** Conceptualization, Writing – review & editing, Supervision, Investigation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

I have shared the link to the data https://physionet.org/content/capslpdb/1.0.0/CAP Sleep Database (Reference data).

Acknowledgments

We acknowledge the Department of Electrical and Electronic Engineering, Bangladesh University of Engineering and Technology for providing the necessary assistance and resources to carry out this research.

References

[1] U.S. Department of Health and Human Services, et al., What are sleep deprivation and deficiency? 2017, https://www.nhlbi.nih.gov/health/health-topics/topics/sdd/. (Online; Accessed 04 February 2022).

[2] S. Redline, G. Yenokyan, D.J. Gottlieb, E. Shahar, G.T. O'Connor, H.E. Resnick, M. Diener-West, M.H. Sanders, P.A. Wolf, E.M. Geraghty, et al., Obstructive sleep apnea–hypopnea and incident stroke: the sleep heart health study, Am. J. Respir. Crit. Care Med. 182 (2) (2010) 269–277.

[3] M.S. Khan, R. Aouad, The effects of insomnia and sleep loss on cardiovascular disease, Sleep Med. Clin. 12 (2) (2017) 167–177.

[4] D. Moser, P. Anderer, G. Gruber, S. Parapatics, E. Loretz, M. Boeck, G. Kloesch, E. Heller, A. Schmidt, H. Danker-Hopfe, et al., Sleep classification according to AASM and Rechtschaffen & Kales: effects on sleep scoring parameters, Sleep 32 (2) (2009) 139–149.

[5] C. Iber, The AASM manual for the scoring of sleep and associated events: Rules, in: Terminology and Technical Specification, American Academy of sleep Medicine, 2007.

[6] A.K. Patel, V. Reddy, J.F. Araujo, Physiology, sleep stages, in: StatPearls, StatPearls Publishing, Treasure Island (FL), 2022.

[7] M.G. Terzano, L. Parrino, A. Sherieri, R. Chervin, S. Chokroverty, C. Guilleminault, M. Hirshkowitz, M. Mahowald, H. Moldofsky, A. Rosa, et al., Atlas, rules, and recording techniques for the scoring of cyclic alternating pattern (CAP) in human sleep, Sleep Med. 2 (6) (2001) 537–553.

[8] F. Machado, F. Sales, C. Santos, A. Dourado, C. Teixeira, A knowledge discovery methodology from EEG data for cyclic alternating pattern detection, BioMed. Eng. OnLine 17 (1) (2018) 1–23.

[9] P. Halasz, Hierarchy of micro-arousals and the microstructure of sleep, Clin. Neurophysiol. 28 (6) (1998) 461–475.

[10] V. Gnoni, P. Drakatos, S. Higgins, I. Duncan, D. Wasserman, R. Kabiljo, C. Mutti, P. Halasz, P.J. Goadsby, G.D. Leschziner, et al., Cyclic alternating pattern in obstructive sleep apnea: A preliminary study, J. Sleep Res. 30 (6) (2021) e13350.

[11] S. Korkmaz, N.T. Bilecenoglu, M. Aksu, T.K. Yoldas, Cyclic alternating pattern in obstructive sleep apnea patients with versus without excessive sleepiness, Sleep Disord. 2018 (2018).

[12] M.G. Terzano, A. Smerieri, A. Del Felice, F. Giglia, V. Palomba, L. Parrino, Cyclic alternating pattern (CAP) alterations in narcolepsy, Sleep Med. 7 (8) (2006) 619–626.

[13] I. Chouvarda, M.O. Mendez, V. Rosso, A.M. Bianchi, L. Parrino, A. Grassi, M.G. Terzano, S. Cerutti, N. Maglaveras, Cyclic alternating patterns in normal sleep and insomnia: structure and content differences, IEEE Trans. Neural Syst. Rehabil. Eng. 20 (5) (2012) 642–652.

[14] G.B. Senel, E.U. Ozcelik, D. Karadeniz, Cyclic Alternating Pattern Analysis in Periodic Leg Movements in Sleep in Patients With Obstructive Sleep Apnea Syndrome Before and After Positive Airway Pressure Treatment, J. Clin. Neurophysiol. 38 (5) (2021) 456–465.

[15] M.Y. Kassab, M.U. Farooq, R. Diaz-Arrastia, P.C. Van Ness, The clinical significance of EEG cyclic alternating pattern during coma, J. Clin. Neurophysiol. 24 (6) (2007) 425–428.

[16] S. Woo, J. Park, J.-Y. Lee, I.S. Kweon, CBAM: Convolutional block attention module, in: V. Ferrari, M. Hebert, C. Sminchisescu, Y. Weiss (Eds.), Computer Vision – ECCV 2018, Springer International Publishing, Cham, 2018, pp. 3–19.

[17] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł.u. Kaiser, I. Polosukhin, Attention is all you need, in: I. Guyon, U.V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett (Eds.), Advances in Neural Information Processing Systems, Vol. 30, Curran Associates, Inc., 2017.

[18] A. Rosa, L. Parrino, M. Terzano, Automatic detection of cyclic alternating pattern (CAP) sequences in sleep: preliminary results, Clin. Neurophysiol. 110 (4) (1999) 585–592, http://dx.doi.org/10.1016/S1388-2457(98)00030-3.

[19] F. Machado, F. Sales, C. Bento, A. Dourado, C. Teixeira, Automatic identification of Cyclic Alternating Pattern (CAP) sequences based on the Teager Energy Operator, in: 2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC, 2015, pp. 5420–5423, http://dx.doi.org/10.1109/EMBC.2015.7319617.

[20] F. Machado, C. Teixeira, C. Santos, C. Bento, F. Sales, A. Dourado, A-phases subtype detection using different classification methods, in: 2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC, 2016, pp. 1026–1029, http://dx.doi.org/10.1109/EMBC.2016.7590877.

[21] F. Karimzadeh, E. Seraj, R. Boostani, M. Torabi-Nami, Presenting efficient features for automatic CAP detection in sleep EEG signals, in: 2015 38th International Conference on Telecommunications and Signal Processing, TSP, 2015, pp. 448–452, http://dx.doi.org/10.1109/TSP.2015.7296302.

[22] S. Hartmann, M. Baumert, Automatic A-phase detection of cyclic alternating patterns in sleep using dynamic temporal information, IEEE Trans. Neural Syst. Rehabil. Eng. 27 (9) (2019) 1695–1703.

[23] E.R. Arce-Santana, A. Alba, M.O. Mendez, V. Arce-Guevara, A-phase classification using convolutional neural networks, Med. Biol. Eng. Comput. 58 (5) (2020) 1003–1014, http://dx.doi.org/10.1007/s11517-020-02144-6.

[24] S. Dhok, V. Pimpalkhute, A. Chandurkar, A.A. Bhurane, M. Sharma, U.R. Acharya, Automated phase classification in cyclic alternating patterns in sleep stages using Wigner–Ville distribution based features, Comput. Biol. Med. 119 (2020) 103691.

[25] F. Mendonça, S.S. Mostafa, F. Morgado-Dias, A.G. Ravelo-García, Cyclic alternating pattern estimation from one EEG monopolar derivation using a long short-term memory, in: 2019 International Conference in Engineering Applications, ICEA, 2019, pp. 1–5, http://dx.doi.org/10.1109/CEAP.2019.8883470.

[26] H. Loh, C. Ooi, S. Dhok, M. Sharma, A. Bhurane, U.R. Acharya, Automated detection of cyclic alternating pattern and classification of sleep stages using deep neural network, Appl. Intell. 52 (2022) http://dx.doi.org/10.1007/s10489-021-02597-8.

[27] S. Hartmann, M. Baumert, Improved A-phase detection of cyclic alternating pattern using deep learning, in: 2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC, 2019, pp. 1842–1845, http://dx.doi.org/10.1109/EMBC.2019.8857006.

[28] F. Mendonça, A. Fred, S.S. Mostafa, F. Morgado-Dias, A.G. Ravelo-García, Automatic detection of cyclic alternating pattern, Neural Comput. Appl. (2018) 1–11.

[29] F. Mendonça, S.S. Mostafa, F. Morgado-Dias, A.G. Ravelo-García, Cyclic alternating pattern estimation based on a probabilistic model over an EEG signal, Biomed. Signal Process. Control 62 (2020) 102063.

[30] S. Murarka, A. Wadichar, A. Bhurane, M. Sharma, U.R. Acharya, Automated classification of cyclic alternating pattern sleep phases in healthy and sleep-disordered subjects using convolutional neural network, Comput. Biol. Med. 146 (2022) 105594, http://dx.doi.org/10.1016/j.compbiomed.2022.105594.

[31] S. Mousavi, F. Afghah, U.R. Acharya, SleepEEGNet: Automated sleep stage scoring with sequence to sequence deep learning approach, PLoS One 14 (5) (2019) e0216456.

[32] A. Supratak, H. Dong, C. Wu, Y. Guo, DeepSleepNet: A model for automatic sleep stage scoring based on raw single-channel EEG, IEEE Trans. Neural Syst. Rehabil. Eng. 25 (11) (2017) 1998–2008.

[33] E. Eldele, Z. Chen, C. Liu, M. Wu, C.-K. Kwoh, X. Li, C. Guan, An attention-based deep learning approach for sleep stage classification with single-channel EEG, IEEE Trans. Neural Syst. Rehabil. Eng. 29 (2021) 809–818.

[34] T. Mahmud, I.A. Khan, T.I. Mahmud, S.A. Fattah, W.-P. Zhu, M.O. Ahmad, Sleep apnea event detection from sub-frame based feature variation in EEG signal using deep convolutional neural network, in: 2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society, EMBC, IEEE, 2020, pp. 5580–5583.

[35] M.G. Terzano, L. Parrino, A. Sherieri, R. Chervin, S. Chokroverty, C. Guilleminault, M. Hirshkowitz, M. Mahowald, H. Moldofsky, A. Rosa, R. Thomas, A. Walters, CAP Sleep Database, physionet.org, 2001, http://dx.doi.org/10.13026/C2VC79.

[36] S. Kiranyaz, O. Avci, O. Abdeljaber, T. Ince, M. Gabbouj, D. Inman, 1D Convolutional Neural Networks and Applications: A Survey, Mech. Syst. Signal Process. 151 (2021) http://dx.doi.org/10.1016/j.ymssp.2020.107398.

[37] E. Arce-Santana, A. Alba, M. Mendez, V. Arce, A-phase classification using convolutional neural networks, Med. Biol. Eng. Comput. 58 (2020) http://dx.doi.org/10.1007/s11517-020-02144-6.

[38] Ö. Türk, M.S. Özerdem, Epilepsy detection by using scalogram based convolutional neural network from EEG signals, Brain Sci. 9 (5) (2019) 115, http://dx.doi.org/10.3390/brainsci9050115.

[39] S. Kiranyaz, T. Ince, R. Hamila, M. Gabbouj, Convolutional neural networks for patient-specific ECG classification, in: 2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC, 2015, pp. 2608–2611, http://dx.doi.org/10.1109/EMBC.2015.7318926.

[40] W. Huang, J. Cheng, Y. Yang, G. Guo, An improved deep convolutional neural network with multi-scale information for bearing fault diagnosis, Neurocomputing 359 (2019) 77–92, http://dx.doi.org/10.1016/j.neucom.2019.05.052.

[41] W. Dai, C. Dai, S. Qu, J.B. Li, S. Das, Very deep convolutional neural networks for raw waveforms, in: 2017 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP, 2017, pp. 421–425.

[42] E. Eldele, Z. Chen, C. Liu, M. Wu, C.-K. Kwoh, X. Li, C. Guan, An attention-based deep learning approach for sleep stage classification with single-channel EEG, IEEE Trans. Neural Syst. Rehabil. Eng. 29 (2021) 809–818, http://dx.doi.org/10.1109/TNSRE.2021.3076234.

[43] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proc. Inst. Radio Eng. 86 (11) (1998) 2278–2323, http://dx.doi.org/10.1109/5.726791.

[44] S. Ioffe, C. Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift, in: Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37, ICML '15, JMLR.org, 2015, pp. 448–456.

[45] D. Hendrycks, K. Gimpel, Bridging nonlinearities and stochastic regularizers with Gaussian error linear units, CoRR abs/1606.08415, 2016, arXiv:1606.08415.

[46] C. Hansen, Activation functions explained - Gelu, Selu, Elu, relu and more, in: Machine Learning from Scratch, 2020, URL https://mlfromscratch.com/activation-functions-explained/#vanishing-gradients-problem.

[47] R. Karim, Illustrated: Self-attention, in: Medium, Towards Data Science, 2022, URL https://towardsdatascience.com/illustrated-self-attention-2d627e33b20a.

[48] M. Sokolova, G. Lapalme, A systematic analysis of performance measures for classification tasks, Inf. Process. Manage. 45 (4) (2009) 427–437.