ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 978-1-7281-6327-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICASSP49357.2023.10096894

ALEATORIC UNCERTAINTY ESTIMATION OF OVERNIGHT SLEEP STATISTICS THROUGH POSTERIOR SAMPLING USING CONDITIONAL NORMALIZING FLOWS

Hans van Gorp[1,2], Merel M. van Gilst[1,3], Pedro Fonseca[1,2],
Sebastiaan Overeem[1,3], Ruud J. G. van Sloun[1,2]

[1] Department of Electrical Engineering, Eindhoven University of Technology, The Netherlands
[2] Philips Research, Eindhoven, The Netherlands
[3] Sleep Medicine Centre, Kempenhaeghe Foundation, Heeze, the Netherlands

ABSTRACT

In sleep staging, a polysomnography is visually scored by a human expert, who creates a hypnogram that classifies the measurement into a sequence of sleep stages, from which overnight sleep statistics, such as total sleep time, are derived. Because inter-scorer agreement between humans is limited, deep learning methods trained to automate sleep staging have aleatoric uncertainty about both hypnogram and overnight statistics. We would like to estimate this aleatoric uncertainty, which can be achieved by means of posterior sampling. Current approaches model the hypnogram through a time-based factorization of categorical distributions over sleep stages. This discards time-dependent information, invalidating posterior sampling of the overnight statistics. Instead of factorizing, we propose to jointly model the sequence of sleep stages, by introducing U-Flow, a conditional normalizing flow network. We compare U-Flow to factorized baselines, leveraging 921 recordings, and show that it achieves similar performance in terms of accuracy and Cohen's kappa on the majority voted hypnograms, while outperforming in terms of uncertainty estimation of the overnight sleep statistics.

Index Terms— Automatic Sleep Staging, Overnight Sleep Statistics, Aleatoric Uncertainty, Normalizing Flows

1. INTRODUCTION

Sleep staging is a crucial element in the diagnosis of many sleep disorders, such as parasomnias, sleep apnea, and narcolepsy. Sleep staging is usually performed after an analysis of the signals collected during a so-called polysomnography (PSG) measurement, which typically includes electroencephalography (EEG), electrooculography (EOG), electromyography (EMG), electrocardiography (ECG), among other modalities, such as oxygen saturation and respiratory effort. Following the American Academy of Sleep Medicine (AASM) manual [1], the PSG measurement is divided into non-overlapping segments of 30 seconds in length, called sleep epochs*. These sleep epochs are scored visually as belonging to one of five discrete sleep stages: Wake (W), Rapid Eye Movement (REM), or non-REM (NREM) stage 1-3. The resulting registration of how the sleep stages evolve during the night is called a hypnogram. From this hypnogram, overnight sleep statistics are calculated, such as total sleep time and the number of awakenings. A physician is usually presented with both the hypnogram and the overnight statistics in a lab report.

Because sleep staging is such a labor-intensive process, great effort has been put into automating the process. Recently, Deep Learning (DL) has enabled many advances in this field, among which the Stanford staging algorithm [2], U-Sleep [3], SleepTransformer [4], AttnSleep [5], and TinySleepNet [6]. We refer the reader to the review paper of Phan and Mikkelsen for an overview of recent methods [7].

DL methods are trained on scorings of humans with imperfect inter-scorer agreement ($\sim 82.6\%$ [8]). From the point of view of the DL method, all scorings are equally 'true'. Consequently, DL methods should express aleatoric uncertainty about what the correct sleep stage is for a given sleep epoch, and accordingly for the overall hypnogram of a PSG recording [9]. This can be expressed as the conditional probability distribution $p(\boldsymbol{h}|\boldsymbol{x})$, where $\boldsymbol{x}$ is the PSG and $\boldsymbol{h}$ is the hypnogram. The final step is then to calculate the overnight sleep statistics s from the hypnogram using $s = f_{stat}(\boldsymbol{h})$, which leads to the conditional distribution $p(s|\boldsymbol{x})$.

Current DL methods used for automatic sleep staging are trained using cross entropy, which assumes a factorization of categorical distributions over the sleep epochs in a recording:

$$p(\boldsymbol{h}_{1:T}|\boldsymbol{y}_{1:T}) = \prod_{t=1}^{T}\prod_{l=1}^{L} y_{t,l}^{[h_t=l]}, \tag{1}$$

where T is the number of sleep epochs in the recording, L is the number of sleep staging labels, h_t is the sleep stage for the sleep epoch at time t, and $y_{t,l}$ is the softmax output of the neural network at time t and for label l. The expression $[h_t =$

*To avoid confusion, we will refer to a 30 second segment of sleep as a 'sleep epoch', and a full iteration over a dataset as a 'dataset epoch'.

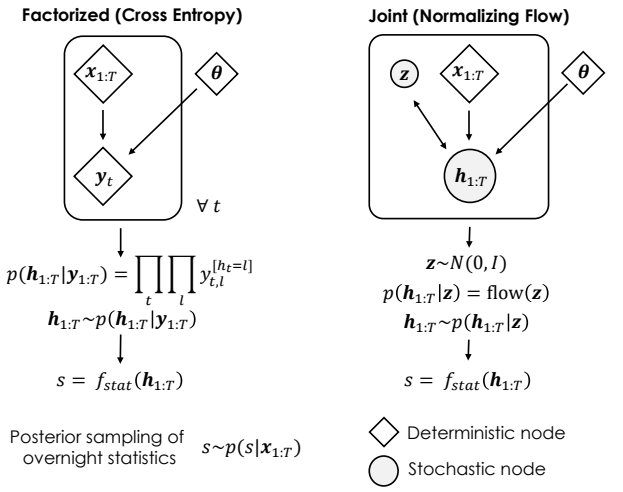

Fig. 1. Graph models of posterior sampling of the overnight statistics. Cross entropy-based methods model the hypnogram in factorized manner, while conditional generative approaches do this jointly. The latter is advantageous for posterior sampling as it can model dependencies between sleep epochs.

$l]$ evaluates to 1, if $h_t = l$, and is 0 otherwise, thereby selecting only the relevant softmax outputs given the hypnogram $\boldsymbol{h}_{1:T}$.

Conventionally, only the most likely hypnogram is presented as the output of the DL network and presented to the user. However, such a point estimate only leads to one point estimate of the overnight statistics. To estimate the aleatoric uncertainty of the overnight statistics, we can perform posterior sampling from $p(\boldsymbol{h}_{1:T}|\boldsymbol{x}_{1:T})$, see Fig. 1. We hypothesize that such posterior sampling with current DL methods in sleep, will however lead to wrong samples $s \sim p(s|\boldsymbol{x})$ since the factorization introduced in (1) removes crucial contextual information about dependencies between sleep epochs.

In this paper, we propose to forgo the assumed factorization of cross entropy to directly model the joint distribution $p(\boldsymbol{h}_{1:T}|\boldsymbol{x}_{1:T})$. To this end, we make use of a conditional deep generative approach, which starts from a randomly sampled latent variable $\boldsymbol{z} \sim N(0, I)$ that is mapped to a hypnogram, where this mapping depends conditionally on the PSG (see Fig. 1). To implement this mapping, we introduce a conditional normalizing flow model [10, 11] that allows for exact evaluation of $p(\boldsymbol{h}_{1:T}|\boldsymbol{x}_{1:T})$, termed U-Flow. U-Flow is based on a combination of the popular normalizing flow network, Glow [11], and the discriminative U-Net model [3]. Our main contributions are:

- We propose U-Flow, combining the multi-scale architecture of Glow with a U-Net.
- We show that joint modeling of the full hypnogram outperforms factorized (cross entropy) based methods in its ability to capture uncertainty of overnight statistics.
- For the first time, we show how multiple scorings per PSG can be used to evaluate the uncertainty prediction of overnight statistics of automated sleep staging models.

2. METHODS[†]

2.1. Datasets

We use four publicly available datasets. The Stanford Sleep Cohort [12, 13] made available through National Sleep Research Resource [14] contains 772 single-scorer PSGs. We excluded 111 recordings because not all selected channels were available (see subsection 2.2). Of the remaining 661 PSGs, 529 (80%) are included in the training set and 132 (20%) are used in the validation set.

The Dreem Open Datasets (DOD) [15] consists of two separate subsets: DOD-H contains 25 PSG recordings of healthy participants, and DOD-O contains 55 recordings of obstructive sleep apnea patients. All recordings were scored by 5 different sleep experts. Both DOD-H and DOD-O are used for training.

From the Institute of Systems and Robotics dataset [16], we use subsets I and III, which contain 110 recordings each scored by 2 sleep experts, and we use it for training.

Lastly, the Inter-Scorer Reliability Cohort [17] contains 70 PSGs, each scored by 6 human experts coming from 3 different sleep centers (2 experts per center). Because this dataset was scored by a variety of experts coming from different sleep centers, we use it exclusively as a hold-out test set.

2.2. Channel selection and preprocessing

We use the following signals from the PSG as input: EEG (C3-M2, C4-M1), EOG (O2-A1, O1-A2), EMG (LOC-A2, ROC-A1, EMG1-EMG2), and ECG. Because sampling frequencies differ by non-integer factors (256 Hz, 250Hz, 200 Hz, and 128 Hz) we employ polyphase filtering to resample all PSGs to 128 Hz. The recordings were zero-padded to a length of $7 * 2^8 = 1792$ sleep epochs for implementation purposes. This additional padding was solely added for training purposes, and not used for the computation of the final results. Additionally, each input PSG signal is rescaled using:

$$\boldsymbol{x} = \text{sign}(\tilde{\boldsymbol{x}}) \cdot \log\left(|\tilde{\boldsymbol{x}}|/P_{95}(\tilde{\boldsymbol{x}}) + 1\right), \tag{2}$$

where $\tilde{\boldsymbol{x}}$ corresponds to each signal, $\boldsymbol{x}$ is the rescaled signal, and $P_{95}(\tilde{\boldsymbol{x}})$ is the 95th magnitude percentile.

2.3. Overnight Statistics

To model the posterior sampling of the overnight statistics we use the hypnogram as an intermediate step, since:

$$s \sim p(s|\boldsymbol{x}_{1:T}) = f_{stat}(\boldsymbol{h}_{1:T} \sim p(\boldsymbol{h}_{1:t}|\boldsymbol{x}_{1:T})). \tag{3}$$

It is thus sufficient to use a DL model to first sample from $p(\boldsymbol{h}_{1:t}|\boldsymbol{x}_{1:T})$ and then use the deterministic f_{stat} function to calculate the overnight statistics. We calculated the following overnight statistics: total sleep time, time spent in each sleep stage, and the number of awakenings in either REM or Non-REM sleep.

[†]https://github.com/HansvanGorp/U-Flow

2.4. U-Flow

Normalizing flows are a type of invertible neural network that learn a mapping from a normal latent distribution $\boldsymbol{z}$ to a desired output signal and back again. We use the popular Glow architecture [11] that was originally proposed for 2d images and adapted here to work on the 1d hypnogram. We additionally condition the Glow model on a learned conditioning vector $\boldsymbol{c}_{0:8}$ created from the corresponding PSG recording at different resolution levels. U-Flow can then be described as:

$$\boldsymbol{z}_{1:T} = f_{Glow}(\boldsymbol{h}_{1:T}|\boldsymbol{c}_{0:8}), \quad \boldsymbol{h}_{1:T} = f^{-1}_{Glow}(\boldsymbol{z}_{1:T}|\boldsymbol{c}_{0:8}). \quad (4)$$

The learned conditioning vector is created by a convolutional neural network that consists of two parts, a sleep epoch encoder and a context encoder:

$$\boldsymbol{c}_{0:8} = (f_{enc} \circ f_{epoch})(\boldsymbol{x}_{1:T}), \quad (5)$$

where $\circ$ denotes the composition of two functions. The sleep epoch encoder takes the full PSG measurement as input, which is of size $(1792 \cdot 30 \cdot 128)$. It applies a convolution of kernel size 7 and then a maxpooling of size 4, which together are repeated 4 times. Then, a convolution of kernel and stride 15 is applied to achieve a sleep epoch encoding of size 1792.

The sleep epoch encoding is then fed into the context encoder, which consists of 8 blocks, each with two convolutional layers of kernel size 7 followed by a max pooling operation of size 2. The output at each resolution level r is used as part of the conditioning vector $\boldsymbol{c}_r$.

In contrast to the encoder that progressively lowers in size, the Glow model progressively increases in size using 8 levels. Each level of Glow consists of 6 flow steps that are conditioned on the $\boldsymbol{c}_r$ of the same resolution. At the lowest scale, $\boldsymbol{c}_8$ is only of size 7, allowing the Glow model to have the entire night in its receptive field by applying a kernel that is also of size 7. For further details regarding the architecture of Glow, we refer the reader to the original paper by Kingma and Dhariwal [11].

U-Flow is trained using the exact negative log-likelihood of the hypnogram as given by the change of variables formula:

$$\begin{aligned}\mathcal{L}_{f_{flow}} &= -\log p_h(\tilde{\boldsymbol{h}}_{1:T}|\boldsymbol{x}_{1:T}) \\ &= -\log(p_z(f_{Glow}(\tilde{\boldsymbol{h}}_{1:T}|\boldsymbol{c})) \cdot |\det Jf_{Glow}(\tilde{\boldsymbol{h}}_{1:T}|\boldsymbol{c})|) \\ &= |\boldsymbol{z}|_2^2 - \log|\det Jf_{Glow}(\tilde{\boldsymbol{h}}_{1:T}|\boldsymbol{c})|, \end{aligned} \quad (6)$$

Where $\det J$ is the determinant of the Jacobian of the Glow model, which accounts for the change of probability density, and $\tilde{h}_t$ is the one-hot encoded ground truth hypnogram.

Because the change of variables formula is defined in continuous space, but the hypnograms are discrete, we apply dequantization using triangular noise:

$$\tilde{\boldsymbol{h}}_{deq} = 0.5\tilde{\boldsymbol{h}} + 0.25\boldsymbol{u}_1 + 0.25\boldsymbol{u}_2, \quad (7)$$

with $\boldsymbol{u}_1$ and $\boldsymbol{u}_2$ i.i.d. samples from the uniform distribution.

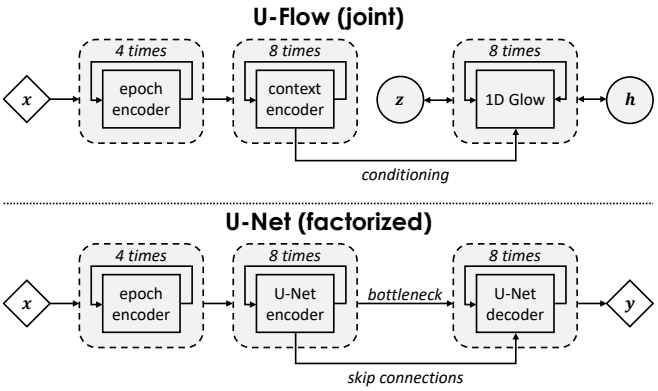

Fig. 2. U-Flow and U-Net have a similar architecture. However, U-flow makes use of a 1D invertible Glow.

During inference with U-Flow, we sample 1024 times from $p(\boldsymbol{h}_{1:t}|\boldsymbol{x}_{1:T})$ by first creating the conditioning using (5), then sampling 1024 i.i.d. normally distributed latent variables $\boldsymbol{z} \sim N(0, I)$, and finally applying the inverse of the Glow model to each latent variable using (4).

2.5. Baseline

We create a U-Net baseline of similar complexity to U-Flow. The baseline uses a similar structure to that of U-Flow, but we change the 1d Glow model to a convolutional decoder. The U-Net decoder follows a similar but mirrored structure to that of the U-Net encoder, employing upconvolutions of size 2 instead of max pooling operations. The outputs of the decoder are the logits $\boldsymbol{\phi}$ (log-probability) of each sleep stage, which can be mapped to the probabilities $\boldsymbol{y}$ using the softmax activation function:

$$\begin{aligned}\boldsymbol{\phi}_{1:T} &= (f_{dec} \circ f_{enc} \circ f_{epoch})(\boldsymbol{x}_{1:T}), \\ \boldsymbol{y}_{1:T} &= \text{softmax}(\boldsymbol{\phi}_{1:T}). \end{aligned} \quad (8)$$

The U-Net is trained using cross entropy, which assumes a factorization of categorical distributions over the sleep epochs:

$$\mathcal{L}_{\text{cross entropy}} = \sum_{t=1}^{T} \tilde{\boldsymbol{h}}_t \cdot \log(\boldsymbol{y}_t). \quad (9)$$

During inference with U-Net, we sample 1024 times from $p(\boldsymbol{h}_{1:t}|\boldsymbol{x}_{1:T})$ leveraging Gumbel sampling [18]:

$$\boldsymbol{h}_t = \arg\max \boldsymbol{\phi}_t + \boldsymbol{\epsilon}_t \quad \forall t, \quad (10)$$

with $\boldsymbol{\epsilon}$ i.i.d. samples from the standard Gumbel distribution.

2.6. Training strategy

Both methods are trained with their respective loss function for 200 dataset epochs using mini-batches of two whole-night PSGs. We use the Adam optimizer [19] with parameters: $lr = 10^{-4}$, $\beta_1 = 0.9$, and $\beta_2 = 0.999$. After 100 dataset epochs the learning rate is lowered to 10^{-5} to aid with convergence.

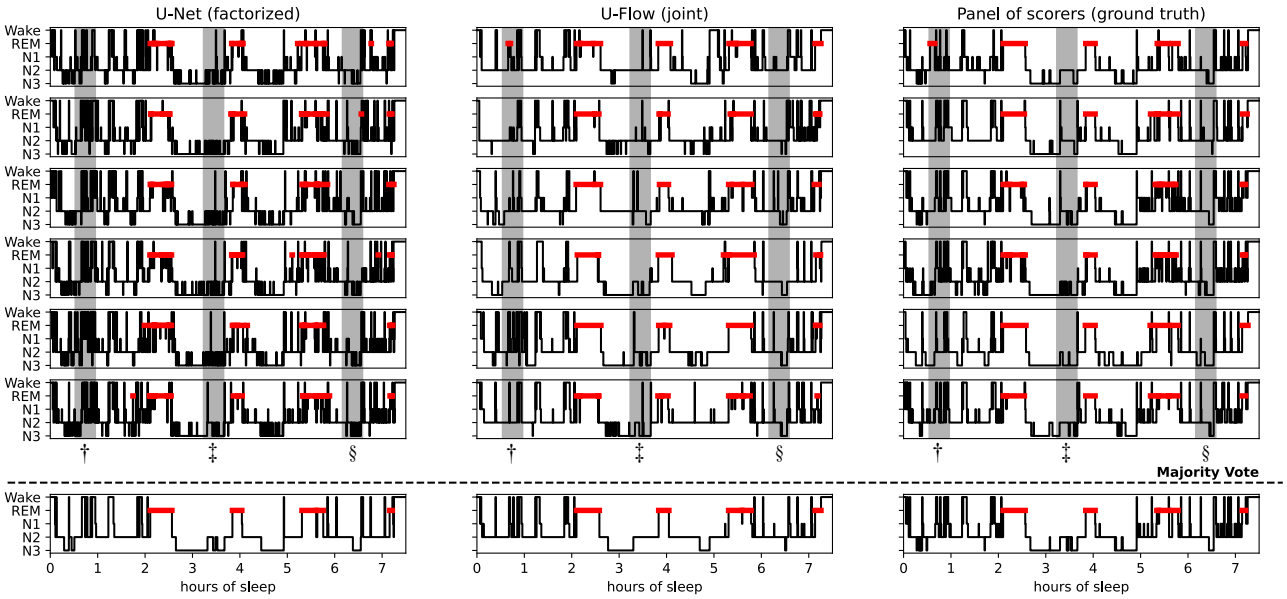

Fig. 3. Example results for the PSG in the test set with median inter-rater agreement (84%). Shown here are six samples from each DL method, as well as the six ground truth hypnograms made by the panel of human scorers. For both DL methods, the majority vote is based on 1024 samples, not just the six samples shown here. † U-Flow predicts uncertainty of potential REM period. ‡ U-Net predicts too many stage transitions. § U-Flow correctly predicts uncertainty about the length of this N3 period.

Table 1. Average Kullback–Leibler divergence between the predictions of the models and the panel, lower is better. As well as the accuracy and Cohen's kappa scores for the majority voted hypnograms, higher is better.

	(factorized)			(**joint**)
	U-Net	Stanford	U-sleep	**U-Flow**
Total sleep time	41.1	59.2	107	**1.34**
Time in N1	13.2	34.7	70.5	**0.78**
Time in N2	68.8	124	231	**1.57**
Time in N3	87.8	332	343	**2.31**
Time in REM	81.2	84.2	124	**1.84**
Awakenings - from REM	1.30	1.71	3.20	**0.66**
Awakenings - from NREM	14.0	5.04	5.81	**1.38**
Accuracy	77.7	**84.1**	83.3	83.2
Cohen's Kappa	0.667	**0.751**	0.747	0.735

3. RESULTS

A qualitative example of the resulting hypnograms for U-Net, U-Flow, and the panel of human scorers is shown in Fig. 3. It can be seen that qualitatively, factorizing leads to too many state transitions over the whole recording, whereas joint modeling only has a lot of state transitions were the human panel also had them.

We quantitatively compare posterior sampling from U-Flow against U-Net, the Stanford staging algorithm [2], and U-Sleep [3]. From the 1024 hypnograms produced by each method and the six hypnograms of the human panel, we calculate the overnight statistics and summarize them using a normal distribution following the observed mean and variance. The distributions of the models are compared to that of the human panel using the Kullback–Leibler (KL) divergence. We also compare the majority vote results of each model to the majority vote of the human panel. We assess both accuracy and Cohen's kappa, which takes agreement due to random chance into account. The quantitative results are shown in Table 1.

4. CONCLUSION

In this manuscript, we proposed to model the relationship between PSG and hypnograms jointly, instead of factorized per sleep epoch as is assumed by cross entropy. We achieve this through conditional deep generative modeling leveraging normalizing flows in a U-Net-like structure, termed U-Flow. We compared the proposed U-Flow model to factorized baselines, concluding that it outperforms in terms of posterior sampling of the overnight sleep statistics.

Here, we only investigated those overnight sleep statistics than can be calculated solely from the hypnogram. However, there are other clinically relevant statistics, such as the apnea-hypopnea index (AHI). Calculation of the AHI not only requires jointly modeling PSG and hypnogram but also disordered breathing events. We leave uncertainty prediction of the AHI to future work. Moreover, as the first application of conditional generative modeling to sleep staging, we only investigated the use of conditional normalizing flows. We also leave the investigation of other conditional generative techniques to future work.

5. REFERENCES

[1] C. Iber, S. Ancoli-Israel, A. Chesson, and S.F. Quan, "The AASM manual for the scoring of sleep and associated events: rules, terminology and technical specifications," *American Academy of Sleep Medicine*, 2007.

[2] J.B. Stephansen et al., "Neural network analysis of sleep stages enables efficient diagnosis of narcolepsy," *Nature Communications*, vol. 9, no. 1, December 2018.

[3] M. Perslev, S. Darkner, L. Kempfner, M. Nikolic, P.J. Jennum, and C. Igel, "U-sleep: resilient high-frequency sleep staging," *npj Digital Medicine*, vol. 4, no. 1, April 2021.

[4] H. Phan, K.B. Mikkelsen, O. Chen, P. Koch, A. Mertins, and M. de Vos, "Sleeptransformer: Automatic sleep staging with interpretability and uncertainty quantification," *IEEE Transactions on Biomedical Engineering*, 2022.

[5] E. Eldele, Z. Chen, C. Liu, M. Wu, C. Kwoh, X. Li, and C. Guan, "An attention-based deep learning approach for sleep stage classification with single-channel eeg," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 29, 2021.

[6] A. Supratak and Y. Guo, "Tinysleepnet: An efficient deep learning model for sleep stage scoring based on raw single-channel eeg," in *EMBC*, 2020.

[7] H. Phan and K. Mikkelsen, "Automatic sleep staging of EEG signals: recent development, challenges, and future directions," *Physiological Measurement*, vol. 43, no. 4, April 2022.

[8] R.S. Rosenberg and S. Van Hout, "The American Academy of Sleep Medicine inter-scorer reliability program: sleep stage scoring," *Journal of Clinical Sleep Medicine*, vol. 9, no. 1, January 2013.

[9] H. van Gorp, I.A.M. Huijben, P. Fonseca, R.J.G. van Sloun, S. Overeem, and M.M. van Gilst, "Certainty about uncertainty in sleep staging: a theoretical framework," *Sleep*, vol. 45, no. 8, June 2022.

[10] C. Winkler, D. Worrall, E. Hoogeboom, and M. Welling, "Learning likelihoods with conditional normalizing flows," *arXiv preprint*, 2019.

[11] D.P. Kingma and P. Dhariwal, "Glow: Generative flow with invertible 1x1 convolutions," *Advances in neural information processing systems*, vol. 31, 2018.

[12] O. Andlauer et al., "Nocturnal rapid eye movement sleep latency for identifying patients with narcolepsy/hypocretin deficiency," *JAMA Neurology*, vol. 70, no. 7, July 2013.

[13] H. Moore IV et al., "Design and validation of a periodic leg movement detector," *PloS one*, vol. 9, no. 12, 2014.

[14] G. Zhang et al., "The national sleep research resource: towards a sleep data commons," *J Am Med Inform Assoc*, vol. 25, no. 10, October 2018.

[15] A. Guillot, F. Sauvet, E.H. During, and V. Thorey, "Dreem open datasets: Multi-scored sleep datasets to compare human and automated sleep staging," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 28, no. 9, 2020.

[16] Sirvan Khalighi, Teresa Sousa, José Moutinho Santos, and Urbano Nunes, "ISRUC-Sleep: A comprehensive public dataset for sleep researchers," *Computer Methods and Programs in Biomedicine*, vol. 124, 2016.

[17] S.T. Kuna et al., "Agreement in computer-assisted manual scoring of polysomnograms across sleep centers," *Sleep*, vol. 36, no. 4, 2013.

[18] I.A.M. Huijben, W. Kool, M.B. Paulus, and R.J.G. van Sloun, "A review of the gumbel-max trick and its extensions for discrete stochasticity in machine learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[19] D.P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *International Conference on Learning Representations (ICLR)*, 2015.