Project on Detecting Sleep Apnea from raw physiological signals

E. Cohen[1], **L. Ravillon**[1]
[1] Ecole Nationale des Ponts et Chaussées

emile.cohen@eleves.enpc.fr
louis.ravillon@eleves.enpc.fr

Abstract

In this paper, we detail the different methods we used to tackle the Data Challenge provided by Ecole Normale Supérieure on the detection of sleep apnea. Our methodology highly relies on the course Time-Series that we followed part of the MVA Master track. Our approach was to create a scalable testing environment to be able to easily perform ablation studies with different preprocessing steps, models, hyperparameters etc. We finally found that using a encoder-decoder with connexity post processing allowed us to get the satisfactory accuracy of 63%.

1 Motivations

Objectives. Sleep Apnea is one of the most common breathing-related sleep disorder. It afflicts arounds 25% of male and 10% of female. Sleep apnea is characterized by repeated episodes of apnea (breathing stops) and hypnopnea (breathing is insuficient and can cause blood oxigen level disruption). Both these type of episodes can lead to fragmented sleep and daytime sleepiness as well as cardiovascular issues.

However sleep apnea is under-diagnosed and according to the AASM, around 80% of sleep apnea cases are left undetected. To diagnose sleep apnea, one must do one night with a Polysomnography (PSG) which records raw physiological signals (Brain activity with EEG, cardiac activity with ECG, respiratory activity and muscle activity with EMG). The night recording will then be manually analyzed by a trained sleep expert who will detect and count the apnea events. The number of apnea events per hour of sleep (AHI) is the clinical marker of apnea severity. The manual scoring is time consuming, requires trained personals and cannot be done easily at scale.

Hence, the development of automated methods to detect Sleep Apnea would make the apnea diagnosis more efficient and could help to detect the numerous undetected cases.

Dataset. Our dataset gathers samples from 44 nights recorded with a polysomnography and scored for apnea events by a consensus of human experts. For each of the 44 nights, 200 windows (without intersection) are sampled with the associated labels (which are binary segmentation masks). Each of these windows contains 90 seconds of signal from 8 physiological signals sampled at 100Hz:

- Abdominal belt: Abdominal contraction
- Airflow: Respiratory Airflow from the subject
- PPG (Photoplethysmogram): Cardiac activity
- Thoracic belt: Record Thoracic contraction
- Snoring indicator
- SPO2: O2 saturation of the blood

- C4-A1: EEG derivation
- O2-A1: EEG derivation

The segmentation mask is sampled at 1Hz and contains 90 labels (0 = No event, 1 = Apnea event as in Figure 1.

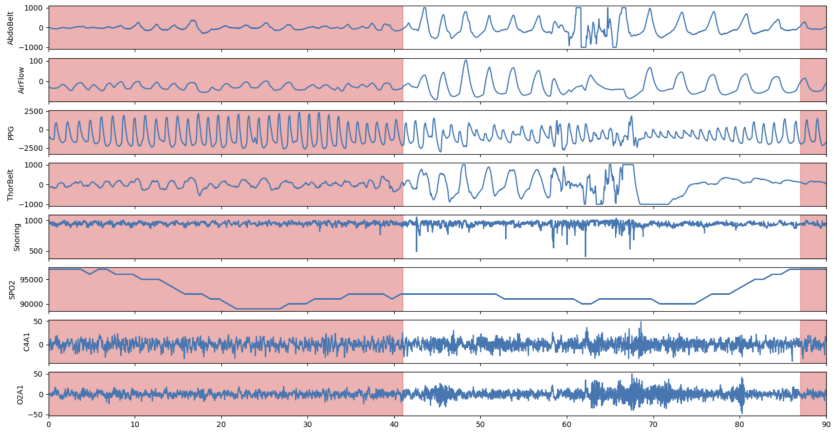

Figure 1: Example of 90 second window (Red: Apnea Event)

2 Related Work

K-complexes and spindles are the two main events occuring during stage 2 of NREM sleep, so they are often use for detection. Multiple automatic algorithms for the detection of micro-events in the sleep EEG like spindles or K-complexes have been proposed in the literature. Methods in the literature often rely on band-pass filtering (11-16 Hz for spindles, 0.5-5 Hz for K-complexes), and the extraction of hand-crafted features.

We can distinguish two main categories of algorithms. A first method relies on extracting the envelope of previously filtered signal and thresholding it. In [3], a threshold is apply on the rectified filtered signal and beginning/end times of events are detected thanks to inflexion points in the rectified filtered amplitude of spindles. The main drawback of this method is that it should be used in specific sleep stages requiring a preliminary visual inspection of the data.

A second category gathers non-linear signal decomposition based methods that attempt to separate the non-rhythmic transients or artifacts from sinusoidal spindle-like oscillations in the single channel sleep EEG. In [1], authors depict a multichannel method for the decomposition. Those methods share the benefit of not being stage dependant.

Although they have proven to be quite effective, the methods mentioned above suffer from several limitations. First, they rely on pre-defined parameters, such as frequency bands for filtering, which may not be optimal for some recordings or subjects.Moroever, their hyper-parameters, such as thresholds, are often selected on the recordings used for evaluating detection performances, introducing a bias in reported results.

To counter those limitations, the computer vision literature is more and more considered, and more specifically deep convolutional neural networks in object detection [8]. Those approaches learn a feature representation used by a prediction module to output both bounding boxes of detected objects and their classes. These approaches can handle objects of multiple classes at any scale and make predictions based on features drawn from entire or subsection of images. Besides, they may make predictions from different features maps of the underlying neural network allowing to handle different resolutions and scales of objects. Translating such methods to detect micro-events in EEG is today one of the main research interests as it would provide a and a method that does not rely on sleep stage and is able to predict locations, durations and types of any micro-event.

3 Methodology

3.1 Organization

Tasks Split. Our objective was to create a scalable and modular environment to be able to test everything quickly without having to change the code. This implies creating configuration files and parameters call in the code as well as using Object oriented programming. This took a long time and was handled by the two of us conjointly. Emile decided to work more closely on the preprocessing steps, while Louis handled the first models architecture. Once the backbone of the overall architecture was created, we could work on ablation studies to find the best preprocessing, model, hyperparameters and even postprocessing step.

Experimental Tools. We decided to stay on *Colab* with GPU enabled as long as the RAM did not explode. It turned out that *Colab* was well fitted to our computation power needs. We coded all python modules locally on *VSCode*

3.2 Preprocessing

We decided to create two different data modules. The first one *SleepApneaDataModule* handles directly the time-series in the time-domain. We decided to normalize each temporal signal patient-wise. The seconde data module called *EmbeddedDataModule* works on the frequency-domain by performing a N-dimensional discrete Fourier Transform (for real input here) after normalizing the time-series patient-wise.

3.3 Metrics

Metrics. As advised in the data challenge, *by event metrics* were used to assess performances for detection and localization of events. These metric rely on the Intersection over Union (IoU) criterion: for a given $\delta \geq 0$, a predicted event was considered as a true positive if it exhibited an $IoU \geq \delta$ with a true event, otherwise it was considered as a false positive. The numbers of positives and true positives were evaluated to compute precision, recall and F1 scores of detectors for different overlapping criterion $\delta \in \{0.1, 0.2, ..., 0.9\}$. We used this metric to evaluate our model in the validation step.

Loss. For the model training, we decided to use the Binary Cross Entropy Loss (BCE Loss) as we are dealing with a binary classification. The main advantage of this loss is that it penalizes wrong confident predictions.

3.4 Models

Overall architecture. We tried different models out of the box. We began with a Recurrent Neural Networks (RNN) [5], but we quickly moved to Long-Short Term Memory Network (LSTM) to train it more easily (problems of vanishing gradients in RNN) and to benefit from the gate architecture as explained in the well-known [6]. We also implemented Convolutional Neural Network (CNN) cells.

On the other hand, the problem can be seen as a Sequence to Sequence problem, because we try to translate the continuous sequence of input into a binary sequence. Following efforts in the literature on encoder-decoder architectures for Seq2seq tasks [4] [2], we decided to implement an encoder-decoder architecture in order to directly learn the embeddings: we tried different encoders and decoders such as 1D Convolutions, 2D Convolutions, LSTM, CNN, Transformer.

Input. To feed our model, we designed our input to have the following tensor shape: $(n\_signals, seq\_length, sampling\_freq)$ with $n\_signals$ being the number of signals considered from 1 to 8, seq_length here 90 points to be coherent with the wanted output sequence size (only 90 labels per window) and $sampling\_freq$ the frequency of sampling (here 100 Hz). This input preprocessing is done in both $SleepApneaDataset$ and $EmbeddedDataset$ classes.

Post Processing. We also implemented a postprocessing script. Indeed we found that some wrong event predictions were very isolated (points "around" had a very low post sigmoid probability). Our idea was thus to enforce connexity on the output by removing confident event predictions surrounded by confident no-event prediction.

Ablation Study. To provide more insight about our model and the influence of several hyperparameters and modules, we conduct extensive ablation studies. Overall we find that ablating any of design components from the default model (*GroupedConv2D + LSTM*) would degrade the performance. The effects are detailed as follows.

4 Experiments Results

4.1 Model

We first decided to train and test our models on the full multivariate signal (8 signals). We compared the results for the Encoder-Decoder, a biLSTM and a RNN. In this case we set a 4-layer CNN as the encoder and a biLSTM as the decoder. Our models were trained for 30 epochs. For all experiments we set our learning rate at 0.001 .

	Validation Accuracy (%)
Encoder-Decoder (CNN + LSTM)	**55.6**
biLSTM	21.4
RNN	9.8

Table 1: Comparison between models

The huge discrepancies but the Encoder-Decoder and the biLSTM and RNN can be explained by the fact that there might too much noise in the whole multivariate signals for those networks. As a result the Encoder-Decoder is able in a first time to extract relevant features compared to raw RNN and biLSTM.

Figure 2: Comparison between models for 8 signals - Training Loss (*top*) and Validation Accuracy (*bottom*)

4.2 Influence of the Signals

We decided to inspect which signals might be the most determinant signals to detect sleep apnea. We examined the signal 1 (*Airflow*), signal 2 (*PPG*) and signal 3 (*Thoracic belt*). The results can be found on Table 2.

These experiments enabled us to see the influence of the Airflow signal (1) compared to the others signals. The Airflow signal standalone achieves performances almost similar to the whole multivariate signal (8 physiological signals). The only other signal that has a reasonable accuracy alone is the Thoracic Belt (3). The best model is achieved using the Airflow (1), PPG (2) and Thoracic Belt (3) signal with the Encoder-Decoder.

	Validation Accuracy (%)
only Airflow	53.7
only Thoracic Belt	39.3
Airflow + Thoracic Belt	58.1
Airflow + PPG	53.6
PPG + Thoracic Belt	46.4
Airflow + PPG + Thoracic Belt	**60.3**
8 signals	55.6

Table 2: Comparison between models

4.3 Ablation Study

Finally we wanted to explore the influence of several hyperparameters and modules such as :

- The encoder model (simple CNN 2D or Grouped CNN 2D)
- The influence of the connex post processing and the different threshold
- The impact of smoothing the input signals with a Gaussian kernel

Encoder. We compared the performance between two types of convolutional network and find indeed that the Grouped CNN is better than a standard CNN.

	Validation Accuracy (%)
Grouped CNN 2D	**60.3**
CNN 2D	56.2

Table 3: Comparison between Encoders

Connexity Post Processing. To capture the influence, we evaluate our method on several signals. The connex post processing is done on by first computing an average pooling on y_{pred}. Then if a prediction is above a certain threshold (thus classified as an event) but its averaged value is below another threshold, we classify it as non-event. And vice versa. We found that such post processing greatly increases the performance of the model.

	Without PP	With PP
[1,2]	53.6	59.6
[1,2,3]	60.2	**63.4**
[1,2,3,4]	57.6	60.4

Table 4: Validation Accuracy (%) for connex post processing

Gaussian smoothing. In order to avoid overconfidence in the predictions, we applied Gaussian smoothing but unfortunately it did improve the performance of the network.

	Validation Accuracy (%)
Without Gaussian Smoothing	**60.3**
With Gaussian Smoothing	48.7

Table 5: Results for label smoothing on signal [1,2,3]

5 Conclusion

We built a encoder-decoder model to tackle this classification problem of detecting sleep anea from raw physiological signals. Our best method combines a CNN and a biLSTM and its performance

ranks it among the best model on the public leaderboard. We studied extensively the impact of several key hyperparameters and modules of the model and were able to leverage the connectedness of our data to boost our perfomances. Several improvements lay in the preprocessing of our signal. Indeed a thorough study of the seasonality and patterns of the 8 physiological signals might benefit the detection of sleep anea.

References

[1] R. S. Osorio A. W. Varga D. M. Rapoport I. Ayappa A. Parekh, I. W. Selesnick. Multichannel sleep spindle detection using sparse low-rank optimization. *Methods 288*, 2017.

[2] Yoshua Bengio Dzmitry Bahdanau, Kyunghyun Cho. Neural machine translation by jointly learning to align and translate. *https://arxiv.org/abs/1409.0473*, 2015.

[3] M. J. Peterson M. Massimini M. Murphy B. A. Riedner A. Watson P. Bria G. Tononi J. Neurosci F. Ferrarelli, R. Huber. Reduced sleep spindle activity in schizophrenia patients. *Am J Psychiatry 164 (3) (2007) 483–492.*

[4] Bart van Merriënboer Dzmitry Bahdanau Yoshua Bengio Fethi Bougares Holger Schwenk Kyunghyun Cho, Caglar Gulcehre. Learning phrase representations using rnn encoder–decoder for statistical machine translation. *https://arxiv.org/abs/1406.1078*, 2014.

[5] Geoffrey E Rumelhart, David E; Hinton and Ronald J Williams. Learning internal representations by error propagation. *Tech. rep. ICS 8504. San Diego, California: Institute for Cognitive Science, University of California.*, Sept. 1985.

[6] Jurgen Schmidhuber Sepp Hochreiter. Long short-term memory. *https://www.bioinf.jku.at/publications/older/2604.pdf*, 1997.

[7] Pierrick J. Arnal Emmanuel Mignot Alexandre Gramfort Stanislas Chambon, Valentin Thorey. Dosed: a deep learning approach to detect multiple sleep micro-events in eeg signal. *https://arxiv.org/abs/1812.04079*, 2018.

[8] R. Girshick K. He B. Hariharan S. Belongie T.-Y. Lin, P. Doll ar. Feature pyramid networks for object detection. *CVPR*, 2017.