COMPARISON OF AUTOMATIC SLEEP STAGE SCORING METHODS USING LIMITED SCORING

Alexei Labrada Tsoraeva[*], Elsa Santos Febles, José Manuel Antelo
Cuban Neuroscience Center, Cuba.

ABSTRACT
The diagnosis of various types of sleep disorders requires the experts to perform sleep stage scoring. However, it is an arduous and repetitive task and, therefore, an important candidate for automation. This work seeks to evaluate several scoring algorithms based on Machine Learning from the scientific literature. The comparison is performed with the same experimental design, using EEG, EOG and EMG signals from the polysomnographic records of the *ISRUC-Sleep* dataset. It is compared the precision, memory and speed of methods based on Linear Discriminant Analysis, Support Vector Machines, Random Forests and Neural Networks. As a result, several of the analyzed algorithms reach high levels of accuracy, exceeding 75%. Also, it is demonstrated that the accuracy can be raised to 85% by skipping the classification of doubtful epochs and still classify 65% of them.

KEYWORDS: Polysomnography, Sleep Stage Scoring, Digital Signal Processing, Machine Learning
MSC: Artificial Intelligence

RESUMEN
El diagnóstico de varios tipos de trastornos del sueño requiere que los especialistas clasifiquen las fases del sueño. Sin embargo, esta es una tarea ardua y repetitiva y, por lo tanto, un importante candidato a automatizarse. El trabajo busca evaluar varios tipos de algoritmos de clasificación basados en Aprendizaje Automático disponibles en la literatura científica. La comparación se efectúa con el mismo diseño experimental, utilizando señales de EEG, EOG y EMG de los registros polisomnográficos del conjunto *ISRUC-Sleep.* Se compara la precisión, memoria y velocidad de métodos basados en Análisis Discriminante Lineal, Máquinas de Vectores de Soporte, Bosques Aleatorios y Redes Neuronales. Se comprueba que varios de los algoritmos analizados alcanzan altos niveles de exactitud, superando el 75%. Además, se demuestra que puede aumentarse la exactitud al 89% si se deja de clasificar las épocas dudosas, aun así clasificando el 65% de las mismas.

PALABRAS CLAVE: Polisomnografía, Fases del Sueño, Procesamiento Digital de Señales, Aprendizaje Automático

1. INTRODUCTION

Sleep stage scoring is a necessary step for diagnosing sleep disorders such as insomnia, sleep apnea, narcolepsy and hypersomnia. According to the American Association of Sleep Medicine (AASM),

[*] labrada.alexei@gmail.com

this operation is performed by dividing a Polysomnographic (PSG) record in consecutive 30 second windows, also called epochs. The stage of each epoch must be classified as Wake (W), REM sleep (R) or one of the non-REM sleep stages: N1, N2 or N3 [16]. Also, the AASM defines a set of guidelines that the experts should follow while scoring a PSG record through the visual inspection of each epoch. A PSG record contains the behavior of several electrophysiological signals during the analyzed time period. The most important signals for sleep scoring are the electric activity of the cerebral cortex, measured using Electroencephalography (EEG), of the facial muscles, using Electromyography (EMG) and the ocular movements, using Electrooculography (EOG). There may be additional signals, such as the cardiac activity or Electrocardiogram (ECG), the respiratory activity and body movements. The scoring rules are based on the identification of several patterns in the signals, including Alpha activity (8-13 Hz), Beta activity (13-35 Hz), Theta activity (4-8 Hz), K-complexes, sleep spindles, Rapid Eye Movements (REM), Slow Eye Movements (SEM) and chin activity.

The PSG records can reach eight hours in duration and, therefore, the number of epochs in a sleep study is close to a thousand. This implies that the scoring process is an arduous and repetitive one and, consequently, it invites potential mistakes. The scientific literature contains many instances of algorithms that automate this process using Machine Learning techniques. However, even though there are established rules for the process, their subjective nature and the low agreement level between experts [5] do not allow the algorithms to be precise enough to completely replace the experts in clinical contexts.

The consulted articles propose the usage of various signal processing methods, including High Order Spectra (HOS) [4], the Continuous Wavelet Transform (CWT) [8], the Discrete Wavelet Transform (DWT) [22], the Fourier Transform and several statistic methods [25, 14]. They also use classification methods based on Random Forest (RF) [9], Support Vector Machines (SVM) [14, 3] and Artificial Neural Networks. This last group includes Multilayer Perceptrons (MLP) [22, 2] and Recurrent Neural Networks (RNN) [27, 6].

In previous works [15], the authors make a performance comparison among classification methods based on LDA, RF, SVM, MLP and RNN, using PSG records from the Sleep-EDFx dataset [10, 12]. It was concluded that an algorithm based on RNN achieves the best results according to the evaluated parameters. But, even though the dataset is a common one among the consulted articles [2, 7, 19, 24, 26], the provided expert-based sleep scoring does not conform to the AASM guidelines.

The objective of this work is to select a sleep scoring algorithm that makes the job of the sleep experts easier. The algorithm must be included in a software system dedicated to the clinical analysis of PSG records. Hence, the selection has to be based on the accuracy of the results, but taking into consideration the execution speed and the memory usage. Consequently, a public dataset is chosen and the performance of several classification methods from the scientific literature is compared in similar conditions.

2. MATERIALS

This work uses PSG records obtained from the *ISRUC-Sleep* public dataset [13], including 10 healthy subjects and 100 with different kinds of sleep disorders, one record from each subject. Also, each one of the records was separately scored by two sleep experts following the AASM guidelines. Table 1 summarizes the distribution of the sleep stages in the records of the dataset. It also shows the agreement level between the experts using the F-score metric.

The records include EEG, EOG and EMG signals with a sampling frequency of 200 Hz. The signals that are considered by the AASM as optional for sleep scoring [16] are not used as, in case of been absent, it would render the algorithms unusable.

	Expert 1		Expert 2		
Stage	Count	Percent	Count	Percent	F1
W	22490	22.90	23554	23.98	0.919
N1	12694	12.92	10181	10.36	0.560
N2	30664	31.22	33158	33.75	0.803
N3	19408	19.76	17882	18.20	0.858
R	12971	13.21	13459	13.70	0.904
Total	98227		98234		0.809

Table 1: Distribution of sleep stages in the dataset records and expert agreement.

3. METHODS

The records are randomly split in two groups, the first one for training and the second one for validation, with a size ratio of 75%-25%. The implemented algorithms are trained separately with the annotations provided by both experts, so the training process produces two trained models from each algorithm. Then, the trained models are tested using the records from the validation group, classifying each epoch twice and comparing the prediction with the annotation of the corresponding expert.

The execution time of the analyzed algorithms can be split in three main phases: Data preprocessing, feature extraction and classification.

3.1. Preprocessing

The goal of the preprocessing phase is to prepare the data for the feature extraction phase. In order to achieve it, the records are segmented in 30 second windows that match the epochs that will be classified later. Also, the epochs that were registered with the lights turned on and the ones with unknown or invalid sleep stages are excluded from consideration.

3.2. Feature Extraction

The feature extraction phase obtains a limited number of descriptive values that reflect the information inside the signals that is relevant for the classification process. The values or features used in this work are obtained by analyzing the signals in each epoch in the time domain, the frequency domain, the time-frequency domain and by some other nonlinear means. Table 2 shows a summary of the extracted features.

Method	EEG(F)	EEG(C)	EEG(O)	EOG	EMG	Total
Kurt	1	1	1	1	1	5
Skew	1	1	1	1	1	5
P75	1	1	1	1	1	5
Act	0	1	0	1	1	3
Mob	0	1	0	1	1	3
Cpx	0	1	0	1	1	3
ShEn	1	1	1	1	1	5
ApEn	0	1	0	1	1	3
LLE	0	1	0	1	1	3
HFD	0	1	0	1	1	3
LZC	0	1	0	1	1	3
FFT	4	6	4	3	0	17
HOS	0	4	0	4	4	12
CWT	0	2	0	0	0	2
DWT	0	6	0	6	0	12
Total	8	29	8	24	15	84

Table 2: Summary of extracted features from each signal, including frontal (F), central (C) and occipital (O) EEG, EOG and EMG. The specific extraction methods are explained in section 3.2.

3.2.1. Descriptive Statistics

This kind of features are obtained by computing descriptive statistics from the signal's samples. The Variance, Kurtosis (Kurt), Skewness (Skew) and 75th Percentile (P75) have been employed in this work.

3.2.2. Hjorth's Parameters

The Hjorth's parameters, called Activity, Mobility and Complexity, reflect the spectral properties of a signal in the time domain [11]. The Activity is equivalent to the variance of the signal ($Act = var(X)$), while the Mobility is defined in equation 3.1,

$$Mob = \sqrt{\frac{var(X')}{var(X)}} \tag{3.1}$$

where X' is the first derivative of the signal X. Complexity is the ratio between the Mobility of the signal's derivative and the signal itself (Equation 3.2).

$$Cpx = \frac{Mob(X')}{Mob(X)} \tag{3.2}$$

3.2.3. Entropy

Entropy is a measure of the irregularity of a signal in the time domain [21]. The equation 3.2.3. shows the formula proposed by Shannon for this measure:

$$ShEn = -\sum_{i=1}^{N} p\left(x_i\right) \log p\left(x_i\right) \tag{3.3}$$

where $p\left(x_i\right)$ is the probability of a signal sample having the value x_i.
There are other estimation methods, including the Approximate Entropy, displayed in equation 3.2.3..

$$ApEn(r, m) = \phi_r^m - \phi_r^{m+1} \tag{3.4}$$

The values of ϕ can be obtained using an algorithm that represents the signal in the phase domain $X_i = \left\{x_i,\, x_{i+1},\, ...,\, x_{i+(m-1)}\right\}$ and calculates the distance between those patterns using the L1 norm. Then,

$$\phi_r^m = \frac{1}{M} \sum_{i=1}^{M} \log \frac{N_r^m(i)}{M} \quad M = N - m + 1 \tag{3.5}$$

where N_r^m is the number of X_j patterns that satisfy $\|X_i - X_j\|_1 \leq r$.
In this work the pattern length (m) is 2 and r is the standard deviation of the signal in the epoch, multiplied by 0.1, as estimated in [21].

3.2.4. Largest Lyapunov Exponent

The Largest Lyapunov Exponent (LLE) is an indicator of how unpredictable a signal is. It has been demonstrated that it can be useful for discriminating the N1 and N2 stages [14]. The algorithm proposed by [20] allows to estimate LLE by calculating the distances between the most similar trajectories, that are also distant in the time domain. Equation 3.2.4. describes this distance,

$$d_j(0) = \min_k \|X_j - X_k\|, \quad |i - j| > \tau \tag{3.6}$$

where τ is the threshold in time domain and $X_i = \left\{x_i,\, x_{i+J},\, ...,\, x_{i+(m-1)J}\right\}$ is a trajectory in phase domain. Once the distances have been calculated, the LLE can be obtained using linear regression with equation 3.2.4..

$$y(i) = \sum_{j=1}^{M} \frac{\log d_j(i)}{T_s M} \quad M = N - (m-1)J \tag{3.7}$$

In our work we use the values 10 and 7 for m and J, respectively, while τ is the mean period of the signal (MNF^{-1}).

3.2.5. Fractal Dimension

The Higuchi Fractal Dimension (HFD) is an estimate of the fractional dimensions of the geometric shape of a signal in the time domain [21]. In [14] it is stated that this measure is especially useful for recognizing the N3 stage.

The Higuchi algorithm calculates the fractal dimension as the slope of the mean squares fit of the values of $\log(L(k))$ against $\log(1/k)$ for k between 1 and k_{max}. The values of $L(k)$ are calculated using the equation 3.2.5.:

$$L(k) = \sum_{m=1}^{k} L_m(k) \tag{3.8}$$

where $L_m(k)$ is the mean length of the sequence

$$x_m^k = (x_m, x_{m+k}, x_{m+2k}, ..., x_{m+N_m^k\,k}),\; N_m^k = \lfloor (N-m)/k \rfloor\, k$$

calculated with equation 3.2.5.:

$$L_m(k) = \frac{(N-1)\sum_{i=1}^{N_m^k} x_{m+i\,k} - x_{m+(i-1)\,k}}{N_m^k\, k} \tag{3.9}$$

In this work we use the value 40 for k_{max}, that was estimated in [21].

3.2.6. Lempel-Ziv Complexity

The Lempel-Ziv complexity (LZC) is an estimation of the complexity of a signal [14, 21]. It can be calculated by first transforming the signal x in a binary sequence x_2 by comparing each sample with a predefined threshold T. Then, the sequence is used to calculate $c(x_2)$, the number of different sub-sequences of x_2, scanning sequentially from left to right. Finally, the value of x_2 is normalized using equation 3.10,

$$LZC = \frac{c(x_2)}{\frac{N}{\log_2 N}} \tag{3.10}$$

where N is the length of the sequence. This works uses the median of the samples of X as the value of T [21].

3.2.7. Discrete Fourier Transform

The Fast Fourier Transform (FFT) algorithm efficiently estimates the frequency spectrum of a signal. The spectrum can be used to obtain the mean frequency of the signal, the spectral entropy and the relative spectral density of the relevant frequency bands.

The mean frequency can be calculated using equation 3.2.7.:

$$MNF = \sum_{i=1}^{M} f_i\, P_i \tag{3.11}$$

where M in the number of frequency bins, f_i are the frequency values and P is the normalized spectral frequency ($\sum P_i = 1$) [18]. Similarly, the spectral entropy of a frequency band can be obtained from equation 3.2.7.:

$$SpEn = -\sum_{i=f_l}^{f_h} \frac{P_i \log P_i}{\log N_f} \tag{3.12}$$

where f_l and f_h are the minimum and maximum frequencies, respectively and N_f is the number of frequency bins in the range $[f_l, f_h]$ [21].

3.2.8. High Order Spectra

The High Order Spectra (HOS) analysis can be employed to extract features related to third order statistics of a signal [4]. Before calculating the features, the Bispectrum has to be estimated using equation 3.2.8.,

$$B(f_1, f_2) = \sum_{i=1}^{W} \frac{X_i(f_1)X_i(f_2)X_i(f_1+f_2)}{W} \tag{3.13}$$

where X_i is the Short-Time Fourier Transform (STFT) of the signal on the i-th window and W is the number of windows. The STFT in a vicinity of x_i is the FFT of the product of the signal and a window function centered on x_i [23]. In our work, we use 2 seconds long Haan windows, with 1 second (50%) of overlap between consecutive windows. The Bispectrum is symmetric in both axes, so its domain of interest is defined in the expression 3.2.8..

$$\Omega = \{(f_1, f_2)|\; f_1 \geq 0 \;, f_1 \geq f_2 \;, f_1 + f_2 \leq 0.5\} \tag{3.14}$$

Once the Bispectrum is calculated, it is possible to calculate its mean amplitude, the Normalized Bispectral Entropy (equation 3.2.8.), its logarithmic sum (equation 3.2.8.) and its mean frequency (equation 3.2.8.):

$$\begin{aligned} BiEn &= -\sum_{n=1}^{N} p_n \log p_n \\ p_n &= \frac{|B(f_1, f_2)|}{\sum_{\Omega} |B(f_1, f_2)|} \end{aligned} \tag{3.15}$$

$$H_1 = \sum_{\Omega} \log |B(f_1, f_2)| \tag{3.16}$$

$$WCOB_1 = \frac{\sum_{\Omega} f_1 \, B(f_1, f_2)}{\sum_{\Omega} B(f_1, f_2)} \tag{3.17}$$

3.2.9. Wavelet Transform

The Wavelet Transforms translate a signal into the time-frequency domain. The transformation approximates the signal inside a time window by a Wavelet base (ψ) using different time scales [23]. The scale factors are inversely proportional to the frequency of the Wavelet base, as stated in equation 3.2.9.,

$$\omega = \frac{f_\psi}{a \, T_s} \tag{3.18}$$

where T_s is the sampling period and f_ψ is the mean frequency of the Wavelet base [8].
The Continuous Wavelet Transform (CWT) can be used to estimate the instantaneous frequency along the time domain of the signal and is computed using equation 3.19

$$L_\psi(a, t) = \frac{1}{\sqrt{|a|}} \int_{-\infty}^{\infty} \bar{\psi}\left(\frac{u-t}{a}\right) f(u) \, du \tag{3.19}$$

where $f(x)$ is the signal and ψ is a Wavelet base [23]. However, computing the CWT across the whole frequency domain of the signal is a very computationally expensive operation. Therefore, in this work the usage of the CWT is limited to the detection of K-Complexes and sleep spindles in the 0.5-1.5 Hz and 12-14 Hz frequency bands, respectively, of the EEG signals.

The Discrete Wavelet Transform (DWT) decomposes the signal in two coefficient vectors with $N/2$ values, satisfying

$$\alpha_1 = H_\psi \, x \qquad \delta_1 = G_\psi \, x \tag{3.20}$$

where H_ψ and G_ψ are dual filters with sub-sampling, related to the Wavelet base [23]. The α_1 vector contains an approximation of the original signal in the frequency range $\left[0, \frac{1}{4} f_s\right]$, while δ_1 is a detail vector in the frequency range $\left[\frac{1}{4} f_s, \frac{1}{2} f_s\right]$, where f_s is the sampling frequency [22]. The DWT can be computed again from vector α_1, in order to obtain the vectors α_2 and δ_2 with frequency ranges $\left[0, \frac{1}{8} f_s\right]$ and $\left[\frac{1}{8} f_s, \frac{1}{4} f_s\right]$, respectively. Thus, successively, the signal can be decomposed in L levels, after which the vectors $\delta_1, \delta_2, ..., \delta_L, \alpha_L$ belong to different frequency bands.

From the transform, the entropy of each relevant frequency band along the epoch in question can be calculated. In our work we use the Daubechies function (*db1*) as the Wavelet base for the EOG signals and the reverse biorthogonal function (*rbio3.3*), for the EEG signals. Given the 100 Hz sampling frequency of the signals, once they are decomposed in 5 levels, the frequencies of the coefficient vectors approximately match the frequency bands of interest.

3.3. Classification

The classification phase is responsible for assigning a sleep stage to each epoch contingent on the features extracted from it. The classification can take into consideration historic information by also using the features extracted from the previous $K - 1$ epochs, given a predefined sequence length K. This work uses classifiers based on LDA, SVM, RF, MLP and RNN. Additionally, it evaluates an ensemble based classifier, using majority Voting (V) of the LDA, SVM and MLP models.

For the implementation of the LDA, SVM, RF, MLP and V classifiers this work uses the Python package Sciki-Learn [17]. On the other hand, the RNN based classifier, a neural network with two Long-Short Term Memory (LSTM) bidirectional layers, is implemented using the TensorFlow [1] package.

Given that the previously mentioned implementations are capable of estimating the probability of the offered predictions, the authors propose an algorithm that takes into consideration that information. Given the classifier C and a minimum probability P_{min}, the algorithm scores epoch E only when the classification probability P_C satisfies $P_C(E) > P_{min}$.

3.4. Evaluation

The performance of each algorithm has been analyzed taking into consideration the accuracy (Acc), Cohen's *Kappa* coefficient and the F-score (F1). Additionally, the amount of space required for the storage of the trained models and execution time are also considered, but as secondary parameters. In the latter case, it is only compared the execution time of the classification phase, as the preprocessing and feature extraction phases are common to all the analyzed algorithms.

4. RESULTS

The estimation of the optimum sequence length (K) and other model-specific hyperparameters is accomplished through 10-fold cross validation, using the first group of PSG records of the *ISRUC-Sleep* dataset. Then, the training is repeated with the estimated hyperparameters and all the records of the group.

After training the classifiers, two instances of each algorithm are obtained, associated to each one of the experts (E1 and E2). Then, the obtained classifiers are tested using the second group of PSG records and the results are compared. Table 3 shows the comparison of the performance of the algorithms.

Algorithm	Acc	Kappa	F1	F1 by stage					Memory	Time
				W	N1	N2	N3	REM	(KB)	(s)
RF-E1	0.7659	0.6983	0.7454	0.8446	0.4330	0.7491	0.8507	0.8494	44874	0.390
RF-E2	0.7696	0.7009	0.7409	0.8508	0.3983	0.7595	0.8444	0.8515	44266	0.389
MLP-E1	0.7497	0.6782	0.7243	0.8253	0.3903	0.7426	0.8475	0.8155	34	0.048
MLP-E2	0.7487	0.6763	0.7309	0.8448	0.4400	0.7394	0.8105	0.8197	34	0.048
LDA-E1	0.7710	0.7059	0.7570	0.8346	0.4798	0.7686	0.8573	0.8449	7	0.012
LDA-E2	0.7720	0.7062	0.7515	0.8423	0.4517	0.7703	0.8509	0.8423	7	0.013
SVM-E1	0.7736	0.7097	0.7451	0.8570	0.4273	0.7559	0.8609	0.8246	10	0.039
SVM-E2	0.7726	0.7092	0.7440	0.8601	0.4257	0.7554	0.8619	0.8167	10	0.038
V-E1	0.7829	0.7212	0.7621	0.8574	0.4688	0.7768	0.8631	0.8426	64	0.187
V-E2	0.7842	0.7223	0.7596	0.8578	0.4532	0.7818	0.8606	0.8402	64	0.189
LSTM-E1	0.7804	0.7181	0.7592	0.8580	0.4541	0.7623	0.8600	0.8615	3759	4.500
LSTM-E2	0.7910	0.7291	0.7476	0.8746	0.3407	0.7825	0.8758	0.8645	3759	4.326

Table 3: Comparison of the performance of the classifiers using the testing records.

Additionally, the performance of the classifiers is tested while limiting the minimum classification probability. Table 4 shows the obtained results using 0.5 and 0.75 as limits, as well as the fraction of epochs that were actually classified in each case. Similarly, Figure 1 shows the effect of different P_min values on the performance of the V and LSTM classifiers.

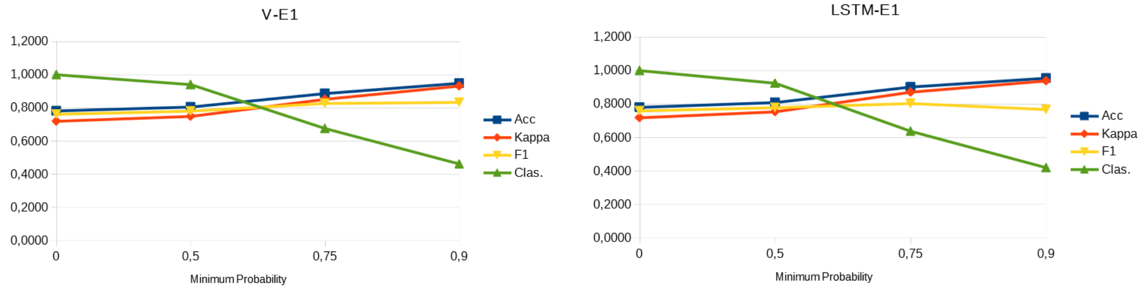

Figure 1: Effect of the minimum probability in the performance of the algorithms

	$P_{min} = 0.5$				$P_{min} = 0.75$			
Algorithm	Acc	Kappa	F1	Classified	Acc	Kappa	F1	Classified
RF-E1	0.8551	0.8082	0.7724	0.7579	0.9638	0.9493	0.7710	0.3214
RF-E2	0.7587	0.6894	0.7303	0.9743	0.9658	0.9536	0.7744	0.3353
MLP-E1	0.7564	0.6861	0.7364	0.9803	0.8185	0.7656	0.7674	0.7885
MLP-E2	0.7783	0.7150	0.7633	0.9805	0.814	0.7593	0.7725	0.8004
LDA-E1	0.7815	0.7179	0.7585	0.9741	0.8394	0.793	0.8066	0.7941
LDA-E2	0.8230	0.7691	0.7624	0.8892	0.8434	0.7964	0.7962	0.7944
SVM-E1	0.8212	0.7657	0.7335	0.8851	0.9358	0.9145	0.8152	0.4960
SVM-E2	0.8551	0.8082	0.7724	0.7579	0.9353	0.9137	0.7698	0.5147
V-E1	0.8054	0.7489	0.7807	0.9405	0.8866	0.8520	0.8272	0.6759
V-E2	0.8068	0.7501	0.7781	0.9366	0.8964	0.8639	0.8238	0.6516
LSTM-E1	0.8092	0.7539	0.7786	0.9248	0.902	0.8712	0.8034	0.6376
LSTM-E2	0.8222	0.7677	0.7530	0.9185	0.9006	0.8687	0.7465	0.6898

Table 4: Comparison of the performance of the classifiers with minimum classification probability P_{min}

5. DISCUSSION

The LSTM algorithm obtains the best results among the individual classifiers, achieving above 0.78 accuracy, 0.71 Kappa coefficient and 0.74 F-score. The amount of memory that occupies this model, approximately 3 Megabytes (MB), is greater than the remaining classifiers, except RF, but not in a significant amount. Its execution time is also greater than the required by the other classifiers, but it is significatively smaller than 1768 seconds, the time it takes to execute the common feature extraction phase. Additionally, the proposed method (V) achieves a precision level similar to the one of LSTM, with 0.78 accuracy, 0.72 Kappa coefficient and 0.75 F-score, but requiring less memory and time.

The analysis of the stage-specific classification accuracy of the methods demonstrates that N1 is significantly harder to predict than the remaining stages. Table 1 demonstrates that the experts have a very low agreement on this stage, so the low N1 classification accuracy is not an exclusive trait of the algorithms.

The obtained results are still below the precision level expected from a sleep expert. This statement is supported by the agreement level between the experts that are involved in the dataset, with 0.8259 accuracy, 0.7747 Kappa coefficient and 0.8088 F-score. Thus, the sleep scoring process requires the supervision of an expert.

Taking this into consideration, by limiting the minimum classification probability, the precision of the performed predictions can be increased and the classification of the most doubtful epochs can be delegated to the experts. For instance, by limiting the probability to 0.5 the accuracy is increased to 0.80, the kappa coefficient to 0.75 and the F-score to 0.78, while the amount of classified epochs remains above 92%. Furthermore, by limiting the probability to 0.75 the accuracy is increased to 0.89, the kappa coefficient to 0.85 and the F-score to 0.80, keeping the amount of classified epochs around

65%. Increasing the minimum probability produces more precise predictions, but it also decreases the number of classified epochs, as can be observed in Figure 1.

6. CONCLUSIONS

This work compares the performance of a wide range of sleep scoring methods that are available in the scientific literature. Its objective is to find out which one is more useful in a clinical context. Consequently, it uses several selection criteria, including scoring precision, memory and speed. The results show that the LSTM and V algorithms achieve the highest precision levels, reaching 78% accuracy, 0.72 Kappa coefficient and 0.75 F-score. Also, among them, V is faster and requires less memory. The accuracy of these automatic scoring methods is close, but inferior to the accuracy of a sleep expert, so the former requires the supervision of the latter. Additionally, by delegating to the experts the classification of the most doubtful epochs, the accuracy can be increased to 89%, the Kappa coefficient to 0.85 and the F-score to 0.80 and still classify 65% of the epochs.

RECEIVED: AUGUST, 2022.
REVISED: FEBRUARY, 2023.

REFERENCES

[1] ABADI, M., AGARWAL, A., BARHAM, P., BREVDO, E., CHEN, Z., CITRO, C., CORRADO, G. S., DAVIS, A., DEAN, J., DEVIN, M., GHEMAWAT, S., GOODFELLOW, I., HARP, A., IRVING, G., ISARD, M., JIA, Y., JOZEFOWICZ, R., KAISER, L., KUDLUR, M., LEVENBERG, J., MANÉ, D., MONGA, R., MOORE, S., MURRAY, D., OLAH, C., SCHUSTER, M., SHLENS, J., STEINER, B., SUTSKEVER, I., TALWAR, K., TUCKER, P., VANHOUCKE, V., VASUDEVAN, V., VIÉGAS, F., VINYALS, O., WARDEN, P., WATTENBERG, M., WICKE, M., YU, Y., AND ZHENG, X. (2015): TensorFlow: Large-scale machine learning on heterogeneous systems Software available from tensorflow.org.

[2] ABOALAYON, K., FAEZIPOUR, M., ALMUHAMMADI, W., AND MOSLEHPOUR, S. (2016): Sleep stage classification using EEG signal analysis: A comprehensive survey and new investigation **Entropy**, 18(9):272.

[3] ABOALAYON, K., OCBAGABIR, H., AND FAEZIPOUR, M. (2014): Efficient sleep stage classification based on EEG signals In **Applications and Technology Conference**.

[4] ACHARYA, R., CHUA, E. C.-P., CHUA, K. C., MIN, L. C., AND TAMURA, T. (2010): Analysis and automatic identification of sleep stages using higher order spectra **International Journal of Neural Systems**, 20(6):509–521.

[5] DAKER-HOPFE, H., ANDERER, P., ZEITLHOFER, J., BOECK, M., DORN, H., GRUBER, G., HELLER, E., LORETZ, E., MOSER, D., PARAPATICS, S., SALETU, B., SCHMIDT, A., AND DOFFNER, G. (2009): Interrater reliability for sleep scoring according to the Rechtschaffen & Kales and the new AASM standard **J Sleep Res**, pages 78–84.

[6] DONG, H., SUPRATAK, A., PAN, W., WU, C., MATTHEWS, P. M., AND GUO, Y. (2018): Mixed neural network approach for temporal sleep stage classification **IEEE Transactions on Neural Systems and Rehabilitation Engineering**, 26(2):324–333.

[7] FIORILLO, L., PUIATTI, A., PAPANDREA, M., RATTI, P.-L., FAVARO, P., ROTH, C., BARGIOTAS, P., BASSETTI, C. L., AND FARACI, F. D. (2019): Automated sleep scoring: A review of the latest approaches **Sleep Medicine Reviews**, 48:101204.

[8] FRAIWAN, L., LWEESY, K., KHASAWNEH, N., FRAIWAN, M., WENZ, H., AND DICKHAUS, H. (2010): Classification of sleep stages using multi-wavelet time frequency entropy and LDA **Methods of information in medicine**, 49:230–7.

[9] FRAIWAN, L., LWEESY, K., KHASAWNEH, N., WENZ, H., AND DICKHAUS, H. (2012): Automated sleep stage identification system based on time–frequency analysis of a single EEG channel and random forest classifier **Computer Methods and Programs in Biomedicine**, 108:10 – 19.

[10] GOLDBERGER, A. L., AMARAL, L. A. N., GLASS, L., HAUSDORFF, J. M., IVANOV, P. C., MARK, R. G., MIETUS, J. E., MOODY, G. B., PENG, C.-K., AND STANLEY, H. E. (2000): PhysioBank, PhysioToolkit, and PhysioNet : Components of a new research resource for complex physiologic signals **Circulation**, 101.

[11] HJORTH, B. (1970): EEG analysis based on time domain properties **Electroencephalography and Clinical Neurophysiology**, 29(3):306–310.

[12] KEMP, B., ZWINDERMAN, A. H., TUK, B., KAMPHUISEN, H. A. C., AND OBERYE, J. J. L. (2000): Analysis of a sleep-dependent neuronal feedback loop: the slow-wave microcontinuity of the EEG **IEEE Transactions on Biomedical Engineering**, 47(9):1185–1194.

[13] KHALIGHI, S., SOUSA, T., SANTOS, J. M., AND NUNES, U. (2016): Isruc-sleep: A comprehensive public dataset for sleep researchers 124:180–192.

[14] KOLEY, B. AND DEY, D. (2012): An ensemble system for automatic sleep stage classification using single channel EEG signal **Computers in Biology and Medicine**, 42(12):1186–1195.

[15] LABRADA, A., FEBLES, E. S., AND ANTELO, J. M. (2022): Comparison of automatic sleep stage classification methods for clinical use **Global Clinical Engineering**, 5(1).

[16] MALHOTRA, R. K. AND AVIDAN, A. Y. (2014): **Atlas of Sleep Medicine**, chapter 3, pages 77–99 Saunders, 2 edition.

[17] PEDREGOSA, F., VAROQUAUX, G., GRAMFORT, A., MICHEL, V., THIRION, B., GRISEL, O., BLONDEL, M., PRETTENHOFER, P., WEISS, R., DUBOURG, V., VANDERPLAS, J., PASSOS, A., COURNAPEAU, D., BRUCHER, M., PERROT, M., AND DUCHESNAY, E. (2011): Scikit-learn: Machine learning in Python **Journal of Machine Learning Research**, 12:2825–2830.

[18] PHINYOMARK, A., THONGPANJA, S., HU, H., PHUKPATTARANONT, P., AND LIMSAKUL, C. (2012): **The Usefullness of Mean and Median Frequencies in Electromiography Analysis, Computational Intelligence in Electromiography Analysis - A perspective on Current Applications and Future Challenges** IntechOpen.

[19] RONZHINA, M., JANOUSEK, O., KOLAROVA, J., NOVAKOVA, M., HONZIK, P., AND PROVAZNIK, I. (2012): Sleep scoring using artificial neural networks **Sleep Medicine Reviews**, 16(3):251–263.

[20] ROSENSTEIN, M. T., COLLINS, J. J., AND LUCA, C. J. D. (1993): A practical method for calculating largest Lyapunov exponents from small data sets **Physica D: Nonlinear Phenomena**, 65(1-2):117–134.

[21] SABETI, M., KATEBI, S., AND BOOSTANI, R. (2009): Entropy and complexity measures for EEG signal classification of schizophrenic and control participants **Artificial Intelligence in Medicine**, 47(3):263–274.

[22] ŞEN, B., PEKER, M., ÇAVUŞOĞLU, A., AND ÇELEBI, F. V. (2014): A comparative study on classification of sleep stage based on EEG signals using feature selection and classification algorithms **Journal of Medical Systems**, 38(3).

[23] STARK, H.-G. (2005): **Wavelets and Signal Processing: An application-based introduction** Springer.

[24] SUPRATAK, A., DONG, H., WU, C., AND GUO, Y. (2017): DeepSleepNet: A model for automatic sleep stage scoring based on raw single-channel EEG **IEEE Transactions on Neural Systems and Rehabilitation Engineering**, 25(11):1998–2008.

[25] ŠUŠMÁKOVÁ, K. AND KRAKOVSKÁ, A. (2008): Discrimination ability of individual measures used in sleep stages classification **Artificial Intelligence in Medicine**, 44(3):261–277.

[26] YILDIRIM, O., BALOGLU, U. B., AND ACHARYA, U. R. (2019): A deep learning model for automated sleep stages classification using PSG signals **Int J. Environ. Res. Public Helth**.

[27] ZHANG, Y., YANG, Z., LAN, K., LIU, X., ZHANG, Z., LI, P., CAO, D., ZHENG, J., AND PAN, J. (2019): Sleep stage classification using bidirectional LSTM in wearable multi-sensor systems.