A Deep Learning approach to detect sleep states

Armando Bringas, Alexis Guerrero

3rd January - 2024

Introduction

Kaggle competition

This paper presents a project proposal rooted in the practical application of a Kaggle competition, aimed at addressing a challenge in pediatric health and neuroscience: the identification of sleep states in children through wrist-worn accelerometer data.

Project implications

The significance of this project lies in its potential to deepen our understanding of sleep and to provide further insights into its importance. For instance, understanding how environmental factors influence sleep, mood, and behavior can aid in formulating personalized strategies tailored to the unique needs of each child.

Literature Review

Models

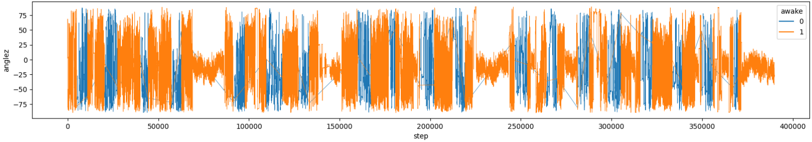
- Random Forest
- Residual Neural Network (RNN)
- Latent Class Analysis (LCA)
- Long Short-Term Memory (LSTM)

Data origin

- Accelerometer data
- Actigraphy and sleep diaries
- Optical plethysmography
- Eelectrocardiography
- Electroencephalography
- Multi channel polysomnogram
  - Electrooculography
  - Electromyography

Dataset

Dataset consists of approximately 500 multi-day recordings from wrist-mounted accelerometers. The accelerometer data in the dataset was processed using R with the GGIR package [**?**]. The recordings are labeled with two event types: 'onset', indicating the start of sleep, and 'wakeup', marking its end. The primary objective is to identify these two events within the accelerometer data series, this primarily represents a binary classification task.



Model - Random Forest

The Random Forest model is initialized with a predetermined number of 100 estimators (n_estimators) and a minimum leaf sample size (min_sample_leaf) set to 300. These initial values are chosen for the purpose of assessing the model's efficiency and will be subject to subsequent adjustments based on its performance. Given the substantial volume of data in this scenario, it is anticipated that the minimum leaf sample size will remain unchanged, whereas the number of estimators will likely need to be increased to optimize performance. An iteration process is set along the Random Forest model, changing the random_state and n_estimators values.

$$\mathcal{X}_{RF} \longrightarrow \boxed{\begin{array}{c}\text{Random Forest}\\ i\text{th iteration}\end{array}} \longrightarrow \mathcal{Y}_{RF_i}(rs_i, Ne((i-1)[\text{mód}3]))$$

Model - LSTM

We are proposing a starting neural network architecture with the following blocks where consists of an LSTM layer of 64 units, ideal for processing sequences by capturing dependencies from prior inputs. This is followed by a Dense layer, the size of which matches the number of classification categories in your problem, in this case for our binary classification problem $n_{\text{classes}} = 2$. The final component is a softmax activation function, applied to convert the output into a probability distribution across the predicted classes.

$$\xrightarrow{\text{input}} \boxed{\text{LSTM}(64)} \rightarrow \boxed{\text{Dense}(n_{\text{classes}})} \xrightarrow{\text{softmax}}$$

Figura: Neural Network for accelerometer data classification

Evaluation Metrics

Average Precision

$$\text{Inputs } \mathcal{X} = (x_1, \dots, x_N)^\intercal$$
$$\text{Outputs } \mathcal{Y} = (y_1, \dots, y_N)^\intercal$$
$$\text{Labels } \mathcal{W} = (w_1, \dots, w_N)^\intercal$$

$$AP = \frac{1}{N} \sum_{i=1}^{N} \mathcal{C}(w_i, y_i)$$

$$\mathcal{C}(w_i, y_i) = \begin{cases} 1 \text{ if } w_i = y_i \forall i \in \mathbb{N}, 1 \leq i \leq N \\ 0 \text{ if } w_i \neq y_i \forall i \in \mathbb{N}, 1 \leq i \leq N \end{cases}$$

$$APF(\mathcal{W}, \mathcal{Y}) = \frac{1}{N} (\omega_1, \dots, \omega_N)(1, \dots, 1)^\intercal$$

Evaluation Metrics

Specificity and Recall

$$\text{Score}(x) = \begin{cases} \text{TP} & \text{if } x \text{ matched and } x > \text{thresh.} \\ \text{FP} & \text{if } x \text{ unmatched pred.} \\ \text{FN} & \text{if } x \text{ unmatched truth} \\ \text{TN} & \text{otherwise} \end{cases}$$

$$\text{Specificity} = SPC = \frac{TP}{TP + FN} \qquad \text{Recall} = SN = \frac{TN}{TN + FP}$$

Final evaluation

$$\text{Model evaluation} = \alpha APF + \beta SPC + \gamma SN$$

Results

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum

Conclusion

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum

Bibliography