Scalable automatic sleep staging in the era of Big Data

Takashi Nakamura, Harry J. Davies and Danilo P. Mandic

***Abstract*— Numerous automatic sleep staging approaches have been proposed to provide an eHealth alternative to the current gold-standard – hypnogram scoring by human experts. However, a majority of such studies exploit data of limited scale, which compromises both the validation and the reproducibility and transferability of such automatic sleep staging systems in the real clinical settings. In addition, the computational issues and physical meaningfulness of the analysis are typically neglected, yet affordable computation is a key criterion in Big Data analytics. To this end, we establish a comprehensive analysis framework to rigorously evaluate the feasibility of automatic sleep staging from multiple perspectives, including robustness with respect to the number of training subjects, model complexity, and different classifiers. This is achieved for a large collection of publicly accessible polysomnography (PSG) data, recorded over 515 subjects. The trade-off between affordable computation and satisfactory accuracy is shown to be fulfilled by an extreme learning machine (ELM) classifier, which in conjunction with the physically meaningful hidden Markov model (HMM) of the transition between the different sleep stages (smoothing model) is shown to achieve both fast computation and highest average Cohen's kappa value of $\kappa = 0.73$ (Substantial Agreement). Finally, it is shown that for accurate and robust automatic sleep staging, a combination of structural complexity (multi-scale entropy) and frequency-domain (spectral edge frequency) features is both computationally affordable and physically meaningful.**

I. Introduction

Polysomnography (PSG) is the gold standard for monitoring sleep and evaluating patients' sleep disorders in a clinic. The recorded physiological data are visually reviewed and manually scored by a human expert, according to some given guidelines, such as the American Academy of Sleep Medicine (AASM) guideline [1]. Automatic sleep staging approaches aim to replicate human scoring based on PSG data, which both provides economic savings, and normally utilise supervised machine learning (ML) techniques with fine-tuned feature extraction for PSG data, and such studies typically propose novel combinations of features and classifiers [2]. Prediction performances of such ML techniques are evaluated using either publicly available datasets (such as [3], [4]) or proprietary data recorded as part of research projects [5]. However, only very few studies have demonstrated the feasibility of automatic classification, while catering rigorously for the generalisation ability for new data.

The obstacles which prevent more widespread generalisation of findings in automatic sleep staging include numerous issues starting with the recording configuration during data acquisition; there are not always the same as the current standard – the AASM. For example, one of the most widely used datasets in automatic sleep staging research is Sleep-EDF [4], however, their EEG montage is Fpz-Cz and Pz-Oz (*cf.* C3-A2 and C4-A1 in AASM). Next, the size of data is usually small, typically recorded from less than 100 subjects; In other words, to become a robust substitution for labour-intensive human scoring, an ideal automatic scoring system needs to accurately capture not only the sleep patterns in its training data, but also to generalise well for unseen sleep data coming from different acquisition montages, different nights of the same subjects and different subjects.

Recently, some large-scale and heterogeneous sleep PSG datasets have been made publicly available [6]. Such 'Big Data' were created with the aim to validate the performance of ML methods when predicting sleep stages based on EEG from diverse patients; some automatic sleep staging approaches have also been rigorously validated using large-scale data [7], [8]. For analyses with large-scale data, the level of scalability of the algorithm becomes a critical issue, together with the computational cost (e.g. execution time, resources), which increase with the size of data. Therefore, for automatic sleep staging in the real-world to become a reality, it is not only the performance of an algorithm but also the computational costs that need to be scrutinised among different methods.

Here, we revisit our recent automatic sleep staging frameworks from the viewpoints of feasibility and compatibility with the Big Data paradigms. This is achieved over the following aspects, which are fundamental for working systems in eHealth but almost ignored in the open literature:

1) Utilise a large-scale sleep data (from 515 individuals) to validate the feasibility of classification methods;
2) Conduct comparative classification over multiple feature extraction methods and classifiers, including the technique proposed in [9], which was validated using the Sleep-EDF dataset (from 61 individuals, the channel configuration was not C3-A2 and C4-A1);
3) Investigate the variability of performance based on the number of training data and model complexity, following suggestions in [8];
4) Compare the performance not only in terms of the accuracy but also the computation time, both of which are critical for Big Data analyses.

Through these comprehensive and fair analyses, we have confirmed the viability of automatic sleep staging with both structural complexity and frequency domain features.

TN, HJD, and DPM are with epartment of Electrical and Electronic Engineering, Imperial College London, SW7 2AZ, UK. {takashi.nakamura14, harry.davies14, d.mandic }@imperial.ac.uk

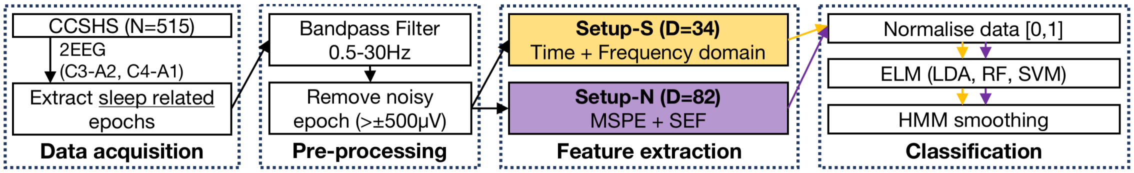


Fig. 1. The flowchart of this study

TABLE I

Proportion of sleep stages. No. of epochs (ratio %)

	Training – $N_{tr} = 258$	Testing – $N_{te} = 257$
W	32557 (11.9%)	35095 (13.0%)
N1	9694 (3.5%)	9278 (3.4%)
N2	125703 (45.8%)	122423 (45.4%)
N3	55737 (20.3%)	54133 (20.1%)
R	50531 (18.4%)	48765 (18.1%)

II. Methods

A. Data sets

We used the dataset from the Cleveland Children's Sleep and Health Study (CCSHS) [10]. It contains PSG recordings from 515 individuals. Two EEG channels (C3-A2 and C4-A1) are used for this study. The data were ordered according to their ID numbers; the first 258 PSGs (from ID001 to ID460) were assigned as training data whereas the rest of 257 PSGs (from ID464 to from ID906) were used for testing.

B. Preprocessing

Some recordings in the CCSHS dataset contain both a large number of pre-sleep wake epochs and post-sleep wake epochs since some of the recordings were conducted over both a day and night period. First, we only extracted 'sleep-related' data; we considered periods between 20 minutes before the first scored sleep epoch, and 20 minutes after the last scored sleep epoch. The so-extracted two channels of EEG data were filtered using a fourth-order Butterworth filter with the passband 0.5-30 Hz. We then deployed another EEG rejection method based on the signal amplitude. Any epoch with absolute voltage of more than 500 μV was removed from further analyses. In total, 1.4 % of the epochs were removed from further analyses. Table I shows the proportion of sleep data in this study.

C. Feature Extraction

For a fair comparison with latest automatic sleep staging results with large scale data [8] and our previous study [9], we extracted features from each 30-second epoch in the same way as in [8] (*Setup-S*) and [9] (*Setup-N*): Setup-S), with time domain features (line length, kurtosis, sample entropy), frequency domain features (spectrogram and kurtosis of spectrogram), Setup-N), with multi-scale permutation entropy (MSPE) and spectral edge frequency (SEF). The details of features extraction can be found in the Supplemental materials[1] and [8], [9]. From the two EEG channels with 30-seconds epoch, 34 features were extracted as Setup-S, while 82 features were extracted for Setup-N. Each feature value, x, was transformed as $\text{sign}(x)\log_{10}(|x+1|)$, and normalised to $[0, 1]$, for each subject.

D. Classifier

After comprehensive testing for accuracy and computational demands, we employed an extreme learning machine (ELM) classifier [11], while linear discriminant analysis (LDA), random forests (RF), and support vector machine (SVM) were used for comparison of prediction performance and computational time. The details of the parameters and setups can be found in the online Supplement[1].

E. Smoothing

The hidden Markov model (HMM) was reported as being effective in improving the prediction accuracy using such sleep transitional information [8]. We assumed that the hidden states were hypnogram scored by a human in the training data, whilst the observations were predicted labels of the same training data. The transition matrix was given by counting the transition of the hypnogram in the training data, and the emission matrix was the obtained confusion matrix of training data (hypnogram vs prediction). We used the Viterbi algorithm to find the sequence of hidden states, which corresponds to 'smoothed' prediction by the algorithm.

III. Results

The preprocessing, feature extraction, and smoothing analyses were performed in Matlab 2016b, and the classifier was implemented in Python 2.7.12, operated on an iMac with 2.8GHz Intel Core i5, and 16GB of RAM. The versions of `sklearn` and `hpelm` were respectively 0.19.1 and 1.0.10.

A. ELM and the number of training subjects

First, we conducted experiments using the ELM classifier and by changing the number of training subjects (N_{tr}), ranging from 10 to 258. For each experiment with the selected number of training subjects, N_{tr}, we repeated the simulation ten times; this was achieved independently and for randomly chosen subsets from training data (except for the experiment of $N_{tr} = 258$, since all training data were used). The initial hidden weights and bias of ELM were randomly generated. For each simulation, we trained the ELM model and computed the Cohen's Kappa value using the confusion matrix derived from the same training subset (training performance); and then, the testing performance was evaluated by computing the kappa values of the confusion matrix, which were derived from all testing data ($N_{te} = 257$). The

[1] https://sites.google.com/site/tkshnakamura/home/materials

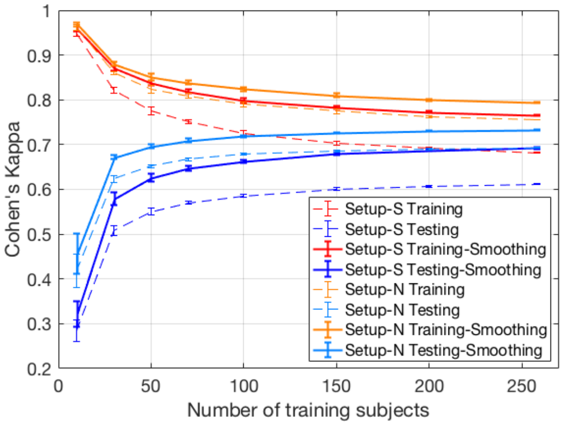


Fig. 2. Cohen's kappa values for different numbers of training subjects in Setup-S and Setup-N ($L = 5000$). The error bar shows standard deviation of over 10 independent iterations.

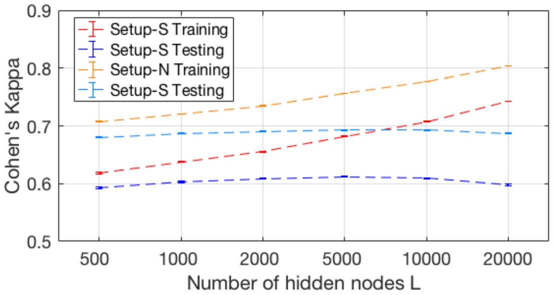


Fig. 3. Cohen's kappa values for different numbers of hidden nodes in Setup-S and Setup-N ($N_{tr} = 258$). The error bar shows standard deviation of over 10 independent iterations.

TABLE II

Computational time of different analyses

Training/	Setup-N			Setup-S
Testing (sec)	$L = 500$	5000	20000	5000
$N_{tr} = 10$	-/-	8/*51*	-/-	8/*50*
150	-/-	98/*51*	-/-	97/*50*
258	7/*3*	165/*51*	1679/*202*	164/*50*

statistical significance was evaluated by the Wilcoxon signed-rank test. The p-values were computed by comparing to the kappa values for each testing subjects from ten independent simulations (2570 samples).

Figure 2 shows the kappa values for different numbers of training subjects N_{tr}, ranging from 10 to 258 subjects, using different features based on [8] (Setup-S) and [9] (Setup-N). The number of hidden nodes within ELM, $L = 5000$, was fixed. The highest averaged testing kappa values were obtained with $N_{tr} = 258$, the values of which were 0.61 and 0.69 in Setup-S and Setup-N, respectively. To evaluate the statistical significance of increasing the number of training subjects, we compared the testing performances. In Setup-S, increasing the number of subjects until 200 led to the improvement of the performance: (p-values: 100 vs 150 – 4.6×10^{-06}, 150 vs 200 – 0.032, 200 vs 258 – 0.14), whereas there was no statistical significance between 150 and 200 in Setup-N: (p-values: 100 vs 150 – 0.011, 150 vs 200 – 0.14, 200 vs 258 – 0.31). The computational time of training increased with the number of training subjects ($N_{tr} = 10$: 8 s, $N_{tr} = 258$: 165 s in Setup-N), whereas the times for testing were the same regardless of N_{tr}, as shown in Table II.

B. ELM and the number of hidden nodes

We next conducted experiments by changing the number of hidden nodes, L, of ELM, ranging from 500 to 20000, in order to evaluate the performance against model complexity. The number of training subjects was fixed to $N_{tr} = 258$ (all training data were used). Figure 3 illustrates the kappa values for different numbers of hidden nodes, L, in Setup-S and Setup-N. In both setups, the highest averaged kappa values were achieved with $L = 5000$, while increasing the number of hidden nodes to more than 5000 negatively affected the performance, which contradicts the results in [8]. The number of hidden nodes, L, also affects computational time, especially for training, as shown in Table II; the averaged training time of $L = 20000$ was 1679 s ($\approx$ 28 mins), which was approximately 10 times larger than simulation, with $L = 5000$ (165 s).

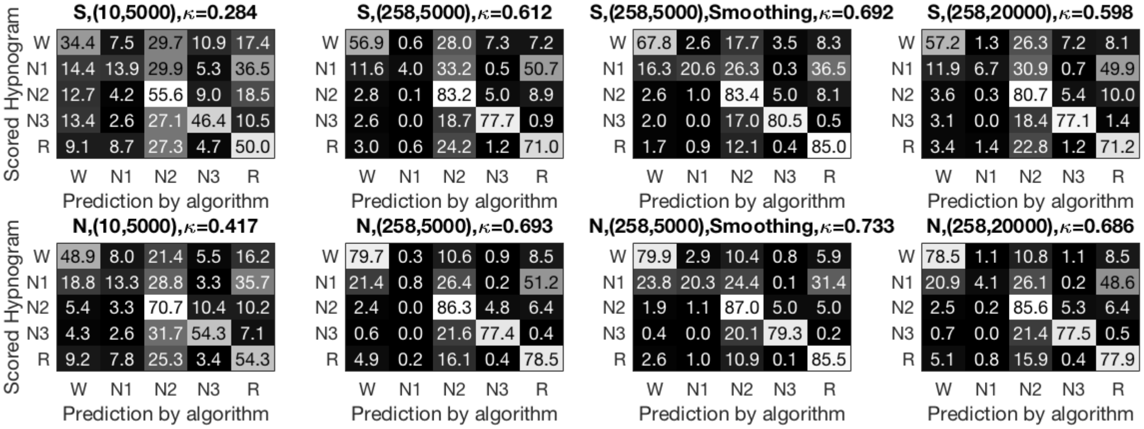
C. HMM smoothing

The predictions after HMM smoothing are given in Figure 2 (bold solid lines). Using all training data ($N_{tr} = 258$), the achieved kappa values were 0.69 (*cf.* 0.61 before smoothing) in Setup-S, and 0.73 (*cf.* 0.69 before smoothing) in Setup-N. Among 257 testing subjects, the prediction performance was improved by smoothing for 96 % of subjects in Setup-S, whereas for 93 % of subjects in Setup-N.

Figure 4 depicts the confusion matrices obtained with different settings of ELM, upon changing the number of training subjects N_{tr}, the number of hidden nodes L, and applying smoothing. The upper confusion matrices were results for Setup-S whilst lower ones were for Setup-N. Comparing to the matrices in the second column (i.e. $N_{tr} = 258$, $L = 5000$, before smoothing) and the third column (the same N_{tr} and L, but after smoothing), the sensitivities of each class improved with smoothing, especially for the N1 stage (from 4.0 % to 20.6 % in Setup-S, and from 0.8 % to 20.3 % in Setup-N). In the hypnogram plot, the advantage of smoothing was also confirmed. Figure 5 depicts an overnight hypnogram of one subject; the upper graph shows the manually scored hypnogram based on the PSG recordings (black) whereas the bottom panel shows the automatically predicted label based on the ELM ($N_{tr} = 258$, $L = 5000$) before smoothing (blue) and after smoothing (red). The smoothing successfully took account of neighbouring epochs and eliminated fragmentation in prediction. The kappa value for this subject was 0.81 after smoothing, improved from 0.67 before.

D. Performance with different classifiers

Finally, we tested the performance using different classifiers, the LDA, RF and SVM. Table III summarises the prediction performance and computational time (Training time/*Testing time*) for different classifiers in Setup-N. The



S,(10,5000),κ=0.284

	W	N1	N2	N3	R
W	34.4	7.5	29.7	10.9	17.4
N1	14.4	13.9	29.9	5.3	36.5
N2	12.7	4.2	55.6	9.0	18.5
N3	13.4	2.6	27.1	46.4	10.5
R	9.1	8.7	27.3	4.7	50.0

S,(258,5000),κ=0.612

	W	N1	N2	N3	R
W	56.9	0.6	28.0	7.3	7.2
N1	11.6	4.0	33.2	0.5	50.7
N2	2.8	0.1	83.2	5.0	8.9
N3	2.6	0.0	18.7	77.7	0.9
R	3.0	0.6	24.2	1.2	71.0

S,(258,5000),Smoothing,κ=0.692

	W	N1	N2	N3	R
W	67.8	2.6	17.7	3.5	8.3
N1	16.3	20.6	26.3	0.3	36.5
N2	2.6	1.0	83.4	5.0	8.1
N3	2.0	0.0	17.0	80.5	0.5
R	1.7	0.9	12.1	0.4	85.0

S,(258,20000),κ=0.598

	W	N1	N2	N3	R
W	57.2	1.3	26.3	7.2	8.1
N1	11.9	6.7	30.9	0.7	49.9
N2	3.6	0.3	80.7	5.4	10.0
N3	3.1	0.0	18.4	77.1	1.4
R	3.4	1.4	22.8	1.2	71.2

N,(10,5000),κ=0.417

	W	N1	N2	N3	R
W	48.9	8.0	21.4	5.5	16.2
N1	18.8	13.3	28.8	3.3	35.7
N2	5.4	3.3	70.7	10.4	10.2
N3	4.3	2.6	31.7	54.3	7.1
R	9.2	7.8	25.3	3.4	54.3

N,(258,5000),κ=0.693

	W	N1	N2	N3	R
W	79.7	0.3	10.6	0.9	8.5
N1	21.4	0.8	26.4	0.2	51.2
N2	2.4	0.0	86.3	4.8	6.4
N3	0.6	0.0	21.6	77.4	0.4
R	4.9	0.2	16.1	0.4	78.5

N,(258,5000),Smoothing,κ=0.733

	W	N1	N2	N3	R
W	79.9	2.9	10.4	0.8	5.9
N1	23.8	20.3	24.4	0.1	31.4
N2	1.9	1.1	87.0	5.0	5.0
N3	0.4	0.0	20.1	79.3	0.2
R	2.6	1.0	10.9	0.1	85.5

N,(258,20000),κ=0.686

	W	N1	N2	N3	R
W	78.5	1.1	10.8	1.1	8.5
N1	20.9	4.1	26.1	0.2	48.6
N2	2.5	0.2	85.6	5.3	6.4
N3	0.7	0.0	21.4	77.5	0.5
R	5.1	0.8	15.9	0.4	77.9

Fig. 4. The averaged confusion matrices over all 257 testing subjects for different scenarios. The title of each confusion matrix denotes: [Setup-{S,N}, (N_{tr}, L), Cohen's kappa value κ]. Observe that the sensitivities of each class improved after HMM smoothing.

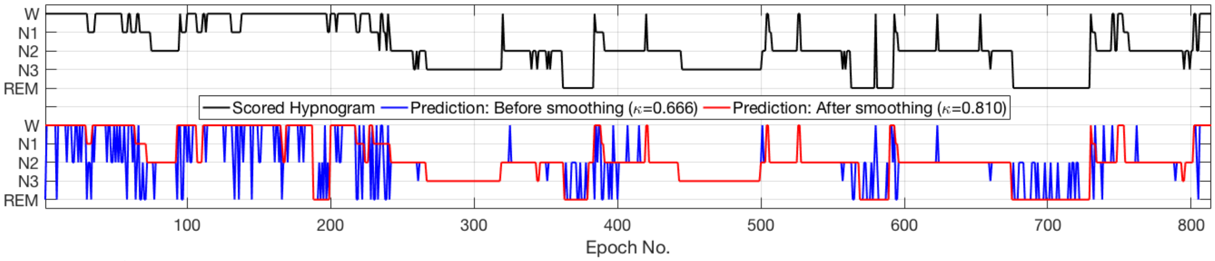


Fig. 5. Scored hypnogram of one subject (black), and the automatically predicted label before/after smoothing (blue/red).

highest kappa value, κ, was obtained by RF (0.695), whereas the RF required the longest training time (878 s).

TABLE III

COMPUTATIONAL TIME OF DIFFERENT CLASSIFIERS

	LDA	RF	SVM	ELM
κ	0.67±0.11	0.69±0.11	0.67±0.11	0.69±0.10
Time (sec)	2/*0.1*	878/*23*	308/*0.1*	165/*51*

IV. Conclusion

We have examined the feasibility of automatic sleep staging using a publicly available large-scale dataset, CC-SHS. Extensive experiments have been conducted to evaluate the scalability of the proposed methods by changing the number of training subjects, model complexity, and choice of classifiers; these analyses were inspired by [8]. Another virtue of this current work is the use of an HMM state machine for physically meaningful discrimination between 'state transition' associated with the evolution of the scored sleep stages, resulting in much smoother and more accurate automatic sleep scores. The extracted features in both structural complexity and frequency domains have been classified by a computationally cheap ELM algorithm, and the achieved average kappa value was 0.73 with the HMM smoothing. The hypnogram from patients with sleep disorder is less 'smooth' than that from the healthy subjects – the smoothing approach for the clinical data will be the subject of future work.

References

[1] C. Iber, *The AASM manual for the scoring of sleep and associated events: Rules, terminology and technical specifications*. American Academy of Sleep Medicine, 2007.

[2] D. Sommer, M. Chen, M. Golz, U. Trutschel, and D. Mandic, "Fusion of state space and frequency-domain features for improved microsleep detection," in *International Conference on Artificial Neural Networks*, pp. 753–759, 2005.

[3] C. O'Reilly, N. Gosselin, J. Carrier, and T. Nielsen, "Montreal archive of sleep studies: An open-access resource for instrument benchmarking and exploratory research," *Journal of Sleep Research*, vol. 23, no. 6, pp. 628–635, 2014.

[4] A. L. Goldberger, L. A. N. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, and H. E. Stanley, "PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, pp. 215–220, 2000.

[5] B. Koley and D. Dey, "An ensemble system for automatic sleep stage classification using single channel EEG signal," *Computers in Biology and Medicine*, vol. 42, no. 12, pp. 1186–1195, 2012.

[6] D. A. Dean, A. L. Goldberger, R. Mueller, M. Kim, M. Rueschman, D. Mobley, S. S. Sahoo, C. P. Jayapandian, L. Cui, M. G. Morrical, S. Surovec, G.-Q. Zhang, and S. Redline, "Scaling up scientific discovery in sleep medicine: The National Sleep Research Resource," *Sleep*, vol. 39, no. 5, pp. 1151–1164, 2016.

[7] A. Patanaik, J. L. Ong, J. J. Gooley, S. Ancoli-Israel, and M. W. Chee, "An end-to-end framework for real-time automatic sleep stage classification," *Sleep*, vol. 41, no. 5, pp. 1–11, 2018.

[8] H. Sun, J. Jia, B. Goparaju, G.-B. Huang, O. Sourina, M. T. Bianchi, and M. B. Westover, "Large-scale automated sleep staging," *Sleep*, vol. 40, no. 10, 2017.

[9] T. Nakamura, T. Adjei, Y. Alqurashi, D. Looney, M. J. Morrell, and D. P. Mandic, "Complexity science for sleep stage classification from EEG," in *Proceedings of IEEE International Joint Conference on Neural Networks (IJCNN)*, pp. 4387–4394, 2017.

[10] NSRR, "National Sleep Research Resource: Cleveland Children's Sleep and Health Study." `https://sleepdata.org/datasets/ccshs/files/datasets`.

[11] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: Theory and applications," *Neurocomputing*, vol. 70, no. 1-3, pp. 489–501, 2006.