A Computationally Efficient Single-Channel EEG Sleep Stage Scoring Approach using Simple Structured CNN

Hongzhe Li*

1School of Chemical Engineering, The University of New South Wales, Sydney, 2052, Australia
*Corresponding author’s e-mail: hongzhe.li@unswalumni.com

Abstract. Automatic sleep stage classification has been a hot trend since hand-crafted feature engineering is highly inefficient. However, current studies of automatic sleep stage scoring focus more on designing complex neural network structures to improve the model performance while neglecting the model efficiency. This causes both lengthy training time and highly demanding hardware are needed for model training, which is not favorable for future industrial applications. This work proves the concept that the simple model, such as a shallow Convolutional Neural Network (CNN) combining the proper data processing techniques, can achieve a comparable model performance (overall accuracy of 79.0%) to the complex model (overall accuracy of 74.9-82.0%). The designed model in this work also significantly improves the model efficiency by reducing the number of learnable parameters in the neural network. This approach provides a new insight into automatic sleep stage scoring study as well as other deep learning studies that the data processing and the model design are equally important.

1. Introduction
Sleep is one of the most vital circadian rhythms of the human body. Keep tracking the sleep rhythm, and its quality has been an inseparable part of neuroscience research and clinical diagnoses [1]. Accurately identifying the corresponding sleep stage from Polysomnography (PSG) signals, which is also commonly known as sleep stage scoring, plays a significant role in this field.

Traditionally, the scoring is done by well-trained technicians. The process is labour-intensive, time-consuming, and easily affected by subject judgment [2, 3]. Nowadays, with the exploding development of computer science, classic machine learning models such as Support Vector Machine (SVM) [4], random forest [5, 6], and Linear Discriminate Analysis (LDA) [7] have been studied. One of the major drawbacks that all these models sharing is that they all require extensive manual feature engineering, which is highly inefficient and does not work well on large datasets [8].

Currently, deep learning has surpassed the heat of machine learning models due to the accelerating development of the theory and supportive hardware. It is known that deep learning can circumvent manual feature engineering and do not restricted by the size of the dataset thusly. There are two cornerstones of the deep learning model: Convolutional Neural Network (CNN) and Recurrent Neural Networks (RNN), both are well-studied [9].

It is reported that many studies have applied CNN to sleep stage scoring [9, 10]. CNN can learn the features of 1D array, such as PSG signals in sleep stage scoring by convoaling the array with any desired filter and stride sizes. The weights within the convolution layer are shift-invariant and the flexibility and generality of the model have enhanced accordingly. Furthermore, the training difficulty of CNN is much lower than the fully-connected deep network [11]. In the meantime, RNN, such as
Long Short-Term Memory (LSTM) Neural Networks, can learn the temporal structures within the series of PSG signals [12]. However, due to its lengthy series of neural network structures, LSTM suffers from an exploding gradient, which may sabotage the prediction results and usually took much longer time to train compared to CNN. Therefore, incorporating LSTM to the deep learning model used in sleep stage scoring would elongate the training time significantly and may not necessarily improve the model accuracy [13].

Recent studies tend to develop more complex and deeper CNN and RNN, such as CNN connected to LSTM in series to enhance the model performance [10], but the complex model would escalate the training difficulty of the model, such as elongated training time and more demanding hardware.

Herein, it is hypothesized that the simple neural network model could perform just as well as a complex model. As the response to the above hypothesis, this work aims to prove that a simple CNN with the help of proper data processing methods could outperform the traditional machine learning methods and complex deep learning model investigated previously. It is expected that a satisfactory overall accuracy rate and macro-averaged F1 score with a promising computational efficiency can be achieved with the simple model presented in this work without relying on any hand-crafted features.

2. Experimental

2.1. Sleep-EDF dataset
In this work, the single-channel EEG signal (Fpz-Cz) was retrieved from the Sleep-EDF Expanded dataset [14]. The dataset consists of two subsets named as Sleep Cassette (SC) and Sleep Telemetry (ST). All subjects were retrieved from SC subset, which consists of healthy Caucasians aged 25-100. Each subject's data contains the PSG recordings with a sampling rate of 100 Hz. Each 30-second epoch of the recording is manually scored by well-trained technicians according to the R&K standards [15]. The scoring matrix has 8 categories which are: N1, N2, N3, N4, Wake, REM, Movement and Unknown. In this work, N4 stages were combined into N3 stages and Movement and Unknown stages were removed just to be consist of the previous work [3]. Other than the sleeping period, only 30 minutes both before and after the sleeping period were included to avoid introducing an excessive number of data with Wake labels.

2.2. Experimental design
A total of 45 subjects were used as a training set, and other independent 6 subjects were evenly split and used as validation and test sets, respectively. Only Fpz-Cz channel Electroencephalography (EEG) was retrieved from the dataset. For the first 20 subjects in the training set, all data were included. However, the classes were highly imbalanced with the presence of an extensive number of N2 stages. Therefore, more data with the rest of the four stages were extracted selectively from the later 25 subjects and were added to the training set to compensate for the imbalanced data distribution. This work also employed Welch's method to convert the raw EEG signal from the time domain into the frequency domain to save computational resources when deploying the neural network.

2.3. Neural network structure
The first neural network structure used in this work is a shallow CNN with only two conv layers followed by a dense layer. For CNN, the first conv layer consists of 100 output channels, 1000 filter size and a stride of 50. Utilizing a large filter size help better grasp the overall trending information from the frequency domain of the EEG signal. The second conv layer consists of 100 output channels, 2 filter sizes and a stride of 1 to have a better organization on the convoluted signal. After flattening the 3D convoluted signal into 2D, the input signal was fed into the dense layer with 5 neurons as the last layer for prediction. ReLu was selected as the activation function for both CNN and dense layer, and cross-entropy loss was used as the loss function in this work. Softmax was used as the activation function for the last layer to estimate the probability of each sleep stage.
As a comparison to the above shallow CNN structure, a Bidirectional LSTM (BiLSTM) was connected to the end of the shallow CNN in series as the second neural network separately used in this work. Using this model shows how well a more complex model is performed when feeding identical data. The input size of BiLSTM was set to 100 to be consistent with the output channel of CNN. The hidden size was set to 256 and 2 layers were used in BiLSTM. Then the output tensor from BiLSTM was flattened and fed into the identical dense layer described in the shallow CNN section.

2.4. Training parameters
Adam was chosen as the solver for weight optimization. After grid research, both weight decay and learning rate were set to 10^-4. The batch size was set to 50 for the model to comprehend the information from the training dataset fully. In this work, dropout layers with the probability of 0.5 were used in the model in both CNN and dense layers. Early stopping was another regularization technique employed in this work to alleviate the overfitting problem. The patience was set to 50, and the overall iteration number for model training was kept at 500 accordingly.

2.5. Implementation
The model was established using Pytorch (version = 1.6.0). An GTX 1080 with up to date NVIDIA driver (version = 451.67) was utilized for training the model. The training was repeated 10 times, it took 3.5 minutes to finish training the model and the model needed around 850 milliseconds for prediction on average.

2.6. Evaluation metrics
To evaluate the performance of the trained model, a classification report was generated, which summarized precision (PR), recall (RE), F1-score (F1) for individual classes by utilizing one versus rest strategy. The overall accuracy and macro-averaged F1 score were also included in the report. In this work, the model efficiency is specifically determined by two factors: The length of training time and the number of learnable parameters

3. Results and discussion

3.1. Dataset
In this work, only a single-channel EEG signal has been used to train the model while excluding other PSG signals such as Electromyography (EMG) and Electrooculography (EOG). Although introducing more signals may improve the performance since both EMG and EOG could capture the respective muscle and eyeball activity of the subject during sleeping, the overall efficiency would rather drop because more time would be needed to train the model. In the Sleep-EDF dataset, two channels of EEG signals were recorded, which are Fpz-Cz and Pz-Oz, both correspond to the front head and back head, respectively. It is reported that due to the position of Fpz-Cz electrode, EEG measured at this channel could capture the delta wave activity and sleep spindles at low-frequency range better, these features are decisive for sleep stage scoring [16]. Consequently, Fpz-Cz channel EEG was selected as the training data for the neural network model.

Table 1. The sleep label data distribution in training set before and after adding an additional 25 subjects with selective labels.

|       | Wake | N1  | N2  | N3   | REM | Total  |
|-------|------|-----|-----|------|-----|--------|
| Before| 3711 | 1475| 9061| 2600 | 3837| 20684  |
| After | 9061 | 3666| 9061| 5245 | 8615| 35648  |

Table 1 summarizes the sleep label distribution in training dataset before and after importing the data of additional 25 subjects. It is evident that the initial training dataset was highly imbalanced, this is a very common issue in sleep stage scoring and no universal solution so far [17]. Directly feeding the highly imbalanced dataset into the neural network would result the model only predicts the major
label. Imbalanced data treatment techniques are not good options in sleep stage scoring due to the fact that the raw EEG signal is quite noisy and cannot be separated without proper data treatment, the under-sampling methods would be futile in this case. On the other hand, the blindly generated samples from over-sampling methods such as Synthetic Minority Over-sampling Technique (SMOTE) could introduce more noise to the dataset, this would mislead the neural network to predict the wrong result [18].

In practice, after testing several methods, it is found that the best solution is to selectively feed more data from minor classes, which are Wake, N1, N3 and REM in this case. The more balanced dataset performed much better than the previous case. The advantage of this method is that no synthetic signal would be added, which maintains the objectivity of the training dataset. The disadvantage of this method is that more training data would be required than usual.

3.2. Model performance and efficiency in this work

Both validation and test sets kept the same data label distribution without introducing any selective data to maintain the predicted results' reliability. Table 2 has summarized the performance of the trained model on the test set. As a further comparison, table 3 had summarized the performance of a trained model when LSTM was added into the model.

The simpler model took around 3.5 minutes to train. Meanwhile, the model only used one GPU without data parallelization, which is not hardware demanding. The high efficiency comes from the relatively fewer number of learnable parameters on CNN, which is around 184K in this model. Compared to table 3, adding LSTM further upsurged the number of learnable parameters to 2.76M, which was a 15-fold increase. More parameters in the model also elongated the training time to around 7.5 minutes, which is undesirable. In the meantime, the performance of the model before and after adding LSTM made very little difference. This shows that further making the neural network structure more complex is not necessary. This unnecessity may come from that this work utilized EEG signal in the frequency domain, its temporal feature within the series might not be as characteristic as the general pattern of the signal, which could already be learned by CNN effectively.

Furthermore, from table 2, we can see that the model performed relatively well on Wake, N2, N3, and REM label but not N1 stage. For example, all Wake, N2, N3 and REMs’ F1 score are greater than 0.70, while the F1 score of N1 stage is less than 0.5. This is because N1 is still a minor class in the training datasets as well as in both validation and test datasets such that the model might not learn the features that distinguish N1 effectively. More importantly, this observation is consistent with the sleep physiology research that the N1 stage is an intrinsically ill-defined stage [3]. Therefore, the model’s inferior performance on N1 stage is tolerable.

| Stage | Predictions | Per-class metrics |
|-------|-------------|-------------------|
|       |              | Precision | Recall | F1   |
| Wake  | 389  83  10  12  31 | 0.87  0.74  0.80 |
| N1    | 15  99  25  2  85 | 0.44  0.44  0.44 |
| N2    | 18  10  931  113 113 | 0.90  0.79  0.84 |
| N3    | 8  0  43  570 1 | 0.82  0.92  0.86 |
| REM   | 15  31  27  0  437 | 0.66  0.86  0.74 |

a Training time: 3.5 minutes on average. Number of learnable parameters: ~184K. Overall accuracy: 0.79. Macro-averaged F1 score: 0.74

| Stage | Predictions | Per-class metrics |
|-------|-------------|-------------------|
|       |              | Precision | Recall | F1   |
| Wake  | 407  65  5  12  36 | 0.87  0.78  0.82 |
| N1    | 26  78  21  2  99 | 0.41  0.35  0.38 |
5

N2  17  10  941  102  115  0.91  0.79  0.85
N3  9   2  43   568  0   0.83  0.91  0.87
REM 11  34  22   0  443  0.64  0.87  0.74

Training time: 7.5 minutes on average. Number of learnable parameters: ~2.76M. Overall accuracy: 0.79.
Macro-averaged F1 score: 0.73

3.3. Model performance and efficiency comparison

Table 4 has compared the model performance in this work to the other similar research. Table 4 shows that the model in this work achieved better overall accuracy than most of the other studies while maintaining a simpler neural network structure with fewer learnable parameters. Although Supratak et al. [10] achieved a better overall accuracy, the complex neural network structure, they employed indubitably raise the barrier of training and hardware requirement. Moreover, the model developed in this work also surpassed most of the traditional machine learning models with hand-crafted feature engineering. It is known that hand-crafted feature engineering requires multiple steps of data processing and involves mathematical extraction of characteristic features such as Power Spectral Intensity and Hjorth Parameters from EEG signal [19]. In some cases, applying feature selection methods such as Minimum redundancy maximum relevance (mRMR) and Recursive Feature Elimination (RFE) is even required in hand-crafted feature engineering [6]. Executing these steps sequentially is vastly time-consuming. Under the current research trend of automatic sleep stage scoring, the manner of hand-crafted feature engineering no longer fulfills the future needs for the industrial application.

The reason why the designed simple neural network outperforms than many complex models is probably that it utilized Welch’s method to convert the EEG signal from the time domain into the frequency domain. When the EEG signal in the time domain was fed into the model, the loss of development set did not decrease at all, which means the model did not learn anything from the training dataset. This is because the raw EEG signal in time domain contains a huge amount of noise, which misleads the model during training [20]. EEG signals in the frequency domain, on the other hand, could circumvent the presence of noises and augment the effect of valuable features such as beta and delta wave activity for sleep stage scoring. In the meantime, Welch’s method is computationally efficient and easy to implement [21]. Therefore, both data pre-processing and neural network structure is equally important when designing the model, especially balancing between model accuracy and efficiency.

Table 4. The comparison of performance and model development manner on different works

| Model type                | Number of parameters | Input channel | Feature extraction | Overall accuracy |
|---------------------------|----------------------|---------------|-------------------|-----------------|
| This work                 |                      |               |                   |                 |
| Single CNN                | 184K                 | Fpz-Cz        | learned           | 79.0            |
| Two CNNs in series        | 496K                 | Fpz-Cz        | learned           | 74.9            |
| ResNet                    | 4.7M                 | Fpz-Cz+EOG    | learned           | 76.8            |
| Parallel CNNs+LSTM in series | 844K              | Fpz-Cz        | learned           | 82.0            |
| Decision trees            | N/A                  | Fpz-Cz        | Hand-crafted      | 75.0            |
| Ensemble SVM              | N/A                  | Fpz-Cz+Pz-Oz  | Hand-crafted      | 78.9            |
| kNN                       | N/A                  | Fpz-Cz+Pz-Oz  | Hand-crafted      | 80.0            |

3.4. Future work

There are limitations in this work that leave several possible future research directions to explore. Firstly, this work only investigated single-channel EEG as the training dataset while excluding other PSG signals. Studies have reported the benefits of combining different signals to have a multimodal input and allow the neural network to learn more extensive features from different signals [3]. Therefore, future work should focus on how to introduce multimodal input while maintaining high efficiency.
Secondly, as mentioned previously, the current strategy of alleviating an imbalanced class problem is to import more data selectively with minor labels. This strategy demands more training data. Thus, future work should focus on training the neural network with a limited amount of data while no compromising on overall model performance.

4. Conclusions
In conclusion, this work proves that using a simple neural network structure, a comparable performance to other complex models for automatic sleep stage scoring could be achieved with a significant improvement in model efficiency by selecting an appropriate data pre-processing method. This approach could be generalized to other deep learning research as well; researchers should not only concentrate on designing complex neural network structure, but also improving the model efficiency by considering proper data pre-processing method without any manual feature extractors. Future work should focus on investigating multimodal input and considering a new strategy alleviating an imbalanced dataset without requiring extensive data feeding.

Acknowledgments
The author would like to express their sincerest gratitude to Prof. Maarten De Vos and Mr. Fei Shen for their kind and valuable guidance.

References
[1] Wulff, K., Gatti, S., Wettstein, J. G., Foster, R. G. (2010). Sleep and circadian rhythm disruption in psychiatric and neurodegenerative disease. Nat. Rev. Neurosci., 11: 589-599.
[2] Faust, O., Razaghi, H., Barika, R., Ciaccio, E. J., Acharya, U. R. (2019). A review of automated sleep stage scoring based on physiological signals for the new millennia. Comput. Meth. Prog. Bio., 176: 81-91.
[3] Phan, H., Andreotti, F., Cooray, N., Chén, O. Y., De Vos, M. (2018). Joint classification and prediction CNN framework for automatic sleep stage classification. IEEE. Trans. Biomed. Eng., 66: 1285-1296.
[4] Koley, B., Dey, D. (2012). An ensemble system for automatic sleep stage classification using single channel EEG signal. Comput. Biol. Med., 42: 1186-1195.
[5] Imtiaz, S. A., Rodriguez-Villegas, E. (2015). Automatic sleep staging using state machine-controlled decision trees. In: the 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Milan. pp. 378-381.
[6] Memar, P., Faradji, F. (2017). A novel multi-class EEG-based sleep stage classification system. IEEE Trans. Neural Syst. Rehabil. Eng., 26: 84-95.
[7] Fraiwan, L., Lweesy, K., Khasawneh, N., Fraiwan, M., Wenz, H. (2010). Classification of sleep stages using multi-wavelet time frequency entropy and LDA. Method. Inform. Med., 49: 230-237.
[8] Zhang, L., Fabbri, D., Upender, R., Kent, D. (2019). Automated sleep stage scoring of the Sleep Heart Health Study using deep neural networks. Sleep., 42: 159.
[9] Tsinalis, O., Matthews, P. M., Guo, Y., Zafeiriou, S. (2016). Automatic sleep stage scoring with single-channel EEG using convolutional neural networks. arXiv:1610.01683.
[10] Supratak, A., Dong, H., Wu, C., Guo, Y. (2017). DeepSleepNet: A model for automatic sleep stage scoring based on raw single-channel EEG. IEEE Trans. Neural Syst. Rehabil. Eng., 25: 1998-2008.
[11] LeCun, Y., Bengio, Y., Hinton, G. (2015). Deep learning. Nature., 521: 436-444.
[12] Malafeev, A., Laptev, D., Bauer, S., Omlin, X., Wierzbicka, A., Wichniak, A., Achermann, P. (2018). Automatic human sleep stage scoring using deep neural networks. Front. Neurosci., 12: 781.
[13] Greff, K., Srivastava, R. K., Koutnik, J., Steunebrink, B. R., Schmidhuber, J. (2016). LSTM: A search space odyssey. IEEE Trans. Neural Netw. Learn. Syst., 28: 2222-2232.
[14] Kemp, B., Zwinderman, A. H., Tuk, B., Kamphuisen, H. A., Oberye, J. J. (2000). Analysis of a sleep-dependent neuronal feedback loop: the slow-wave microcontinuity of the EEG. IEEE Trans. Biomed. Eng., 47: 1185-1194.

[15] Hobson, J. A. (1969). A manual of standardized terminology, techniques and scoring system for sleep stages of human subjects. Elsevier publishing, Washington, DC.

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

[17] Mousavi, S., Afghah, F., Acharya, U. R. (2019). SleepEEGNet: Automated sleep stage scoring with sequence to sequence deep learning approach. PloS ONE, 14: 0216456.

[18] Sáez, J. A., Luengo, J., Stefanowski, J., Herrera, F. (2015). SMOTE–IPF: Addressing the noisy and borderline examples problem in imbalanced classification by a re-sampling method with filtering. Inf. Sci., 291: 184-203.

[19] Bao, F. S., Liu, X., Zhang, C. (2011). PyEEG: an open source python module for EEG/MEG feature extraction. Comput. Intell. Neurosci., 2011: 406391.

[20] Pijn, J. P., Van Neerven, J., Noest, A., da Silva, F. H. L. (1991). Chaos or noise in EEG signals; dependence on state and brain site. Electroencephalogr Clin Neurophysiol., 79: 371-381.

[21] Welch, P. (1967). The use of fast Fourier transform for the estimation of power spectra: a method based on time averaging over short, modified periodograms. IEEE Trans. Audio Electroacoust., 15: 70-73.

[22] Andreotti, F., Phan, H., Cooray, N., Lo, C., Hu, M. T., De Vos, M. (2018). Multichannel sleep stage classification and transfer learning using convolutional neural networks. In: the 2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)., Honolulu. pp. 171-174.

[23] Sanders, T. H., McCurry, M., Clements, M. A. (2014). Sleep stage classification with cross frequency coupling. In: the 2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society. Illinois. pp. 4579-4582.

[24] Rodríguez-Sotelo, J. L., Osorio-Forero, A., Jiménez-Rodríguez, A., Cuesta-Frau, D., Cirugeda-Roldán, E., Peluffó, D. (2014). Automatic sleep stages classification using EEG entropy features and unsupervised pattern analysis techniques. Entropy., 16: 6573-6589.