Journal of Engineering and Applied Sciences 14 (16): 5906-5916, 2019
ISSN: 1816-949X

ROM-based Inference Method Built on Deep Learning for Sleep Stage Classification

Mohamed H. AlMeer, Hanadi Hassen and Naveed Nawaz
Department of Computer Science and Engineering, Qatar University,
P.O. Box 2713, Doha, Qatar
almeer@qu.edu.qa, nnawaz@qu.edu.qa

Abstract: We used a classical Deep Feed Forward Neural Network (DFFNN) for an automatic sleep stage scoring based on a single-channel EEG signal. We used an open-available dataset, randomly selecting one healthy young adult for both training (≈5%) and evaluation (≈95%). We also, augmented the validation by using 5-fold cross validations for the result comparisons. We introduced a new method for inferring the trained network based on a ROM module (memory concept), so, it would be faster than directly inferring the trained Deep Neural Network (DNN). The ROM content is filled after the DNN network is trained by the training set and inferred using the testing set. An accuracy of 97% was achieved in inferring the test datasets using ROM when compared to the classic trained DNN inference process.

Key words: PSG, sleep stages, deep neural networks, DNN, FFNN, ROM content

INTRODUCTION

Sleep is essential to human health and when a person undergoes a reduced sleep period, abnormal sleep patterns or suffers a sleep illness such as desynchronized circadian rhythms, he will face cognitive, somatic and cognitive symptoms (Medic *et al*., 2017). There exists a relation between abnormal sleep patterns and neuro diseases (Abbott and Videnovic, 2016). Recent research shows that the detection of all sleep abnormalities such as circadian disruption, could be a clear indicator of a risk potential for the early stages of neurodegenerative illnesses such as alzheimer and parkinson diseases (Wulff *et al*., 2010). Sleep experts judge sleep quality using electrical sensors attached to the different parts of a person's body. Those signals comprise an Electroencephalogram (EEG), an Electrooculogram (EOG) an Electromyogram (EMG) and an Electrocardiogram (ECG). A Polysomnogram (PSG) is the name for the entire set of those related signals recorded through these sensors.

The PSG data segments all recordings into 30 sec epochs and the sleep stage experts assign different stages according to Rechtschaffen and Kale's (R&K) (Hori *et al*., 2001), sleep manual as well as the American Academy of Sleep Medicine (AASM) (Berry *et al*., 2012). The process is a time-consuming and labor-intensive full-manual approach with multiple sensors having a 100 Hz sampling rate responsible for increasing the amount of data collected. Human experts doing the manual scoring demand specialized training which makes them expensive to hire. Additionally, the rating quality depends on the rater's experience and the accuracy is <90% in most cases (Rosenberg and Van Hout, 2013).

According to the R&K rules in each 30 sec epoch, the sleep EEG signals were annotated as belonging to one of 5 stages: WA, NREM1 (N1), NREM2 (N2), NREM3 (N3) and NREM4 (N4; or SWS) and REM. Not every epoch is a 100% fit in a specific stage. A neurologist specialized in sleep analysis is assigned to assess these stages. In detail, the Wake stage (WA) is considered the normal body function stage. The NREM1 is believed to be the beginning of sleep where the eyes are closed. While in the NREM2 stage, the light sleep stage both heart rate and body temperature are slowed down. In NREM3, the person falls into the deep sleep stage, during which the body repairs and regrows tissues. Finally, the REM sleep period called the dream period is characterized by faster brain activity, breathing and heart rate. In this study, we altered the Rechtschaffen and Kales sleep staging criteria (Hori *et al*., 2001) merging the criteria for N1 and N2 into N1 and merging N3 and N4 into a single stage (N2). The WA and REM stages are unchanged.

Several studies attempting to develop automated scoring methods for sleep stages based on multiple biosignals (EOG, EEG and EMG) have emerged recently. They incorporate methods that extract the frequency-domain, time-domain or frequency-time-domain

Corresponding Author: Mohamed H. AlMeer, Department of Computer Science and Engineering, Qatar University, P.O. Box 2713, Doha, Qatar, almeer@qu.edu.qa

features from each recorded epoch. In many of the studies using such multiple signals, the different features concatenate into a single feature vector composing the training features of the epochs (Lajnef *et al*., 2015; Huang *et al*., 2014 and Gunes *et al*., 2010).

In this study, we present a novel, classic DFFNN framework for automatic sleep stage scoring using a single channel of EEG and then we evaluate it. The single EEG channel within the PSG signals and their spectral components for the estimation of sleep features is solely used. Four normalized power frequency spectrums are extracted from the 500-temporal data (5 sec long acquired EEG signal), constituting both training and testing epochs.

Most researches have used the classical method of training a single or a mix of the following deep networks (feedforward, convolutional neural network, recurrent neural network or long-short-term memories) or a combination for basic training and inference operations. However, instead of inferring the same-trained network, we will alternatively accomplish the inference of a full-trained DNN for sleep stage classification, using a concept of basic storage memory (or simply a ROM). The content of every location within the ROM is labeled with the corresponding sleep stage class but its indices are identified from a prior training process acting on the proposed network.

We compared the performance of the built neural network model with that of the ROM model. We achieved a state of-the-art accuracy of (81and 97% resemblance) but had some difficulty comparing the results to other studies, since, to our knowledge there was absolutely no such previous usage of this new inference.

PSG data set: The data used to train, test and evaluate the model is a publicly available sleep-EDF database from the physionet repository. All the subject's signals are whole-night polysomnographic sleep recordings embedding EEG_Fpz-Cz and EEG_Pz-Oz electrodes, EOG_horizontal and a submental EMG chin signal besides other signals related to breathing rate, oxygen concentration and body movements. The PSG recordings for the PSG_SC_4002E subject in particular is obtained from a healthy Caucasian male volunteer taking no medication this subject was randomly selected for the research. Additionally, it is worth noting that all signals were technically digitized using a 100 Hz sampling rate.

Although, the original PSG-EEG data was divided into 30 sec epochs adequate for offline analysis, we have rearranged it to 5 sec epochs instead. This increased the number of features generated x6 while preserving the feature representation of the hypnogram's 5 stages as will be shown in the results study. The EEG data collected consists of 16,947 sets each 500 samples long (8,503,974 samples stored in a 132,908 MB file). Randomly selected sets from the PSG_SC_4002E subject were used to form the training batches which are composed of 4.7% of the total epochs (800 epochs) while the remaining constitutes the test set. Table 1 and 2 presents a summary of the data for this subject. The record was acquired for the whole day for both the WA and other sleep stages. Figure 3 for the training to testing to validation ratios.

Table 1: Sleep stage EEG confusion matrix for Deep learning 4J Java Model

	Prediction			
Sleep stages	W	NREM 1-2	NREM 3-4	REM
Original				
W	**9685**	499	534	592
NREM 1-2	37	**1631**	466	458
NREM 3-4	25	164	**1591**	2
REM	77	320	42	**851**

Bold values are significant values

Table 2: Sleep stage epoch distribution with accuracies for original data, DL4J model and ROM model

Sleep stages	No. of epochs	Percentage	DL4 stage acc. (%)	ROM stage acc. (%)	ROM/ DL4J(%)
W	11.310	66.6	85.6	90.6	105.8
NREM 1-2	2.592	15.3	62.9	47.9	76.1
NREM 3-4	1.782	10.5	89.2	83.1	93.2
REM	1.290	7.6	65.9	41.3	62.6
Total	16.974	100.0	81	79.6	98.3

System description: In this study, an overall system architecture is presented in detail. The system was divided into a training module and a deployment module. The system architecture is shown in Fig. 1.

DNN training module: The objective of the model-training module is to find an accurate prediction of a DNN that can take the EEG signal as input and intelligently generate a sequence of the sleep stages using one stage label assigned to each 5 sec epoch. To generate such a prediction algorithm we rely on the model-training module to extract features from the EEG data and then find classification algorithms to identify the best feature and algorithm configuration. Model training is described (Fig.2 and 3).

Independent inference modules (non-DNN or ROM-based module): After the same pattern goes through a feature extraction process, it is mapped to a binary discretized integer or index known as a memory address. Those addresses are applied to a memory module where classification labels are stored representing the sleep stages (numbered between 0-3). When the deployment process finishes a direct substitution in the memory module produces a direct prediction. The inference model is described in the study.

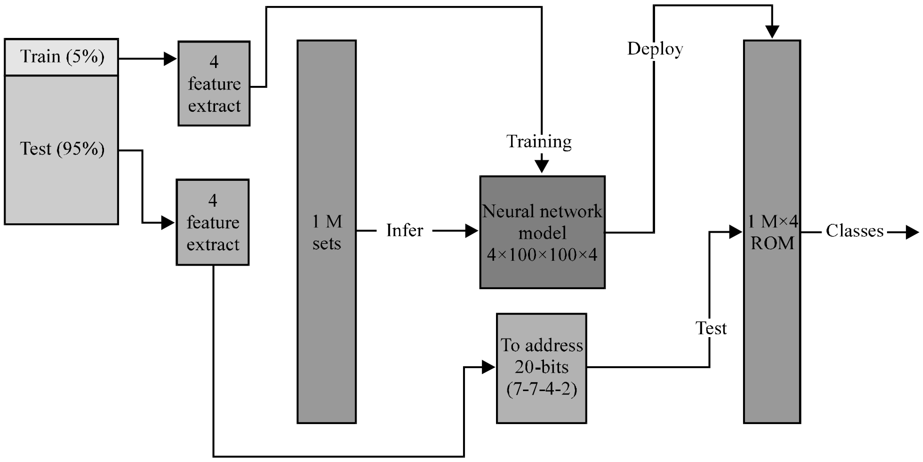

Fig. 1: Overview of the proposed approach

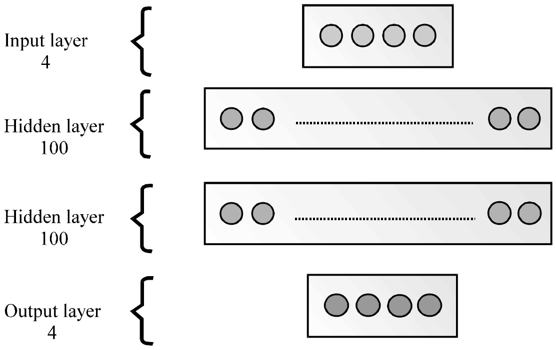

Fig. 2: Illustration of the DNN architecture

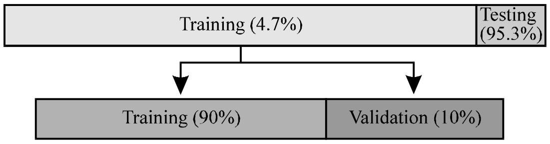

Fig. 3: The allocation of the EEG data used for training and testing the proposed algorithm

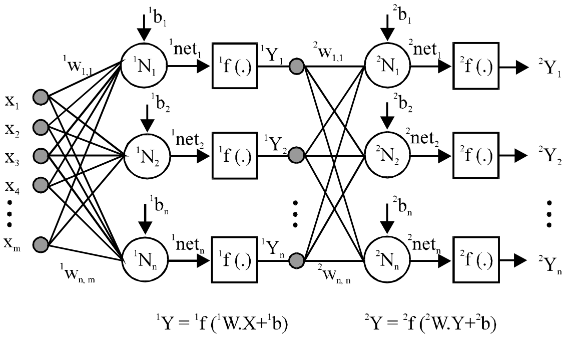

Fig. 4: Multilayer perceptron network (Siddique and Adeli, 2013)

Deep Feed Forward Neural Network (DFFNN): Artificial neural networks inspired by biological neurons to solve some prediction in many classification and recognition application such as for instance, vision, voice and natural language. With their recent successor, the Deep Neural Networks or (DNN) they have achieved close to 100% success in different pattern recognition fields in recent years.

DNN has an architecture of multiple layers, input layer with a descriptor X_1, L hidden layers and an output layer to enforce a prediction. Multi-layer feed forward neural networks consist of neurons into layers. Neurons on all layers are fully connected to all neurons to adjacent layer. In each layer all neurons use the same activation function. The input to neurons of the other layers is the output (activation) of the previous layer's neurons, except for the input layer.

Figure 4 Siddique and Adeli (2013) shows a multilayer feedforward neural network. The input this network is shown below with its weights matrix and bias vectors. Figure 2 shows the multilayer neural network in which every layer contains a weight matrix W beside a bias vector b beside the output vector y as shown in Eq. 1:

$$x = \begin{bmatrix} x1 \\ x2 \\ .. \\ .. \\ .. \\ xn \end{bmatrix}, \quad W = \begin{bmatrix} w1,1 & w1,2 & ..... & w1,m \\ w2,1 & w2,2 & ..... & w2,m \\ & .. & & \\ & .. & & \\ & .. & & \\ wn,1 & wn,2 & ..... & wn,m \end{bmatrix}, \quad b = \begin{bmatrix} b1 \\ b2 \\ .. \\ .. \\ .. \\ bn \end{bmatrix} \tag{1}$$

Each of the m components of the input vector x = {x1, x2, ..., xm} feeds forward to the n neurons. The first hidden layer generates an output as ${}^1Y = {}^1f\,({}^1W \bullet X + {}^1b)$ and the final output of the network is given by ${}^2Y = {}^2f\,({}^2W \bullet {}^1Y + {}^2b)$. It passes through the activation function that could be a sigmoidal, a tan sigmoidal (TANH), a hyperbolic tangent function (ReLU), a Leaky ReLU or a Softmax all defined as shown:

$$\text{Sigmoida,}\quad f(x) = \frac{1}{1+e^{x}}$$

$$\text{TANH}\qquad f(x) = \frac{1-e^{-x}}{1+e^{-x}}$$

$$\text{ReLU}\quad f(x) = \max(0, x)$$

$$\text{LeakyReLU}\quad f(x) = ax, x<0,, = 0, x \geq 0$$

$$\text{Soft max}\quad \sigma(z) = \frac{e^{-zj}}{\sum_{1}^{K} e^{zk}}\ \text{for } j = 1, \ldots, K$$

The output of these functions is linearly combined with weights into a network output f (x). The strategy of how this network processes information is deeply dependent on its building architecture and the number of neurons as well as the correct choice of the transfer functions and their diversities among layers has the biggest impact for training.

MATERIALS AND METHODS

In this study, we present a novel approach to automatically inferring a trained DNN to detect the WAKE, NREM1-2, NREM3-4 and REM of sleep classes within a single EEG record. A Feed Forward Deep Neural Network (FFDNN) model is trained with the 4 sets and inferred using a ROM module. Figure 2 shows a graphical representation of the approach.

Pre-processing: No preprocessing is applied here except for the manual selection of trained epochs to ensure they contained no deformed signals or severe noise, a precaution to ensure correct training. Although, the generic data presented in the sleep-EDF database has wake and REM stages in more length than other classes, the current study used the complete number of sets without trimming.

Training pattern matrix construction: The classification of sleep stages was based on time segments 5 sec long and the neurologist specified the target classes. The frequency content of PSG-EEG channels advised by the specific literature was adopted for this current study and used for classification (Hsu *et al*., 2013). The physiological nature of the signals dictates the selection of those frequencies and they covered a range from (0.5, 20) Hz. Specifically, the Δ (0.5-4 Hz) θ, (4-8 Hz), α (8-12 Hz) and σ (12-20 Hz) bands were taken. The normalized image of the 4-frequency power density is utilized as feature vectors.

To extract those frequency-domain features successfully, we first segment each 5 sec epoch into 500-long readings of the temporal subpatterns. Then, we estimate the power spectral density of each subepoch, resulting in 250 frequency bins covering the range from 0-50 Hz each 5 bins representing 1 Hz. We then normalize these power estimates over a total power range of 0-20 Hz. The next formula shows the relative spectral feature FS of a signal segment which the discrete fourier transform evaluates, resulting in a relative power in the specified frequency band (fc1 and 2) by relation:

$$FS = \frac{\sum_{k\in\theta} |Y(k)|^2}{\sum_{k=0}^{\frac{N}{2}} |Y(k)|^{2'}},\quad Y(k) = \sum_{n=0}^{N-1} y(n)\, e^{-jkn\frac{2\pi}{N}} \qquad (2)$$

where, θ is the indices group for the frequency values fk = k/N fs belongs to (fc1, fc2). This will form a 2D training pattern of four columns (vectors) by Q rows where, Q denotes the number of training epochs. For every feature vector, the corresponding sleep class specified by the neurologist is appended.

ROM-based inference process: The main objective of this part of the data preparation process is to convert the 4-spectrum bands to a 20-bit integer index used to infer a memory module for class content. First, every 5 sec interval of the EEG signal (or 500 EEG readings) is extracted and labeled in accordance with the sleep states the SPG_SC_4002E subject undergoes. Second, the Fast Fourier Transform (FFT) is applied to the time series, set by set and 250 frequency vector bins are generated, covering a range of 0-50 Hz. The sampling frequency is 100 Hz with every 1 Hz represented by 5 frequency readings each 0.2 Hz. Third, the four different frequency bands identified with the notations (α, β, θ, and σ) are then computed as shown in Eq. 1. Fourth, the power densities are then normalized over the total power content from 0-20 Hz, resulting in a normalized or relative power density pattern distribution across a 2D matrix.

Figure 5 shows the four snapshots of four sets of EEG signals covering the four stages of sleep for a random subject. It shows a frequency content of 250 Hz.

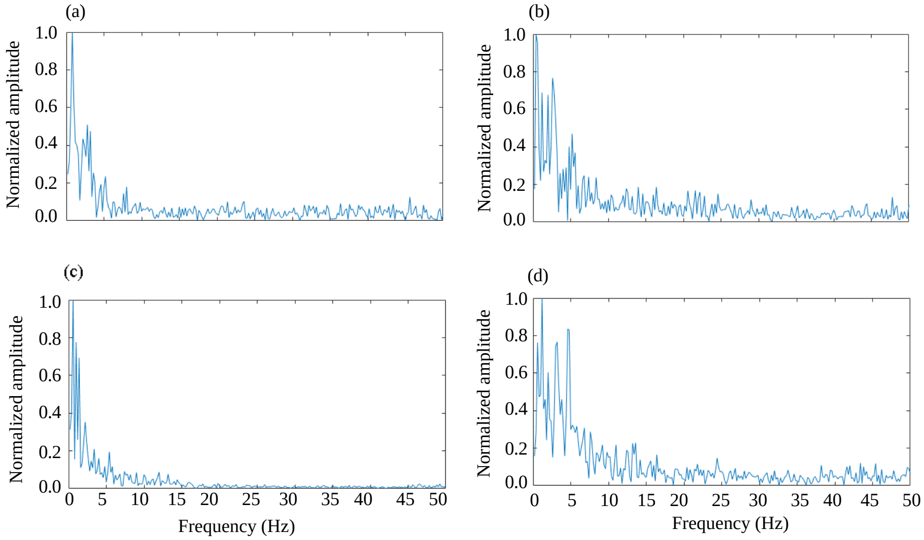

Fig. 5: Example of EEG frequency spectrum of the 4 sleep stages belonging to PSG_SC_4002E subject. Upper left, upper right, lower left and lower right will be for sleep stage "Wake", "NREM 1, 2", "NREM 3, 4" and "REM", respectively; a) Wake stage; b) NREM 1.2 stage; c) NREM 13.4 stage and d) REM stage

Table 3: Results for inference for subject PSG_CS_4002E using deep learning 4J Model

Accuracy (%)	Precession (%)	Recall (%)	F1 score (%)
81	66.5	76	70

Table 4: MATLAB classifier learner accuracies for different classification algorithms for subject PSG_SC_4002E with 16,974 sets

Classification learners	Prediction accuracy (%)
Trees	
Fine tree	84.8
Medium tree	78.1
Coarse tree	73.8
SVM (Support Vector Machine)	
SVM linear	88.2
SVM final Gaussian	89.8
SVM coarse Gaussian	88.7
Ensembles	
Ensemble boosted trees	83.9
Ensemble bagged trees	89
Subspace discriminant	66.1
Subspace kNN	75.7
RUS boosted trees	79.6
KNNs (k-Nearest Neighborhood)	
Fine	86
Medium kNN	89.2
Coarse	88.5

The spectrum plots show the density of power near 0 Hz which drops sharply as the frequency approaches 15 Hz, eventually reaching zero-close values around 250 Hz. Based on this observation, 7-bit is assigned to represent the first 2 bands, α and β, since, their content is highly dense, 4-bit and 2-bit are assigned to the remaining other 2 bands, θ and σ, respectively, since, their contents are less dense (Table 3 and 4).

Next, each of the four relative frequency power densities, F4 (σ), F3 (θ), F2 (β) and F1 (α) are converted to an integer ranging from 0 to 2^2-1, 2^4-1, 2^7-1 and 2^7-1, respectively and finally rounded. This process results in an integer from 0-3, 0-15, 0-127 and 0-127, expressing values for the frequency spectrum bands, F4, F3, F2 and F4, respectively. All 4 integers are extracted and substituted in the weighting formula below to form a single large-range integer value most suitable for a memory module index (address):

$$\text{Index} = F4*4*\left(2^{18}\right)+F3*16*\left(2^{14}\right)+F2*128*\left(2^{7}\right)+F1*128$$

where, F1-F4 represents the four normalized spectrum values. This process finally generates a memory space of 1,048,576 (1M) different storages, each holds one 4-class symbol while simply being addressed by the above index of Eq. 2. This index supposedly points to the class level. In this way, we have totally converted the inference process of a fully trained DNN to a simple memory address-content problem (Table 5).

Memory content generation process: Here, in this data preparation stage, we use the reverse engineering concept to effectively compute the corresponding features of 4 frequency bands synthetically resulting from a given index value. We start using an index from 1 to (2^{20}-1) or the so-called full-memory address range. The calculation of the 4 synthetic frequency bands, each represented by different binary bits is generated when substituted in next equation:

Table 5: Summary of sleep stages classification related work

References	Title	Dataset	Purpose of the study	Features	Classifier	Results
Jatupaiboon *et al.*(2013)	SleepNet: automated sleep staging system via. deep learning	PSGs of 10,000 patients from the Massachusetts General Hospital (MGH) sleep laboratory	Sleep stages classification	-Raw EEG features Spectrogram features Expert defined features	-CNN -RNN -RNN-CNN	Using RNN Average accuracy of 85.76% Algorithm-expert Inter-Rater Agreement (IRA) of $\kappa = 79.46\%$
Hsu *et al.* (2013)	Real-time EEG-based happiness detection system	EEG recording using 14-channels wireless EMOTIV 600 samples per participant	Happy and unhappy emotions classification	Power Spectral Density (PSD)	SVM	Accuracy of subject-dependent model 75.62% and subject-independent model 65.12%
Zhang *et al.* (2016)	A recurrent neural sleep-stage classifier using energy Features of EEG signals	Sleep-EDF database	Sleep stages classification	Energy features extracted on 30 sec epochs from EEG signals	Recurrent neural classifier	Average classification accuracy 87.2%
Tsinalis *et al.* (2016a)	Automatic sleep stage classification based on sparse deep belief net and combination of multiple classifiers	UCD database	Sleep stages classification	Deep belief network features	Combination of HMM, KNN and SVM	91.31%
Tsinalis *et al.* (2016b)	Automatic sleep stage scoring with single-channel EEG using convolutional neural networks	Sleep PSG dataset	Sleep stage scoring Neural Networks (CNNs)	Convolutional Neural Networks (CNNs)	Convolutional Neural Networks (CNNs)	Overall accuracy: 74% F1-score: 81%
Yulita *et al.* (2017)	Automatic sleep stage scoring using time-frequency analysis and stacked sparse autoencoders	Sleep PSG dataset	Sleep stage scoring	Time-frequency-based analysis using Morlet wavelets	Stacked sparse autoencoders	Overall accuracy: 78% Mean F1-score: 84%
Supratak *et al.* (2017)	Bi-directional long short-term memory using quantized data of deep belief networks for sleep stage classification	Vincent's University Hospital/University College Dublin's Sleep Apnea Database	Sleep stage scoring features	28 handcrafted Bi-directional	Quantization Recall:72.1% Long Short-Term Memory (qBi-LSTM)	Precision: 86% F-measure:75.27%
Langkvist *et al.* (2012)	Deep sleep net: A model for automatic sleep stage scoring based on raw single-channel EEG	Montreal Archive of Sleep Studies (MASS) and sleep-EDF	Sleep stage scoring	Deep learning features	CNN and BLSTM	MASS: Overall accuracy:86.2% Macro F1-score: 81.7 Sleep-EDF: Overall accuracy: 82.0% Macro F1-score: 76.9
Sors *et al.* (2018)	Sleep stage classification using unsupervised feature learning	Vincent's university hospital and university college dublin	Sleep stage classification	DBNs features	HMM	Accuracy (mean ± std): 72.2±9.7
Paisarnsrisomsuk *et al.* (2018)	A convolutional neural network for sleep stage scoring from raw single-channel EEG	Data from the Sleep Heart Health Study (SHHS) (Single Channel)	Sleep scoring	CNN	CNN	Accuracy: 0.87 kappa: 0.81
Simonyan and Zisserman (2014)	Deep sleep:convolutional neural networks for predictive modeling of human sleep time-signals	Sleep-EDF database Synthetic data	Sleep stages classification	CNN	CNN	Overall classification accuracy of 81%
Prochazka *et al.* (2018)	Multi-class sleep stage analysis and adaptive pattern recognition	Database of 184 polysomnography overnight observations	Sleep stages classification	5 energy features and 12 energy features	Bayesian neural network classifier	Mean classification accuracy on single channel: 88.7% On multimodal channels: 98.6%
Stephansen *et al.* 2017)	The use of neural networks in the analysis of sleep stages and the diagnosis of narcolepsy				Convolutional (CNN) and Recurrent (RNN) Neural Networks	

$$\text{Synthetic band\_4} = \text{shift\_right}\left(\text{Index}\,\&\left(0\times\text{C0000}\right),\ \text{18-bit}\right)/4.0$$

$$\text{Synthetic band\_3} = \text{shift\_right}\left(\text{Index}\,\&\left(0\times\text{3C000}\right),\ \text{14-bit}\right)/16.0$$

$$\text{Synthetic band\_2} = \text{shift\_right}\left(\text{Index}\,\&\left(0\times\text{03F80}\right),\ \text{7-bit}\right)/128.0$$

$$\text{Synthetic band\_1} = \text{Index}\,\&\left(0\times\text{0007F}\right)/128.0$$

This should generate a 2D matrix of four values representing a normalized synthetic power spectrum, repeated over the full range of a 20-bit binary address or 1,048,576 (Fig. 6 and 7).

This 1M-by-4 memory matrix is prepared for inference (or in other words for testing the network). This network is supposedly the same network which is trained using the subject's training patterns prepared earlier as explained in study. The inferred or predicted classification classes from the previous operation are used as a content to fill the memory module using next simple equation:

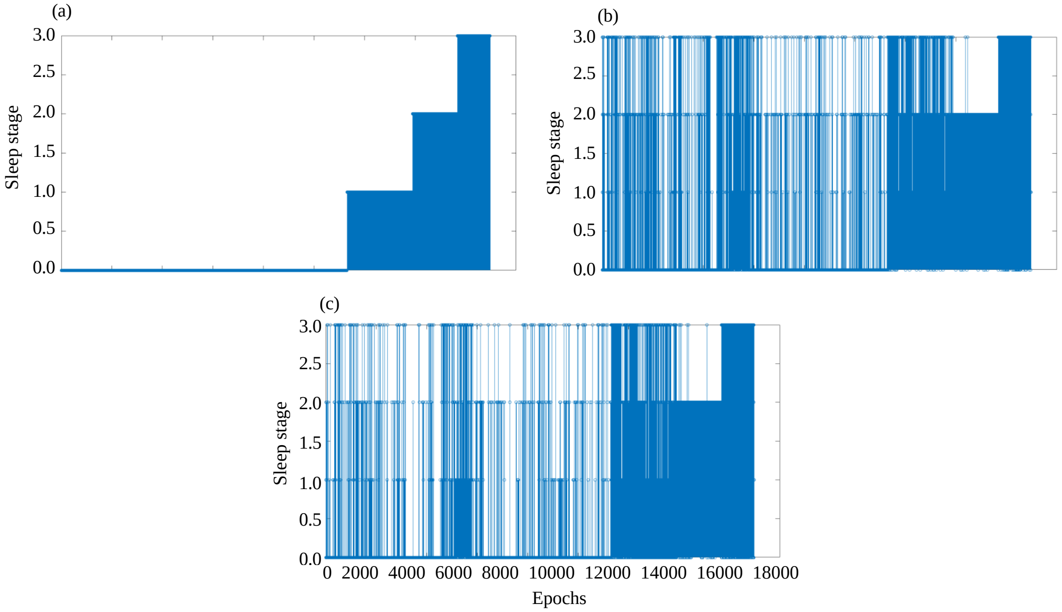

Fig. 6: a) Target sleep classes forming desired DNN output; b) Resulting classes emerging from DNN-trained Deeplearning4J Java inference Model; c) Resulting classes emerging from DNN-trained ROM-based inference model; a) subject SC-4002 manually scored hyphogram-ordered-100% accuracy (Given); b) subject SC-4002 estimated using Deeplearning4j java-81% accuracy and c) Subject SC-4002 estimated-using ROM-based inference-MATLAB-79.9% accuracy

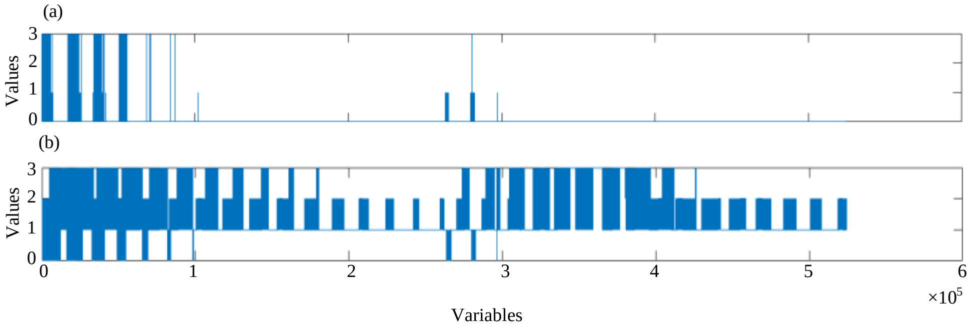

Fig. 7: Comparison between the ROM content with the training set filled (upper image) and when filled after the network is inferred with total range of 524,287 sets (lower image)

Memory [Index] = prediction_class

This should prepare the inference operation separately from the DNN used for training and enable a faster inference operation. Figure 7 shows ROM content when training patterns are firstly mapped in (800 sets) and after it is fully mapped with the generated synthetically from the inference process (524287 sets).

DNN architecture and training parameters: All kernels or parameters reported in this study are settled after some trial-and-error attempts. Figure 2 illustrates the graphical representation of the DNN structure with normalized 4 frequency bands chosen as input samples. This proposed DNN architecture encompasses one input layer, followed by two fully connected hidden layers and finishes with 1 output layer. The activation functions used in the input and the two hidden layers are the tanh function while the one used in the output layer is the softmax function. The weight initialization algorithm used is Xavier while the regularization used to overcome overfitting was 12 (1e-4). The loss function used in the output layer was chosen to be negative log likelihood.

A conventional backpropagation training algorithm is employed to train the DNN with a batch size of 800. In the training algorithm, the batch size is used to denote the number of epochs or the number of signals used for each training update of the network's parameters. Here, the batch size is taken to be equal to the number of sets used in training. Back propagation calculates the gradient of the loss function with respect to the weights. Error signals emerging from each pass are passed backward through the network during training to update the weights. The batch size of 100 was used in this research. The learning rate was commonly tested from 10^{-1}-10^{-3} for the Adam optimizer. The implementation is based on the Deeplearning4j Java framework, especially designed for creating, testing and adjusting hyperparameters of different deep neural architectures.

RESULTS AND DISCUSSION

The proposed DNN is implemented on a ThinkPad Laptop Intel® Core™ i7-5600U CPU@ 2.6 GHz with 8 GB RAM using the MATLAB programming software to simulate the ROM inference model and Deeplearning4j Java framework model. It took about 15 min to complete all epochs of training with 30,000 iterations.

The confusion matrix across all stages is represented in Table 1. We found that 85% of the EEG signals are normal and correctly classified as wake signals. Small percentages of 4.4, 4.7 and 5.2% of the normal EEG signals are wrongly classified as N1-2, N3-4 and REM, respectively, 62% of the EEG signals are correctly classified as NREM1-2 signals, 1.4% of the EEG signals were wrongly classified as WA, NREM3-4 (18%) and REM (17.6%), 89% of the EEG signals are correctly classified as NREM3-4 with 1.4% wrongly classified as WA, NREM1-2 (9.2%) and REM (0.1%). The performance (Accuracy, precision and F1-score) of the proposed model can be seen in Table 3 for Subject PSG_SC_4002E when the Deeplearning4J Model is used for tested inference.

Moreover, when the same test patterns undergo the proposed ROM inference model a close resemblance in results is reached: 79.6% from 81-98.3% of the expected accuracy. Table 2 shows the accuracies for the Deeplearning4J platform and ROM Model as well as the relative resemblance rate. Figure 6 lists the results of testing the two models, the DNN and ROM-based using 95.3% of the subject sets. Clearly, the resemblance between the responses of the two models is imminent and reaches 98%.

Table 4 shows the cross-validation done using the MATLAB classification learner tool with 5-folds validation when the full-set of the 16,974 subject sets is tested. Different classifiers used included trees, ensembles Support Vector Machine (SVM) and k-Nearest Neighbors (KNN). The best accuracy reached was for SVM (Final grain Gaussian) at 89.8% besides the kNN (medium grained) algorithm for 89.2%.

Table 4 represents a summary of the conducted studies in an automated detection of sleep stages during the last decade. According to Hillman *et al*. (2006), around 50-70 million adults in the United States are affected by sleep disorders such as parasomnias, disorders and hypersomnia. The overnight PSG is used to diagnose sleep disorders including brain monitoring using EEG. Trained technologists conventionally conduct the PSG analysis. Recently, the PSG analysis has been automated with the help of machine-learning algorithms trained using physiological datasets. The EEG signal is not limited to the sleep study, it is also applied in many other studies such as happy and unhappy emotions studies as by Hsu *et al*. (2013) where they first registered the EEG signal using 14-channels wireless EMOTIV. After filtering the signal they decompose it, using a window of 1 sec to 5 frequency bands resulting in 70 features normalized by scaling between 0 and 1. Then, the SVM is used to classify the happy and unhappy emotions.

One of the tools deployed for the automatic annotation of sleep staging is sleepnet proposed by Biswal *et al*. (2017). They apply a deep Recurrent Neural Network (RNN) for automatic sleep staging annotation and achieved a performance comparable to the human level. Three types of EEG features were used: the raw waveform, the spectrogram and expert-defined features. To evaluate the system, conventional classifiers such as logistic regression, tree boosting and multilayer perceptron were compared to deep learning-based classifiers an RNN and a Convolutional Neural Network (CNN) (Stephansen *et al*., 2017). The RNN achieved the best accuracy for sleep stages classification. Another research applying an RNN for sleep stages classification is proposed by Hsu *et al*. (2013). They train the RNN using energy features that are calculated by taking the summation of the magnitudes of the squared components of the signal.

Instead of using handcrafted features, the deep learning is recently employed as a successful unsupervised feature learning method. Zhang *et al*. (2016) the EEG, EOG and EMG signals are filtered then divided into segments of 30 sec with zero overlaps. These signals are then passed to a three layer sparse deep belief network for features extraction. The classification of the extracted features was achieved by combining multiple classifiers in particular, the Hidden Markov Model

(HMM), SVM and kNN. The combination of these classifiers with the application of classification entropy voting resulted in a 91% accuracy.

Another research employing Deep Belief Nets (DBNs) for features extraction was presented by Paisarnsrisomsuk *et al.* The DBNs were trained using a dataset of EEG, EOG and EMG signals. All signals were first preprocessed by notch filtering then three different experiments were run using these filtered signals. In the first experiment (feat-GOHMM experiment) 28 handcrafted features were used to train a Gaussian Observation Hidden Markov Model (GOHMM). The feature set was reduced using a Principle Component Analysis (PCA). In the second experiment (feat-DBN experiment), they trained the DBNs using the 28 handcrafted features then the HMM was used on top of the DBNs. In the last experiment (raw-DBN), the DBN used raw data instead of handcrafted features then the HMM was built on top of it. The comparative analysis between the three experiments showed that the feat-DBN performs better than the raw-DBN and feat-GOHMM.

The quantization Bidirectional Long Short-Term Memory (qBi-LSTM) was applied by Supratak *et al.* (2017) for the classification of sleep stages. After filtering the EEG, EOG and EMG signals, 28 handcrafted features were extracted and normalized. In addition to the qBi-LSTM, the reasearchers applied BLSTM and a DBN. The performance of the qBi-LSTM outperformed the other two models.

A deep learning approach for both the feature extraction and classification of sleep stages was proposed by Langkvist *et al.* (2012). They employed 2 CNNs for extracting time-invariant features from raw EEG. Each CNN consists of 4 convolutional layers and 2 max-pooling layers. The temporal information of the sleep stages transitions was encoded using Bi-LSTM. They evaluated their model using two datasets: the first was Montreal Archive of Sleep Studies (MASS) and the second was sleep-EDF.

Recently, the CNNs successful in many computer vision tasks have been applied to the task of sleep stage classification. A CNN with two convolutional and pooling layers two fully connected layers and a softmax layer was implemented by Tsinalis *et al.* (2016 a, b) to achieve sleep stage scoring using a PSG dataset. They compared CNN results with their previous study (Tsinalis *et al.*, 2016a, b) in which they performed a time-frequency-based analysis by using morlet wavelets for feature extraction. The classification was achieved using a special type of neural network, stacked Sparse Auto-Encoders (SAE). Their results showed that the SAE Model outperforms the end-to-end CNN training using a PSG dataset.

Another application of CNNs on sleep staging is proposed by Sors *et al.* (2018). Using single-channel EEG raw signals, the CNNs were used to learn features and perform the classification of sleep stages. The proposed network consists of 12 convolutional layers followed by two fully connected layers trained on Sleep Heart Health Study (SHHS) data. The proposed network achieved an accuracy of 0.87 and Cohen kappa of 0.81.

Instead of using only single-channel EEG, the researhers by Paisarnsrisomsuk *et al.* proposed the use of a CNN with multiple-channel signals. The CNN model they built is based on a VGG network proposed by Simonyan and Zisserman (2014). It consists of 17 convolutional layers separated by max-pooling layers followed by two fully connected layers for the classification of the features extracted by the previous convolutional structure. The data used is sleep-EDF database and synthetic data. Compared to, Rosenberg and Van Hout (2013) which employs the CNN with single-channel EEG this multiple-channel approach with a much deeper network achieved a better performance even after testing the network with single-channel EGG data.

A Bayesian neural network classifier was implemented by Prochazka *et al.* to achieve sleep stages classification. The proposed 2-layer network was trained on a database of 184 PSG. The preprocessing of the signals included noise filtering and artifact removal. Two types of features were extracted: in the first, 5 energy features were extracted from the single EEG channel and in the second 12 energy features were extracted from 3 multimodal (EEG, EOG, breathing (Flow)) channels. The classification accuracy on the multimodal channels is higher than that on the single channel.

In this research, the main novelty is the implementation of a DNN to automate the classification of combined four sleep stages using a single EEG channel into wake, NREM1 and NREM2, NREM3 and NREM4 and REM stages. A classic DFFNN is adopted to train 4.7% of a subject set. By using the techniques of reverse engineering, 524,287 of testing sets are synthetically generated for inferring the trained network. When done, the corresponding class predicted is stored in a ROM module. This module is separately used to infer the DNN for the remaining 95.3% of the subject's data set.

More speed is expected when inferring a DNN using this model than with the network used for training, therefore, we expect this method to be more suitable for real-time applications. We estimate that for embedded systems characterized for small memory capacity and low power computation this technique would be efficient in implementing extrahuge DNNs.

CONCLUSION

The proposed model is introducing a solution for inferring a fully trained network for sleep stage classification from a single-source brain EEG signal by using a ROM model instead of the trained network. Features has been extracted from the 500-sample epoch set when the FFT operation is applied, resulting in a 4-spectrum power density for the α, β, θ and σ regions. An average accuracy of 81% for the Deeplearning4J Model and 79.6% for the deployment ROM model was reached when test sets were performed even though a reduction of the input features was 500-to-4. The trained pattern constituted of 4.7% (800 sets) of the PSG_SC_4002E subject undergoing this experiment while the test set was 95.3% (16,174 sets). The new ROM model used a trained set to fill 524,287 classes, covering the entire range of the anticipated 4-frequency spectrum. The advantage of the proposed model presented in this study, however is the separate intake for the inference process away from the trained DNN network.

REFERENCES

Abbott, S.M. and A. Videnovic, 2016. Chronic sleep disturbance and neural injury: Links to neurodegenerative disease. Nat. Sci. Sleep, 8: 55-61.

Berry, R.B., R. Brooks, C.E. Gamaldo, S.M. Harding and C.L. Marcus *et al*., 2012. The AASM Manual for the Scoring of Sleep and Associated Events: Rules, Terminology and Technical Specifications. American Academy of Sleep Medicine, Darien, Illinois, USA.,.

Biswal, S., J. Kulas, H. Sun, B. Goparaju and M.B. Westover *et al*., 2017. SLEEPNET: Automated sleep staging system via deep learning. Mach. Learn., 1: 1-17.

Gunes, S., K. Polat and S. Yosunkaya, 2010. Efficient sleep stage recognition system based on EEG signal using K-means clustering based feature weighting. Expert Syst. Appl., 37: 7922-7928.

Hillman, D.R., A.S. Murphy, R. Antic and L. Pezzullo, 2006. The economic cost of sleep disorders. Sleep, 29: 299-305.

Hori, T., Y. Sugita, E. Koga, S. Shirakawa and K. Inoue *et al*., 2001. Proposed supplements and amendments to a manual of standardized terminology, techniques and scoring system for sleep stages of human subjects the Rechtschaffen and Kales (1968) standard. Psychiatry Clin. Neurosci., 55: 305-310.

Hsu, Y.L., Y.T. Yang, J.S. Wang and C.Y. Hsu, 2013. Automatic sleep stage recurrent neural classifier using energy features of EEG signals. Neurocomput., 104: 105-114.

Huang, C.S., C.L. Lin, L.W. Ko, S.Y. Liu and T.P. Su *et al*., 2014. Knowledge-based identification of sleep stages based on two forehead electroencephalogram channels. Front. Neurosci., 8: 1-12.

Jatupaiboon, N., S. Pan-ngum and P. Israsena, 2013. Real-time EEG-based happiness detection system. Sci. World J., 2013: 1-12.

Lajnef, T., S. Chaibi, P. Ruby, P.E. Aguera and J.B. Eichenlaub *et al*., 2015. Learning machines and sleeping brains: Automatic sleep stage classification using decision-tree multi-class support vector machines. J. Neurosci. Methods, 250: 94-105.

Langkvist, M., L. Karlsson and A. Loutfi, 2012. Sleep stage classification using unsupervised feature learning. Adv. Artif. Neural Syst., 2012: 1-10.

Medic, G., M. Wille and M.E. Hemels, 2017. Short-and long-term health consequences of sleep disruption. Nat. Sci. Sleep, 9: 151-161.

Prochazka, A., J. Kuchynka, O. Vysata, P. Cejnar and M. Valis *et al*., 2018. Multi-class sleep stage analysis and adaptive pattern recognition. Appl. Sci., 8: 1-14.

Rosenberg, R.S. and S. Van Hout, 2013. The American Academy of Sleep Medicine inter-scorer reliability program: Sleep stage scoring. J. Clin. Sleep Med., 9: 81-87.

Siddique, N. and H. Adeli, 2013. Computational Intelligence: Synergies of Fuzzy Logic, Neural Networks and Evolutionary Computing. John Wiley & Sons, Hoboken, New Jersey, USA., ISBN:9781118337844, Pages: 532.

Simonyan, K. and A. Zisserman, 2014. Very deep convolutional networks for large-scale image recognition. Proceedings of the Conference on Learning Representations (ICLR'15), May 7-9, 2015, Hilton San Diego Resort & Spa, San Diego, California, pp: 1-14.

Sors, A., S. Bonnet, S. Mirek, L. Vercueil and J.F. Payen, 2018. A convolutional neural network for sleep stage scoring from raw single-channel EEG. Biomed. Signal Process. Control, 42: 107-114.

Stephansen, J.B., A. Ambati, E.B. Leary, H.E. Moore and O. Carrillo *et al*., 2017. Neural network anlalysis of sleep stages enables efficient diagnosis of narcolepsy. Neural Evol. Comput., 1: 1-15.

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

Tsinalis, O., P.M. Matthews and Y. Guo, 2016. Automatic sleep stage scoring using time-frequency analysis and stacked sparse autoencoders. Ann. Biomed. Eng., 44: 1587-1597.

Tsinalis, O., P.M. Matthews, Y. Guo and S. Zafeiriou, 2016. Automatic sleep stage scoring with single-channel EEG using convolutional neural networks. Mach. Learn., 1: 1-12.

Wulff, K., S. Gatti, J.G. Wettstein and R.G. Foster, 2010. Sleep and circadian rhythm disruption in psychiatric and neurodegenerative disease. Nat. Rev. Neurosci., 11: 589-599.

Yulita, I.N., M.I. Fanany and A.M. Arymuthy, 2017. Bi-directional long short-term memory using quantized data of deep belief networks for sleep stage classification. Procedia Comput. Sci., 116: 530-538.

Zhang, J., Y. Wu, J. Bai and F. Chen, 2016. Automatic sleep stage classification based on sparse deep belief net and combination of multiple classifiers. Trans. Inst. Meas. Control, 38: 435-451.