Project Abstract

Rakesh Joshi

Multimodal time series sleep stage classification using a deep learning

In order to diagnose and treat sleep problems, it's crucial to correctly define sleep phases. The ability to identify sleep phases from nightly Polysomnography (PSG) recordings is critical for diagnosing and treating sleep disorders. In this work we overview an end-to-end deep learning strategy for sleep stage categorization that learn without using spectrograms or hand-crafted features, and that takes advantage of all polysomnography (PSG) signals, multivariate and multimodal (EEG, EMG, and EOG), which can take use of the temporal context of each data frame of 30 seconds. PSG consists of electroencephalogram (EEG), electrooculogram (EOG), an electromyogram (EMG) and electrocardiogram (ECG) [1]. PSG or single-channel EEG recordings are often separated into 30-second segments, with each segment manually examined by sleep experts and subsequently categorized into one of six phases, namely waking (W), rapid eye movement (REM), and four non-REM stages (N1, N2, N3, and N4). We use deep network architecture to perform sleep stage classification from multivariate time series without temporal context has three key features: linear spatial filtering to estimate so called virtual channels, convolutive layers to capture spectral features and separate pipelines. The first layer of the network implements a spatial filtering, and it is a time-independent linear operation that outputs a set of virtual channels. Then two blocks of temporal convolution followed by non-linearity and max pooling are consecutively applied [2]. We use the popular publicly available Sleep-EDF-20 and Sleep-EDF-78 dataset [3] for our analysis. The Sleep-EDF-20 contains data files for 20 participants, whereas the Sleep-EDF-78 provides data files for 78 patients.
The Adam optimizer with $\beta1 = 0.9$ and $\beta2 = 0.999$ and learning rate $1 \times 10-3$ was used. In this work data set is imbalanced and one way to address this issue is to reweight the model loss function so that the cost of making an error on a rare sample is larger. The choice of Optimization Algorithms and Loss Functions or criterion are the two critical pieces of the training process for a deep learning model can play a big role in producing optimum and faster results Here, we use the standard multi-class cross-entropy loss function or criterion. The optimizer implements the parameter update procedure. Here, we use Adam, a popular adaptive gradient descent optimizer for deep neural networks. For comparison of classification models Cohen's kappa is more informative than overall accuracy when working with unbalanced data. Thus, we use Cohen's kappa (instead of the standard accuracy) As more training epochs are performed, we expect the loss to decrease and the Cohen's kappa to increase. To give a better estimate of the generalization performance of our Convolution Network, we measure the performance on the test set, we inspect the results using confusion matrix which tells us how ConvNet perform on each class which was not seen during training. Further, discuss some classification metrics of the confusion matrix we use Precision, Recall and F1-score.

References

[1] Chambon, S., Galtier, M. N., Arnal, P. J., Wainrib, G., & Gramfort, A. (2018). A deep learning architecture for temporal sleep stage classification using multivariate and multimodal time series. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 26(4), 758-769.

[2] Banville, Hubert, et al. "Uncovering the structure of clinical EEG signals with self-supervised learning." Journal of Neural Engineering 18.4 (2021): 046020..

[3] Ary L Goldberger, Luis AN Amaral, Leon Glass, and et al. Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals. circulation, 101(23), 2000.