Automatic sleep stage classification with CNN and LSTM

Carlos Fabbri
Universidad del Pacífico
Lima, Perú
c.fabbrigarcia@alum.up.edu.pe

ABSTRACT

A good night sleep serves as the chief propeller of the complex systems in human health and well-being. In recent decades, a global sleep deprivation epidemic has made it necessary to improve sleep quantity and quality. To do so, and adequately immerse in the research and treatment of sleep health, the classification of sleep into its multiple stages (Awake, N1, N2, N3 and REM) is a crucial activity. Automatically performing this classification with the use of deep learning techniques could significantly reduce error rates and increase the efficiency of the task. This study proposes and compares novel methods for automatically identifying sleep epochs based on a single EEG Fpz-Cz channel extracted from the Sleep-EDFx database. The best performing method, an ensemble of CNN and LSTM networks, achieves results that are comparable with the current state of the art (ACC: 85.3%, MF1: 79.6%). Furthermore, a cooperative system between classifier and human sleep scorer is also put forward and proven to have the potential of raising the overall classification accuracy.

1 INTRODUCTION

In light of the overriding contemporary information regarding sleep's effect on health [1, 2], mental capacity [3, 4] and emotional stability [5, 6], the focus of sleep as a public health topic has progressively expanded. In this line of thought, there have been several studies in recent years reporting a significant decrease in sleep quality and quantity among society [**Reference20 Reference24**, 7]. This actuality is of special concern since research in the last decades has been able to translate the effect of poor sleep and sleep deprivation into economic and social problems. For example, Gibson & Shrader [8] concluded that employees that reduce their sleep time by one hour in average tend to see their salaries reduced by 4.9% in the long term. Furthermore, Hafner *et al.* [9] have estimated that the societal costs caused by sleep deprivation take a toll of 2.28% worth of PIB in the case of USA (1.9% in the UK and 2.9% in Japan).

Based on the cited evidence, it is a public urgent matter to take care of human sleep. A key process common to both treatment and research of sleep related problems is sleep monitoring (also called sleep staging or sleep state detection). Sleep monitoring consists in classifying an individual's sleep period into its different possible stages. Based on this classification, doctors and researchers are able to analyze a patient's sleep profile, recognize clinical disorders and prescribe treatment. In addition, in sleep laboratories, annotated sleep is useful to explore and describe pathologies. Therefore, it can be stated that sleep monitoring is relevant to improve sleep health and sleep research.

Sleep monitoring as a clinical technique was introduced in 1968 by Kales and Rechtschaffen (K&R) [10] and until today has remained as a relatively manual task. Health staff is required to observe the signals recorded during a polysomnography[1] which include an electroencephalogram (EEG), electrooculogram (EOG) and electromyogram (EMG), among other signals. The current standard for sleep staging, published by the American Academy of Sleep Medicine (AASM) [11], indicates that sleep should be split into periods of 30-s or 60-s called epochs and subsequently classified into one of 5 possible stages. This stages are: Awake (A), Non-REM 1 (N1), Non-REM 2 (N2), Non-REM 3 (N3) and REM (R). Most of the classification criteria is based on the frequency and amplitude of the signals registered in the EEG. The AASM also provides EMG and EOG criteria that helps to better distinguish stages.

Since it is so dependent on human sight and subjectivity, the activity of classifying sleep stages is tedious and prone to error or reproducibility issues. Classifying a single record of sleep lasts, on average, around 2 hours [12] and the average rate of consistency between sleep scorers is of 80% [13] (1 out of every 5 sleep epochs is misclassified). These two figures portray some of the drawbacks of performing sleep monitoring manually.

Numerous studies have proposed methods to automate this classification task with the objective of making it more efficient and accurate [14–16]. Most of them rely on hand engineered features that are too dependant on the dataset's characteristics and/or haven't been able to reach human precision. A more recent set of studies [17, 18], based in deep learning techniques, has achieved more promising results. While these studies have used several input signal channels to classify sleep (based on combinations of EEG, EOG and EMG), only a few of them attempt to classify stages using a single channel. Being able to classify sleep with one or few channels would decrease the cost of hardware for sleep analysis and eventually enable the manufacture of equipment for at-home and automatic sleep monitoring. The research of Supratak *et al.* [17] proposed a novel method of classification based on Convolutional Neural Networks (CNN) and Long-Short Term Memory Networks (LSTM) that achieved state of the art results in the task. The approach used by the authors shows that using a single channel EEG Fpz-Cz can be enough to achieve human-level precision.

The present paper proposes a method of classification based on CNN and LSTM that uses only the EEG Fpz-Cz channel. The network's architecture, based on the approach from [17], plus a novel set of hyperparameters and training technique allow to achieve results (ACC: 85.3%, MF1: 79.6%) comparable with the current state of the art. The network can perform classification of a new 8-hour sleep recording in less than a second. The study also suggests a system of interaction between automatic classification and human sleep scorer supervision. The purpose of said system is to ensure that the sleep epochs that are harder to classify with the model

[1] A polysomnography is a multi-parametric sleep study used as a diagnostic tool in sleep medicine.

are derived to sleep experts for manual revision. This cooperation between machine and human raises the overall accuracy of the solution.

The paper is structured as follows. Chapter 2 presents the data and classifiers used, as well as the experimental design. The interaction system based on a threshold analysis is also explained. Chapter 3 presents the results of the data preparation and classification, together with a benchmark against current state of the art and human accuracy. The potential use of the interaction system is also proven. Chapter 4 elaborates on the interpretation and implications of the results achieved. Chapter 5 summarizes the insights from the research and puts forward ideas for future research.

2 MATERIALS AND METHODS

The following chapter presents the theoretical standpoint for sleep and its stages, the dataset that will be used for the experiments, the design of the 2 classifiers that will be tested and the experimental design for training and testing them. Additionally, further argumentation and elucidation for the confidence threshold analysis is developed. Finally, the technical specifications for the implementation is laid out.

2.1 Sleep stages

Sleep can be defined as a period of reduced sensibility that all complex organisms experience as part of the daily routine [19]. The sleep process has an internal structure that can be described by sleep stages that are distinguished based on patterns found in cerebral, ocular and muscular activity. In 2007, the AASM published the updated criteria for sleep classification in what is now called the AASM manual [11]. According to the AASM manual, each sleep stage can be recognized by the frequencies of the brain activity recorded by an EEG. Also, each sleep stage occurs in different durations throughout the night. Table 1 shows the normal range of wave frequency for each sleep stage and the average duration of the stage (relative to the full sleep period) according to [20, 21]. As a complement, Figure 1 shows the characteristic EEG wave pattern for each sleep stage.

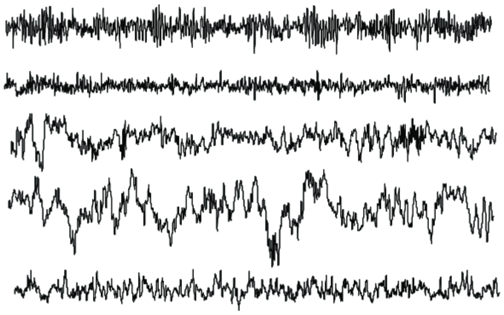

Figure 1: Characteristic EEG wave pattern for each sleep stage. Extracted from Khalighi [22]

Table 1: Sleep stage frequency range and duration

Stage	EEG Frequency	Duration relative to full night of sleep
W	>50% of the epoch should be 8 - 13 Hz.	5%
N1	<50% should be 8 - 13 Hz. No sleep spindles or K-complex should be present.	2% - 5%
N2	<20% should be 0 - 4 Hz. Presence of sleep spindles or K-complex	45% - 55%
N3	>20% should be 0 - 4 Hz.	15% - 25%
REM	Low-voltage mixed-frequency	20% - 25%

2.2 Data

The dataset used for this study is extracted from the Sleep-EDF's "Sleep Cassette Study Data" which is a result of the study conducted by Kemp *et al.* in [23]. The data was collected using "walkman-like devices" that measured the subjects' brain activity (EEG) and other signals for approximately 20 hour periods that include sleep in the middle. The data, in form of time series, is annotated with corresponding sleep stage by a team of experts according to the K&R rules for sleep classification. In total, 39 annotated sleep recordings from 20 subjects are available in PhysioNet[2] in an EDF format[3]. The selection of Sleep EDF's database as source for the dataset is also based on the fact that it is one of the most common databases in recent works and guarantees a fair benchmark of the results.

Although the recordings include several signals from each subject, we will consider only the EEG Fpz-Cz. Different studies [14, 17, 25] have proved that using only this channel can be enough to classify sleep with human-level accuracy. Tsinalis *et al.* explain that the Fpz-Cz is an ideal channel because it captures most of the brain's activity phenomena that are necessary to discriminate between stages [14]. As was mentioned beforehand, if we can continue to demonstrate the sufficiency of the EEG Fpz-Cz channel, new possibilities of highly accurate at-home automatic sleep monitoring can appear to detect and aid certain sleep disorders. Also, the signal is fragmented into periods of 30 seconds called epochs that will be considered the data points of the dataset.

2.3 Experiment 1: CNN with concatenated epochs (CNN Concat)

The first experiment conducted in the study uses a CNN for feature extraction of a concatenation of neighboring epochs (30 second sleep periods) and a classification with a final Softmax layer. This approach is inspired from the results reported by Chambon *et al.* [25] which proved that when a sleep classifier has access to features of previous and future epochs for a given epoch, the performance increases.

[2] https://physionet.org/content/sleep-edfx/1.0.0/

[3] For detail about EDF format please refer to [24].

The architecture of the CNN is based on [17] and is as follows. Two main branches are defined. One branch extracts temporal information through short filters and strides; while the other, through longer filters and strides, learns frequency information from the waves represented in the time series. Each branch, independently, has the next sequence: 1D-CONV →MP →DROP →1D-CONV →1D-CONV →1D-CONV →MP. The specific parameters of each layer are defined in Figure 2. Each 1D convolution layer has 3 operations: convolution, batch normalization and ReLU activation function. Finally, the outputs from both branches are flattened and concatenated before being fed into a Softmax layer of 5 units (one for each possible sleep stage).

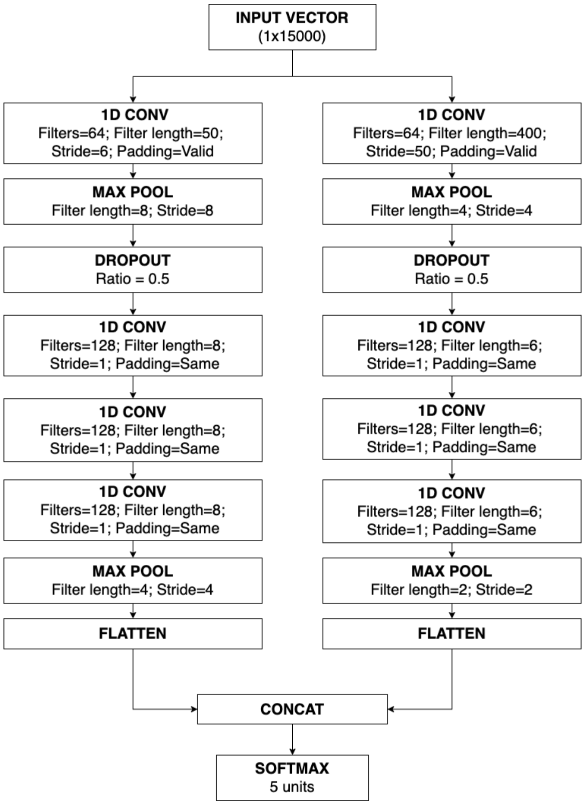

Figure 2: Architecture of the CNN Concat classifier based on [17]

The network's left branch learns temporal information and sudden changes of signal amplitude and the right branch learns to extract frequency information.

For training and prediction, data points are arranged sequentially and concatenated in sets of 5. For each set, the sleep stage from the middle data point was taken as the target variable. For example, for the prediction of data point (X_i, Y_i), the new vector $X_{i'}$ comes from the concatenation of vectors $X_{i-2}, X_{i-1}, X_i, X_{i+1}, X_{i+2}$ where i represents the index of the data point; while the new target vector $Y_{i'}$ is the same as Y_i. Consequently, each vector or data point changes its shape to five times its original size because each input vector is now a representation of a period of 150 seconds, instead of 30. Figure 3 shows a diagram of how the data points or sleep epochs are concatenated. Each sleep recording is appropriately zero padded to avoid the classifier mixing data points from different recordings.

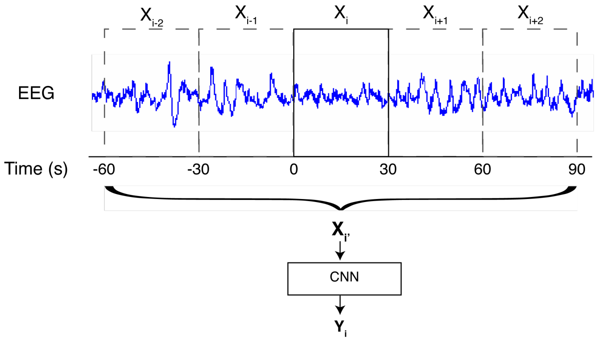

Figure 3: Data point concatenation for the CNN's input

For each original data point X_i, we concatenate $X_{i-2}, X_{i-1}, X_i, X_{i+1}, X_{i+2}$ to create a new vector$X_{i'}$ that contains information from neighboring data points.

2.4 Experiment 2: CNN + LSTM

The second experiment conducted is an ensemble of CNN and LSTM. The results of this experiment are compared with the results from the first experiment to determine which of the two strategies, CNN with vector concatenation or CNN+LSTM, is more effective in learning temporal dependencies for sleep stage classification.

For this experiment, the same CNN architecture presented in Figure 2 is used but with changes to the inputs and the outputs. In this case, each data point is fed directly into the CNN without any previous concatenation with neighboring epochs. Also, instead of using the outputs from the Softmax layer, the activations from the penultimate layer are stored into lists and used as features vectors from each datapoint. In this experiment, the CNN is used mainly as a feature extractor so the Softmax layer's activations are not needed. The activations of the previous layer, however, serve as a summary of the information that was extracted from each raw data point. Therefore, these activations are considered the feature vectors of each data point and are subsequently used as input vectors for the LSTM. The shape of the feature vectors is of 1×2688.

Since each data point is fed independently into the CNN, the resulting feature vectors are then arranged sequentially in timesteps of 5 by 5 to obtain new input vectors of shape 5×2688 for the LSTM component. The architecture of the LSTM consists of a main sequence of layers: DROP →BiLSTM →DROP →BiLSTM →DROP. In parallel, the output of the first dropout layer is processed with a fully connected (FC) layer. Then, to join both paths, a shortcut connection performs addition of the outputs of the first dropout layer and the FC layer. The result of the addition then follows the final sequence of DROP →FLATTEN and Softmax as a final layer for classification. The architecture of the CNN + LSTM ensemble with detailed parameters is presented in Figure 4. Bidirectional LSTM cells are used because the network should be able to learn from temporal dependencies in both directions since the classification of the sleep recordings is performed at the end of night, when the EEG signal for the complete sleep duration is available. Additionally, it is important to highlight that each component of the system is trained independently, i.e., the CNN is being used as a feature extractor and the LSTM as the classifier.

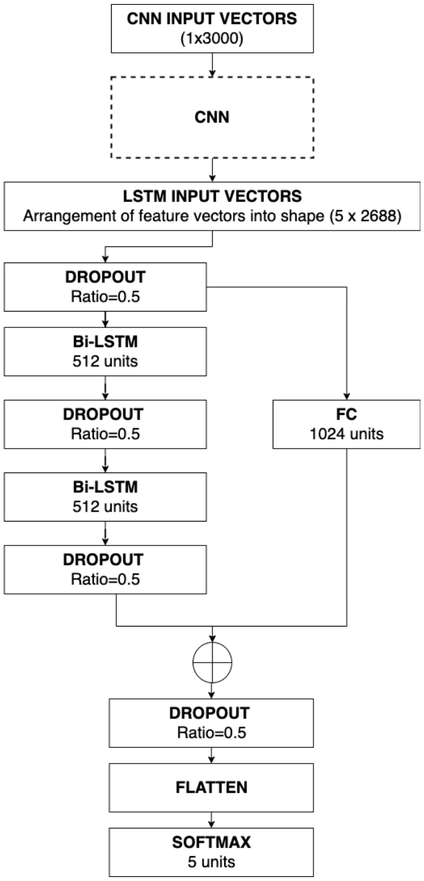

Figure 4: Architecture of the CNN + LSTM classifier based on [17]

The CNN block represents the CNN architecture presented in Fig. 2.

2.5 Experimental design and metrics

In both experiments the training and validation is done with a 20-fold stratified cross validation based on the Scikit-learn implementation. In each fold, the models are trained with 95% of the dataset and produce predictions for the 5% remaining. After the 20 iterations, the predictions of all folds are concatenated into a $\hat{Y}$ vector and compared with the Y vector of true values to estimate performance metrics.

The main metrics used to compare the classifiers are: accuracy (ACC), precision per class (P_C), recall per class (R_C), F1 score per class ($F1_C$) and Macro-F1 score (MF1). Time is also measured in minutes and seconds (mm:ss) to compare training and classification speed. The following formulas define how each metric is computed.

$$ACC = \frac{\sum_{C=1}^{C} TP_C}{N}$$

$$P_C = \frac{TP_C}{TP_C + FP_C}$$

$$R_C = \frac{TP_C}{TP_C + FNC}$$

where TP_C, FP_C, FN_C are the true positives, false positives and false negatives of class C, respectively; and N is the number of all the data points in the dataset.

$$F1_C = \frac{2}{\frac{1}{P_C} + \frac{1}{R_C}}$$

$$MF1 = \frac{\sum_{C=1}^{C} F1_C}{C}$$

where $F1_C$ is the F1 score of class c and C is the number of classes, in this case sleep stages.

2.6 Confidence Threshold

Another important aspect of the study is to analyze whether the classifier can establish a cooperative relationship with human sleep scorers. In this sense, the use of a confidence threshold is proposed. The idea behind this threshold is to distinguish between high confidence and low confidence classifications. In the case of the CNN + LSTM classifier, the activations from the final Softmax layer are considered to represent the degree of confidence with which the network produced each prediction. For instance, given a data point with corresponding softmax layer of $[0.01; 0.1; 0.9; 0.2; 0.01]$, we consider that the model has classified the vector as an N2 epoch (third unit in the layer) with a 0.9 of confidence, meaning, the probability of the network being correct.

Intuitively, predictions with low confidence are likely to correspond to data points that exhibit uncommon signals and frequencies that the classifier did not have the chance to properly learn to recognize during the training phase. Therefore, these sleep epochs should be derived to sleep scoring experts for manual and careful revision and annotation. By deriving this less obvious ("trickier") epochs to manual scorers, it is expected that the general performance of the classifier will increase.

The aim of the confidence threshold analysis is to define threshold values that allows to derive a portion of the dataset but not the majority of it. For example, deriving 10% or 20% of the least confident predictions could be an ideal result. In Section 3.5, an empiric analysis is held to define a set of threshold values to accomplish said percentages.

2.7 System and runtime

Training and validation of the models were implemented using Tensorflow [26] web) with a Keras backend in a Python environment from Google's Colaboratory [4]. The virtual machine deployed is available for free and presents the following characteristics: an Nvidia Tesla P100 GPU, a dual-core Intel Xeon processor and 12 GB of memory. The confidence threshold analysis is performed in the same machine.

[4] https://colab.research.google.com/

3 RESULTS

This chapter report the most relevant results. These include the results of the data processing, the 2 proposed classifiers and the confidence threshold analysis. The performance metrics of the classifiers are also compared with current state of the art and human-level performance.

3.1 Data processing

The data was loaded in Python using the MNE library[5] which has a variety of classes and methods for processing biosignals. From the 20 hour long recordings, the EEG Fpz-Cz channel was filtered and the portion of the recording corresponding to sleep was selected and fragmented into 30-second epochs to make up the data points. Each epoch was annotated with their corresponding sleep stage that was extracted from the recording's hypnogram. Epochs annotated as N4 stage were edited to N3 stages to meet current AASM standards of sleep staging. To include the awake stage for each recording, 30 epochs before the first sleep epoch and 30 epochs after the last sleep epoch were added. No signal filtering or processing was applied since the signals had been already filtered with low (0.5 Hz) and high (100 Hz) pass filters. The EEG Fpz-Cz present in the database was originally recorded with a sampling rate of 100 Hz: this means that each 30 second-epoch is a time series of 3000 observations (30 x 100).

After applying all mentioned steps to the database, 42,308 data points were obtained for the dataset. On average, each sleep recording yields 1,084 data points. Each data point is made up of a numerical time series of dimensions 1x3000 and a scalar that represents the sleep stage that can take the value of 0 to 4. Each data point can be plotted to reveal the 30-s EEG that represents. Figure 5 shows a data point plotted. The distribution of the data points by sleep stage is shown in Table 2. It can be seen that the percentage of data points varies for each sleep stage, as is expected because of the different durations of stages during sleep (as was discussed in Section 2.1). Particularly, stage N1 is the least frequent stage, with 6.63% of the data points.

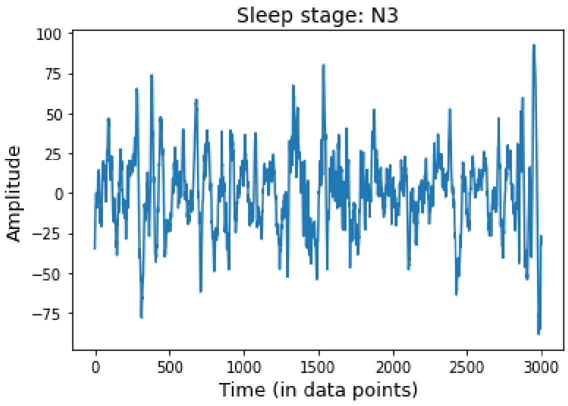

Figure 5: Plot of a data point

Table 2: Distribution of sleep stages in the dataset

Sleep stage	Label	Quantity	Percentage of total
W	0	8,285	19.58
N1	1	2,804	6.63
N2	2	17,799	42.07
N3	3	5,703	13.48
REM	4	7,717	18.24
Total	-	**42,308**	**100**

3.2 CNN Concat

Training of the CNN with concatenated epochs (CNN Concat) was performed by minimizing the categorical cross entropy. The Adam Optimizer was used with parameters $lr = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.99$. The training was performed in mini batches with a size of 100 data points per batch over 6 epochs.

The classifier attains an accuracy of 81.78% and MF1 score of 75.79%. The classifier's metrics in detail are presented in Table 3 and its confusion matrix can be found in Figure 6. The average training time for each fold was of 1:26 (1 minute and 26 seconds) and the average prediction time for a new sleep recording was less than 1 second.

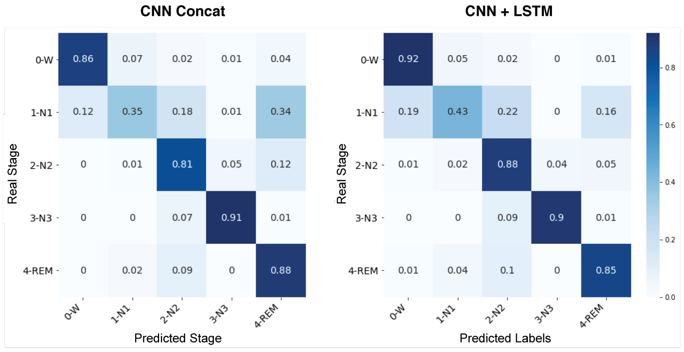

Figure 6: Confusion matrices for CNN Concat (left) and CNN + LSTM (right) classifiers

Values have been normalized horizontally dividing by the total amount of real labels. The main diagonal, therefore, represents the recall score for each sleep stage.

3.3 CNN + LSTM

The CNN + LSTM classifier was trained in two separate phases. First, The CNN used for feature extraction is trained with the same parameters as mentioned in the previous section for 19 epochs. Afterwards, the LSTM network is trained by minimizing the categorical cross entropy with an Adam optimizer ($lr = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.99$). The LSTM training is performed in mini batches with a size of 100 data points over 5 epochs.

The classifier achieves an accuracy of 85,30% and a MF1 score of 78,56%. The detail of the classifier performance is shown in Table

[5] https://mne.tools/stable/index.html

Table 3: Comparison of classifiers with state of the art and human performance

Experiment/ Reference	ACC	MF1	F1 per class					Training time (mm:ss)	Classification time
			0 - W	1 - N1	2 - N2	3 - N3	4 - REM		
CNN Concat	81.78	75.79	89.93	40.83	85.01	87.72	75.46	1:26	<1s
CNN + LSTM	85.30	79.56	91.40	46.77	88.02	88.78	82.82	2:18	<1s
Supratak *et al.* [17]	82.00	76.90	84.70	46.60	85.90	84.80	82.40	-	-
Human performance [12, 13]	80.00	-	-	-	-	-	-		2h - 4h

In each column, the classifier with the best performance appears underlined.

3, together with a comparison against current state of the art and estimated human-level accuracy. The confusion matrix for the CNN + LSTM classifier is presented in Figure 6. The average training time for each fold was of 2:18 (2 minutes and 18 seconds) and the average prediction time for a new sleep recording was less than 1 second.

To further exhibit the accuracy of the CNN + LSTM classifier, a comparison between a real hypnogram and the classifier's predicted hypnogram is shown in figure 7. The hypnogram shown corresponds to a sleep recording in which the classifier obtains an accuracy that is similar to the overall accuracy of the classifier. In particular, the hypnogram exposed originates from sleep recording SC4142E0, in which the model reached an 85.80% of accuracy which can also be considered the percentage of similarity between both line plots.

3.4 Comparison with state of the art and human performance

To establish a benchmark with the state of the art, we considered studies that met certain criteria to make the results comparable. First, they have to be studies that use a single EEG channel as input for the classifier. Second, they have to train the classifier or classifiers with the Sleep-EDF database to guarantee that the data comes from the same distribution and level of detail. Third, the features from the data should be extracted with non-supervised techniques. This means that studies that use hand-engineered features with the scoring rules embedded in them are not comparable with our results. Fourth, the results reported in the studies should proceed from a cross-validation or a hold-out set.

After looking for studies that met this criteria, we concluded that the current state of the art is achieved by Supratak *et al.*, whose classifier achieves $ACC = 82.0\%$ and $MF1 = 76.9\%$. As it has been reported in previous sections, the CNN + LSTM classifier presented in this paper achieves comparable performance of $ACC = 85.3\%$ and $MF1 = 79.6\%$. Table 3 shows a comparison with state of the art performance.

Although there is no exact answer, some studies have attempted to quantify the accuracy of human scorers when classifying sleep stages as well as estimating the average duration of the task when performed by a human. Danker-hopfe *et al.* [13] estimate that the average inter-scorer consistency [6] is in the order of 80%; this number can be considered the accuracy of classification from human expert scorers. Time-wise, Malhotra *et al.* [12] indicate that an expert sleep scorer invests at least 2 hours to classify a single sleep recording. These estimations are included in Table 3 to enable the comparison between automatic classifiers and human sleep scorer.

3.5 Confidence threshold analysis

Following what was explained in Section 2.6, this study aims to deliver a demonstration of how the classifier could be used cooperatively with human sleep scorers in the context of a sleep laboratory.

Once the CNN + LSTM predictions were computed for the whole dataset, the activations of the output (Softmax) layer were extracted as an indicator of the confidence for each prediction. Afterwards, thresholds values were empirically set to split the predictions between high confidence and low confidence. Each threshold value produces a different percentage of epochs derived for manual scoring as well as a new accuracy measured for the remaining epochs. The threshold values 0.584; 0.702; and 0.874 were found to derive (or discard) the 5, 10 and 20 percent, respectively, of the least confident predictions according to the CNN + LSTM classifier. In each case, with the epochs remaining, a new accuracy was computed. The accuracies associated with each threshold are shown in Table 4. It becomes evident that as the amount of derived epochs gets higher, the accuracy of the remaining epoch increases. The implications will be discussed in the next chapter.

Table 4: Derived epochs and new accuracy for 3 possible threshold values

Threshold value	Derived epochs	Percentage of derived epochs	ACC
0 (without threshold)	0	0	85.30
0.5843	2,116	5	87.50
0.7017	4,203	10	89.32
0.8742	8,462	20	92.73

4 DISCUSSION

The following chapter discusses the interpretations and implications of the results.

Starting with the results revealed in Table 3, it is clear that the CNN + LSTM classifier achieves the best classification performance

[6] The "inter-score consistency" is the percentage of epochs that get identified with the same sleep stage by 2 or more human expert scorers.

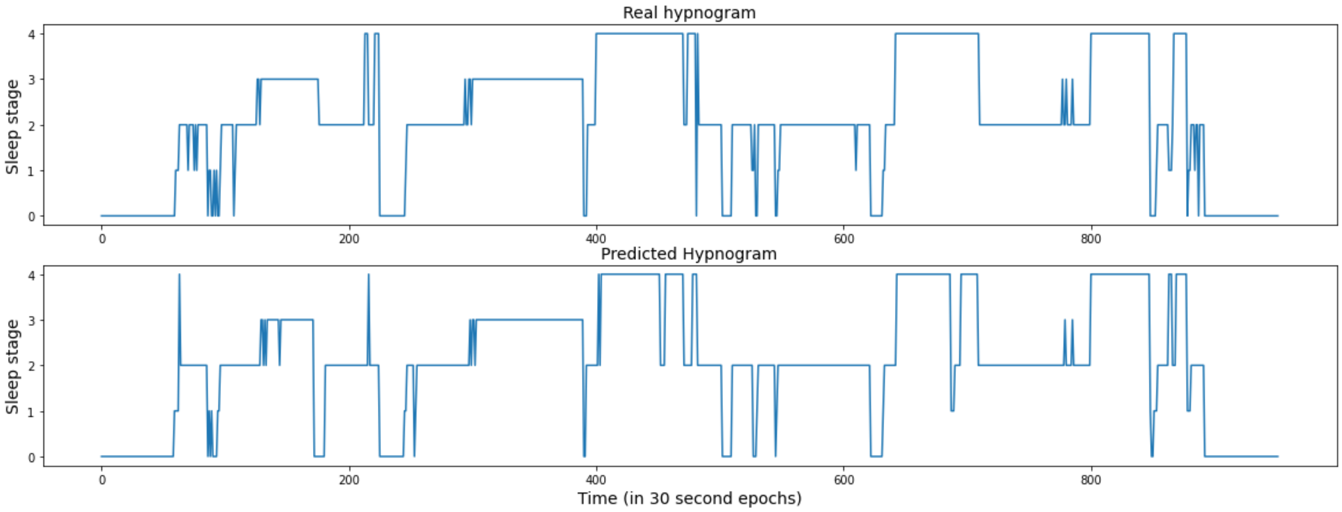

Figure 7: Real hypnogram (top) and predicted hypnogram (bottom) for sleep recording that is representative of the general accuracy of the CNN + LSTM classifier

of the methods explored. Specifically, accuracy is improved in 3.52 percentage points and MF1 in 3.77, with respect to the CNN with concatenated features. For an 8-hours night of sleep, this can mean approximately 34 additional correctly classified epochs. This leads to the conclusion that, for sleep staging, an LSTM component is more effective in learning time dependencies than just concatenating the data points as input for the classifier.

Continuing with the interpretation of Table 3, it is noticed that N1 Stage is the one with the lowest F1 score in comparison with the rest of sleep stages (46.77 in CNN + LSTM). This is expected because it is the least common stage in the dataset with just 6.63% of frequency (see Table 2) and morphologically is not very different from other stages like awake stage (see Figure 1). Therefore, achieving an acceptable F1 for N1 stage has proven to always be a challenge in past studies. In the case of this paper, the author believes that using a stratified sampling for the cross validation propitiated promising results that allowed to reach a higher $F1_{N1}$ score in comparison to the state of the art.

Considering now the hypnogram comparison presented in Figure 7, it is observed that, in general terms, they are similar plots. A dissimilarity found is that at the beginning of the recording the classifier predicts incorrectly REM (code 4) epochs. The fact that the classifier predicted REM stage at the beginning of the recording is not entirely rare, since it was observed in the dataset that some sleep recordings can begin transitioning directly from Awake stage to REM stage. For the remaining of the hypnogram, the plots don't differ significantly, i.e., the errors don't alter sharply the overall sleep profile for the patient.

In comparison with the current state of the art, which is considered to be set in [17], is it concluded that the present study is competitive against it and even presents a slight improvement. In regard to the benchmark against human accuracy, assuming the estimations done by Danker-hopfe *et al.* and Malhotra *et al.* [12] are correct in relation to human precision and classification speed, then the CNN + LSTM classifier opens the chance to increase the accuracy by 5.3 points and reduce the duration of the task by more than 99%. It is noteworthy that this increase in accuracy is achieved by using a single EEG channel. Furthermore, these results help to lay the foundations for the production of hardware for at-home, automatic sleep monitoring with the same precision as a polysomnography.

Finally, the confidence threshold method proposed was proven to be a valid way of combining the strengths of the automatic classifier and the human scorers. By working with a threshold, the network's speed and objectivity can be harnessed to classify the majority of the epochs; while, at the same time, the expert scorer's insight and meticulousness can be of help for the more complex and confusing portions of the sleep recording. It should be noted that, since sleep scorers don't have to classify the entirety of the recordings, they should be less worn out when their help is needed to classify the more complex epochs. Further, this is a chance to give doctors the ability to allocate their time to other highly necessary activities, instead of engaging in repeated sleep recording classification. This can include duties such as focusing on patients' diagnosis or advanced medical research, which are not redundant in nature and require more meticulous human analysis to be completed.

5 CONCLUSION

This paper has proven that an automatic classifier based on CNN and LSTM is capable of classifying sleep stages with a higher precision that human-level sleep scoring and in significantly less time. In particular, the methodology proposed in this study can be compared with the current state of the art from papers with similar characteristics.

The confidence threshold analysis allows to suggest that a cooperation between automatic sleep classifier and expert sleep scorer should be established to enhance the overall performance of the classification by requiring human intervention only for the epochs that are truly confusing and complex.

Some of the work's limitations will be illustrated as future work. First, The low amount of N1 samples can be addressed by accessing a larger database or by applying data augmentation techniques. Additionally, to further exploit the convolution operations in the CNN, certain techniques can be explored to map the 1-dimensional representation of the brain's activity into 2-dimensional, image-like, arrays. Sharma *et al.* [27] present a technique that has recently been proven to enhance results in similar applications.

REFERENCES

[1] Amneet Sandhu, Milan Seth, and Hitinder S Gurm. "Daylight savings time and myocardial infarction". In: *Open Heart* 1.1 (2014).

[2] Aric A Prather et al. "Behaviorally assessed sleep and susceptibility to the common cold". In: *Sleep* 38.9 (2015), pp. 1353–1359.

[3] Seung-Schik Yoo et al. "A deficit in the ability to form new human memories without sleep". In: *Nature neuroscience* 10.3 (2007), p. 385.

[4] Matthew P Walker. "Sleep to Remember: The brain needs sleep before and after learning new things, regardless of the type of memory. Naps can help, but caffeine isn't an effective substitute". In: *American Scientist* 94.4 (2006), pp. 326–333.

[5] Matthew P Walker and Els van Der Helm. "Overnight therapy? The role of sleep in emotional brain processing." In: *Psychological bulletin* 135.5 (2009), p. 731.

[6] Els Van Der Helm, Ninad Gujar, and Matthew P Walker. "Sleep deprivation impairs the accurate recognition of human emotions". In: *Sleep* 33.3 (2010), pp. 335–342.

[7] Emmanuel H During. "The Epidemic of Sleep Deprivation: A Modern Curse". In: (2017).

[8] Matthew Gibson and Jeffrey Shrader. "Time use and productivity: The wage returns to sleep". In: (2014).

[9] Marco Hafner et al. "Why sleep matters—the economic costs of insufficient sleep: a cross-country comparative analysis". In: *Rand health quarterly* 6.4 (2017).

[10] A Rechtschaffen and A Kales. "A manual of standardized terminology, technique and scoring system for sleep stages of human sleep". In: *Brain Information Service, Los Angeles* (1968).

[11] Conrad Iber. *The AASM manual for the scoring of sleep and associated events: rules, terminology and technical specifications.* Vol. 1. American Academy of Sleep Medicine Westchester, IL, 2007.

[12] Atul Malhotra et al. "Performance of an automated polysomnography scoring system versus computer-assisted manual scoring". In: *Sleep* 36.4 (2013), pp. 573–582.

[13] Heidi Danker-hopfe et al. "Interrater reliability for sleep scoring according to the Rechtschaffen & Kales and the new AASM standard". In: *Journal of sleep research* 18.1 (2009), pp. 74–84.

[14] Orestis Tsinalis, Paul M Matthews, and Yike Guo. "Automatic sleep stage scoring using time-frequency analysis and stacked sparse autoencoders". In: *Annals of biomedical engineering* 44.5 (2016), pp. 1587–1597.

[15] Rajeev Sharma, Ram Bilas Pachori, and Abhay Upadhyay. "Automatic sleep stages classification based on iterative filtering of electroencephalogram signals". In: *Neural Computing and Applications* 28.10 (2017), pp. 2959–2978.

[16] Ahnaf Rashik Hassan and Abdulhamit Subasi. "A decision support system for automated identification of sleep stages from single-channel EEG signals". In: *Knowledge-Based Systems* 128 (2017), pp. 115–124.

[17] Akara Supratak et al. "DeepSleepNet: A model for automatic sleep stage scoring based on raw single-channel EEG". In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 25.11 (2017), pp. 1998–2008.

[18] Siddharth Biswal et al. "Expert-level sleep scoring with deep neural networks". In: *Journal of the American Medical Informatics Association* 25.12 (2018), pp. 1643–1650.

[19] Chiara Cirelli and Giulio Tononi. "Is sleep essential?" In: *PLoS biology* 6.8 (2008), e216.

[20] James D Geyer, Paul R Carney, and Troy Payne. *Atlas of polysomnography.* Lippincott Williams & Wilkins, 2012.

[21] Mary A Carskadon, William C Dement, et al. "Normal human sleep: an overview". In: *Principles and practice of sleep medicine* 4 (2005), pp. 13–23.

[22] Sirvan Khalighi et al. "Automatic sleep staging: A computer assisted approach for optimal combination of features and polysomnographic channels". In: *Expert Systems with Applications* 40.17 (2013), pp. 7046–7059.

[23] Bastiaan Kemp. "Model-based monitoring of human sleep stages". In: (1987).

[24] Bob Kemp et al. "A simple format for exchange of digitized polygraphic recordings". In: *Electroencephalography and clinical neurophysiology* 82.5 (1992), pp. 391–393.

[25] S. Chambon et al. "A Deep Learning Architecture for Temporal Sleep Stage Classification Using Multivariate and Multimodal Time Series". In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 26.4 (2018), pp. 758–769. ISSN: 1534-4320. DOI: 10.1109/TNSRE.2018.2813138.

[26] Martín Abadi et al. "Tensorflow: A system for large-scale machine learning". In: *12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16).* 2016, pp. 265–283.

[27] Alok Sharma et al. "DeepInsight: A methodology to transform a non-image data to an image for convolution neural network architecture". In: *Scientific reports* 9.1 (2019), pp. 1–7.