SleepFCN: A Fully Convolutional Deep Learning Framework for Sleep Stage Classification Using Single-Channel Electroencephalograms

Narjes Goshtasbi, Reza Boostani, and Saeid Sanei, *Senior Member, IEEE*

***Abstract*— Sleep is a vital process of our daily life as we roughly spend one-third of our lives asleep. In order to evaluate sleep quality and potential sleep disorders, sleep stage classification is a gold standard method. In this paper, we introduce a novel fully convolutional neural network architecture (SleepFCN) to classify sleep stages into five classes using single-channel electroencephalograms (EEGs). The framework of SleepFCN includes two major parts for feature extraction and temporal sequence encoding namely multi-scale feature extraction (MSFE) and residual dilated causal convolutions (ResDC), respectively. These are then followed by convolutional layers of 1-sized kernels instead of dense layers to build the fully convolutional neural network. Due to the imbalance in the distribution of sleep stages, we incorporate a weight corresponding to the number of samples of each class in our loss function. We evaluated the performance of SleepFCN using the Sleep-EDF and SHHS datasets. Our experimental results show that the proposed method outperforms state-of-the-art works in both classification correctness and learning speed.**

***Index Terms*— CNN, Deep Learning, EEG, Single-channel, Sleep stage classification.**

I. Introduction

SLEEP is an important brain state. Having a good sleep quality is essential for guaranteeing normal body performance and mental health. During sleep, the brain goes to several stages. In the non-rapid eyes movements (Non-REM) stage, the neural systems (e.g., emotional and sensory systems) of the brain get rest and calibrated. Inadequate sleep can cause significant problems such as increasing the risk of Alzheimer's, diabetes, and cancer. In addition, disturbing the sleep rhythm can lead to anxiety, depression, and even invoke suicidal thoughts [1], [2]. As a result, specialists need to analyze sleep patterns to identify sleep-related conditions such as drowsiness, fatigue, or sleep disorders including narcolepsy, insomnia, and sleep apnea [3]. Several attempts have been made to distinguish sleep stages using polysomnography to quantitatively measure the duration of each of the sleep stages [4]. Polysomnography measures a set of physiological signals such as electroencephalography (EEG), electromyogram (EMG), electrooculogram (EOG), and electrocardiogram (ECG) from each subject during their sleep. Sleep experts visually classify successive 30-second intervals of the EEG signals based on a set of rules like those provided by Rechtschaffen & Kales [5], scoring sleep into six stages, including Wake, S1, S2, S3, S4, and REM. Manual scoring by an expert is time-consuming and involves a degree of uncertainty [6]–[8]. To make this subjective problem objective, researchers used intelligent techniques to automatically classify the sleep stages. These methods need to extract sleep patterns, select the more important ones, and then apply the selected features to an efficient classifier in order to differentiate sleep stages. EEG features can be extracted in the time, frequency, and time-frequency domains [9]. Although these methods have achieved reasonable performance, they are highly dependent to hand engineering features based on knowledge of sleep data characteristics. Hence, establishing an automated method to extract more general features seems inevitable [10], [11].

Some studies have explored deep learning algorithms to address this issue in recent years. These algorithms have been shown good performance in various fields of EEG analysis [12]–[14], most of which used convolutional [15]–[18] and recurrent neural networks [19]–[22]. Having a hard look over the literature, we can see that the studies can be classified into two categories. Some researchers try to adopt different modalities of polysomnography (e.g., EEG, EMG) [23]–[25]. It is obvious such multi-modal approaches impose high computational complexity on the recognition system at the cost of achieving better accuracy [26], [27]. On the other hand, using multiple modalities of PSG needs more intrusive devices to be attached to the subject during sleep and are more expensive [16]. Among other electrophysiological signals, EEGs are of high interest in various cognitive tasks, since they are non-invasive, portable, and low in cost [28], [29]. Thus, most studies are conducted to decode the information captured through single-channel EEG signals in order to differentiate the sleep stages [30], [31]. Tsinalis *et al.* [15] used a convolutional neural network (CNN) to classify sleep stages from the raw Fpz-cz EEG channel. In [16], the authors used a relatively deeper CNN model to show the effect of model depth on the performance. In most research findings, an input signal is fed into the model in batches of 30 s EEG signals. In some research, the interval of 90 s is

Narjes Goshtasbi and Reza Boostani are with CSE & IT Department of Electrical and Computer Engineering, Shiraz University, Shiraz, Iran. (E-mail. n.Goshtasbi@shirazu.ac.ir, Boostani@shirazu.ac.ir).
Saeid Sanei is with School of Science and Technology, Nottingham Trent University, Nottingham NG11 8NS, UK (E-mail. saeid.sanei@ntu.ac.uk).

also considered to explore the effect of input length [32]. Moreover, Zhu *et al.* [33] fed 2 s epochs to their model in order to capture events such as K-complexes and sawtooth waves that occur rapidly (in 1–2 s). Olesen *et al.* [34] built a slightly deeper CNN model with four blocks of convolution layers and used skip connections between them and Andereotti *et al.* [35] incorporated a naïve inception-like module [36] in their CNN to take advantage of multiple time-frequency-based contexts. Various models used recurrent neural networks (RNNs), specially long-short term memory (LSTM), to capture sequential characteristics of sleep EEGs. For instance, Dong *et al.* [37] used a hybrid neural network structure consisting of the multi-layer perceptron neural networks and LSTM. In addition, Some studies considered techniques to handle the imbalanced class problem of sleep scoring. In [21], the authors have proposed a cascaded model of RNNs with two blocks of LSTMs which can separately classify the majority and minority stages and Supratak *et al.* [38] applied a data augmentation technique to the rarest stage to overcome its low performance.

Here is the description of our contributions in this study:

1) We constructed a novel fully convolutional neural network (FCN) that consists of three major parts. I. We built a custom multi-representational model of convolutional layers to extract local time-invariant features of sleep data in a multi-scale way, which we call multi-scale feature extraction (MSFE) module. II. Referring to the sequential order of sleep stages, we need to learn from these sequential sleep cycles, in order to accurately detect the sleep stages. To achieve this goal, we built a residual dilated causal convolutional module (ResDC), which models the sequence of sleep with a remarkably simpler and faster structure than RNNs . III. We integrated our fully convolutional architecture by the use of convolutional filters with kernel size = 1 instead of dense layers on the top of our model. This strategy could help us to overcome the disadvantages of dense layers that have been deployed in almost all the models in the sleep staging processes.
2) In order to deal with the imbalance problem, we proposed a loss function that takes the weight of each stage into account to correct the prediction errors more evenly.
3) We assessed our model with two public datasets and showed that it outperforms state-of-the-art models in the sleep stage classification task.
4) Most studies in the domain of sleep scoring obtained their results using data from healthy subjects. Despite being a potentially useful baseline for sleep scoring, this constrains the task and might reduce the generality of the model. In a supplementary experiment, we trained our model with unhealthy subjects to learn useful patterns for identifying abnormalities and disorders.

The rest of the paper is organized in the following order: in section II, we introduce our model and explain its components in detail together with the description of datasets and the preprocessing steps. Section III contains experimental results for different datasets as well as the comparison between our model and the previous ones. Finally, the paper is concluded in section IV.

II. Materials and Method

A. Data and Preprocessing

We used two publicly available datasets in this work. The first one, namely Sleep-EDF, which is widely used in sleep scoring, is derived from the PhysioBank [39]. It comprises the data from 20 healthy subjects, aged 25–101, 10 females and 10 males. The polysomnography was recorded for about 20 hours for each individual during two consecutive nights and includes two channels of EEG signals (Fpz-cz, Pz-oz) sampled at 100 Hz in addition to the EMG and EOG channels. Similar to previous studies [40], we use the single-channel Fpz-cz EEG as the input of our model. The second dataset is obtained from the sleep heart health study (SHHS) database [41], [42], comprising 6,441 men and women aged more than 40 years. Each subject's data has been recorded during about six hours of sleep, and sampled at 125 Hz. The subjects of this dataset suffer from sleep-correlated breathing diseases resulting from lung and cardiovascular abnormalities. To reduce the impact of disorders, we follow the work in [43] and select 120 subjects whose Apnea-Hypopnea-Indexes (AHI) is lower than 5. This shows that the subjects had almost a regular sleep pattern. We have taken the C4-A1 EEG of the selected subjects from the two EEG channels provided in this dataset (C4-A1, C3-A2). In another experiment, we selected and applied the data from another 120 subjects whose AHI is above 5, indicating a mild to severe state of obstructive sleep apnea, and fed them to the proposed model to explore the effect of breath-correlated abnormalities on the sleep stage classification.

Preprocessing includes these steps: we remove unknown stages and movements from signals and merge stages 3 and 4 into one stage according to the American Academy of Sleep Medicine (AASM) guideline [44]. Therefore, we have five sleep stages namely W, N1, N2, N3, and REM whereby W represents the state of wakefulness, ranging from full alertness to early drowsiness and REM stands for rapid eye movement sleep in which sharp and irregular eye movements are observable. Stages N1, N2 and N3 are the stages of non-REM sleep. By way of explanation, stage N1 is characterized by low amplitude and mixed frequency activities (4–7 Hz), stage N2 is characterized by the presence of sleep spindles and K-complexes, while during stage N3, the subject experiences a state of deep sleep associated with slow wave activity. Then, we exclude the redundant wake times before and after the sleep onset. In other words, we keep in-bed times which include 30 minutes of wake stages before and after the sleeping time. EEG signals were filtered by a 5^{th} order Butterworth filter with a cutoff frequency of 30 Hz to reduce artifacts and then standardized by removing the mean and scaling to unit variance. After all, each data was segmented into successive 30 s windows. Table I shows the per-class and the total number of 30 s epochs of each dataset.

B. Proposed Method

There are three main challenges in sleep stage classification. First, we should control the trade-off between the time and

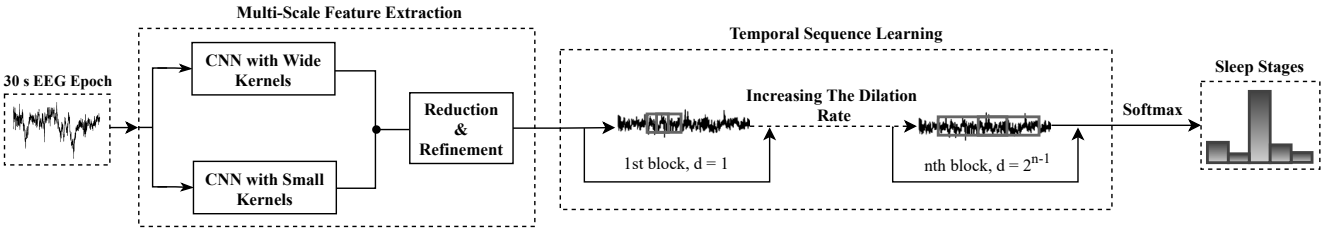

Fig. 1. The overall framework of the proposed method. Parameter d indicates the dilation rate of the convolutional layers and n is experimentally set to 4.

TABLE I
THE NUMBER OF PER-STAGE AND TOTAL 30 S EPOCHS FOR EACH DATASET.

Dataset	W	N1	N2	N3	REM	Total
Sleep-EDF	8285	2804	17799	5703	7717	42308
SHHS	5252	1279	17874	7405	7961	39771

frequency resolutions during the feature extraction phase. The second is to learn the temporal characteristics of EEG signals to capture the dependencies of samples that lead to the transition between sleep stages. Eventually, we should incorporate some techniques in our model to overcome the imbalanced distribution of samples between sleep stages. To deal with these challenges, we propose a novel FCN called SleepFCN, the components of which are explained in subsequent parts. The overall framework of SleepFCN is shown in Fig. 1.

1) Time-Frequency Resolution Trade-Off: As the first part of the model, we construct MSFE module containing two convolutional branches with different sizes of kernels, inspired by [38], [45] and the fact that each sleep stage is associated with a specific frequency band [46]. In this way, we will be able to capture information from different frequency bands of EEG signals and local time-invariant features in the time domain. In one branch of this block, we use small kernels of size 25, and in the other, wide kernels of size 200 as the first layer. These layers are able to capture frequency band information directly from input EEG samples. To illustrate more, imagine an input signal sampled at frequency 100 Hz, so each kernel size of 200 can capture 2 s of the EEG signal. In the frequency domain, these kernels slide on groups of samples with a frequency of about 0.5 Hz, i.e., slow waves that occur in a deep sleep. Furthermore, the filter with a kernel size of 25, can capture frequencies about 4 Hz. Therefore, with respect to the stride, theta and alpha or even beta bands can be learned. In order to make these features more interpretable for a classifier, we need to refine them after extraction. Hence, these convolutions are followed by batch normalization [47], which is a regularization technique for training deep neural networks, normalizing the input to each layer for mini-batches of the data. This method results in a more stable learning process since the prediction errors can be propagated backward more effectively through the layers. The number of epochs required for training the network can

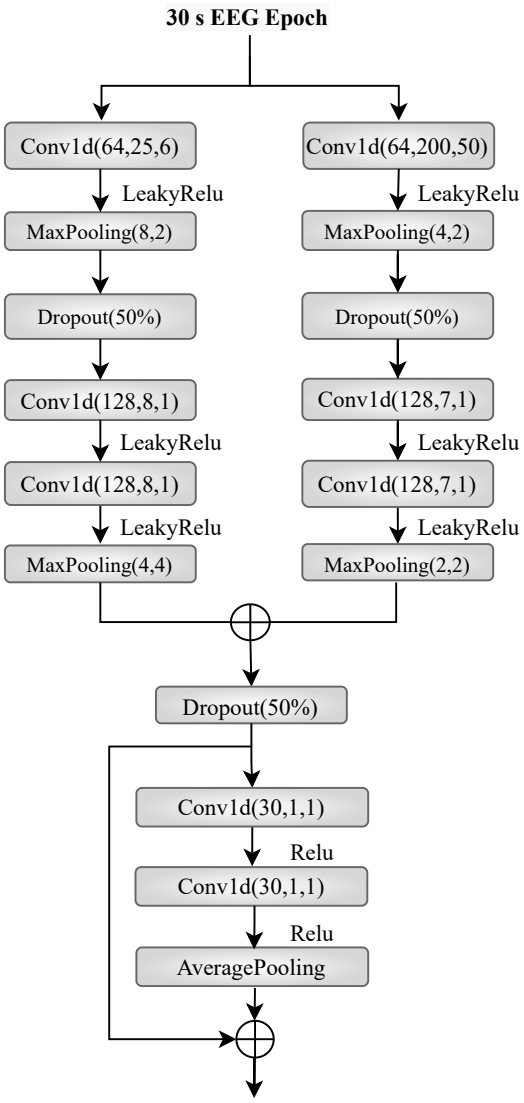

Fig. 2. The structure of the MSFE module.

be significantly reduced as a result. The outputs then are passed through a Leaky Rectified Linear Unit (LeakyReLU) activation function that allows the negative outputs of neurons to be passed through the layer. Consequently, we do not lose such information as opposed to the ReLU that is used in similar works. Essentially, the initial convolutional layer of CNNs can be thought of as extracting basic features of the samples, and as we delve deeper into the network, the

features become more representative. Thus, in each branch, we add convolutional and pooling layers to extract more details of the signal and prevent the excessive increase of the parameters, respectively. Features extracted by these two branches then are concatenated into one array and are passed through the convolutional layers activated by ReLU, and an average pooling mechanism for refinement and reduction purposes. It is worth mentioning that negative values are no longer a concern since we have included and processed them within the feature space through the two branches of the MSFE module. In order to facilitate faster convergence, we opt for ReLU activation in the converging part of the module. The whole MSFE module is shown in Fig. 2.

2) Sequential Feature Learning: Sleep stage classification is naturally a sequential problem. According to the AASM guideline, an epoch is labeled as N2 if some events such as K-complexes or sleep spindles occur in the 2_{nd} half of the previous epoch [44]. Furthermore, the REM stage depends on the presence of mixed frequency activities in EEG signals without the occurrence of K-complexes or sleep spindles in the prior epoch. Indeed, it can be distinguished even without rapid eye movements, based on information learned by the sequence [21]. Consequently, after extracting the features of interest, we should make some decisions to capture the sequential characteristics of the sleep data, i.e., implement a sequential modeling scheme. For this purpose, we develop the ResDC module inspired by [48]. It could control the trade-off between accurate localization and context-aware understanding, including learning the sequence's order of occurrence, to capture the transition rules between sleep stages.

There are two fundamental principles behind the ResDC module. Firstly, it provides a causal sequence modeling scheme, in which an output at the time t is produced by performing convolution on the inputs at time t and earlier, i.e., its past sequence [48]. Secondly, the model is required to produce an output sequence with the same size as the input. This can be accomplished by setting the shape of hidden layers equal to the input layer and adding zero pads of length (kernel size-1). The problem is, however, that these settings require a very deep network and/or extensive filters in order to capture the history of sequence, which is difficult when using classic convolutional layers. Dilated convolutions are employed to solve this problem [49], which allow the network to broaden the filter's field of view, incorporating a larger context. For a sequence input x and a filter f, the dilated convolution operation D on point s is defined as:

$$D(s) = \sum_{i=0}^{k-1} f(i).x_{s-d.i} \tag{1}$$

where k stands for kernel size and d is the dilation factor, assuming $d = 1$ results in classical convolution operation. For $d > 1$, the receptive field of the convolution effectively grows such that a broader range of features at the beginning of the module could be flowed through consecutive layers [48]. To achieve this aim, we increase the dilation rate exponentially, concerning the depth of the ResDC module. Furthermore,

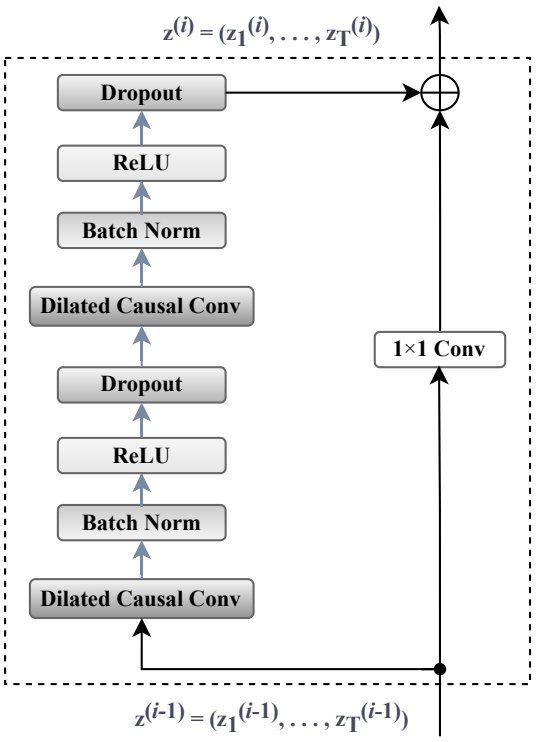

Fig. 3. A residual dilated causal convolutional block. The entire ResDC module is built by connecting a series of such blocks with an exponentially increasing dilation rate.

there are residual connections between multiple blocks in the ResDC module. These connections help the model learn identity function besides the dilated causal convolutions which is a more complex transformation. Having demonstrated good performance in the literature, this can be a wise choice to help the model become more generalized [48]. The architecture of each residual block of the ResDC module includes two dilated causal convolutions, each followed by batch normalization and non-linearities, Fig. 3.

The network is then followed by two convolutional layers with a kernel size of 1. This idea is proposed to eliminate the disadvantages of dense layers. Because the number of parameters must be determined to construct a dense layer, using such layers necessitates that the model's input dimensions be fixed. In contrast, a convolutional layer receives inputs of arbitrary size and produces outputs of the appropriate size. It should also be noted that the spatial information could be lost in dense layers, whereas this is not the case for convolutional layers, given that the convolution is a local operation [50]. After all, an adaptive average pooling mechanism is applied to these outputs, and a softmax classifier activates them to classify the features learned for different sleep stages so far.

3) Imbalanced Classification Problem: One challenging problem in classifying sleep epochs into stages of desire is that the number of samples in different stages is not equal. The stage N1, as the AASM guideline states, contains the transition episodes between wake and sleep. Therefore, this stage is

the minority class in almost every dataset. Moreover, most individuals spend more sleep time in the stages N2 and N3 rather than REM. Hence, these are the most major classes in many sleep datasets [44]. Classifiers are always biased toward the majority class since common loss functions correct errors of all classes with the same significance. To overcome this issue, we construct a loss function sensitive to the number of samples in each class. Further, the loss function should learn the proportion of the least and the most represented stages. We exploit the well-known cross-entropy function as the foundation for our loss function, which is expressed as (2) in a multi-class fashion.

$$Cr = -\frac{1}{N}\sum_{s=0}^{S-1}\sum_{n=0}^{N-1} y_{ns}.log(\hat{y}_{ns}), \tag{2}$$

where N and S stand for the number of 30 s EEG epochs and sleep stages, y and $\hat{y}$ represent the actual and predicted stages, respectively, and Cr is the categorical cross-entropy loss function. This loss function is an excellent criterion for distinguishing between two probability distributions (i.e., the actual and predicted stages). We reformulate (2) into (4) by reweighting it relative to the following two factors: the ratio of the minor class (num_{minor}) to the major one (num_{major}), and the inverse proportion of samples in the whole dataset $(\frac{N}{N_s})$. In brief, the weight for each stage is shown as (3).

$$W_s = \frac{N}{N_s}.\frac{num_{minor}}{num_{major}} \tag{3}$$

$$Loss = -\frac{1}{N}\sum_{s=0}^{S-1}\sum_{n=0}^{N-1} W_s.y_{ns}.log(\hat{y}_{ns}) \tag{4}$$

III. Experimental Results

A. Performance Metrics

We use overall and per-class metrics to evaluate the performance of our model. Although one of these metrics is accuracy, due to the fact that we are faced with an imbalanced problem, i.e., sleep scoring, the widely-used conventional accuracy metric can not fully represent the model's correctness. By way of illustration, suppose we have a dataset, samples of which belong to two classes, with the first-class containing 90 percent of all samples. In such circumstances, even if the classifier mistakenly classifies all second-class samples with the label of first-class, the accuracy criterion will give us a 90 percent correctness rate. It is a biased interpretation that can be deceptive, especially in clinical diagnostic tasks such as sleep scoring. Hence, we need to employ the F1-score to interpret the model performance precisely. Macro F1-score is a faultless measurement of classification performance in the imbalanced problems. It establishes a compromise between how much the results are relevant (i.e., precision) and the percentage of predictions truly classified by the algorithm (i.e., recall). We also used Cohen's kappa to compare our model to other works as another common metric [51]. With the True Positives (TP), False Positives (FP), True Negatives (TN), and False Negatives (FN) of each class which can be yielded from the confusion matrix, the metrics stated above are calculated as follows:

$$PR_s = \frac{TP_s}{TP_s + FP_s} \tag{5}$$

$$RE_s = \frac{TP_s}{TP_s + FN_s} \tag{6}$$

$$mf1 = \frac{1}{S}\sum_{s=0}^{S-1}\frac{PR_s \times RE_s}{PR_s + RE_s} \tag{7}$$

$$acc = \frac{\sum_{s=0}^{S-1} TP_s}{N} \tag{8}$$

where PR_s and RE_s represent the precision and recall of each class s, respectively, $mf1$ indicates the macro F1-score and acc stands for the accuracy.

B. Sleep Stage Classification Performance

Table II and III show the confusion matrices and per-class metrics obtained by 20-fold cross-validation on Sleep-EDF and SHHS datasets, respectively. It can be inferred from Table II that the stages N1 and REM have the most misclassified number of samples among all classes. The stage REM is mostly misclassified with stage N2, and the stage N1 has been confused with the stages Wake, N2, and REM. Another fact derived from the tables is that stage N3 was only confused with stage N2. The model's overall accuracy and macro F1-score on the Sleep-EDF dataset were 84.8% and 78.8%, respectively. Per-category precision, recall, and F1-score reported in the tables show that the performance of all classes except N1 is relatively high and reliable. Stage N1 is significantly low in F1-score and accuracy. The reason for this issue is the small number of samples in this class, and incidentally, for the same reason, it does not affect the overall accuracy strongly. However, this poor ability to classify stage N1 reduces the overall macro F1-score. Fig. 4 shows the sleep stages for 250 subsequent epochs (about 2 hours) of subject 9 in the Sleep-EDF dataset. This figure shows the information inferred from the confusion matrix more clearly. For example, it can be seen in the figure that stage N3 is only misclassified as N2. It has also some unique information, e.g., it is evident from the figure that the most wrong predictions occur when the transition is from one stage to another.

TABLE II
THE CONFUSION MATRIX OF SLEEPFCN APPLIED TO SLEEP-EDF DATASET (CHANNEL FPZ-CZ)

	Predictions					Per-Class Metrics		
	W	**N1**	**N2**	**N3**	**REM**	**PR**	**RE**	**F1**
W	7373	397	240	19	148	90.2	89.1	89.6
N1	554	1275	492	2	594	43.7	45.5	44.6
N2	106	527	15666	424	683	90.1	88.0	89.1
N3	26	6	591	5241	4	89.3	91.9	90.6
REM	226	599	810	17	6288	79.2	81.5	80.3

TABLE III
THE CONFUSION MATRIX OF SLEEPFCN APPLIED TO SHHS DATASET (CHANNEL C4-A1)

	Predictions					Per-Class Metrics		
	W	N1	N2	N3	REM	PR	RE	F1
W	4502	253	176	55	266	75.4	85.7	80.2
N1	178	337	215	0	549	25.1	26.3	25.7
N2	537	355	14581	1032	1369	89.6	81.9	85.4
N3	417	2	799	6177	10	84.8	83.4	84.1
REM	337	395	495	16	6718	75.4	84.4	79.6

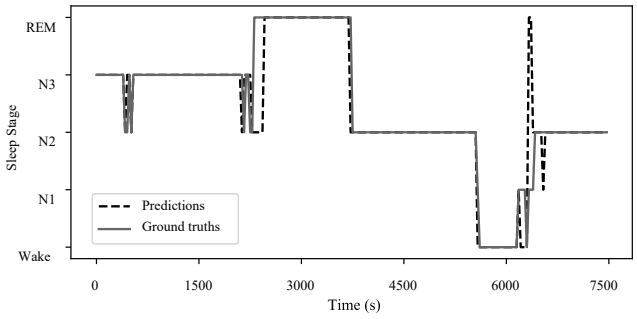

Fig. 4. Comparison between the scores provided by expert clinicians and predictions of the SleepFCN model.

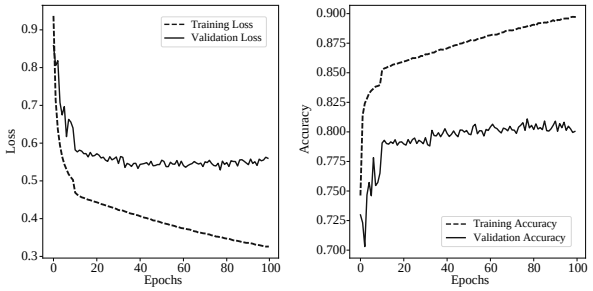

Fig. 5. The learning curve obtained by fold 10 (subject 16) of Sleep-EDF dataset.

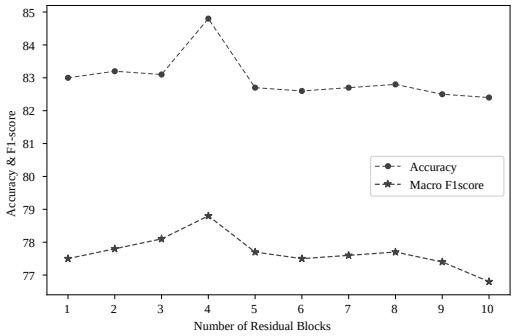

Fig. 6. The performance of the SleepFCN model with varying number of residual blocks used to construct the ResDC module.

C. Experimental Design

The model was trained and validated through a 20-fold cross-validation method. That is to say, the dataset is divided into 20 sets (folds), and then the model is trained 20 times, each time 19 sets are fed to the model for training, and one set is held out to evaluate the model. It is identical to the leave-one(patient)-out scheme for the Sleep-EDF dataset as it contains 20 subjects. Consequently, one subject is held out as the validation set in each fold. We scored the overall performance by aggregating the predictions obtained based on the validation data from each fold after running 20-fold cross-validation.

Using PyTorch, we built the model and trained it on an Nvidia GeForce-RTX1070 GPU. Here are some settings in the training procedure. We chose Adam optimizer [52] with a decaying learning rate whose initial point is 0.001 and the decay rate of 0.1 every 15 epochs given that the model requires a high learning rate at the start to reduce large errors. Whereas a lower learning rate is required to avoid getting stuck in local minimums as the training progresses. We set the batch size to 128 and initial weights of all convolutional layers to the random normal distribution with a mean of zero and standard deviation of 0.05. The number of epochs was set to 100, as the model converges in less than 100 epochs. The learning curve in Fig. 5 shows that the validation error and accuracy almost stabilize before reaching 100 epochs. This reflects the robustness of the SleepFCN against the overfitting problem. There are also small oscillations due to the small size of validation data compared to the training data. We also explored the effect of the number of residual blocks forming ResDC module by setting them to multiple numbers in the range of 1 to 10. After training the models, we observed that with four of such blocks, the best performance in regard to accuracy and macro F1-score had reached, Fig. 6. The reason is that the classification of sleep stages is based on a relatively short series of patterns (e.g., a few epochs prior to a 30 s epoch) arising in the data and does not encompass the entire sequence. Therefore, the construction of a large ResDC structure containing a large number of residual blocks for capturing a long sequence, no longer contributes to enhancing performance but rather increases the training time. It should be noted that each convolutional layer of the ResDC module contains 32 filters.

D. Comparison with Other Methods

To show the validity of the SleepFCN model, we compared its performance to some state-of-the-art studies in terms of accuracy, macro F1-score, Cohen's kappa, and the speed of training. We considered some of their results as the baseline for our work. All of them used a customized representational learning module. While each of them used a different module as temporal context learning, in [38], the authors used LSTM layers (DeepSleepNet), Eldele *et al.* [40] employed self-attention (AttnSleep), and the authors of [53] built a module that included both RNNs and attention mechanism (SleepEEGNet). Based on these works, we built our representation module as explained in Section II. We then captured and encoded the temporal context by proposing an alternative way, i.e., ResDC module. We also presented a different approach proposed in [33] that we named it CNNAttention;

TABLE IV
COMPARISON BETWEEN METHODS IN TERMS OF PER-CLASS F1-SCORE AND OVERALL MACRO F1-SCORE, ACCURACY AND COHEN'S KAPPA FOR TWO DATASETS.

Dataset	Method	Per-Class F1-score					Overall Metrics			Training Time
		W	N1	N2	N3	REM	acc	mf1	kappa	(Per-Fold)
Sleep-EDF	NaïveCNN [15]	65.4	43.7	80.6	82.1	74.5	73.6	69.3	0.70	20 mins
	DeepSleepNet [38]	86.7	45.5	85.1	83.3	**82.6**	81.9	76.6	0.76	2.5 hours
	SleepEEGNet [53]	89.4	44.4	84.7	84.6	79.6	81.7	76.5	0.75	1.5 hours
	CNNAttention [33]	89.3	**46.1**	86.0	82.1	83.2	81.5	77.3	0.77	1.9 hours
	AttnSleep [40]	88.6	42.4	88.2	89.2	80.8	84.0	77.8	0.78	35 mins
	SleepFCN *(Ours)*	**89.6**	44.6	**89.1**	**90.6**	80.3	**84.8**	**78.8**	**0.79**	**27 mins**
SHHS	NaïveCNN [15]	60.1	26.5	76.2	78.4	74.5	73.0	63.1	0.67	17 mins
	DeepSleepNet [38]	81.0	29.6	81.3	79.2	81.5	80.1	70.5	0.74	2.4 hours
	SleepEEGNet [53]	81.9	29.2	81.5	80.9	79.6	79.6	70.6	0.72	1.2 hours
	CNNAttention [33]	81.9	**29.8**	82.1	79.8	**82.3**	79.6	71.2	0.72	1.8 hours
	AttnSleep [40]	**82.1**	29.1	84.9	83.5	79.2	81.0	71.8	0.73	32 mins
	SleepFCN *(Ours)*	81.7	29.0	**85.4**	**84.1**	79.6	**81.3**	**72.0**	**0.74**	**25 mins**

they did not use a representation learning similar to the mentioned studies but a simpler single-resolution one. To capture the context of sleep, they used an attention mechanism. We brought this model to our comparison scheme to show that the multi-resolution CNN is able to achieve a higher performance comparable to single-resolution ones. In addition, we presented the results of a model proposed in [15], which is almost the first work which employed CNNs in the field of sleep staging, to show how far the research has come along with this task. We named this model NaïveCNN as it is obtained by placing convolutional and pooling layers, followed by fully connected ones. It has a low number of parameters and thus the average time of training for each fold is lower than those of other works. Nevertheless, it is inferior amongst other models in terms of accuracy, macro F1-score, and Cohen's kappa. To have a fair comparison, especially in terms of training speed, we used the published codes of DeepSleepNet [38], SleepEEGNet [53], and AttnSleep [40] models, and re-implemented the architectures of NaïveCNN [15], and CNNAttention [33], codes of which have not been published publicly. We re-ran all these works through a 20-fold cross-validation scheme. As it can be seen in Table IV, our model outperforms others in terms of overall metrics for the Sleep-EDF and SHHS dataset. The SleepFCN also outperforms other per-stage F1-score for Wake, N2, and N3 but not for N1 and REM on the Sleep-EDF dataset. As is shown in Table IV, CNNAttention [33] and DeepSleepNet [38] have reached a higher F1-score for N1 and REM stages, respectively. This might be due to the use of data augmentation in their works.

For the second dataset, SHHS, the results have got slightly lower, which could be because of differences in subjects and channel configurations. SHHS EEG channels are recorded on C4-A1 channel, resulting in slightly different signal records from Fpz-cz channel in Sleep-EDF dataset. Additionally, the age and gender of subjects are different for different datasets, which is another contributing factor to the differences in performance between models based on these two datasets. Although the overall results of the SleepFCN model have remained superior to the compared models, it is lower than CNNAttention [33] and AttnSleep [40] in regard to N1 and Wake F1-scores, respectively. In an overview of these results, it can be concluded that the weights we used in our loss function, as well as the layout of the proposed model, have done well to achieve proper performance in sleep scoring. On the other hand, the average training time per-fold in the SleepFCN decreased noticeably compared to other related works due to the development of ResDC module instead of RNNs or even attention mechanisms which add more computational complexities to the model. In order to statistically examine superiority of our results, we performed the Friedman test [54]. This is a non-parametric measure which ranks different algorithms based on their performance on various datasets. In practice, the Friedman test demonstrated the significance of the proposed model.

As a supplementary experiment, we trained our model using 120 extra subjects with a mild stage of obstructive sleep apnea from the SHHS dataset. To explore the effect of sleep disorders on different sleep stage classification models, we also trained AttnSleep [40] with the same subjects as it performed better than its previous models in overall. Table V, VI, and VII show the normalized confusion matrices obtained by applying 20-fold cross-validation to the SleepFCN and AttnSleep [40]

TABLE V
THE NORMALIZED CONFUSION MATRIX OF SLEEPFCN APPLIED TO 120 UNHEALTHY SUBJECTS OF SHHS DATASET.

0.82	0.03	0.10	0.02	0.03
0.24	**0.24**	0.32	0	0.20
0.05	0.02	**0.82**	0.06	0.05
0.01	0	0.2	**0.79**	0
0.06	0.08	0.21	0	**0.65**

TABLE VI
THE NORMALIZED CONFUSION MATRIX OF ATTNSLEEP [40] APPLIED TO 120 UNHEALTHY SUBJECTS OF SHHS DATASET.

0.81	0.03	0.10	0.02	0.04
0.27	**0.21**	0.29	0	0.23
0.05	0.03	**0.80**	0.06	0.06
0.01	0	0.20	**0.76**	0.03
0.06	0.08	0.22	0.01	**0.63**

TABLE VII
COMPARISON BETWEEN THE SLEEPFCN AND ATTNSLEEP [40] MODELS ON 120 UNHEALTHY SUBJECTS OF SHHS DATASET.

Method	Per-Class F1-scores					Overall Metrics		
	W	N1	N2	N3	REM	acc	mf1	kappa
AttnSleep [40]	**82.5**	18.5	79.8	77.4	65.8	76.1	64.8	66.2
SleepFCN *(Ours)*	82.1	**21.2**	**80.1**	**78.1**	**68.9**	**76.6**	**66.1**	**67.1**

using random subjects, and per-class F1-scores and overall metrics. As is shown in the tables, the model performance reduces with random subjects because it is harder for a model to classify stages with abnormal patterns in contrast with standard patterns of healthy individuals (e.g., fewer fluctuations between sleep stages). However, our model is still superior to AttnSleep [40] in both overall and per-class metrics.

IV. CONCLUSION

In this study, a novel fully CNN called SleepFCN is proposed to classify sleep patterns into five classes according to the AASM manual [44]. The SleepFCN incorporates a module for feature extraction whose main idea is to capture multi-frequency band information as well as local time-invariant features of sleep data called MSFE. Another module called ResDC is developed for temporal context learning that employs causal and dilated convolutional layers that enables the model to capture the history, i.e., the context of the signal effectively. The proposed model was trained using single-channel EEG signals obtained by two publicly available datasets, namely Sleep-EDF and SHHS. A weighted loss function is applied to the training algorithm to reduce the data's skewed distribution effect. Experimental results show that such a model could be faster and more accurate than other methods using RNNs or even attention mechanisms. Moreover, the SleepFCN exhibits more generality for the breath-related distortions in individuals' sleep. Furthermore, avoiding fully connected layers provides the model flexibility to varying dimensions.

It is beneficial for sleep technicians to have a relatively accurate and fast model to train and test over unseen data. Therefore, the motives for future research in the field of sleep stage classification can be pruning the model to reduce the parameters for simplicity as well as applying transfer learning to mitigate the need for large training data.

REFERENCES

[1] M. Walker, *Why we sleep: Unlocking the power of sleep and dreams*. Simon and Schuster, 2017.

[2] M. W. Mahowald and C. H. Schenck, "Insights from studying human sleep disorders," *Nature*, vol. 437, no. 7063, pp. 1279–1285, 2005.

[3] F. Karimzadeh, M. Nami, and R. Boostani, "Sleep microstructure dynamics and neurocognitive performance in obstructive sleep apnea syndrome patients," *Journal of Integrative Neuroscience*, vol. 16, no. 2, pp. 127–142, 2017.

[4] A. Procházka, J. Kuchyňka, O. Vyšata *et al.*, "Sleep scoring using polysomnography data features," *Signal, Image and Video Processing*, vol. 12, no. 6, pp. 1043–1051, 2018.

[5] A. Rechtschaffen, "A manual for standardized terminology, techniques and scoring system for sleep stages in human subjects," *Brain Information Service*, 1968.

[6] A. Malhotra, M. Younes, S. T. Kuna *et al.*, "Performance of an automated polysomnography scoring system versus computer-assisted manual scoring," *Sleep*, vol. 36, no. 4, pp. 573–582, 2013.

[7] S. M. Mohammadi, S. Kouchaki, M. Ghavami *et al.*, "Improving time–frequency domain sleep EEG classification via singular spectrum analysis," *Journal of Neuroscience Methods*, vol. 273, pp. 96–106, 2016.

[8] F. Karimzadeh, R. Boostani, E. Seraj *et al.*, "A distributed classification procedure for automatic sleep stage scoring based on instantaneous electroencephalogram phase and envelope features," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 26, no. 2, pp. 362–370, 2017.

[9] R. Boostani, F. Karimzadeh, and M. Nami, "A comparative review on sleep stage classification methods in patients and healthy individuals," *Computer Methods and Programs in Biomedicine*, vol. 140, pp. 77–91, 2017.

[10] Y. Sun, B. Wang, J. Jin *et al.*, "Deep convolutional network method for automatic sleep stage classification based on neurophysiological signals," in *2018 11th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*. IEEE, 2018, pp. 1–5.

[11] E. Khalili and B. M. Asl, "Automatic sleep stage classification using temporal convolutional neural network and new data augmentation technique from raw single-channel EEG," *Computer Methods and Programs in Biomedicine*, vol. 204, p. 106063, 2021.

[12] S. Sanei and J. A. Chambers, *EEG signal processing and machine learning*. John Wiley & Sons, 2021.

[13] M. Dehghani, A. Mobaien, and R. Boostani, "A deep neural network-based transfer learning to enhance the performance and learning speed of BCI systems," *Brain-Computer Interfaces*, vol. 8, no. 1-2, pp. 14–25, 2021.

[14] E. Amirizadeh and R. Boostani, "CDEC: a constrained deep embedded clustering," *International Journal of Intelligent Computing and Cybernetics*, 2021.

[15] O. Tsinalis, P. M. Matthews, Y. Guo *et al.*, "Automatic sleep stage scoring with single-channel EEG using convolutional neural networks," *ArXiv Preprint ArXiv:1610.01683*, 2016.

[16] A. Sors, S. Bonnet, S. Mirek *et al.*, "A convolutional neural network for sleep stage scoring from raw single-channel EEG," *Biomedical Signal Processing and Control*, vol. 42, pp. 107–114, 2018.

[17] S. Hashempour, R. Boostani, M. Mohammadi *et al.*, "Continuous scoring of depression from EEG signals via a hybrid of convolutional neural networks," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 2022.

[18] H. Phan, F. Andreotti, N. Cooray *et al.*, "Joint classification and prediction CNN framework for automatic sleep stage classification," *IEEE Transactions on Biomedical Engineering*, vol. 66, no. 5, pp. 1285–1296, 2018.

[19] S. Afshar, R. Boostani, and S. Sanei, "A combinatorial deep learning structure for precise depth of anesthesia estimation from EEG signals," *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 9, pp. 3408–3415, 2021.

[20] E. Bresch, U. Großekathöfer, and G. Garcia-Molina, "Recurrent deep neural networks for real-time sleep stage classification from single channel EEG," *Frontiers in Computational Neuroscience*, vol. 12, p. 85, 2018.

[21] N. Michielli, U. R. Acharya, and F. Molinari, "Cascaded LSTM recurrent neural network for automated sleep stage classification using single-channel EEG signals," *Computers in Biology and Medicine*, vol. 106, pp. 71–81, 2019.

[22] X. Zhang, W. Kou, I. Eric *et al.*, "Sleep stage classification based on multi-level feature learning and recurrent neural networks via wearable device," *Computers in Biology and Medicine*, vol. 103, pp. 71–81, 2018.

[23] S. Chambon, M. N. Galtier, P. J. Arnal *et al.*, "A deep learning architecture for temporal sleep stage classification using multivariate

and multimodal time series," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 26, no. 4, pp. 758–769, 2018.

[24] Z. Cui, X. Zheng, X. Shao *et al.*, "Automatic sleep stage classification based on convolutional neural network and fine-grained segments," *Complexity*, vol. 2018, 2018.

[25] F. Andreotti, H. Phan, N. Cooray *et al.*, "Multichannel sleep stage classification and transfer learning using convolutional neural networks," in *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2018, pp. 171–174.

[26] J. Jin, Y. Miao, I. Daly *et al.*, "Correlation-based channel selection and regularized feature optimization for mi-based bci," *Neural Networks*, vol. 118, pp. 262–270, 2019.

[27] J. Jin, R. Xiao, I. Daly *et al.*, "Internal feature selection method of CSP based on L1-norm and Dempster–Shafer theory," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 11, pp. 4814–4825, 2020.

[28] M. Xu, X. Xiao, Y. Wang *et al.*, "A brain–computer interface based on miniature-event-related potentials induced by very small lateral visual stimuli," *IEEE Transactions on Biomedical Engineering*, vol. 65, no. 5, pp. 1166–1175, 2018.

[29] J. Jin, Z. Wang, R. Xu *et al.*, "Robust similarity measurement based on a novel time filter for SSVEPs detection," *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

[30] J. Zhang, R. Yao, W. Ge *et al.*, "Orthogonal convolutional neural networks for automatic sleep stage classification based on single-channel EEG," *Computer Methods and Programs in Biomedicine*, vol. 183, p. 105089, 2020.

[31] S. Kouchaki, K. Eftaxias, and S. Sanei, "An adaptive filtering approach using supervised ssa for identification of sleep stages from EEG," *Frontiers in Biomedical Technologies*, vol. 1, no. 4, pp. 233–239, 2014.

[32] H. Phan, F. Andreotti, N. Cooray *et al.*, "SeqSleepNet: end-to-end hierarchical recurrent neural network for sequence-to-sequence automatic sleep staging," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 27, no. 3, pp. 400–410, 2019.

[33] T. Zhu, W. Luo, and F. Yu, "Convolution-and attention-based neural network for automated sleep stage classification," *International Journal of Environmental Research and Public Health*, vol. 17, no. 11, p. 4152, 2020.

[34] A. N. Olesen, P. Jennum, P. Peppard *et al.*, "Deep residual networks for automatic sleep stage classification of raw polysomnographic waveforms," in *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2018, pp. 1–4.

[35] F. Andreotti, H. Phan, and M. De Vos, "Visualising convolutional neural network decisions in automatic sleep scoring," in *CEUR Workshop Proceedings*. CEUR Workshop Proceedings, 2018, pp. 70–81.

[36] C. Szegedy, W. Liu, Y. Jia *et al.*, "Going deeper with convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1–9.

[37] H. Dong, A. Supratak, W. Pan *et al.*, "Mixed neural network approach for temporal sleep stage classification," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 26, no. 2, pp. 324–333, 2017.

[38] A. Supratak, H. Dong, C. Wu *et al.*, "DeepSleepNet: A model for automatic sleep stage scoring based on raw single-channel EEG," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 11, pp. 1998–2008, 2017.

[39] A. L. Goldberger, L. A. Amaral, L. Glass *et al.*, "PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, pp. e215–e220, 2000.

[40] E. Eldele, Z. Chen, C. Liu *et al.*, "An attention-based deep learning approach for sleep stage classification with single-channel EEG," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 29, pp. 809–818, 2021.

[41] G.-Q. Zhang, L. Cui, R. Mueller *et al.*, "The national sleep research resource: towards a sleep data commons," *Journal of the American Medical Informatics Association*, vol. 25, no. 10, pp. 1351–1358, 2018.

[42] S. F. Quan, B. V. Howard, C. Iber *et al.*, "The sleep heart health study: design, rationale, and methods," *Sleep*, vol. 20, no. 12, pp. 1077–1085, 1997.

[43] P. Fonseca, N. Den Teuling, X. Long *et al.*, "Cardiorespiratory sleep stage detection using conditional random fields," *IEEE Journal of Biomedical and Health Informatics*, vol. 21, no. 4, pp. 956–966, 2016.

[44] C. Iber, "The AASM manual for the scoring of sleep and associated events: Rules," *Terminology and Technical Specification*, 2007.

[45] M. X. Cohen, *Analyzing neural time series data: theory and practice*. MIT press, 2014.

[46] S. Kouchaki, S. Sanei, E. L. Arbon *et al.*, "Tensor based singular spectrum analysis for automatic scoring of sleep EEG," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 23, no. 1, pp. 1–9, 2014.

[47] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International Conference on Machine Learning*. PMLR, 2015, pp. 448–456.

[48] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," *ArXiv Preprint ArXiv:1803.01271*, 2018.

[49] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," *ArXiv Preprint ArXiv:1511.07122*, 2015.

[50] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of The IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.

[51] J. Cohen, "A coefficient of agreement for nominal scales," *Educational and Psychological Measurement*, vol. 20, no. 1, pp. 37–46, 1960.

[52] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *ArXiv Preprint ArXiv:1412.6980*, 2014.

[53] S. Mousavi, F. Afghah, and U. R. Acharya, "SleepEEGNet: Automated sleep stage scoring with sequence to sequence deep learning approach," *PloS One*, vol. 14, no. 5, p. e0216456, 2019.

[54] M. Friedman, "The use of ranks to avoid the assumption of normality implicit in the analysis of variance," *Journal of the American Statistical Association*, vol. 32, no. 200, pp. 675–701, 1937.