SLEEPGPT: LEARNING TRANSFERABLE SLEEP STAGE DYNAMICS FROM HYPNOGRAMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Sleep stage sequences (hypnograms) encode temporal structure that is useful for both resolving ambiguous stage predictions and characterizing whole-night sleep architecture. However, existing sleep models typically learn temporal dependencies jointly with physiological signal representations, leaving open whether stage dynamics can be learned independently and transferred across models and tasks. We introduce SleepGPT, a compact autoregressive transformer pretrained exclusively on 5.86 million expert-scored stage annotations from 5,793 overnight recordings. SleepGPT learns stage dynamics through next-stage prediction and supports two complementary forms of transfer. For sleep staging, the frozen model provides a sequence prior that is combined with signal-based predictions from independently trained PSG classifiers; for sleep disorder classification, its pretrained encoder represents local hypnogram segments within a hierarchical whole-night model. Across seven external staging comparisons involving three classifiers and three datasets, SleepGPT improves accuracy by 1.6-4.2 percentage points without target-cohort retraining. Improvements persist for classifiers that already model temporal EEG context and for simultaneous PSG and wearable EEG recordings. For hypnogram-based disorder classification, SleepGPT initialization improves balanced accuracy over the same hierarchical architecture trained from scratch, with the gains extending to independent external cohorts. These results show that transferable temporal representations can be learned directly from expert-scored hypnograms, independently of physiological signal modeling, and reused for both epoch-level inference and recording-level prediction.

1 INTRODUCTION

Sleep assessment requires both accurate sleep staging and interpretation of the resulting sleep architecture. Specialists assign each 30-second polysomnography (PSG) epoch to one of five stages: wakefulness (W), N1, N2, N3, or rapid eye movement (REM) sleep. For ambiguous epochs, scoring decisions can depend on the stages of surrounding epochs, rather than on the current signal alone (Iber et al., 2007). Automated staging models incorporate temporal information through neighboring EEG epochs (Supratak et al., 2017; Phan et al., 2019; 2022a), and some explicitly account for stage transitions during training (Phyo et al., 2023). These approaches integrate context into the staging model. A complementary approach is to learn how stage labels evolve directly from scored sequences, independently of the signal classifier, and use this knowledge to refine its predictions.

At the whole-night level, a sleep stage sequence, or hypnogram, describes sleep macrostructure through stage proportions, durations, transitions, and the organization of sleep cycles. Hypnograms and derived measures are central components of PSG reports, providing evidence for the assessment of sleep disorders and associated medical conditions (Iber et al., 2007; Mahowald & Schenck, 2005). This clinical role motivates using stage sequences as inputs for sleep disorder classification. Prior work has demonstrated the diagnostic value of sleep architecture through hypnodensity-derived features (Stephansen et al., 2018). However, reducing a sequence to selected summary measures does not preserve its full temporal organization. Learning from ordered stage tokens offers an alternative, while pretraining on stage annotations allows the model to use recordings without requiring corresponding disorder labels.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Together, these observations motivate a model that learns sleep-stage dynamics directly from expert-scored hypnograms and transfers this knowledge to both staging refinement and sleep disorder classification. To investigate this possibility, we introduce SleepGPT, a compact autoregressive transformer pretrained exclusively on 5.86 million stage annotations from 5,793 overnight recordings in the Sleep Heart Health Study (SHHS) (Quan et al., 1997). Rather than learning physiological representations from EEG or PSG signals, SleepGPT treats sleep stages as discrete tokens and learns their temporal dynamics through next-stage prediction. The key distinction from conventional signal-based sleep models is therefore not the use of a transformer architecture itself, but the decoupling of stage-sequence modeling from physiological signal modeling. This separation allows the learned sequence knowledge to be reused independently of the signal representation or classifier that produced the stage predictions: for staging, the frozen model's next-stage scores are combined with those of an existing PSG-based classifier at inference time, analogous to a language model complementing acoustic evidence in speech recognition (Chan et al., 2016; Karita et al., 2019); for disorder classification, SleepGPT initializes the local encoder of a hierarchical transformer that aggregates segment representations across the night and is fine-tuned using disorder labels.

Our experiments evaluate whether the resulting stage-dynamics knowledge transfers across classifiers, recording cohorts, and prediction tasks. Across seven external staging comparisons involving YASA, TinySleepNet, and XSleepNet, SleepGPT improves accuracy by 1.6–4.2 percentage points without retraining the staging models. Temporal-module ablations show gains even when the base classifier already models EEG context, and evaluation on simultaneous PSG and headband recordings shows improvements in both modalities. For sleep disorder classification, SleepGPT initialization improves balanced accuracy over the same hierarchical classifier trained from scratch on abnormal-versus-normal sleep classification and type-1 narcolepsy identification. Models trained on the augmented CAP cohort also transfer to abnormal-versus-normal classification on ISRUC and MNC.

2 Related Work

Automated sleep staging. Deep learning has advanced automated sleep staging by learning features from raw PSG signals and their time–frequency representations. DeepSleepNet combines convolutional feature extraction with recurrent sequence modeling (Supratak et al., 2017), while SeqSleepNet jointly classifies sequences of epochs using a hierarchical recurrent architecture (Phan et al., 2019). XSleepNet integrates raw-signal and spectrogram representations within a multi-view sequential model (Phan et al., 2022a). Temporal context is modeled at different scales: IITNet models intra- and inter-epoch dependencies, SleepTransformer uses attention across epochs, and L-SeqSleepNet extends sequence modeling to whole sleep cycles (Seo et al., 2020; Phan et al., 2022b; 2023). These methods learn temporal dependencies as part of the mapping from physiological signals to sleep stages. A separate model of stage sequences may offer a complementary way to incorporate sleep architecture into the scoring process.

Sleep-stage transitions and sequence refinement. The continuity and transition structure of sleep have also been incorporated through explicit scoring rules and probabilistic models. Liang et al. (2012) combined rule-based classification with temporal smoothing. HMM-based approaches used stage-transition probabilities to refine classifier outputs (Malafeev et al., 2018; Sun et al., 2017), but typically represent sequence structure through local state-transition dependencies. More recently, TransSleep introduced an auxiliary stage-transition detection task to improve the representation of ambiguous epochs (Phyo et al., 2023). SleepGPT adopts the autoregressive formulation for expert-scored hypnograms: rather than embedding temporal dependencies within a particular PSG classifier, it learns a reusable model directly from ordered stage annotations and transfers the resulting sequence knowledge across classifiers, analogous to the use of a separately trained language model in speech recognition (Karita et al., 2019).

Automated sleep disorder assessment. Automated disorder detection has used both physiological signal features and measures of sleep architecture. Examples include ECG wavelet features for insomnia identification (Sharma et al., 2021) and EEG cross-frequency coupling with random forests for sleep disorder detection (Dimitriadis et al., 2021). At the macrostructural level, stage durations, transition patterns, and sleep-stage latencies provide complementary information about the organization of sleep. Bianchi et al. (2010) characterized altered transition dynamics in obstructive sleep

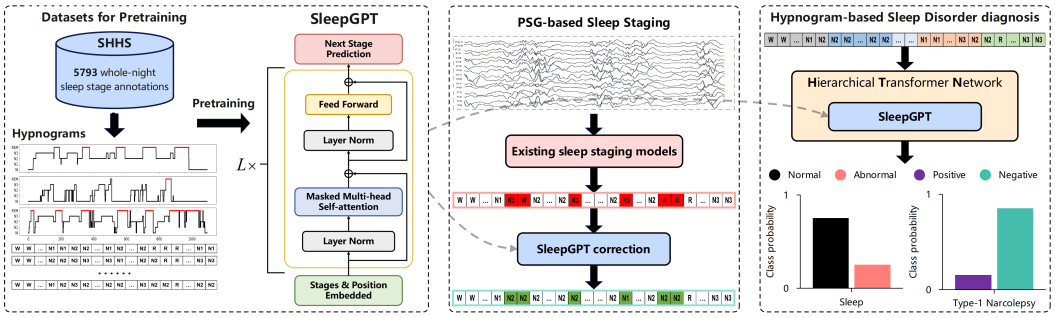


Figure 1: Pretraining and two forms of transfer in SleepGPT. SleepGPT is pretrained exclusively on expert-scored stage sequences through next-stage prediction. For sleep staging, probabilistic transfer uses the frozen model as a sequence prior to complement the current-epoch evidence from a PSG classifier. For sleep disorder classification, representational transfer reuses the pretrained encoder to represent local hypnogram segments, which are aggregated by a global transformer for recording-level prediction.

apnea, and Stephansen et al. (2018) identified narcolepsy using features derived from hypnodensity graphs, including unusual overlap between sleep stages. These findings motivate learning from the temporal organization of sleep rather than relying exclusively on signal morphology. In contrast to selected summary measures or features derived from stage probabilities, SleepGPT learns contextual representations directly from ordered stage annotations and transfers them to whole-night hypnogram classification.

3 METHOD

As illustrated in Figure 1, SleepGPT is pretrained on expert-scored hypnograms to learn stage dynamics independently of physiological signals. The pretrained model is then transferred in two complementary ways. In probabilistic transfer, its next-stage distribution serves as a frozen sequence prior for sleep staging. In representational transfer, its pretrained encoder provides local stage-sequence representations for whole-night disorder classification.

3.1 AUTOREGRESSIVE PRETRAINING ON SLEEP-STAGE SEQUENCES

SleepGPT adopts the decoder-only architecture of GPT-2 and learns to predict the next sleep stage from preceding stages (Radford et al., 2019). An overnight hypnogram is represented as $Y = (y_1, \ldots, y_T)$, where each token corresponds to a 30-second epoch and takes one of five stage labels, $\mathcal{V} = \{\mathrm{W}, \mathrm{N1}, \mathrm{N2}, \mathrm{N3}, \mathrm{REM}\}$, encoded as integers from 0 to 4. For pretraining, we use 5,863,207 annotations from 5,793 SHHS-1 recordings (Quan et al., 1997). Sleep stages were mapped to a five-class vocabulary comprising W, N1, N2, N3, and REM, with R&K stages N3 and N4 merged into N3 (Wolpert, 1969; Iber et al., 2007).

Training pairs are extracted separately from each recording using a sliding window with stride one. For a window starting at position s, we denote the input by $U = (u_1, \ldots, u_K)$ and its next-stage targets by $U^+ = (u_1^+, \ldots, u_K^+)$, where $u_j = y_{s+j-1}$ and $u_j^+ = y_{s+j}$ for $j = 1, \ldots, K$. Thus, the target sequence is shifted one stage ahead of the input. A recording of length T provides $T - K$ input–target pairs. Token and positional embeddings are processed by L transformer blocks, followed by a linear prediction head:

$$\begin{aligned} H_0 &= \mathrm{Embed}(U; W_e) + W_p, \\ H_l &= \mathrm{TransformerBlock}_l(H_{l-1}), \quad l = 1, \ldots, L, \\ Q_\theta(U) &= \mathrm{softmax}(H_L W_s^\top). \end{aligned} \tag{1}$$

Here, W_e and W_p are the token and positional embeddings, and W_s is the output projection. Each block contains masked multi-head self-attention and a feed-forward network with residual connections and layer normalization. At position j, causal masking allows the model to use only $(u_1, \ldots, u_j)$

to predict the next stage u_j^+. Softmax is applied over stage classes, so $[Q_\theta(U)]_{j,k}$ is the predicted probability of stage k at that position.

Let $\mathcal{D}_{\text{pre}}$ be the collection of window pairs (U, U^+). The model parameters θ are optimized by minimizing the cross-entropy between each next-stage prediction and its target u_j^+:

$$\mathcal{L}_{\text{pre}}(\theta) = -\frac{1}{K|\mathcal{D}_{\text{pre}}|} \sum_{(U,U^+)\in\mathcal{D}_{\text{pre}}} \sum_{j=1}^{K} \log[Q_\theta(U)]_{j,u_j^+}. \tag{2}$$

The target stages are obtained from the input hypnograms themselves, so this pretraining objective requires neither PSG signals nor disorder labels. Model dimensions and optimization settings are reported in Appendix B.

3.2 Sequence Refinement for Sleep Staging

A PSG-based sleep staging model (SSM) estimates stage probabilities from physiological signals. SleepGPT serves as a sleep language model (SLM), estimating the next-stage probabilities from the preceding stage sequence. These probabilities, denoted by P_{SLM}, are obtained from the last-position output of Q_θ for the input history. The two models' outputs are combined by weighted probability fusion:

$$P_t(k) = \alpha P_{\text{SSM}}(y_t = k \mid \mathbf{x}_t) + (1-\alpha) P_{\text{SLM}}(y_t = k \mid h_t), \tag{3}$$

where $\mathbf{x}_t$ contains the PSG epochs used by the staging model, and $h_t = (\hat{y}_{\max(1,t-K)}, \ldots, \hat{y}_{t-1})$ contains up to K most recent predicted stages. The weight $\alpha \in [0,1]$ balances signal evidence and stage-sequence context. The final prediction is $\hat{y}_t = \arg\max_{k\in\mathcal{V}} P_t(k)$.

Both models remain fixed during staging. The first epoch, for which no preceding stage is available, is predicted solely by the SSM. For each subsequent epoch, SleepGPT uses all available preceding predicted stages as context until the history reaches the maximum length K; thereafter, only the most recent K predictions are retained. The fused prediction from Equation 3 is appended to the history for the next epoch.

3.3 Hierarchical Sleep Disorder Classification

For sleep disorder classification, the input is a whole-night hypnogram. In the experiments reported here, we use expert-scored hypnograms to isolate the contribution of sequence representation from upstream staging errors. In principle, the framework can also operate on automatically generated hypnograms, including sequences produced by a PSG classifier followed by SleepGPT refinement. Whole-night hypnograms form long stage sequences, whereas SleepGPT is pretrained on windows of K stages. To represent the full sequence, we adopt a hierarchical transformer network (HTN), following the segment-level encoding and document-level aggregation strategy used in long-text classification (Yang et al., 2016; Pappagari et al., 2019). Specifically, Y is partitioned into non-overlapping segments $S_1, \ldots, S_M$ of at most K stages. The pretrained SleepGPT encoder extracts a representation from each segment, and a global transformer integrates these representations across the night. The segment representations are arranged in temporal order and prepended with a classification token:

$$\begin{aligned} \mathbf{z}_i &= f_\theta(S_i), \quad i = 1, \ldots, M, \\ Z &= [\mathbf{z}_{\text{CLS}}, \mathbf{z}_1, \ldots, \mathbf{z}_M], \\ [\mathbf{v}_{\text{CLS}}, \mathbf{v}_1, \ldots, \mathbf{v}_M] &= g_\phi(Z), \\ p(c \mid Y; \Phi) &= \text{softmax}(W_o \mathbf{v}_{\text{CLS}} + b_o)_c. \end{aligned} \tag{4}$$

Here, f_θ is the SleepGPT feature extractor, g_ϕ is the global transformer encoder, and $\mathbf{v}_{\text{CLS}}$ is the whole-night representation used to predict class c. The classification head has parameters W_o and b_o, and Φ denotes all trainable parameters of the hierarchical model. The next-stage prediction head is discarded. The retained SleepGPT encoder, global encoder, classification token, and classification head are jointly optimized using cross-entropy between the predicted class probabilities and recording-level disorder labels. Padding and attention masks accommodate variable-length sequences in mini-batches. Appendix E illustrates the HTN architecture.

Table 1: Datasets and evaluation roles. Counts distinguish recordings, subjects, and scored epochs.

Dataset	Size used	Role
SHHS-1	5,793 recordings	Stage-token pretraining
SleepEDF	78 subjects; 199,352 epochs	Staging; added controls for CAP task
MASS	200 subjects; 229,040 epochs	Staging and cross-cohort transfer
Physio2018	994 recordings; 892,262 epochs	Staging and cross-cohort transfer
BOAS	128 nights; 120,095 epochs	Paired PSG/headband staging
CAP (augmented)	186 subjects	Abnormal versus normal classification
ISRUC (subset)	86 subjects	External abnormal-sleep classification
MNC (three cohorts)	407 subjects	T1N classification; abnormal-sleep test

4 Experiments and Results

4.1 Experimental Setup

Data and transfer protocols. We use eight datasets across pretraining and evaluation (Table 1). The SHHS-1 pretraining corpus is separate from downstream evaluation cohorts. For staging, fixed YASA models are evaluated on SleepEDF, MASS, and Physio2018; TinySleepNet and XSleepNet are trained on MASS and evaluated on Physio2018, and vice versa. Each comparison evaluates the same base outputs with and without SleepGPT. These protocols test inference-time transfer rather than target-cohort retraining. A separate within-cohort experiment removes the temporal LSTM modules from the two neural staging models, testing whether the benefit depends on the availability of signal context. BOAS provides simultaneous PSG and headband EEG for modality-specific cross-validation. Dataset preprocessing and configurations appear in Appendices A and B.

Classification tasks. The hypnogram-based classification experiments use expert-scored sleep-stage sequences as inputs. The augmented CAP task combines 92 abnormal and 16 normal CAP recordings with 78 SleepEDF controls. Because additional normal controls are drawn from SleepEDF, the augmented CAP experiment may retain cohort-related differences between diagnostic groups. We therefore complement within-cohort cross-validation with external evaluation on ISRUC and MNC, where the CAP-trained models are applied without target-cohort retraining. For external evaluation, models trained on the full augmented CAP dataset are applied to 76 obstructive sleep apnea patients and 10 controls from ISRUC, and to 125 patients and 282 controls from MNC. These evaluations assess abnormal-versus-normal classification across cohorts with different disorder compositions. The second task distinguishes 82 type-1 narcolepsy (T1N) cases from 43 other hypersomnia cases and 282 controls in the MNC cohorts. T1N results are within-cohort cross-validation results.

Baselines and metrics. Staging comparisons use YASA, TinySleepNet, and XSleepNet (Vallat & Walker, 2021; Supratak & Guo, 2020; Phan et al., 2022a). We report accuracy, macro-F1 (MF1), and Cohen's κ. Classification baselines include XGBoost on hypnogram statistics, XGBoost on hypnodensity features for MNC (Stephansen et al., 2018), and a fastText-style embedding-and-averaging network (BaseNet) (Joulin et al., 2017). We compare the hierarchical transformer (HTN) with and without SleepGPT initialization using balanced accuracy (BACC), sensitivity, and specificity. Classification tables report bootstrap estimates and 95% confidence intervals.

Training. SleepGPT is pretrained for 50 epochs with Adam, batch size 256, learning rate 5×10^{-4}, and cosine decay. HTN training uses 100 epochs, batch size 16, and initial learning rate 5×10^{-5}. We varied the local context length from 5 to 120 stage tokens and selected $K = 60$ to balance predictive performance and computational cost. The same segment length is used for the HTN local encoder. For within-cohort experiments, α was selected from $\{0, 0.1, \ldots, 1.0\}$ based on validation accuracy using subjects from the training fold only, and was then fixed for evaluation on the corresponding held-out test fold. For cross-cohort experiments, the same procedure was performed using the source-cohort validation set, and the selected α was fixed when evaluating the target cohort. Ten-fold subject-level cross-validation is used for the within-cohort staging ablations, BOAS staging experiments, and within-cohort classification on augmented CAP and MNC. Repeated recordings from the same subject are kept in the same split. Cross-cohort experiments instead evaluate source-trained models

Table 2: External staging performance. Entries show the base result → the result with SleepGPT. Source and target refer to the PSG classifier; SleepGPT is pretrained on SHHS-1 throughout. ACC and MF1 are percentages.

Classifier	Source → target	ACC	MF1	κ
YASA	Pretrained → SleepEDF	72.7 → 76.9	63.3 → 68.2	.611 → .677
	Pretrained → MASS	78.6 → 80.2	71.2 → 73.6	.700 → .724
	Pretrained → Physio2018	71.4 → 73.5	66.2 → 68.6	.613 → .640
TinySleepNet	Physio2018 → MASS	74.1 → 77.0	67.1 → 71.3	.617 → .665
	MASS → Physio2018	70.5 → 72.1	66.3 → 68.0	.600 → .621
XSleepNet	Physio2018 → MASS	74.4 → 76.7	68.5 → 71.3	.626 → .661
	MASS → Physio2018	70.3 → 72.2	65.8 → 67.4	.594 → .617

Table 3: Effect of the PSG model's temporal module under within-cohort evaluation. Entries are accuracy (%) without → with SleepGPT; parentheses show gains in percentage points.

Classifier	EEG temporal module	SleepEDF	MASS
TinySleepNet	Removed	80.5 → 82.6 (+2.1)	81.7 → 83.2 (+1.5)
TinySleepNet	Present	83.1 → 83.6 (+0.5)	84.2 → 84.8 (+0.6)
XSleepNet	Removed	80.3 → 82.8 (+2.5)	82.1 → 83.4 (+1.3)
XSleepNet	Present	83.7 → 84.3 (+0.6)	85.3 → 85.7 (+0.4)

on independent target cohorts. SleepGPT parameters are not updated during the staging experiments. Further implementation details are provided in Appendix B.

4.2 Sleep Staging Refinement

Cross-dataset evaluation. Table 2 reports the seven external staging comparisons. SleepGPT improves accuracy, MF1, and κ in every comparison. YASA gains 4.2, 1.6, and 2.1 accuracy points on SleepEDF, MASS, and Physio2018, respectively. With Physio2018 as the source and MASS as the target, TinySleepNet gains 2.9 points and XSleepNet gains 2.3 points. In the reverse direction, the gains are 1.6 and 1.9 points. These results show that a single stage-token model can complement both feature-based and neural signal classifiers without being retrained on their target datasets.

The benefit is not uniform across stages. For example, YASA's SleepEDF wake accuracy decreases from 92.6% to 89.1%, even as overall accuracy and MF1 improve. N1 remains difficult, with accuracy increasing from 13.2% to 20.2% in this setting. Appendix C reports complete stage-specific values.

Contribution beyond EEG temporal context. Removing the temporal LSTM yields non-sequential versions of TinySleepNet and XSleepNet that classify individual EEG epochs. Table 3 compares these variants with the original sequential models under within-cohort evaluation. The gains are larger for the non-sequential models: 2.1–2.5 points on SleepEDF and 1.3–1.5 points on MASS. However, the sequential models also improve by 0.4–0.6 points. These results indicate that stage-sequence context complements the temporal EEG representations learned by the staging models.

Wearable EEG evaluation. We evaluate SleepGPT on the simultaneous PSG and headband EEG recordings in BOAS, with the base staging models trained and evaluated separately for each modality. As shown in Table 4, SleepGPT improves accuracy, MF1, and κ for both TinySleepNet and XSleepNet in both modalities. On headband EEG, accuracy increases from 80.7% to 81.3% for TinySleepNet and from 83.0% to 83.9% for XSleepNet; the corresponding MF1 scores increase from 62.5% to 64.5% and from 65.7% to 67.8%. On PSG, accuracy increases from 80.9% to 81.6% and from 83.4% to 84.3%, respectively. These results show that stage-sequence refinement also benefits classifiers using wearable EEG inputs. Figure 2 summarizes the improvements and presents two headband hypnogram examples comparing predictions before and after refinement with ground truth.

Table 4: BOAS cross-validation on simultaneous PSG and headband EEG. Entries show the base result → the result with SleepGPT. ACC and MF1 are percentages.

Modality	Classifier	ACC	MF1	κ
PSG	TinySleepNet	80.9 → 81.6	64.1 → 65.6	.660 → .673
	XSleepNet	83.4 → 84.3	68.8 → 70.4	.715 → .728
Headband	TinySleepNet	80.7 → 81.3	62.5 → 64.5	.652 → .667
	XSleepNet	83.0 → 83.9	65.7 → 67.8	.694 → .712

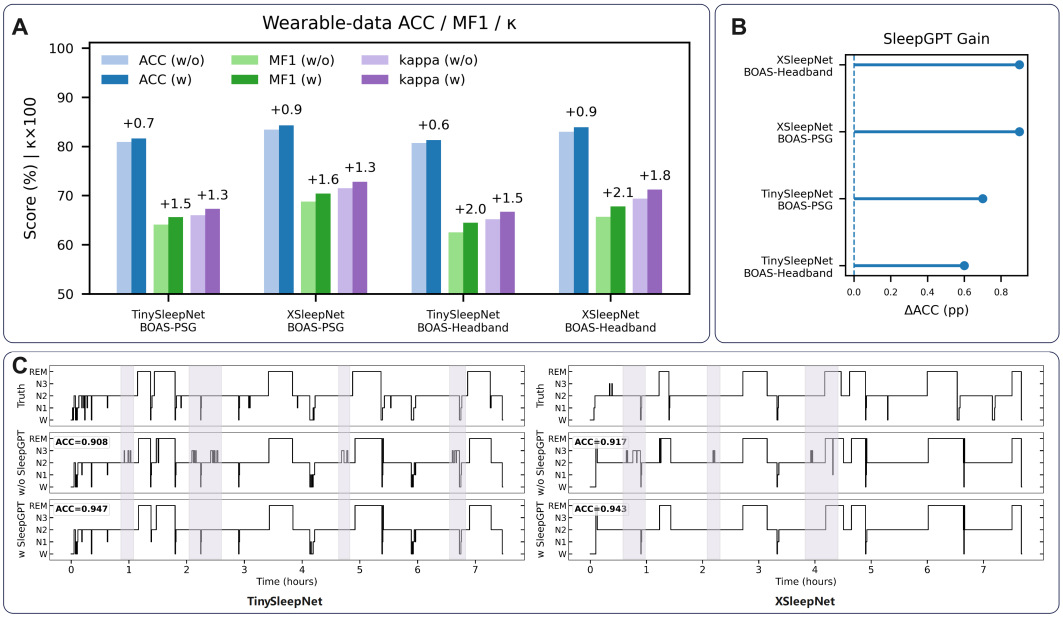


Figure 2: BOAS staging results and selected headband examples. (A) ACC, MF1, and κ with and without SleepGPT. (B) Accuracy gains in percentage points. (C) Two examples of stage predictions and expert reference sequences; shaded areas highlight selected refinements.

4.3 Sleep Disorder Classification

Abnormal sleep classification. We first evaluate normal-versus-abnormal classification on augmented CAP and external cohorts (Table 5). In CAP cross-validation, the pretrained HTN achieves BACC 0.962, compared with 0.893 for the same architecture trained from scratch, 0.910 for the hypnogram-feature baseline, and 0.849 for BaseNet. Initializing the local encoder with SleepGPT therefore improves BACC by 6.9 percentage points over training the hierarchy from scratch.

Because the augmented CAP task includes controls from SleepEDF, cohort-related differences could contribute to its within-cohort performance. We addressed this concern by evaluating models trained on the full augmented CAP dataset, without target-cohort retraining, on two independent cohorts: ISRUC, whose abnormal group comprises patients with obstructive sleep apnea, and MNC, whose abnormal group comprises patients with narcolepsy or hypersomnia. The improvement persisted under both external evaluations, with BACC increasing from 0.766 to 0.899 on ISRUC and from 0.800 to 0.837 on MNC. These results indicate that the benefit is not confined to distinguishing CAP abnormalities from SleepEDF controls, although they do not completely exclude cohort-related effects. On ISRUC, sensitivity is 1.000 and specificity is 0.797; the specificity interval is wide (95% CI: 0.500–1.000), consistent with the small control group. Complete sensitivity and specificity results are reported in Appendix D.

Type-1 narcolepsy classification. The second task distinguishes T1N from other hypersomnia cases and controls using cross-validation on MNC. The pretrained HTN achieves BACC 0.929, compared

Table 5: Sleep disorder classification: BACC with 95% bootstrap confidence intervals. CAP denotes the augmented CAP task. The first three result columns evaluate abnormal versus normal sleep; the final column evaluates T1N versus other hypersomnia cases and controls. CV denotes cross-validation.

	Abnormal vs. normal			T1N vs. others
Method	CAP CV	CAP → ISRUC	CAP → MNC	MNC CV
Hypnogram	.910 (.868–.948)	.804 (.638–.945)	.778 (.735–.820)	.843 (.797–.885)
Hypnodensity	–	–	–	.851 (.805–.894)
BaseNet	.849 (.793–.899)	.670 (.504–.830)	.749 (.697–.806)	.797 (.745–.850)
HTN, scratch	.893 (.848–.935)	.766 (.599–.921)	.800 (.758–.842)	.839 (.787–.888)
HTN, pretrained	**.962** (.935–.985)	**.899** (.750–1.000)	**.837** (.801–.875)	**.929** (.896–.960)

with 0.839 for the same hierarchy trained from scratch, 0.851 for the hypnodensity-feature baseline, 0.843 for the hypnogram-feature baseline, and 0.797 for BaseNet (Table 5). The 9.0-percentage-point improvement over training from scratch, together with the CAP results, supports the use of next-stage pretraining for recording-level disorder classification.

Attention visualizations for both tasks are provided in Appendix F. In the selected T1N examples, the global encoder assigns greater attention to segments containing short wake-to-REM latency or direct wake-to-REM transitions.

5 Discussion

The central finding of this study is that sleep-stage dynamics can be learned independently of physiological signal representations and subsequently reused across prediction settings. SleepGPT is intentionally compact and operates only on stage tokens; its value therefore lies not in model scale, but in providing a transferable sequence model decoupled from the PSG classifier. Expert-scored hypnograms can thus serve as a pretraining resource for both sleep staging refinement and disorder classification. Although its shared learning objective is next-stage prediction, the pretrained model is useful both for providing local stage-sequence priors and for initializing representations of whole-night sleep organization. This extends the role of stage annotations beyond supervision for PSG classifiers or inputs to predefined sleep statistics, supporting the reuse of temporal structure learned directly in the output-label space across signal classifiers and downstream tasks.

For staging, the results indicate that stage-sequence context complements rather than duplicates the temporal information learned from EEG. The same frozen SleepGPT model improves YASA, TinySleepNet, and XSleepNet across external datasets, including variants that already model neighboring EEG epochs. The larger gains for the non-sequential variants show that the sequence prior is especially useful when signal-based temporal context is absent, while the gains for the original models show that it remains informative when such context is present. This complementarity likely reflects the different learning signals: PSG classifiers optimize signal discrimination, whereas SleepGPT captures regularities in expert-scored stage sequences. The BOAS results further show that this combination transfers to headband EEG, supporting sequence refinement as a modality-agnostic addition to existing staging systems.

For disorder classification, the hierarchical architecture uses the pretrained local encoder to relate short stage patterns to whole-night sleep organization. Unlike methods based on selected hypnogram statistics, it learns both segment representations and their aggregation from recording-level labels. Its improvement over the same hierarchy trained from scratch on abnormal-versus-normal classification and T1N identification suggests that next-stage prediction provides an informative initialization even without disorder labels. The advantage also persists when CAP-trained models are evaluated on ISRUC and MNC, despite differences in cohort composition. Attention visualizations highlight segments with short wake-to-REM latency or direct wake-to-REM transitions, offering hypotheses about relevant patterns but not evidence that they are diagnostic biomarkers.

Operating on stage tokens makes downstream modeling independent of raw waveform inputs, but limits it to information preserved in the hypnogram. The classification experiments here use expert-scored hypnograms; applying the classifier to automatically generated sequences introduces staging

errors that require separate evaluation. Moreover, discrete stages omit within-epoch physiological detail. Combining learned hypnogram representations with complementary signal features is therefore a natural extension, particularly for disorders whose distinguishing features are not fully expressed in sleep macrostructure.

Several limitations remain. Pretraining on SHHS-1 may favor stage patterns prevalent in that adult cohort, and feeding predictions back into the history can propagate errors or suppress valid atypical transitions. Broader pretraining cohorts and uncertainty-aware fusion could help address these effects. Although the context-length analysis indicates that SleepGPT benefits from extended stage histories, the present study does not isolate which specific higher-order transition patterns account for this benefit. Future work could characterize these learned dependencies more explicitly and compare alternative sequence priors under matched transfer settings. For classification, adding SleepEDF controls to CAP and pooling MNC cohorts introduce potential cohort-related confounding; larger, independently collected cohorts are needed to assess robustness. Finally, prospective evaluation is needed to determine clinical utility, and attention patterns require validation before they can be interpreted as biomarkers.

6 Conclusion

SleepGPT learns sleep-stage dynamics through autoregressive pretraining on expert-scored hypnograms. The pretrained model improves predictions from existing sleep staging classifiers across datasets and recording modalities without retraining those classifiers. It also provides an effective initialization for hierarchical hypnogram classification, improving abnormal-sleep and type-1 narcolepsy classification over the same architecture trained from scratch. These findings show that expert-scored hypnograms can serve as a reusable pretraining resource for learning temporal structure directly in the stage-label space, complementing physiological signal modeling across both epoch-level staging and recording-level classification.

AI Use Statement

Generative AI tools were used to assist with English-language editing and LaTeX formatting. The authors reviewed and verified all AI-assisted revisions.

Ethics Statement

This work analyzes existing sleep datasets with expert annotations. Dataset access is subject to the terms and approvals of the original providers; the source studies report their own participant consent and ethics procedures. No new participants were recruited for this study. Sleep recordings and diagnostic labels are sensitive health data, and the models are intended for research rather than standalone clinical decision-making.

Reproducibility Statement

Detailed information for reproducing the experiments, including dataset composition, preprocessing steps, architectural settings, and optimization parameters, is provided in the Appendix. Source code implementing the models, training procedures, and evaluation pipelines is provided in the supplementary materials.

References

Matt T. Bianchi, Sydney S. Cash, Joseph Mietus, Chung-Kang Peng, and Robert Thomas. Obstructive sleep apnea alters sleep stage transition dynamics. *PLOS ONE*, 5(6):1–12, 06 2010. doi: 10.1371/journal.pone.0011356. URL https://doi.org/10.1371/journal.pone.0011356.

William Chan, Navdeep Jaitly, Quoc Le, and Oriol Vinyals. Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4960–4964, 2016. doi: 10.1109/ICASSP.2016.7472621.

S. I. Dimitriadis, C. I. Salis, and D. Liparas. An automatic sleep disorder detection based on EEG cross-frequency coupling and random forest model. *Journal of Neural Engineering*, 18 (4), 2021. doi: 10.1088/1741-2552/abf773. URL https://www.ncbi.nlm.nih.gov/pubmed/33848982.

E. Eldele, Z. Chen, C. Liu, M. Wu, C. K. 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*, 29:809–818, 2021. doi: 10.1109/TNSRE.2021.3076234. URL https://www.ncbi.nlm.nih.gov/pubmed/33909566.

M. M. Ghassemi, B. E. Moody, L. H. Lehman, C. Song, Q. Li, H. Sun, R. G. Mark, M. B. Westover, and G. D. Clifford. You snooze, you win: the physionet/computing in cardiology challenge 2018. *Computing in Cardiology (2010)*, 45, 2018. doi: 10.22489/cinc.2018.049.

Ary L. Goldberger, Luis A. N. Amaral, Leon Glass, Jeffrey M. Hausdorff, Plamen Ch. Ivanov, Roger G. Mark, Joseph E. Mietus, George B. Moody, Chung-Kang Peng, and H. Eugene Stanley. PhysioBank, PhysioToolkit, and PhysioNet. *Circulation*, 101(23):e215–e220, 2000. doi: 10.1161/01.CIR.101.23.e215. URL https://www.ahajournals.org/doi/abs/10.1161/01.CIR.101.23.e215.

Conrad Iber, Sonia Ancoli-Israel, A.L. Chesson, and Stuart Quan. The AASM manual for the scoring of sleep and associated events: Rules, terminology and technical specifications. *American Academy of Sleep Medicine*, 01 2007.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pp. 427–431, Valencia, Spain, April 2017. Association for Computational Linguistics. URL https://aclanthology.org/E17-2068.

Shigeki Karita, Nelson Enrique Yalta Soplin, Shinji Watanabe, Marc Delcroix, Atsunori Ogawa, and Tomohiro Nakatani. Improving transformer-based end-to-end speech recognition with connectionist temporal classification and language model integration. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 2019-September:1408–1412, 2019. doi: 10.21437/Interspeech.2019-1938.

B. Kemp, A.H. Zwinderman, B. Tuk, H.A.C. Kamphuisen, and J.J.L. Oberye. Analysis of a sleep-dependent neuronal feedback loop: the slow-wave microcontinuity of the EEG. *IEEE Transactions on Biomedical Engineering*, 47(9):1185–1194, 2000. doi: 10.1109/10.867928.

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*, 124:180–192, 2016. doi: https://doi.org/10.1016/j.cmpb.2015.10.013. URL https://www.sciencedirect.com/science/article/pii/S0169260715002734.

Sheng-Fu Liang, Chin-En Kuo, Yu-Han Hu, and Yu-Shian Cheng. A rule-based automatic sleep staging method. *Journal of Neuroscience Methods*, 205(1):169–176, 2012. doi: https://doi.org/10.1016/j.jneumeth.2011.12.022. URL https://www.sciencedirect.com/science/article/pii/S016502701100759X.

Eduardo López-Larraz, María Sierra-Torralba, Sergio Clemente, Galit Fierro, David Oriol, Javier Minguez, Luis Montesano, and Jens G. Klinzing. Bitbrain open access sleep dataset. OpenNeuro dataset, 2024.

M. W. Mahowald and C. H. Schenck. Insights from studying human sleep disorders. *Nature*, 437 (7063):1279–85, 2005. doi: 10.1038/nature04287. URL https://www.ncbi.nlm.nih.gov/pubmed/16251953.

Alexander Malafeev, Dmitry Laptev, Stefan Bauer, Ximena Omlin, Aleksandra Wierzbicka, Adam Wichniak, Wojciech Jernajczyk, Robert Riener, Joachim Buhmann, and Peter Achermann. Automatic human sleep stage scoring using deep neural networks. *Frontiers in Neuroscience*, 12, 2018. URL https://www.frontiersin.org/articles/10.3389/fnins.2018.00781.

S. Mousavi, F. Afghah, and U. R. Acharya. SleepEEGNet: Automated sleep stage scoring with sequence to sequence deep learning approach. *PLoS One*, 14(5):e0216456, 2019. doi: 10.1371/journal.pone.0216456. URL https://www.ncbi.nlm.nih.gov/pubmed/31063501.

Christian O'Reilly, Nadia Gosselin, Julie Carrier, and Tore Nielsen. Montreal Archive of Sleep Studies: an open-access resource for instrument benchmarking and exploratory research. *Journal of Sleep Research*, 23(6):628–635, 2014. doi: https://doi.org/10.1111/jsr.12169. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/jsr.12169.

Raghavendra Pappagari, Piotr Zelasko, Jesús Villalba, Yishay Carmiel, and Najim Dehak. Hierarchical transformers for long document classification. In *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 838–844, 2019. doi: 10.1109/ASRU46091.2019.9003958.

H. Phan, F. Andreotti, N. Cooray, O. Y. Chen, and M. De Vos. SeqSleepNet: End-to-end hierarchical recurrent neural network for sequence-to-sequence automatic sleep staging. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 27(3):400–410, 2019. doi: 10.1109/TNSRE.2019.2896659. URL https://www.ncbi.nlm.nih.gov/pubmed/30716040.

H. Phan, O. Y. Chen, M. C. Tran, P. Koch, A. Mertins, and M. De Vos. XSleepNet: Multi-view sequential model for automatic sleep staging. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(9):5903–5915, 2022a. doi: 10.1109/TPAMI.2021.3070057. URL https://www.ncbi.nlm.nih.gov/pubmed/33788679.

Huy Phan, Kaare Mikkelsen, Oliver Y. Chén, Philipp Koch, Alfred Mertins, and Maarten De Vos. SleepTransformer: Automatic sleep staging with interpretability and uncertainty quantification. *IEEE Transactions on Biomedical Engineering*, 69(8):2456–2467, 2022b. doi: 10.1109/TBME.2022.3147187.

Huy Phan, Kristian P. Lorenzen, Elisabeth Heremans, Oliver Y. Chén, Minh C. Tran, Philipp Koch, Alfred Mertins, Mathias Baumert, Kaare B. Mikkelsen, and Maarten De Vos. L-SeqSleepNet: Whole-cycle long sequence modeling for automatic sleep staging. *IEEE Journal of Biomedical and Health Informatics*, 27(10):4748–4757, 2023. doi: 10.1109/JBHI.2023.3303197.

Jaeun Phyo, Wonjun Ko, Eunjin Jeon, and Heung-Il Suk. TransSleep: Transitioning-aware attention-based deep neural network for sleep staging. *IEEE Transactions on Cybernetics*, 53(7):4500–4510, 2023. doi: 10.1109/TCYB.2022.3198997.

Stuart F. Quan, Barbara V. Howard, Conrad Iber, James P. Kiley, F. Javier Nieto, George T. O'Connor, David M. Rapoport, Susan Redline, John Robbins, Jonathan M. Samet, and Patricia W. Wahl. The Sleep Heart Health Study: Design, rationale, and methods. *Sleep*, 20(12):1077–1085, 12 1997. doi: 10.1093/sleep/20.12.1077. URL https://doi.org/10.1093/sleep/20.12.1077.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Hogeon Seo, Seunghyeok Back, Seongju Lee, Deokhwan Park, Tae Kim, and Kyoobin Lee. Intra- and inter-epoch temporal context network (IITNet) using sub-epoch features for automatic sleep scoring on raw single-channel EEG. *Biomedical Signal Processing and Control*, 61:102037, 2020. doi: https://doi.org/10.1016/j.bspc.2020.102037. URL https://www.sciencedirect.com/science/article/pii/S1746809420301932.

M. Sharma, H. S. Dhiman, and U. R. Acharya. Automatic identification of insomnia using optimal antisymmetric biorthogonal wavelet filter bank with ecg signals. *Computers in Biology and Medicine*, 131:104246, 2021. doi: 10.1016/j.compbiomed.2021.104246. URL https://www.ncbi.nlm.nih.gov/pubmed/33631498.

J. B. Stephansen, A. N. Olesen, M. Olsen, A. Ambati, E. B. Leary, H. E. Moore, O. Carrillo, L. Lin, F. Han, H. Yan, Y. L. Sun, Y. Dauvilliers, S. Scholz, L. Barateau, B. Hogl, A. Stefani, S. C. Hong, T. W. Kim, F. Pizza, G. Plazzi, S. Vandi, E. Antelmi, D. Perrin, S. T. Kuna, P. K. Schweitzer, C. Kushida, P. E. Peppard, H. B. D. Sorensen, P. Jennum, and E. Mignot. Neural network analysis of sleep stages enables efficient diagnosis of narcolepsy. *Nature Communications*, 9(1):5229, 2018. doi: 10.1038/s41467-018-07229-3. URL https://www.ncbi.nlm.nih.gov/pubmed/30523329.

Haoqi Sun, Jian Jia, Balaji Goparaju, Guang-Bin Huang, Olga Sourina, Matt Travis Bianchi, and M. Brandon Westover. Large-scale automated sleep staging. *Sleep*, 40(10):zsx139, 2017. doi: 10.1093/sleep/zsx139.

A. Supratak and Y. Guo. TinySleepNet: An efficient deep learning model for sleep stage scoring based on raw single-channel EEG. In *The 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, pp. 641–644, 2020. ISBN 2694-0604. doi: 10.1109/EMBC44109.2020.9176741.

A. Supratak, H. Dong, C. Wu, and Y. Guo. 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, 2017. doi: 10.1109/TNSRE.2017.2721116. URL https://www.ncbi.nlm.nih.gov/pubmed/28678710.

Mario Giovanni Terzano, Liborio Parrino, Adriano Sherieri, Ronald Chervin, Sudhansu Chokroverty, Christian Guilleminault, Max Hirshkowitz, Mark Mahowald, Harvey Moldofsky, Agostino Rosa, Robert Thomas, and Arthur Walters. Atlas, rules, and recording techniques for the scoring of cyclic alternating pattern (CAP) in human sleep. *Sleep Medicine*, 2(6):537–553, 2001. doi: https://doi.org/10.1016/S1389-9457(01)00149-6. URL https://www.sciencedirect.com/science/article/pii/S1389945701001496.

Raphael Vallat and Matthew P Walker. An open-source, high-performance tool for automated sleep staging. *eLife*, 10:e70092, oct 2021. doi: 10.7554/eLife.70092.

Edward A. Wolpert. A manual of standardized terminology, techniques and scoring system for sleep stages of human subjects. *Archives of General Psychiatry*, 20(2):246–247, 02 1969. doi: 10.1001/archpsyc.1969.01740140118016. URL https://doi.org/10.1001/archpsyc.1969.01740140118016.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1480–1489, San Diego, California, June 2016. Association for Computational Linguistics. doi: 10.18653/v1/N16-1174. URL https://aclanthology.org/N16-1174.

Guo-Qiang Zhang, Licong Cui, Remo Mueller, Shiqiang Tao, Matthew Kim, Michael Rueschman, Sara Mariani, Daniel Mobley, and Susan Redline. The National Sleep Research Resource: towards a sleep data commons. *Journal of the American Medical Informatics Association*, 25(10):1351–1358, 05 2018. doi: 10.1093/jamia/ocy064. URL https://doi.org/10.1093/jamia/ocy064.

A Datasets and Preprocessing

A.1 Pretraining Corpus

The Sleep Heart Health Study (SHHS) is a multicenter study of sleep-disordered breathing (Quan et al., 1997; Zhang et al., 2018). SleepGPT pretraining uses SHHS-1, containing 5,793 subjects aged 39–90 years. Recordings contain 360–1,199 scored epochs (mean 1,012), totaling 5,863,207 epochs. Sleep stages were mapped to a five-class vocabulary comprising W, N1, N2, N3, and REM, with R&K stages N3 and N4 merged into N3 (Wolpert, 1969; Iber et al., 2007). These annotations provide the pretraining targets; PSG waveforms and diagnostic labels are not pretraining inputs.

A.2 Staging Datasets

SleepEDF. The 2018 SleepEDF Expanded release contains 153 recordings from 78 subjects (Kemp et al., 2000; Goldberger et al., 2000). The evaluated EEG channel is Fpz-Cz. Stages N3 and N4 are merged according to AASM guidelines. All recordings from the same subject are assigned to the same split.

MASS. The Montreal Archive of Sleep Studies contains recordings from 200 subjects in five subsets (O'Reilly et al., 2014). SS1 and SS3 use AASM scoring, whereas SS2, SS4, and SS5 use R&K scoring. Labels are harmonized to the five-stage vocabulary. Signal segments were prepared according to the input requirements of each pretrained staging model. The evaluated EEG derivations are C4-A1/C3-A2.

Physio2018. The labeled training partition of the PhysioNet 2018 Challenge supplies 994 recordings for evaluation (Ghassemi et al., 2018). The evaluated channel is C3-A2. The dataset provides five-stage annotations under AASM scoring and is used both as a source and as a target for cross-cohort staging experiments.

BOAS. The Bitbrain Open Access Sleep dataset contains 128 nights with simultaneous PSG and wearable headband EEG (López-Larraz et al., 2024). The PSG system records F3, F4, C3, C4, O1, and O2; the headband records AF7 and AF8. Both systems sample at 256 Hz. Three scorers annotate sleep stages, with a fourth expert establishing consensus. The reported comparison uses C4 PSG and AF7 headband data and evaluates the modalities separately against the scored stages.

Table 6: Stage distributions for the three external staging datasets after preprocessing. Entries are numbers of scored epochs, not subjects.

Dataset	W	N1	N2	N3	REM	Total
SleepEDF	69,824	21,522	69,132	13,039	25,835	199,352
MASS	31,184	19,359	107,930	30,383	40,184	229,040
Physio2018	157,945	136,978	377,870	102,592	116,877	892,262

A.3 Classification Cohorts

Augmented CAP. The original CAP Sleep Database comprises 108 recordings: 92 associated with sleep disorders and 16 normal recordings (Terzano et al., 2001). Table 7 lists the original groups. Because several disease groups are small, the task is binary abnormal-versus-normal classification. The evaluated augmented cohort adds one recording from each of 78 SleepEDF control subjects, yielding 94 normal and 92 abnormal subjects.

ISRUC. ISRUC-Sleep includes recordings of subjects with sleep disorders and healthy controls (Khalighi et al., 2016). The subset used here contains 76 obstructive sleep apnea patients and 10 controls. It is an external test of the CAP-trained abnormal-versus-normal classifier.

Table 7: Original CAP groups, before adding SleepEDF controls.

Group	Subjects
Bruxism	2
Sleep-disordered breathing	4
Insomnia	9
Narcolepsy	5
Nocturnal frontal lobe epilepsy	40
Periodic leg movements	10
REM behavior disorder	22
No pathology	16

MNC. We use the CNC, DHC, and SSC cohorts of the Mignot Nature Communications dataset (Stephansen et al., 2018; Zhang et al., 2018), comprising 407 subjects with the annotations and labels required for these experiments. The T1N task combines other hypersomnia and non-narcolepsy controls into the negative class. For external evaluation of CAP-trained models, T1N and other hypersomnia are instead combined into the abnormal class. Both tasks use the same 407 subjects with these task-specific class definitions.

Table 8: MNC subjects included in the classification experiments.

Group	CNC	DHC	SSC	Total
Type-1 narcolepsy	54	21	7	82
Other hypersomnia	0	38	5	43
Non-narcolepsy control	23	20	239	282
Total	77	79	251	407

B Implementation Details

B.1 Pretraining and Downstream Optimization

Table 9 records the model sizes and optimization settings. Implementation uses PyTorch and the HuggingFace Transformers library. All downstream staging experiments reuse pretrained SleepGPT weights; the classification hierarchy fine-tunes its local encoder.

Table 9: Architectural and optimization settings. A stage token corresponds to one 30-second epoch.

Setting	SleepGPT	Neural staging models	Classification HTN
Local transformer blocks	3	Model-dependent	3
Global transformer blocks	–	–	1
Attention heads	6	Model-dependent	6
Embedding/hidden dimension	48	Model-dependent	48
Local stage context	60	–	60
Training epochs	50	200	100
Batch size	256	256	16
Initial learning rate	5×10^{-4}	5×10^{-4}	5×10^{-5}
Optimizer	Adam	Adam	Adam
Schedule	Cosine decay	Cosine decay	Cosine decay

For the temporal-module ablation, removing the LSTM gives the non-sequential TinySleepNet and XSleepNet variants. The sequential models use 15 EEG epochs (the current epoch and 14 preceding epochs). Within-cohort evaluation uses ten-fold cross-validation. The staging and classification protocols use subject-level separation, keeping repeated recordings within their subject's split.

We evaluated the effect of local context length by varying K from 5 to 120 stage tokens. Performance generally improved as the context was extended from shorter histories and remained broadly stable across $K = 30$–90. Longer contexts offered limited additional benefit while increasing computational cost. We therefore selected $K = 60$ as a balance between predictive performance and computational cost and used it throughout the remaining experiments, including as the segment length for the HTN local encoder.

B.2 Probability Fusion and Predicted History

The staging model and SleepGPT provide probability distributions over the five sleep stages. These distributions are combined using Equation 3, where α weights the staging model and $1 - \alpha$ weights SleepGPT. The fusion weight α was selected using validation data only, following the validation protocol described in the experimental setup. The initial stage history is obtained from the base classifier. Subsequent predictions use the most recent K selected stages as context, and each selected stage is added to the history. Both models remain fixed during staging refinement.

B.3 Hierarchical Sequence Encoder

The local encoder uses the hidden state at the last non-padding token in each segment, followed by normalization and projection. Segment vectors are combined with a learned classification token and positional embeddings. The global transformer processes the resulting sequence, and the output at the classification token is passed to the classification head. Padding masks are used at both token and segment levels. HTN models were trained using standard cross-entropy loss without class weighting or class-balanced sampling; balanced accuracy was used as the primary evaluation metric to account for class imbalance.

The BaseNet comparison uses a fastText-style embedding layer, an averaging operation, and a hidden layer (Joulin et al., 2017). The feature-based baselines use XGBoost on stage percentages, stage latency and duration, sleep efficiency, and transition statistics. The MNC hypnodensity baseline uses XGBoost on features derived from stage-probability distributions, whereas the HTN takes discrete stage labels as input (Vallat & Walker, 2021; Stephansen et al., 2018).

B.4 METRICS

Staging accuracy is the fraction of correctly labeled epochs. MF1 averages classwise F1 scores over the five stages, and Cohen's κ adjusts observed agreement for chance. For binary classification, sensitivity is the true-positive rate, specificity is the true-negative rate, and balanced accuracy is their arithmetic mean. The positive class is abnormal sleep for CAP transfer and T1N for MNC cross-validation. Classification table entries are bootstrap means. We draw 1,000 bootstrap samples by resampling evaluation subjects with replacement, using a random seed of 42 and excluding samples containing only one class. For each retained sample, BACC, sensitivity, and specificity are computed at a probability threshold of 0.5. The mean across retained samples is reported with a 95% percentile confidence interval defined by the 2.5th and 97.5th percentiles. Confusion matrices instead report counts and row-normalized percentages from the original evaluation predictions, so their percentages can differ slightly from the bootstrap means.

ROC curves are computed from the original prediction scores, while the AUC values displayed in Figure 3 are bootstrap means. Each comparator is tested against the pretrained HTN using a two-sided paired DeLong test on the original prediction scores.

C Additional Staging Results

Tables 10 and 11 provide the full staging metrics. *w* and *w/o* denote with and without SleepGPT. ACC and MF1 are percentages; W, N1, N2, N3, and REM are the reported class-specific accuracies.

Table 10: Results of three baseline staging methods with and without SleepGPT when performing cross-dataset sleep staging on the SleepEDF (Kemp et al., 2000), MASS (O'Reilly et al., 2014), and Physio2018 (Ghassemi et al., 2018) datasets. **Dataset**: **Source** → **Target** indicate that the staging model is trained from the **Source** dataset and evaluated on the **Target** dataset.

Method	Dataset	SleepGPT	ACC	MF1	κ	W	N1	N2	N3	REM
YASA	YASA → SleepEDF	w/o	72.7	63.3	0.611	92.6	13.2	73.9	73.2	64.9
		w	76.9	68.2	0.677	89.1	20.2	81.3	81.4	76.8
	YASA → MASS	w/o	78.6	71.2	0.700	92.8	21.7	82.0	91.6	84.3
		w	80.2	73.6	0.724	95.4	26.1	82.3	94.0	86.5
	YASA → Physio2018	w/o	71.4	66.2	0.613	92.5	20.5	77.3	78.1	77.4
		w	73.5	68.6	0.640	93.6	22.8	80.7	79.3	77.7
TinySleepNet	Physio2018 → MASS	w/o	74.1	67.1	0.617	66.1	27.4	89.2	74.1	69.6
		w	77.0	71.3	0.665	66.2	39.1	90.1	82.9	71.8
	MASS → Physio2018	w/o	70.5	66.3	0.600	94.7	35.0	75.9	48.6	81.6
		w	72.1	68.0	0.621	95.1	36.4	78.2	51.3	81.8
XSleepNet	Physio2018 → MASS	w/o	74.4	68.5	0.626	64.3	39.1	88.6	83.0	61.4
		w	76.7	71.3	0.661	68.8	43.0	90.0	86.5	62.0
	MASS → Physio2018	w/o	70.3	65.8	0.594	94.2	41.8	77.1	38.8	77.0
		w	72.2	67.4	0.617	95.0	42.9	80.6	39.0	77.8

Table 11: Performance of baseline sleep staging methods and the proposed SleepGPT-powered models on the BOAS wearable dataset (López-Larraz et al., 2024). The best results are highlighted in bold, while the results with the SleepGPT-powered model are gray-shaded.

Dataset	Method	SleepGPT	ACC	MF1	κ	W	N1	N2	N3	REM
PSG	TinySleepNet	w/o	80.9	64.1	0.660	78.0	20.8	90.8	48.0	69.4
		w	81.6	65.6	0.673	78.4	23.1	**91.2**	51.1	70.5
	XSleepNet	w/o	83.4	68.8	0.715	88.7	27.0	88.2	53.9	81.1
		w	**84.3**	**70.4**	**0.728**	**89.0**	**27.9**	89.2	**56.3**	**81.7**
Headband	TinySleepNet	w/o	80.7	62.5	0.652	78.5	18.3	91.1	39.5	68.8
		w	81.3	64.5	0.667	79.5	20.8	91.0	**48.0**	69.4
	XSleepNet	w/o	83.0	65.7	0.694	84.7	23.9	91.8	36.8	73.4
		w	**83.9**	**67.8**	**0.712**	**87.0**	**26.9**	**91.9**	43.8	**74.5**

C.1 WITHIN-COHORT TEMPORAL-MODULE COMPARISON

Table 12 expands the within-cohort ablation in the main text. Published comparator results are reproduced from their cited studies and may use different experimental configurations. In particular, the IITNet result uses only MASS SS3 and is not a matched all-MASS comparison. The paired rows for each implemented model provide the with/without SleepGPT comparison.

Table 12: Performance of baseline sleep staging methods and the proposed SleepGPT-powered models on the SleepEDF and MASS datasets. Results with SleepGPT are gray-shaded. **SleepGPT**: with (w) or without (w/o) SleepGPT, **ACC**: accuracy (%), **MF1**: macro-averaging F1-score (%), κ: Cohen's kappa coefficient. **W**, **N1**, **N2**, **N3** and **REM**: sleep stage-specific accuracies (%). **TinySleepNet-Nonseq**: TinySleepNet without an LSTM module. **XSleepNet-Nonseq**: XSleepNet without an LSTM module.

Dataset	Method	SleepGPT	ACC	MF1	κ	W	N1	N2	N3	REM
SleepEDF	DeepSleepNet (Supratak et al., 2017)	w/o	77.8	71.8	0.700	90.8	44.8	78.5	67.9	71.3
	SleepEEGNet (Mousavi et al., 2019)	w/o	80.0	73.6	0.730	91.7	44.1	82.5	73.5	76.1
	AttnSleepNet (Eldele et al., 2021)	w/o	81.3	75.1	0.740	92.0	42.0	85.0	82.1	74.2
	SeqSleepNet (Phan et al., 2019)	w/o	82.6	76.4	0.760	-	-	-	-	-
	TinySleepNet-Nonseq	w/o	80.5	71.9	0.727	93.7	29.1	87.6	70.1	74.1
		w	82.6	74.3	0.755	94.7	30.2	91.4	73.0	74.8
	TinySleepNet (Supratak & Guo, 2020)	w/o	83.1	76.9	0.764	93.3	41.6	87.9	76.8	80.5
		w	83.6	77.6	0.772	93.6	44.2	88.1	77.1	81.1
	XSleepNet-Nonseq	w/o	80.3	72.7	0.725	90.4	31.8	88.5	70.9	76.6
		w	82.8	76.3	0.761	93.3	40.7	88.5	72.0	80.4
	XSleepNet (Phan et al., 2022a)	w/o	83.7	77.9	0.774	93.6	45.0	87.7	78.6	81.4
		w	84.3	78.7	0.781	93.7	47.2	88.3	79.3	81.5
MASS	DeepSleepNet (Supratak et al., 2017)	w/o	83.9	78.9	0.769	86.3	53.9	88.2	79.6	86.8
	IITNet* (Seo et al., 2020)	w/o	86.3	80.5	0.790	85.4	54.1	91.3	86.8	84.8
	TinySleepNet-Nonseq	w/o	81.7	74.3	0.736	86.5	29.4	89.4	79.1	84.5
		w	83.2	76.0	0.756	87.0	30.2	91.8	78.7	86.2
	TinySleepNet (Supratak & Guo, 2020)	w/o	84.2	79.7	0.774	87.5	51.5	89.2	82.4	85.2
		w	84.8	80.2	0.784	87.9	53.1	89.5	84.5	85.6
	XSleepNet-Nonseq	w/o	82.1	75.1	0.742	86.7	32.3	89.7	79.5	83.7
		w	83.4	77.0	0.761	88.3	36.6	90.6	80.9	84.5
	XSleepNet (Phan et al., 2022a)	w/o	85.3	80.8	0.791	88.9	53.8	90.1	83.3	86.5
		w	85.7	81.5	0.797	88.7	56.1	90.2	83.7	87.3

* The results are not directly comparable since they were evaluated on the SS3 subset with 62 healthy subjects in the MASS dataset.

D Additional Classification Results

The following tables report classification performance for the tasks evaluated in the main text. CAP means the augmented CAP cohort. Cross-cohort tests use CAP-trained models to classify abnormal versus normal sleep. MNC cross-validation instead classifies T1N versus other hypersomnia and non-narcolepsy controls.

Table 13: Sleep disorder classification performance using cross-validation on the augmented CAP dataset (Terzano et al., 2001) and external validation on the ISRUC (Khalighi et al., 2016) and MNC (Stephansen et al., 2018) datasets for abnormal versus normal sleep classification.

Dataset	Method	BACC	SENS	SPEC
CAP (CV)	Hypnogram	$0.910_{(0.868–0.948)}$	$0.957_{(0.912–0.990)}$	$0.863_{(0.787–0.931)}$
	BaseNet	$0.849_{(0.793–0.899)}$	$0.868_{(0.796–0.930)}$	$0.830_{(0.750–0.905)}$
	From scratch	$0.893_{(0.848–0.935)}$	$0.902_{(0.835–0.957)}$	$0.884_{(0.816–0.947)}$
	Pretrained	$\mathbf{0.962}_{(0.935–0.985)}$	$\mathbf{0.989}_{(0.962–1.000)}$	$\mathbf{0.936}_{(0.884–0.979)}$
CAP → ISRUC	Hypnogram	$0.804_{(0.638–0.945)}$	$0.908_{(0.846–0.962)}$	$0.700_{(0.375–1.000)}$
	BaseNet	$0.670_{(0.504–0.830)}$	$0.842_{(0.759–0.921)}$	$0.498_{(0.182–0.800)}$
	From scratch	$0.766_{(0.599–0.921)}$	$0.934_{(0.877–0.987)}$	$0.597_{(0.250–0.909)}$
	Pretrained	$\mathbf{0.899}_{(0.750–1.000)}$	$\mathbf{1.000}_{(1.000–1.000)}$	$\mathbf{0.797}_{(0.500–1.000)}$
CAP → MNC	Hypnogram	$0.778_{(0.735–0.820)}$	$0.825_{(0.756–0.889)}$	$0.731_{(0.675–0.781)}$
	BaseNet	$0.749_{(0.697–0.806)}$	$0.671_{(0.570–0.778)}$	$\mathbf{0.828}_{(0.786–0.864)}$
	From scratch	$0.800_{(0.758–0.842)}$	$0.841_{(0.771–0.902)}$	$0.758_{(0.708–0.805)}$
	Pretrained	$\mathbf{0.837}_{(0.801–0.875)}$	$\mathbf{0.879}_{(0.825–0.929)}$	$0.795_{(0.749–0.841)}$

Values are bootstrap means with 95% percentile confidence intervals (shown as subscripts), computed using the procedure described in Appendix B. **BACC**, balanced accuracy; **SENS**, sensitivity; **SPEC**, specificity. **Hypnogram** denotes the XGBoost classifier using hypnogram-derived statistical features. **BaseNet** denotes a baseline neural network. **From scratch** refers to the hierarchical transformer network (HTN) trained without pretraining. **Pretrained** denotes the HTN initialized with pretrained SleepGPT parameters. Bold values indicate the best performance for each dataset.

Table 14: Type-1 narcolepsy classification performance. The model was evaluated using cross-validation on the MNC dataset (Stephansen et al., 2018) for type-1 narcolepsy (T1N) versus other hypersomnia and non-narcolepsy controls.

Dataset	Method	BACC	SENS	SPEC
MNC (CV)	Hypnogram	$0.843_{(0.797–0.885)}$	$0.828_{(0.738–0.907)}$	$0.858_{(0.819–0.894)}$
	Hypnodensity	$0.851_{(0.805–0.894)}$	$0.816_{(0.729–0.896)}$	$0.886_{(0.854–0.917)}$
	BaseNet	$0.797_{(0.745–0.850)}$	$0.671_{(0.574–0.775)}$	$0.924_{(0.895–0.951)}$
	From scratch	$0.839_{(0.787–0.888)}$	$0.754_{(0.656–0.848)}$	$0.923_{(0.896–0.950)}$
	Pretrained	$\mathbf{0.929}_{(0.896–0.960)}$	$\mathbf{0.915}_{(0.854–0.971)}$	$\mathbf{0.944}_{(0.917–0.967)}$

Values are bootstrap means with 95% percentile confidence intervals (shown as subscripts), computed using the procedure described in Appendix B. BACC, balanced accuracy; SENS, sensitivity; SPEC, specificity. Hypnodensity denotes an empirical feature-based XGBoost classifier using hypnodensity features derived from sleep stage probability distributions. Bold values indicate the best performance for each dataset.

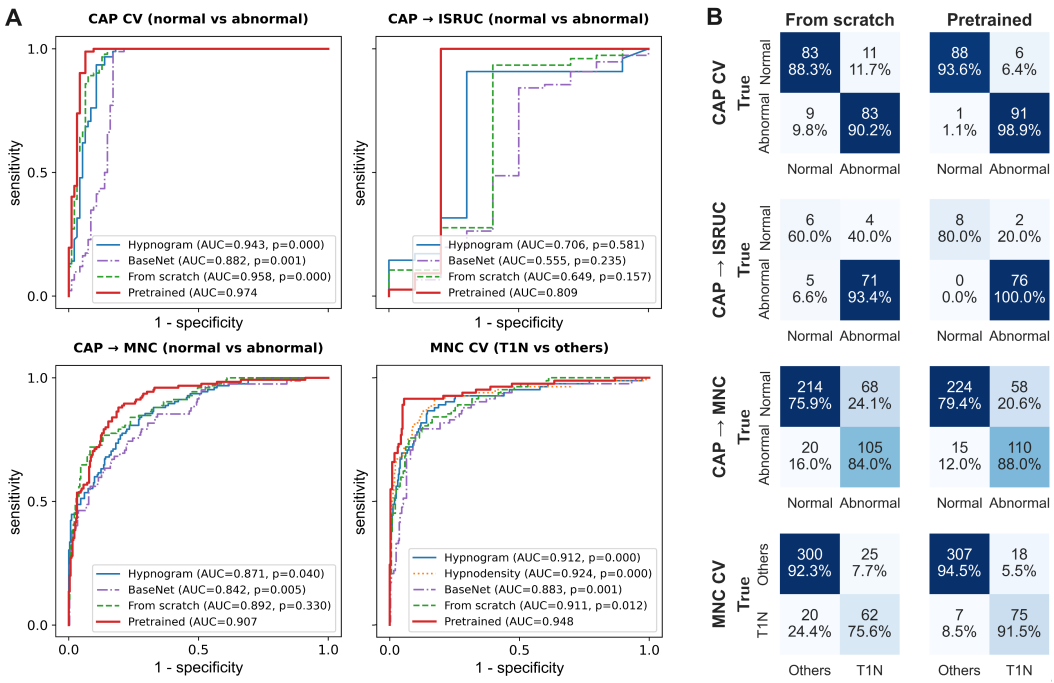


Figure 3: ROC curves and confusion matrices for sleep disorder classification. (A) ROC curves for abnormal-versus-normal classification on augmented CAP (cross-validation), CAP-to-ISRUC and CAP-to-MNC external evaluation, and T1N classification on MNC (cross-validation). Legends report bootstrap mean AUCs and unadjusted two-sided DeLong p values comparing each baseline with the pretrained HTN. Values displayed as $p = 0.000$ reflect rounding to three decimal places and indicate $p < 0.001$. (B) Confusion matrices comparing HTNs trained from scratch and initialized with SleepGPT. Rows indicate reference labels and columns indicate predicted labels; cells show counts and row-normalized percentages calculated directly from those counts. For MNC T1N classification, others denotes other hypersomnia cases and controls.

E ARCHITECTURE

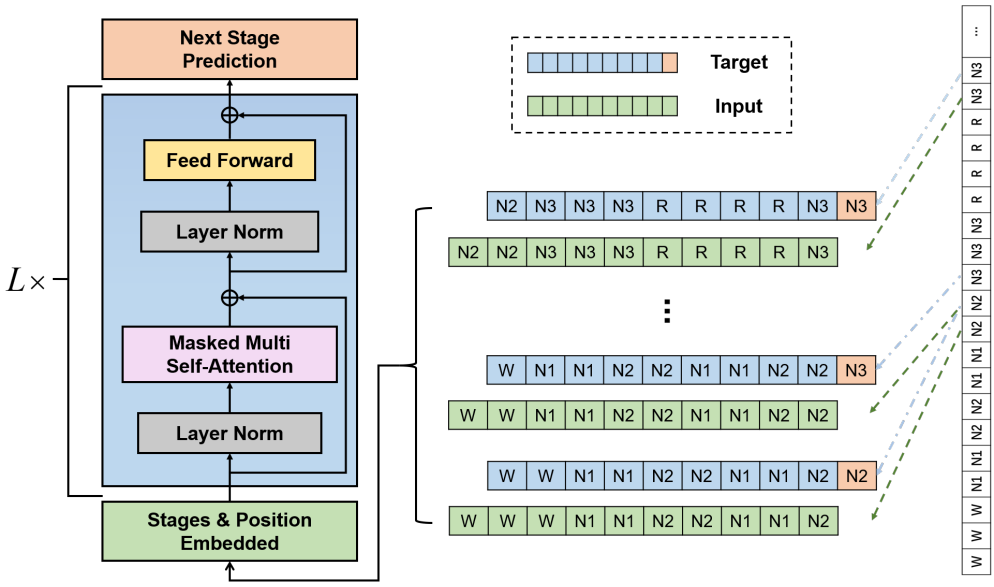


Figure 4: SleepGPT's local causal transformer and shifted stage-token training windows. The inputs and prediction targets are expert-scored stage sequences during pretraining. At staging inference, the model receives previously predicted stage tokens.

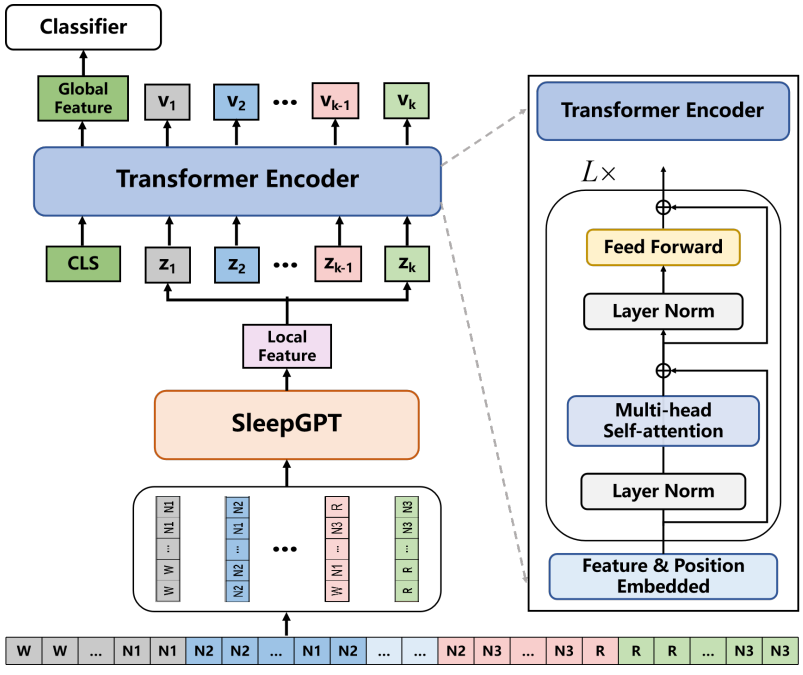


Figure 5: Hierarchical transformer for hypnogram classification. SleepGPT encodes non-overlapping local stage segments; a global transformer combines segment representations and predicts a recording-level label.

F QUALITATIVE EXAMPLES

Figure 2 illustrates selected staging refinements on headband EEG. Figures 6 and 7 show hypnograms and global-transformer attention weights for selected classification examples. Darker attention bars indicate larger weights. Gray bands highlight selected sequence regions, and red portions in the attention figures indicate REM stages.

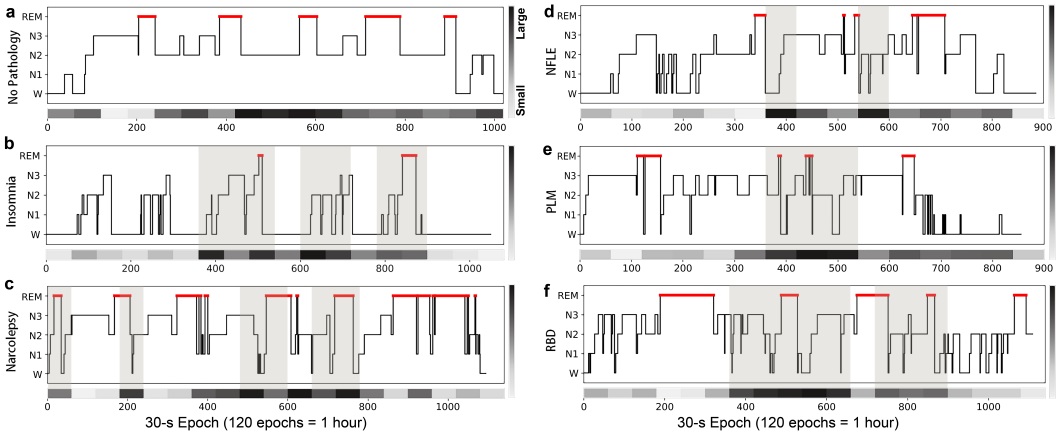


Figure 6: CAP hypnograms and attention weights for a normal subject and subjects with insomnia, narcolepsy, nocturnal frontal lobe epilepsy (NFLE), periodic leg movements (PLM), and REM behavior disorder (RBD). The horizontal axis counts 30-second epochs.

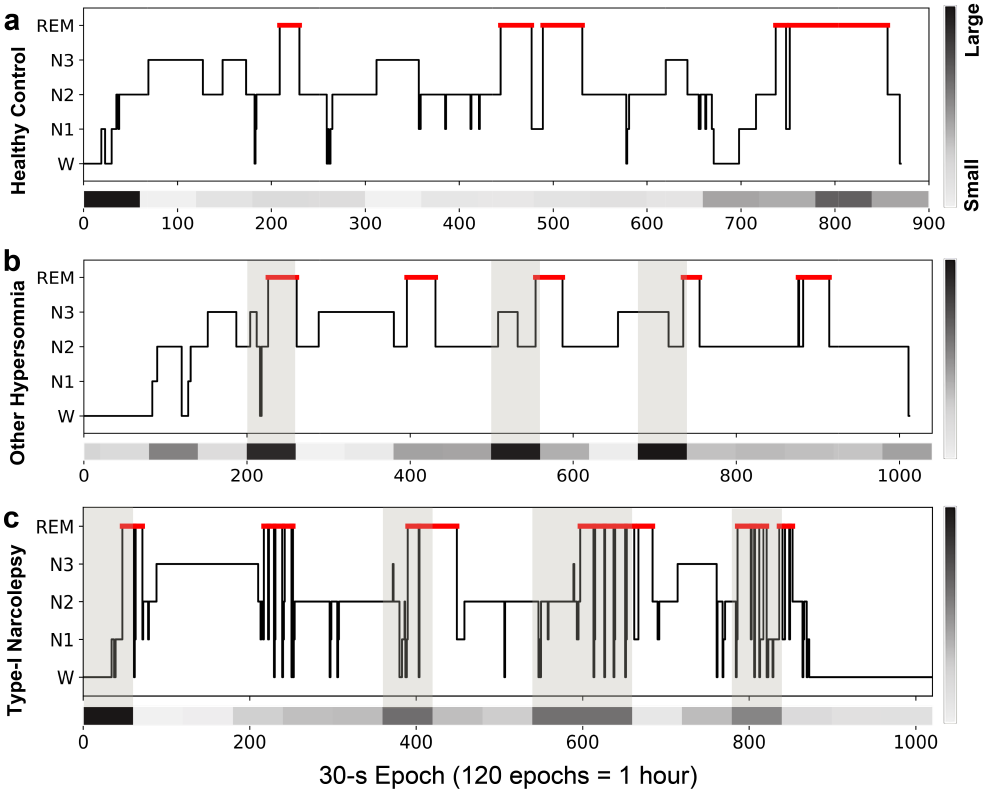


Figure 7: MNC hypnograms and attention weights for a control, an other-hypersomnia case, and a T1N case. The horizontal axis counts 30-second epochs. The selected T1N example includes short wake-to-REM latency and repeated wake/REM transitions.

G DATA ACCESS

SHHS and MNC are available through the National Sleep Research Resource under the providers' access conditions: https://www.sleepdata.org. SleepEDF, Physio2018, and CAP are distributed by PhysioNet:

- SleepEDF: https://physionet.org/content/sleep-edfx/1.0.0/.
- Physio2018: https://physionet.org/content/challenge-2018/1.0.0/.
- CAP: https://physionet.org/content/capslpdb/1.0.0/.

MASS is available at http://ceams-carsm.ca/mass/; BOAS at https://openneuro.org/datasets/ds005555/versions/1.0.0; and ISRUC at https://sleeptight.isr.uc.pt/. Access and reuse remain subject to each dataset's terms.