REPLACING EXPERTS WITHOUT RETRAINING: CONTRACT-AWARE COMPOSITION OF SLEEP MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Many real-world AI applications require multiple models for different tasks within a larger system. The growing availability of pretrained and task-specific models makes it possible to assemble such systems from existing experts. While this reduces the amount of training during initial deployment, these systems should also evolve as stronger models, new architectures, and revised preprocessing become available. Ideally, replacing one expert in the system should not require retraining every other component, yet independently developed models may differ in their input and output contracts. We study this problem in medical sleep analysis, where related time-series tasks use data from the same patient but require different modalities and temporal resolutions. We propose an explicit model contract that specifies the required inputs and temporal meaning of a model's predictions. We use these contracts to execute models over a common prediction envelope and to properly align class probabilities. We study four interacting polysomnography tasks with experts from three different model families. In our experiments, we score a total of 62.8 million windows from 1,820 recordings. Across all three families and four tasks, our contract-based alignment improves the F1 score by up to 0.2 points compared to naive alignment. More importantly, we show that zero-shot exchange of a strong expert from one family into another one can improve the performance of the system by up to 0.17 F1 score without impacting other tasks. To our knowledge, this is the first complete-night study of zero-shot replacement among heterogeneous, interacting sleep experts and the first systematic study towards model replacement in ML systems without fine-tuning or retraining.

1 INTRODUCTION

Modern machine learning is increasingly moving from isolated prediction models to deployed systems in which several learned components interact. Autonomous robots combine perception, planning, and control modules Urmson et al. (2009); Agha et al. (2022); agentic systems coordinate language models, retrieval, and tools Wei et al. (2022); Schick et al. (2023); and medical analysis often depends on multiple models using different sensor modalities Nasiri et al. (2025). The growing availability of pretrained and task-specific models makes it possible to assemble these systems from existing experts rather than training one monolithic network. While this is attractive at initial deployment, a updatable system must also remain updatable when a stronger expert, a revised input pipeline, or an additional task becomes available. Maybe even more frustrating, even small component changes not necessarily related to the ML components can already cause system-level regressions (Srivastava et al., 2020; Xie et al., 2021). Updating and exchanging models is easy when components are truly independent. In a composed system, a downstream model may rely on an upstream prediction, while a learned fusion layer may co-adapt to the exact distribution emitted by a particular expert. Even before learning becomes relevant, two prediction tensors are not necessarily compatible: their entries may refer to different physical times, inputs, class orders, or preprocessings. A modular system must state what signals a model consumes, how they are transformed, which physical time each prediction describes, and how the model can be executed to cover a requested time span.

We study sleep analysis as one concrete case where these requirements naturally occur. The gold standard in sleep analysis is a polysomnography (PSG). A PSG records brain activity, eye and muscle activity, cardiac activity, respiration, and blood oxygen saturation over one night. In order to analyze PSG data, several, partially dependent tasks are solved: Sleep staging partitions the night into distinct

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.

sleep categories, including wakefulness and REM sleep. This provides context for tagging arousals (i.e., micro awakenings during the night). Respiratory events like apneas or hypopneas (which are complete or partial stops of breathing, respectively) are further classified and may be accompanied by oxygen desaturation or arousals (Berry et al., 2012). Models for these tasks are often developed as separate experts because they use different signal groups, temporal windows, output resolutions, and architectures. Foundational and joint models exist (Thapa et al., 2024; 2026; Shuai et al., 2026; Huang et al., 2026), but changing one task or input in a monolithic model generally changes the complete system. Conversely, running specialized experts independently preserves modularity but cannot exploit cross-task information.

We study the middle ground: a single system assembled from independently trained experts, where every expert retains its original input and output. We introduce an executable model contract that makes the information required for the composition of experts explicit: the input signals and preprocessing expected by an expert, its native temporal context, its output classes and resolution, as well as the physical interval described by each prediction. These contracts allow heterogeneous experts to be executed over a common prediction window and their resulting probability time series to be aligned without changing their native inputs or outputs. Finally, small task-specific heads combine the aligned probabilities with each expert's original prediction, leading to a system with exchangeable experts that can leverage cross-task information.

Our study uses the Human Sleep Project (HSP) (Sun et al., 2023) at a substantial scale. The training split alone contains 19,595 recordings from 14,200 patients. We train and then lock experts for sleep staging, arousal detection, respiratory-event classification, and oxygen desaturation from three families: task-specific models trained with our proposed Sleepwalker tooling, and linear task heads for the SleepFM (Thapa et al., 2024) and OSF (Shuai et al., 2026) foundation models. We compare independent and composed experts within each family, followed by zero-shot replacement of the SleepFM and OSF sleep and breathing experts with their stronger Sleepwalker counterparts. Our largest common-cohort evaluation spans 1,820 recordings from 1,418 patients, approximately 13,000 hours of scored sleep-data, and 62.8 million task-specific prediction intervals per system. Our main result is a system with built-in model replacement support rather than a large gain from composition alone. Contract-aware composition preserves the performance of already strong independent experts and modestly improves some dependent tasks while being significantly better than typical tensor-based alignment. More importantly, a deployed composition can adopt a stronger expert without retraining the other experts. Replacing the SleepFM sleep expert with the Sleepwalker expert raises complete-night sleep macro-F1 from 0.610 to 0.782 zero-shot, and replacing its breathing expert raises breathing macro-F1 from 0.493 to 0.584 and desaturation macro-F1 from 0.579 to 0.615. The corresponding OSF-host exchanges raise sleep macro-F1 from 0.703 to 0.785 and breathing macro-F1 from 0.457 to 0.699. Refitting only the small composition heads raises the exchanged breathing scores further to 0.710 for SleepFM and 0.712 for OSF. Our contributions are:

- a contract-aware system that composes and exchanges heterogeneous time-series experts while preserving their native inputs and outputs
- a controlled evaluation across three expert families and four tasks in which tensor-based alignment underperforms contract-aware execution in all 12 family–task comparisons
- a complete-night study covering up to 62.8 million task-specific intervals per system, where zero-shot breathing-expert exchange raises macro-F1 from 0.493 to 0.584 in SleepFM and from 0.457 to 0.699 in OSF.

2 Related Work

Multi-task learning and model composition. Research on multi-task learning has largely followed two branches. The first makes task interaction part of training. Its introduction in (Caruana, 1997) established the benefit of learning a shared representation for multiple tasks. Cross-stitch networks have been proposed as an extension that learns where task-specific branches should exchange features (Misra et al., 2016) and mixtures of experts are used to learn how computation should be routed across specialized modules (Jacobs et al., 1991; Shazeer et al., 2017). More recent modular approaches make trained components reusable through adapter fusion or local module composition (Pfeiffer et al., 2021; Ostapenko et al., 2021) and progressively weaken the requirement of a monolithic architecture. However, the underlying assumption is still that components use either a shared representation

or a shared training context. The second branch composes models after they have been trained. Stacked generalization showed that existing predictions can become the input to another predictor (Wolpert, 1992). Later work moved beyond final outputs by learning common representations from black-box experts or amalgamating heterogeneous teachers into a new student (Hoang et al., 2020; Luo et al., 2019). Model stitching goes further inside the models and learns a mapping from one network's intermediate representation to another's (Bansal et al., 2021; Ding et al., 2026). This progression trades portability for information. Output probabilities form a small and universal interface between experts, while internal representations expose more information but couple the composition to each model. Learned communication bottlenecks and compositional interfaces offer intermediate alternatives (Shao et al., 2022; Luketina et al., 2024). What remains underexplored is how this coupling affects the practical replacement of experts that solve different but dependent tasks and cannot assume a shared backbone, input contract, representation size, or output timeline. We use probability stacking as a deliberately simple communication mechanism and instead ask what temporal and preprocessing contract is required for a replacement to remain meaningful.

Sleep analysis. Sleep analysis provides a concrete modular setting. Sleep staging and event detection have motivated specialized architectures such as SleepTransformer and U-Time (Phan et al., 2022; Perslev et al., 2019), while sleep foundation models such as SleepFM, SleepGPT, and OSF learn representations that transfer across PSG tasks (Thapa et al., 2024; 2026; Huang et al., 2026; Shuai et al., 2026). These directions improve an expert or shared backbone but do not specify how independently released task models should be versioned, composed, and replaced when each retains its own signals, preprocessing, and timeline. Clinical scoring makes this omission observable: arousal and respiratory endpoints are interpreted in the context of predicted sleep, and desaturation is interpreted alongside respiratory events. We use these dependencies as a case study in maintaining a heterogeneous expert system.

Model contracts and exchange. Existing work addresses complementary parts of reusable model artifacts. Model Cards and FactSheets document intended use, evaluation, limitations, and provenance for human consumers (Mitchell et al., 2019; Arnold et al., 2019). Production pipeline and data-validation systems instead make transformations, schemas, and training as well as serving assumptions reproducible or testable (Baylor et al., 2017; Breck et al., 2019). Exchange formats such as ONNX standardize computation graphs and tensor-level input and output signatures (ONNX Contributors, 2026). Yet updating a model can introduce incompatible predictions in downstream pipelines even when aggregate accuracy improves, including without distribution shift (Srivastava et al., 2020; Xie et al., 2021). These layers do not jointly specify the domain semantics needed to compose scientific time-series experts: sensor identity and units, sampling and preprocessing, temporal context, and the physical interval represented by each output. We use *model contracts* that attach this meta information to the model so that we can replace a single model inside a system without negatively impacting the other models.

3 METHOD

We consider systems of related time-series classification tasks and use sleep analysis as our running example. As already mentioned, a polysomnography (PSG) records synchronized physiological signals, including EEG, EOG, EMG, ECG, respiration, and blood oxygen saturation over a complete night (see Figure 1 as an example). A scorer assigns sleep stages and annotates arousals, apneas, hypopneas, and oxygen desaturations that are later reviewed by a doctor to form a diagnosis and potentially select an intervention. While these targets concern the same underlying physiological process, they are defined on different signals and time scales. Sleep stages are scored in 30-second epochs and models commonly use several minutes of surrounding EEG context, whereas arousals and respiratory events are localized at substantially finer resolutions typically ranging from 0.5-1s(American Academy of Sleep Medicine, 2023; Berry et al., 2012; Phan et al., 2022; Perslev et al., 2019). Second, their clinical definitions couple these tasks: for example, an arousal requires preceding sleep, while a hypopnea is scored in association with oxygen desaturation or arousal.

Our goal is to build an updateable system that automatically annotates a full-night PSG. In particular, we want to easily update parts of the system without retraining or influencing other parts of it while still leveraging cross-task information. More formally, we consider a system of expert models that address a set of tasks V and our goal is to replace experts $f_v \colon \mathbb{R}^{L_v \times d_v} \to \mathbb{R}^{T_v \times C_v}$ with $v \in V$

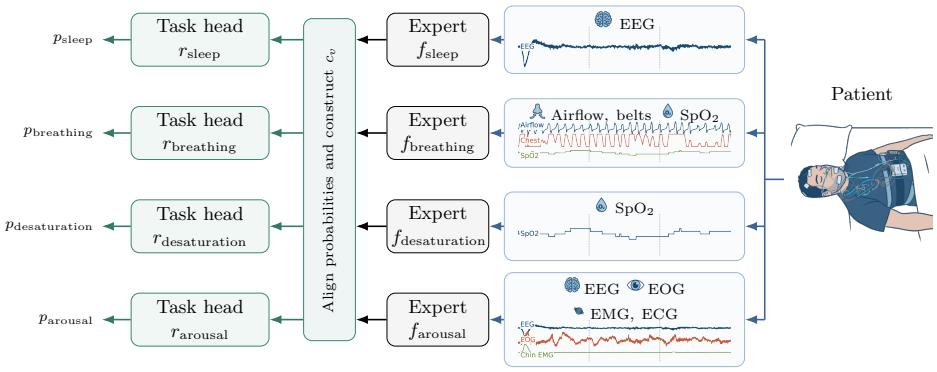

Figure 1: A PSG analysis with four expert models (sleep, breathing, desaturation, and arousal). Each expert retains its native physiological inputs and produces logits and class probabilities. The probabilities are aligned by timestamp and concatenated separately for four task-specific heads.

without retraining or fine-tuning of the entire system. Each expert operates on d_v physiological signals sampled at a frequency ν_v over a duration D_v. After the model-specific input preparation (e.g. normalization, filtering etc.), this gives $x_v \in \mathbb{R}^{L_v \times d_v}$ with $L_v = \nu_v D_v$. For each of T_v time points, the expert predicts a probability vector for C_v classes via

$$q_v = \text{softmax}(z_v) = \text{softmax}(f_v(x_v)) \tag{1}$$

Experts may differ in any of these quantities. Some differences follow directly from the target definition, while others reflect modeling and training choices: an expert may use different signals, sample them at a different frequency, observe a longer context, or produce more predictions at a finer resolution. Equation 1 is the typical model description once x_v has been prepared and the entries of q_v have an agreed interpretation (e.g. ordering of classes). This description is, however, insufficient to properly apply models. As a naive example, consider the setting where we have been given three tensors q_1, q_2, q_3 of unmatching length that span different physical times from three different experts. Simply stacking those tensors discards any temporal dependencies between tensor entries. While a subsequent system might be able to learn that the input is unaligned, exchanging one expert with a potentially different prediction resolution requires retraining the stacking model. To allow exchangeability in this case, one natural heuristic is to use nearest-neighbor interpolation to upsample short tensors and non-overlapping average pooling to downsample tensors when necessary. Unfortunately, this also results in misaligned data, as shown in the example in Figure 2 and leads to worse performance, as shown in our experiments.

3.1 Model Contracts

We associate each expert with an *executable model contract* $K_v = (K_v^{\text{in}}, K_v^{\text{out}})$. The input contract defines how the physiological recording is transformed into x_v and the output contract defines how q_v is interpreted. While the input contract essentially describes how to prepare the data for the model (e.g. re-sampling, normalization etc.), the output contract describes how to interpret the predictions. A model's prediction does not need to overlap with its input, and often there is a significant temporal difference between both. For example, a sleep staging model that consumes 630s of input might use these to score a 30s interval in the center of the input, i.e. $(630s - 30s)/2 = 300s$ after its relative start. Hence, the output contract not only describes the classes and their ordering, but also the temporal span Δ_v , its offset o_v and resolution of each expert's prediction T_v. This way, the contract turns each output tensor into a timestamped probability series, i.e. every row of q_v is paired with the physical time it describes. Since each expert operates on potentially different time spans, the composed model may request different output spans from each expert to properly align them. We call the smallest interval containing all of them the *common prediction envelope*. Each expert is invoked as many times as necessary to cover this envelope with its native predictions. Thus, an expert that emits one 30-second prediction is invoked three times to cover an 80-second envelope, whereas an expert that emits ten consecutive 30-second predictions covers it in one invocation. Every invocation still uses the expert's own input duration, signals, sampling frequency, and preprocessing. The

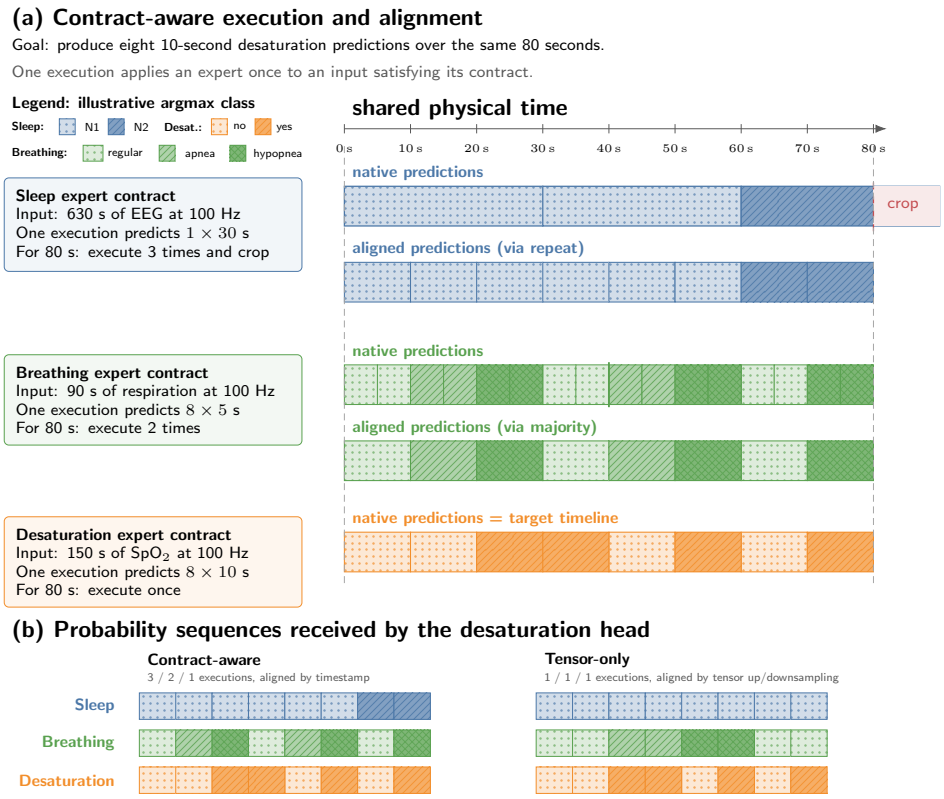

Figure 2: Contract-aware temporal alignment for an 80-second desaturation prediction. The contracts determine that sleep, breathing, and desaturation cover 30, 40, and 80 seconds per execution and therefore require three, two, and one executions, respectively. Sleep predictions are repeated and adjacent breathing predictions are averaged on the target's ten-second intervals. Tensor-only alignment instead produces matching shapes from one execution per expert, although their entries describe different physical times. Shading denotes the argmax class of illustrative probability vectors.

common envelope, therefore, coordinates execution without replacing the individual input contracts. Once all native predictions cover the common envelope, they are converted separately to the output timeline of each task. Coarser probabilities are repeated by nearest-neighbor interpolation, whereas finer probabilities are averaged within the requested intervals. Figure 2 presents an example of our approach and compares it to naive up-/downsampling of tensors. It uses three contracts from our study to make it concrete. The sleep expert maps 630 seconds of EEG to one central 30-second prediction, the breathing expert maps 90 seconds of respiratory signals to eight five-second predictions, and the desaturation expert maps 150 seconds of SpO_2 to eight ten-second predictions. All operate at 100 Hz, yet their input signals, temporal context, prediction offsets, output spans, and output rates differ. The naive tensor alignment does not correctly align tensors across physical time, whereas our contract does.

3.2 Exchangeable probability composition

The composed system of models consists of the native experts $\{f_v\}_{v \in V}$ and one task-specific correction head r_v for every task (see also Figure 1). All experts first produce their native logits z_v and probabilities q_v over the common envelope. For task v, we align the probabilities of every other expert to its output timeline and concatenate them as $c_v = \mathrm{concat}_{u \in V \setminus \{v\}} q_{v|u}$, where $q_{v|u}$ denotes the predictions of expert u on task v's timeline. The final prediction is

$$p_v = \mathrm{softmax}(z_v + r_v(c_v)) \,. \tag{2}$$

All expert predictions are computed before the correction heads, so composition is simultaneous. The native logits remain the reference prediction and r_v learns only a correction based on the other

tasks. We instantiate each r_v as a linear map and initialize it near zero, such that the initial system remains close to independent expert execution. Using class probabilities provides a deliberately restricted interface. Their dimensions and ordered semantics are fixed by the task contract rather than by the expert architecture. Latent representations may retain more information, but their dimension, temporal organization, and coordinate system generally change with the architecture, and hence, exchanging such an expert consequently requires a new adapter and retraining. We use separate correction heads because the tasks have different class sets and output timelines, and because their parameters should remain independently adaptable. A single joint output head would couple every task to the complete expert set: adapting it after one replacement would also change the outputs of otherwise unaffected tasks. With task-specific heads, each native expert remains independently executable, and the heads that consume a changed probability interface can be retained for zero-shot exchange or adapted without retraining the other experts. For training we use the mean task loss: Given N examples, let $x_{i,v}$ and $y_{i,v}$ denote the native input and label for the task v of example i. We compute $z_{i,v} = f_v(x_{i,v})$, obtain $c_{i,v}$ from the aligned probabilities of the other experts, and then minimize

$$\mathcal{L} = \frac{1}{N|V|} \sum_{i=1}^{N} \sum_{v \in V} \mathcal{L}_v(z_{i,v} + r_v(c_{i,v}), y_{i,v}) \,, \tag{3}$$

where $\mathcal{L}_v$ is the class-weighted cross-entropy for task v. We compare three variants of this system. *Independent* executes each expert under its native contract without coupling and without correction heads. This is our baseline. *Frozen* keeps all experts fixed and optimizes only the correction heads. *Joint* optimizes both the correction heads and the experts through the native logits and probability interface. Joint optimization permits co-adaptation, but makes the system more dependent on the particular experts used during training.

4 Experimental Protocol

Our experiments ask three questions. First, can cross-task correlations be exploited in a modular system where models can be swapped? Second, is naive tensor-only alignment sufficient, and what does the complete prediction contract add? Third, can stronger experts be inserted in a zero-shot fashion, without any adaptation?

4.1 Tasks, data, and experts

We study four sleep-related tasks: five-class sleep staging (wake, N1, N2, N3, REM), binary arousal detection (yes, no), three-class respiratory-event classification (apnea, hypopnea, regular breathing), and binary oxygen-desaturation detection (yes, no) using the HSP 2.0 (Sun et al., 2023) dataset. We split the data into three parts: training, fine-tuning, and test, and make sure that patients only occur in one of these sets. After excluding unusable recordings, the data split contains 19,595 training recordings from 14,200 patients, 2,421 fine-tuning recordings from 1,775 patients, and 2,410 test recordings from 1,775 patients. Individual experts are trained only on the training partition and, in case of joint optimization, fine-tuned on the fine-tuning set. Composition and replacement adaptation use only the fine-tuning partition. The test partition is used for testing only. We performed manual hyperparameter tuning during development based on the training and fine-tuning losses. We use three families of expert models: We turn two foundational sleep models, SleepFM (Thapa et al., 2024) and OSF (Shuai et al., 2026) into individual experts by using their respective embeddings to train individual classifier heads for each task. Moreover, we train our own expert models from scratch using state-of-the-art models: for the sleep expert, we use SleepTransformer (Phan et al., 2022) using EEG inputs. For arousal, breathing, and desaturation experts, we use a U-Time inspired model (Perslev et al., 2019) with task-specific EEG/EOG/EMG/ECG, respiratory, and SpO_2 inputs. We denote these models as *Sleepwalker* experts due to our internal Sleepwalker software used for training. Note that these systems differ substantially in their input modalities and temporal contracts: Sleepwalker uses task-specific windows and sampling, OSF processes 30-second segments, and SleepFM processes five minutes of multimodal PSG at 128 Hz. Similarly, the output contracts differ significantly, ranging from 30s sleep staging to 1s for arousal tagging. Details about the models and their contracts are given in the appendix. Each training epoch samples 250,000 native input windows from a pool of 512 patients per epoch. The Sleepwalker experts are trained for 100 epochs and therefore have seen 25 million sampled windows per expert, whereas the foundation-models' classification heads are

trained for 10 epochs using a total of 2.5 million sampled windows per run in addition to their original training data. Because the experts usually return sequences, one input window contributes multiple target intervals. After target masking, a 250,000-window training epoch contains 250,000 sleep-stage labels, around 9.60 million arousal labels, 1.92 million breathing labels, and 1.90 million desaturation labels for a representative Sleepwalker expert. Note that these numbers differ slightly for each expert and task due to the variations in input/output contracts. The corresponding class distributions are 17.7% wake, 12.4% N1, 45.3% N2, 12.3% N3, and 12.3% REM for sleep staging; 0.9% arousal; 3.4% apnea, 4.1% hypopnea, and 92.5% regular breathing; and 5.5% desaturation. Training and evaluation run in PyTorch with each run assigned one NVIDIA A30 GPU with 24 GB of memory on a system with an AMD EPYC 9124 processor. The 60 completed composition and evaluation runs reported here with retained timing records account for approximately 476 A30 GPU-hours, giving a TDP-based upper-bound estimate of approximately 79 kWh for these jobs at the A30's 165 W rated maximum power. Somewhat counterintuitively, evaluation took much longer in our experiments than training. As mentioned, during training a set of windows is sampled, whereas evaluation scores every available prediction interval over every retained night. For example, one contract-aligned evaluation comprises 62.8 million task-specific intervals for Sleepwalker, 56.4 million for SleepFM, and 63.1 million for OSF, including 47.1, 42.3, and 47.3 million arousal intervals, respectively. More details can be found in the appendix.

For evaluation, we are interested in the performance of the entire system and not necessarily of individual experts within that system. Hence, we extract all windows from all test patients, execute the entire system on them, and then perform a small rule-based correction that matches the workflow inside a sleep laboratory. In particular, this correction accounts for errors in sleep staging according to the AASM (American Academy of Sleep Medicine, 2023), where arousals, apneas, and desaturations are only clinically relevant if a patient is asleep. Hence, whenever a sleep-staging model predicts that a patient is not asleep, the other experts' predictions are forced to their default class (i.e., no apnea/hypopnea, no arousal, no desaturation). Our primary metric is macro-F1 due to the imbalanced nature of most of these tasks. Additional metrics such as Cohen's Kappa are reported in the appendix. To account for variations in the test cohort, we additionally resample distinct test patients 2,000 times with replacement, keeping all recordings from each selected patient together. Then, for each resample, we pool its recording-level confusion matrices before computing macro-F1 to estimate uncertainty in the following tables. The appendix reports 95% confidence intervals. We use the following six systems: *Independent* uses each expert under its native contract. *Naive Alignment* invokes every expert once from a shared origin, resizes its output tensor, and uses no learned correction. *Naive + head fine-tuning* does the same, but additionally fits the four correction heads for 10 epochs. *Contract Alignment* likewise has no correction head but executes each expert on its native prediction timeline. *Contract + head fine-tuning* freezes the experts and again fits the four correction heads; *Contract + joint fine-tuning* also updates the permitted expert parameters. Comparisons use a common cohort within, but not across expert families. Due to the differences in input/output contracts of the models, not all windows of a patient can be scored (e.g., if a model requests 630s as context and scores 30s intervals at the center of this context, then it cannot score the first $(630 - 30)/2 = 300 = 10 \cdot 30s$ windows). Hence, for each family, we drop a recording whenever we cannot cover at least 95% of its expected windows with all models in that family. While this reduced the evaluation cohort substantially in some cases (e.g., for SleepFM we evaluate $1,626$ recordings compared to $1,820$ for Sleepwalker), we find that the overall number of windows is still large enough to justify an in-family and a modest across-family comparison. For example, Sleepwalker and SleepFM are evaluated over 62.8 and 56.4 million scored windows per contract-aligned system, respectively. Hence, cross-family values should not be interpreted as strictly paired measurements, but due to the large number of scored windows, we feel confident that some cross-family comparisons are justified.

4.2 Results

Table 1 shows the results of independent experts compared to contract-based and naive alignment. We see three patterns: First, the task-specific Sleepwalker family has the strongest models across all four tasks, whereas the SleepFM and OSF experts lag behind their published paper results. This is consistent with Sleepwalker's task-specific architectures and inputs but also highlights why an executable contract matters when using released models. For example, SleepFM currently (September 2026) does not provide a clear model release: its bundled base checkpoint uses 300-second inputs and 640-sample patches (five seconds at 128 Hz), whereas the released sleep-staging configuration refers

Table 1: Complete-night macro-F1 on each family's common test cohort. Uncertainty (given as ± one bootstrap standard deviation) is computed via patient resampling. The best in-table point estimate for each family and task is bold.

Expert family	System	Sleep	Arousal	Breathing	Desaturation
Sleepwalker	Independent	**.788**±.003	.630±.003	.707±.004	.652±.004
	Naive Alignment	.581±.002	.504±.001	.631±.003	.582±.003
	Naive + head fine-tuning	.588±.002	.506±.001	.642±.003	.594±.003
	Contract Alignment	.783±.003	.635±.004	.711±.004	.650±.004
	Contract + head fine-tuning	.786±.003	**.640**±.004	**.714**±.004	.672±.004
	Contract + joint fine-tuning	.779±.003	.626±.004	.713±.004	**.678**±.004
SleepFM	Independent	.610±.003	.562±.002	.478±.003	.562±.004
	Naive Alignment	.572±.003	.508±.001	.397±.002	.544±.004
	Naive + head fine-tuning	.575±.003	.506±.001	.398±.002	.562±.004
	Contract Alignment	.605±.004	.566±.002	.490±.003	.560±.004
	Contract + head fine-tuning	.610±.004	.563±.002	.493±.003	.579±.004
	Contract + joint fine-tuning	**.680**±.003	**.617**±.003	**.575**±.004	**.624**±.004
OSF	Independent	.694±.003	.504±.002	**.481**±.004	.529±.004
	Naive Alignment	.669±.003	.500±.001	.424±.004	.517±.003
	Naive + head fine-tuning	.677±.003	.509±.001	.420±.004	.549±.005
	Contract Alignment	.694±.003	.509±.001	.461±.004	.527±.004
	Contract + head fine-tuning	.703±.003	**.536**±.002	.457±.004	**.561**±.005
	Contract + joint fine-tuning	**.729**±.003	.505±.001	.450±.004	.543±.004

to a different checkpoint with 3,840-sample patches[1]. We tried to reproduce the configuration from the paper as best as we could: 13 HSP channels are assigned to the four published modality groups, the checkpoint returns 60 five-second tokens per modality, and contiguous tokens are averaged to the output interval of each task before fitting a linear head. The SleepFM values should thus be read as results for this explicit construction rather than as a reproduction of its published downstream experiments. A clear contract-based release would have improved reusability here. Second, contract-based alignment outperforms naive alignment in the direct comparisons in Table 1: Sleepwalker sleep macro-F1 falls from 0.783 to 0.581, SleepFM breathing from 0.490 to 0.397, and OSF breathing from 0.461 to 0.424. With experts frozen, head fine-tuning modestly improves Sleepwalker's arousal, breathing, and desaturation scores over vanilla contract alignment. The effect is mixed for OSF: relative to independent, arousal rises from 0.504 to 0.536 and desaturation from 0.529 to 0.561, but breathing falls from 0.481 to 0.457. Joint fine-tuning improves all four SleepFM tasks, including breathing, from 0.493 to 0.575 relative to head fine-tuning. For OSF it improves sleep from 0.703 to 0.729 but lowers all three event scores. The contract-based advantage also remains when both alignments have fitted heads, although a rate-only baseline that executes experts under their native (input-)contracts but aligns predictions by tensor position performs comparably to full contract alignment in most settings. We hypothesize that correct and aligned input processing leads to the largest performance gains, although our contract alignment makes both, input and output contracts, explicit. See the appendix for additional ablations. Third, compared with Independent, contract alignment with head fine-tuning improves three of four tasks in each family. The exceptions are Sleepwalker sleep (0.786 versus 0.788), SleepFM sleep (0.610 versus 0.610), and OSF breathing (0.457 versus 0.481). For SleepFM, arousal rises by 0.001 and breathing by 0.015.

We next test whether a stronger Sleepwalker expert can replace its SleepFM or OSF counterpart without retraining the other experts. Table 2 shows the results for this experiment. Again, we compare contract alignment with naive alignment for zero-shot replacement and report head fine-tuning for both. The left table exchanges the sleep expert, whereas on the right we exchange the breathing expert. Contract alignment improves the exchanged task without severely impacting the other tasks, whereas naive alignment gives mixed results. For the sleep exchange, zero-shot performance is nearly saturated: sleep macro-F1 rises from 0.610 to 0.782 in SleepFM and from 0.703 to 0.785 in OSF, while adapting the correction heads adds only 0.001 in each system. The other task scores change by at most 0.008. Relative to contract alignment before replacement, naive zero-shot exchange lowers sleep macro-F1 by 0.032 and 0.117, respectively. For breathing, the zero-shot

[1]We used `sleepfm/checkpoints/model_base/best.pt` at revision `2bcbae0`.

Table 2: Complete-night macro-F1 after sleep-expert exchange (left) and breathing-expert exchange (right). ± denotes one patient-bootstrap standard deviation for the exchanged task. Every Δ is relative to the frozen contract host (first row); separately rounded values may differ in the last digit.

System	Sleep	Arousal	Breath.	Desat.	Δ Sleep
SleepFM	.610±.004	0.563	0.493	0.579	–
Contract zero-shot	.782±.003	0.565	0.501	0.581	+0.172
Contract fine-tuning	**.783**±.003	0.565	0.502	0.580	**+0.173**
Naive zero-shot	.578±.002	0.505	0.399	0.561	-0.032
Naive fine-tuning	.582±.003	0.505	0.399	0.561	-0.027
OSF	.703±.003	0.536	0.457	0.561	–
Contract zero-shot	.785±.003	0.538	0.456	0.562	+0.081
Contract fine-tuning	**.786**±.003	0.537	0.460	0.562	**+0.082**
Naive zero-shot	.587±.002	0.509	0.414	0.545	-0.117
Naive fine-tuning	.590±.002	0.510	0.420	0.549	-0.113

System	Sleep	Arousal	Breath.	Desat.	Δ Breath.
SleepFM	0.610	0.563	.493±.003	0.579	–
Contract zero-shot	0.610	0.563	.584±.004	0.615	+0.091
Contract fine-tuning	0.615	0.563	**.710**±.004	0.611	**+0.216**
Naive zero-shot	0.575	0.506	.422±.002	0.594	-0.071
Naive fine-tuning	0.583	0.506	.643±.004	0.588	+0.150
OSF	0.703	0.536	.457±.004	0.561	–
Contract zero-shot	0.702	0.534	.699±.004	0.590	+0.243
Contract fine-tuning	0.704	0.541	**.712**±.004	0.619	**+0.255**
Naive zero-shot	0.674	0.509	.625±.003	0.570	+0.169
Naive fine-tuning	0.680	0.512	.644±.003	0.601	+0.188

exchange raises macro-F1 from 0.493 to 0.584 in SleepFM and from 0.457 to 0.699 in OSF. The head adaptation raises these to 0.710 and 0.712. The zero-shot swap also improves desaturation from 0.579 to 0.615 and from 0.561 to 0.590, respectively, consistent with respiratory evidence benefiting a dependent task. Naive alignment shows a more mixed picture. Relative to contract alignment before replacement, SleepFM breathing falls by 0.071, whereas OSF breathing rises by 0.169. Both scores remain below the corresponding contract-aligned zero-shot exchanges. Expectantly, refitting the naive heads after exchange can help to raise the performance. For breathing, macro-F1 improves to 0.643/0.644 (SleepFM/OSF), but sleep macro-F1 only to 0.582/0.590, while all four remain below the corresponding contract-aligned fine-tuned scores.

5 Conclusion

Many real-world AI applications require multiple models for different tasks within a larger system. With the growing availability of pretrained and task-specific models, these systems are often not trained end-to-end anymore but assembled from available expert components. In this paper, we studied expert composition from a system-maintenance perspective in sleep analysis: independently developed models should cooperate while allowing model replacement. We proposed executable input and output contracts that let heterogeneous sleep-analysis experts retain their native signals and time scales while exchanging aligned class probabilities. Across three families, a naive tensor-based alignment underperforms, while contract-aware execution generally performs best. More importantly, we showed that stronger experts can be inserted zero-shot into systems including foundation-model hosts, substantially improving the replaced task without retraining the others or substantially hurting their performance. In some cases, this replacement improved the performance on related tasks as well.

AI use statement

Generative AI tools were used to assist with code generation, debugging, experimental design, literature discovery, and manuscript drafting. The authors verified all generated code, references, analyses, and claims.

Ethics statement

PSG is sensitive medical data. All experiments use data under their respective access conditions and report only aggregate results. Only anonymized data has been used in this research. The models are research artifacts and are not intended for autonomous diagnosis.

Reproducibility statement

We fully intend to publish our source code upon acceptance. We further added the current version of the code, including the fixed split, and the script used to generate every reported result, to the supplementary material for reviewers.

REFERENCES

Ali Agha et al. Nebula: Team costar's robotic autonomy solution that won phase ii of darpa subterranean challenge. *Field Robotics*, 2(1):1432–1506, 2022. URL https://doi.org/10.55417/fr.2022047.

American Academy of Sleep Medicine. *The AASM Manual for the Scoring of Sleep and Associated Events: Rules, Terminology and Technical Specifications*. American Academy of Sleep Medicine, Darien, IL, version 3 edition, 2023. URL https://aasm.org/clinical-resources/scoring-manual/.

Matthew Arnold et al. Factsheets: Increasing trust in ai services through supplier's declarations of conformity. *IBM J. Res. Dev.*, 63(4/5):6:1–6:13, 2019. doi: 10.1147/JRD.2019.2942288. URL https://doi.org/10.1147/JRD.2019.2942288.

Yamini Bansal, Preetum Nakkiran, and Boaz Barak. Revisiting model stitching to compare neural representations. In *NeurIPS*, volume 34, pp. 225–236, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/01ded4259d101feb739b06c399e9cd9c-Abstract.html.

Denis Baylor, Eric Breck, Heng-Tze Cheng, Noah Fiedel, Chuan Yu Foo, Zakaria Haque, Salem Haykal, Mustafa Ispir, Vihan Jain, Levent Koc, Chiu Yuen Koo, Lukasz Lew, Clemens Mewald, Akshay Naresh Modi, Neoklis Polyzotis, Sukriti Ramesh, Sudip Roy, Steven Euijong Whang, Martin Wicke, Jarek Wilkiewicz, Xin Zhang, and Martin Zinkevich. Tfx: A tensorflow-based production-scale machine learning platform. In *KDD*, pp. 1387–1395, 2017. doi: 10.1145/3097983.3098021. URL https://www.wikidata.org/entity/Q99800318.

Richard B. Berry, Rohit Budhiraja, Daniel J. Gottlieb, David Gozal, Conrad Iber, Vishesh K. Kapur, Carole L. Marcus, Reena Mehra, Sairam Parthasarathy, Stuart F. Quan, Susan Redline, Kingman P. Strohl, Sally L. Davidson Ward, and Michelle M. Tangredi. Rules for scoring respiratory events in sleep: Update of the 2007 AASM manual for the scoring of sleep and associated events. *Journal of Clinical Sleep Medicine*, 8(5):597–619, 2012. doi: 10.5664/jcsm.2172.

Eric Breck, Neoklis Polyzotis, Sudip Roy, Steven Whang, and Martin Zinkevich. Data validation for machine learning. In *SysML*, volume 1, pp. 334–347, 2019. URL https://proceedings.mlsys.org/paper_files/paper/2019/hash/928f1160e52192e3e0017fb63ab65391-Abstract.html.

Rich Caruana. Multitask learning. *Mach. Learn.*, 28(1):41–75, 1997. doi: 10.1023/A:1007379606734. URL https://www.wikidata.org/entity/Q56031197.

Yueli Ding, Linwei Hu, Fei Cheng, Quan Wang, Ziheng Zhu, and Pengfei Yang. Bridging the gap in neural network quantization with feature correlation coding. In *Proceedings of UniReps: the Third Edition of the Workshop on Unifying Representations in Neural Models*, volume 35 of *Proceedings of Machine Learning Research*, pp. 129–139. PMLR, 2026. URL https://doi.org/10.1142/s0218126625504092.

Trong Nghia Hoang, Thanh Lam, Bryan Kian Hsiang Low, and Patrick Jaillet. Learning task-agnostic embedding of multiple black-box experts for multi-task model fusion. In *ICML*, volume 119 of *Proceedings of Machine Learning Research*, pp. 4282–4292. PMLR, 2020. URL http://proceedings.mlr.press/v119/hoang20b.html.

Weixuan Huang, Yan Wang, Hanrong Cheng, Wei Xu, Tingyue Li, Xiuwen Wu, Hui Xu, Pan Liao, Zaixu Cui, Qihong Zou, and Jia-Hong Gao. A unified time-frequency foundation model for sleep decoding. *Nature Communications*, 17:1198, 2026. doi: 10.1038/s41467-025-67970-4.

Robert A. Jacobs, Michael I. Jordan, Steven J. Nowlan, and Geoffrey E. Hinton. Adaptive mixtures of local experts. *Neural Comput.*, 3(1):79–87, 1991. doi: 10.1162/neco.1991.3.1.79. URL https://www.wikidata.org/entity/Q56505849.

Jelena Luketina, Jack Lanchantin, Sainbayar Sukhbaatar, and Arthur Szlam. Compositional interfaces for compositional generalization. In *CoLLAs*, volume 274 of *Proceedings of Machine Learning Research*, pp. 692–709. PMLR, 2024. URL https://proceedings.mlr.press/v274/luketina25a.html.

Sihui Luo, Xinchao Wang, Gongfan Fang, Yao Hu, Dapeng Tao, and Mingli Song. Knowledge amalgamation from heterogeneous networks by common feature learning. In *IJCAI*, pp. 3087–3093, 2019. doi: 10.24963/ijcai.2019/428. URL https://doi.org/10.24963/ijcai.2019/428.

Ishan Misra, Abhinav Shrivastava, Abhinav Gupta, and Martial Hebert. Cross-stitch networks for multi-task learning. In *CVPR*, pp. 3994–4003, 2016. URL https://doi.ieeecomputersociety.org/10.1109/CVPR.2016.433.

Margaret Mitchell, Simone Wu, Andrew Zaldivar, Parker Barnes, Lucy Vasserman, Ben Hutchinson, Elena Spitzer, Inioluwa Deborah Raji, and Timnit Gebru. Model cards for model reporting. In *FAT*, pp. 220–229, 2019. doi: 10.1145/3287560.3287596. URL https://www.wikidata.org/entity/Q103724290.

Samaneh Nasiri et al. CAISR: achieving human-level performance in automated sleep analysis across all clinical sleep metrics. *Sleep*, 48(8), August 2025.

ONNX Contributors. ONNX intermediate representation specification, 2026. URL https://onnx.ai/onnx/repo-docs/IR.html.

Oleksiy Ostapenko, Pau Rodríguez, Massimo Caccia, and Laurent Charlin. Continual learning via local module composition. In *NeurIPS*, volume 34, pp. 30298–30312, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/fe5e7cb609bdbe6d62449d61849c38b0-Abstract.html.

Mathias Perslev, Michael Hejselbak Jensen, Sune Darkner, Poul Jørgen Jennum, and Christian Igel. U-time: A fully convolutional network for time series segmentation applied to sleep staging. *Advances in Neural Information Processing Systems*, 32:4417–4428, 2019. URL http://papers.nips.cc/paper/8692-u-time-a-fully-convolutional-network-for-time-series-segmentation-applied-to-

Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. Adapterfusion: Non-destructive task composition for transfer learning. In *EACL*, pp. 487–503. Association for Computational Linguistics, 2021. doi: 10.18653/v1/2021.eacl-main.39. URL https://aclanthology.org/2021.eacl-main.39/.

Huy Phan, Kaare B. Mikkelsen, Oliver Y. Chén, Philipp Koch, Alfred Mertins, and Maarten De Vos. Sleeptransformer: Automatic sleep staging with interpretability and uncertainty quantification. *IEEE Trans. Biomed. Eng.*, 69(8):2456–2467, 2022. doi: 10.1109/TBME.2022.3147187. URL https://doi.org/10.1109/TBME.2022.3147187.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *NeurIPS*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/d842425e4bf79ba039352da0f658a906-Abstract-Conference.html.

Jiawei Shao, Yuyi Mao, and Jun Zhang. Learning task-oriented communication for edge inference: An information bottleneck approach. *IEEE J. Sel. Areas Commun.*, 40(1):197–211, 2022. doi: 10.1109/JSAC.2021.3126077. URL https://doi.org/10.1109/JSAC.2021.3126087.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc V. Le, Geoffrey E. Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *ICLR (Poster)*, 2017. URL https://openreview.net/forum?id=B1ckMDqlg.

Zitao Shuai, Zongzhe Xu, David Yang, Wei Wang, and Yuzhe Yang. Towards a general-purpose foundation model for sleep, 2026.

Megha Srivastava, Besmira Nushi, Ece Kamar, Shital Shah, and Eric Horvitz. An empirical analysis of backward compatibility in machine learning systems. In *KDD*, pp. 3272–3280, 2020. doi: 10.1145/3394486.3403379. URL https://doi.org/10.1145/3394486.3403379.

Haoqi Sun et al. The human sleep project (version 2.0). Brain Data Science Platform, 2023.

Rahul Thapa, Bryan He, Magnus Ruud Kjær, Hyatt E. Moore IV, Gauri Ganjoo, Emmanuel Mignot, and James Zou. Sleepfm: Multi-modal representation learning for sleep across brain activity, ecg and respiratory signals., 2024. URL https://openreview.net/forum?id=QXqXGDapkQ.

Rahul Thapa, Magnus Ruud Kjær, Bryan He, Ian Covert, Hyatt Moore, Umaer Hanif, Gauri Ganjoo, M. Brandon Westover, Poul Jennum, Andreas Brink-Kjær, Emmanuel Mignot, James Zou, et al. A multimodal sleep foundation model for disease prediction. *Nature Medicine*, 32:752–762, 2026. doi: 10.1038/s41591-025-04133-4.

Chris Urmson et al. Autonomous driving in urban environments: Boss and the urban challenge. *J. Field Robotics*, 25(8):1–59, 2009. doi: 10.1002/ROB.20255. URL https://doi.org/10.1007/978-3-642-03991-1_1.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh (eds.), *NeurIPS*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/9d5609613524ecf4f15af0f7b31abca4-Abstract-Conference.html.

David H. Wolpert. Stacked generalization. *Neural Networks*, 5(2):241–259, 1992. doi: 10.1016/S0893-6080(05)80023-1. URL https://www.wikidata.org/entity/Q56058040.

Yuqing Xie, Yi-An Lai, Yuanjun Xiong, Yi Zhang, and Stefano Soatto. Regression bugs are in your model! measuring, reducing and analyzing regressions in nlp model updates. In *ACL/IJCNLP (1)*, pp. 6589–6602, 2021. doi: 10.18653/v1/2021.acl-long.515. URL https://aclanthology.org/2021.acl-long.515.

A Experimental Details

A.1 Cohort, targets, and class frequencies

The fixed HSP partition contains 19,595 training recordings from 14,200 distinct patients, 2,421 fine-tuning recordings from 1,775 patients, and 2,410 test recordings from 1,775 patients. Patients do not cross partitions, but a patient may contribute more than one recording. The four tasks have different scoring units: one sleep stage per 30 seconds, arousal at one-second resolution in the Sleepwalker and SleepFM experts, breathing at five-second resolution, and desaturation at ten-second resolution. OSF instead predicts one label per 30-second interval for every task. All families use the same five sleep classes, two arousal classes, three breathing classes, and two desaturation classes. Class assignment follows a "50% rule": For an event interval I_t, the training target assigns each annotated class the fraction of I_t occupied by that class, whereas the unoccupied fraction belongs to the configured negative class. For example, an apnea covering two seconds of a five-second breathing interval gives target overlaps of 0.4 for apnea and 0.6 for regular breathing, provided no other event overlaps it. The task loss uses an interval only when scored sleep occupies at least half of that interval. This mask is computed from reference annotations during training. The same event-boundary and mask rules are used when fitting individual experts and composition heads.

Table 3 shows the distribution of the classes. Counts were estimated from 250,000 sampled input windows for a representative expert of each task. An event input window contributes several output intervals, and event targets at boundaries are fractional. Consequently, the displayed counts are rounded sums of target mass, not counts of unique annotated events. The four denominators differ by task and by the number of valid output intervals retained after masking. These frequencies determine the fixed inverse-log class weights used for all comparable training runs.

A.2 Expert architectures and native prediction scales

Sleepwalker uses a SleepTransformer (Phan et al., 2022) for sleep staging and three task-specific, U-Time-inspired (Perslev et al., 2019) convolutional models for event detection. Its sleep expert

Table 3: Training class distribution in representative samples of 250,000 input windows per task. Counts are rounded. Percentages use the unrounded task total.

Task	Class	Target mass	Within-task share (%)
Sleep	Wake	44,256	17.70
	N1	31,027	12.41
	N2	113,132	45.25
	N3	30,822	12.33
	REM	30,763	12.31
Arousal	No arousal	9,509,827	99.10
	Arousal	86,141	0.90
Breathing	Apnea	64,123	3.35
	Hypopnea	78,825	4.11
	Regular breathing	1,772,702	92.54
Desaturation	Desaturation	104,047	5.48
	No desaturation	1,794,418	94.52

processes 21 consecutive 30-second EEG epochs, applies frame-level and epoch-level attention, and predicts the stage of the central epoch. The event experts use separate physiological inputs: EEG, EOG, chin EMG, and ECG for arousal; abdomen, chest, airflow, and SpO_2 for breathing; and SpO_2 alone for desaturation. Each model retains its own filtering, normalization, and temporal context. The U-Time architecture keeps the original one-dimensional encoder–decoder, skip connections, nearest-neighbor upsampling, and segment-level predictions, but is not an exact reproduction. The four encoder levels use 64, 128, 128, and 256 channels, with kernel widths 5, 5, 3, and 3 and pooling factors 5, 5, 3, and 3. Each block has two ordinary convolutions with an ELU activation; the decoded central output span is averaged over one-, five-, or ten-second bins and a linear layer produces task logits. The original U-Time used dilated convolutions with a wider effective kernel, pooling factors 10, 8, 6, and 4, and batch normalization (Perslev et al., 2019). Here arousal and breathing use channel-wise normalization across the time axis, whereas desaturation uses batch normalization. All three event models have zero convolutional dropout.

SleepFM (Thapa et al., 2024) uses 13 PSG channels grouped into brain/eye, respiratory, cardiac, and muscle modalities. Its frozen backbone converts a 300-second, 128-Hz input into 60 five-second tokens per modality. The sleep head averages each group of six neighboring tokens to produce ten 30-second predictions, then applies a shared linear classifier. The arousal, breathing, and desaturation heads instead flatten the complete multimodal token sequence and apply one linear map to all output logits of the task window. Thus a linear head does not necessarily have few trainable weights: the arousal head predicts 40 two-class intervals from a long embedding vector. We fit these heads on HSP rather than use the released whole-night sleep head, whose training identities cannot be reconciled with the HSP partition. OSF (Shuai et al., 2026) embeds one 30-second, 64-Hz multimodal segment with a frozen backbone and fits a linear classification head for each task. The OSF inputs in this study comprise 12 channels and do not include a direct SpO_2 channel, even for desaturation. Unlike the Sleepwalker and SleepFM event experts, all three OSF event heads output only one prediction per 30 seconds. This difference in temporal resolution should be kept in mind when interpreting event metrics. Table 4 gives the actual input duration, sampling rate, and native output timeline for each expert. The output entry $T \times \delta$ means T consecutive predictions of duration δ; the offset is relative to the start of the input. For example, the Sleepwalker sleep expert describes seconds $[300, 330)$ of a 630-second EEG window, while the OSF sleep expert describes its full 30-second input. Equal tensor lengths therefore do not imply equal physical intervals.

The number of fitted parameters also differs sharply across families (Table 5). Sleepwalker trains each complete expert. The foundation backbones remain frozen; only the task heads are fitted. SleepFM's full-window event heads have substantially more parameters than its temporally pooled sleep head, while OSF's single-segment heads are much smaller.

Table 4: Native temporal contracts of the twelve trained task experts.

Family	Task	Input rate (Hz)	Input (s)	Output	Offset (s)
Sleepwalker	Sleep	100	630	1×30 s	300
	Arousal	100	60	40×1 s	10
	Breathing	100	90	8×5 s	25
	Desaturation	100	150	8×10 s	35
SleepFM	Sleep	128	300	10×30 s	0
	Arousal	128	300	40×1 s	130
	Breathing	128	300	8×5 s	130
	Desaturation	128	300	8×10 s	110
OSF	Sleep	64	30	1×30 s	0
	Arousal	64	30	1×30 s	0
	Breathing	64	30	1×30 s	0
	Desaturation	64	30	1×30 s	0

Table 5: Trainable parameters when fitting each individual expert. For Sleepwalker the count includes the full network; for SleepFM and OSF it includes only the task head.

Family	Sleep	Arousal	Breathing	Desaturation
Sleepwalker	3,834,053	1,439,850	1,440,363	1,435,341
SleepFM	2,565	2,457,680	737,304	491,536
OSF	3,845	1,538	2,307	1,538

A.3 Training and evaluation settings

Table 6 reports the hyperparameters used for the experiments in the main paper. The frozen-composition rows apply to naive, rate-only, and contract alignment. Every epoch samples 250,000 input windows from a pool of 512 recording paths. The optimizer is AdamW throughout, with weight decay 10^{-4}. The fixed random seed is 17. Individual experts use the training partition for training, and fine-tuning is done on the fine-tuning dataset. The test partition is reserved for complete-night evaluation. In the joint regime, the pretrained foundation backbones remain frozen even though their HSP task heads can update. Every composition has four task-specific linear heads, initialized with weights from $\mathcal{N}(0, 0.005^2)$ and zero biases. Together they have 114 trainable parameters when the experts are frozen. A joint model updates its permitted expert parameters through both the native task loss and incoming probability messages. An exchange-adapted model retains the four experts and fits the correction heads to the new combination.

A.4 Compute and energy accounting

We sum wall-clock durations from all 60 completed, single-A30 jobs reported here with retained timing records: 19 composition or adaptation fits (177.3 GPU-hours) and 41 complete-night evaluations (298.8 GPU-hours), including the eight naive-exchange evaluations and four post-exchange head refits. The total is 476.1 A30 GPU-hours. Multiplication by the A30's rated maximum board power of 165 W[2] gives approximately 79 kWh as a TDP-based upper-bound estimate for *these recorded GPU jobs*, not a direct energy measurement. Initial expert fitting, foundation-model pretraining, earlier or discarded development runs, CPU and storage power, and the unfinished multi-task run are excluded. The 476.1 GPU-hours are therefore not a full-project compute total, and 79 kWh is not an upper bound on full-project energy consumption.

[2]NVIDIA A30 datasheet.

Table 6: Optimization settings for the final comparison. Initial / peak denotes a OneCycle schedule; a single value denotes a constant learning rate. All runs use 250,000 sampled input windows per epoch and AdamW with weight decay 10^{-4}.

Run	Epochs	Batch size	Learning rate	Updated parameters
Sleepwalker experts	100	128	10^{-5} / $5 \cdot 10^{-4}$	Full expert
SleepFM and OSF task heads	10	32	10^{-3}	Linear task head
Frozen composition: Sleepwalker	10	128	10^{-5} / $5 \cdot 10^{-4}$	Correction heads
Frozen composition: SleepFM / OSF	10	64	10^{-5} / $5 \cdot 10^{-4}$	Correction heads
Joint composition: Sleepwalker	10	32	10^{-5} / $5 \cdot 10^{-4}$	Experts and correction heads
Joint composition: SleepFM / OSF	10	16	10^{-5} / $5 \cdot 10^{-4}$	Task heads and correction heads
Exchange adaptation: SleepFM / OSF	10	64	10^{-5} / $5 \cdot 10^{-4}$	Correction heads

B Additional Results

B.1 Cohen's κ and uncertainty for the six systems

Table 7 gives Cohen's κ for the six systems in Table 1. For each family, the common cohort includes recordings with all four tasks initialized and at most 5% missing expected input windows in every completed system. We pool confusion matrices over these recordings before computing each metric, rather than averaging recording-level scores.

Table 7: Complete-night Cohen's κ for the systems in Table 1. The common cohorts comprise 1,820 Sleepwalker, 1,626 SleepFM, and 1,785 OSF recordings.

Family	System	Sleep	Arousal	Breathing	Desaturation
Sleepwalker	Independent	.747	.260	.619	.307
	Naive Alignment	.484	.009	.492	.167
	Naive + head fine-tuning	.490	.014	.509	.188
	Contract Alignment	.740	.269	.625	.304
	Contract + head fine-tuning	.743	.279	.629	.345
	Contract + joint fine-tuning	.734	.252	.625	.357
SleepFM	Independent	.528	.125	.264	.128
	Naive Alignment	.482	.015	.128	.094
	Naive + head fine-tuning	.484	.017	.132	.126
	Contract Alignment	.520	.133	.293	.126
	Contract + head fine-tuning	.524	.129	.302	.160
	Contract + joint fine-tuning	.619	.236	.407	.249
OSF	Independent	.637	.009	.267	.072
	Naive Alignment	.609	.005	.169	.051
	Naive + head fine-tuning	.611	.019	.164	.107
	Contract Alignment	.637	.021	.233	.067
	Contract + head fine-tuning	.641	.073	.229	.128
	Contract + joint fine-tuning	.673	.014	.219	.098

Table 8: 95% patient-cluster bootstrap confidence intervals for macro-F1 in Table 1. Each interval is the 2.5th–97.5th percentile of 2,000 resamples.

Family	System	Sleep	Arousal	Breathing	Desaturation
Sleepwalker	Independent	(.783,.793)	(.624,.637)	(.699,.714)	(.643,.660)
	Naive Alignment	(.576,.585)	(.504,.506)	(.624,.637)	(.576,.589)
	Naive + head fine-tuning	(.583,.593)	(.505,.507)	(.635,.649)	(.587,.601)
	Contract Alignment	(.777,.787)	(.628,.642)	(.703,.718)	(.642,.659)
	Contract + head fine-tuning	(.781,.791)	(.633,.646)	(.707,.721)	(.663,.680)
	Contract + joint fine-tuning	(.774,.785)	(.619,.633)	(.706,.720)	(.669,.686)
SleepFM	Independent	(.603,.616)	(.558,.567)	(.471,.485)	(.555,.569)
	Naive Alignment	(.565,.578)	(.506,.509)	(.393,.402)	(.536,.551)
	Naive + head fine-tuning	(.568,.582)	(.505,.507)	(.393,.402)	(.554,.571)
	Contract Alignment	(.598,.612)	(.562,.571)	(.482,.496)	(.552,.568)
	Contract + head fine-tuning	(.602,.616)	(.558,.567)	(.486,.499)	(.570,.587)
	Contract + joint fine-tuning	(.674,.687)	(.612,.623)	(.568,.582)	(.617,.631)
OSF	Independent	(.687,.700)	(.501,.509)	(.472,.490)	(.521,.537)
	Naive Alignment	(.663,.675)	(.499,.502)	(.416,.431)	(.511,.524)
	Naive + head fine-tuning	(.671,.683)	(.507,.511)	(.412,.427)	(.539,.559)
	Contract Alignment	(.687,.700)	(.506,.512)	(.453,.468)	(.520,.534)
	Contract + head fine-tuning	(.697,.710)	(.532,.540)	(.449,.464)	(.551,.571)
	Contract + joint fine-tuning	(.722,.735)	(.503,.507)	(.443,.457)	(.535,.550)

Table 9: 95% patient-cluster bootstrap confidence intervals for Cohen's κ in Table 7. Each interval is the 2.5th–97.5th percentile of 2,000 resamples.

Family	System	Sleep	Arousal	Breathing	Desaturation
Sleepwalker	Independent	(.740,.754)	(.247,.273)	(.609,.628)	(.290,.324)
	Naive Alignment	(.478,.491)	(.007,.011)	(.483,.501)	(.154,.181)
	Naive + head fine-tuning	(.484,.496)	(.012,.016)	(.500,.517)	(.175,.202)
	Contract Alignment	(.734,.747)	(.256,.283)	(.615,.634)	(.287,.321)
	Contract + head fine-tuning	(.736,.749)	(.265,.293)	(.620,.638)	(.328,.362)
	Contract + joint fine-tuning	(.726,.741)	(.238,.266)	(.616,.634)	(.340,.374)
SleepFM	Independent	(.519,.536)	(.116,.134)	(.252,.276)	(.114,.142)
	Naive Alignment	(.473,.491)	(.013,.018)	(.119,.136)	(.079,.110)
	Naive + head fine-tuning	(.475,.493)	(.015,.019)	(.123,.140)	(.110,.143)
	Contract Alignment	(.511,.529)	(.124,.142)	(.279,.305)	(.111,.142)
	Contract + head fine-tuning	(.515,.533)	(.120,.138)	(.289,.314)	(.144,.177)
	Contract + joint fine-tuning	(.610,.627)	(.226,.247)	(.394,.418)	(.234,.264)
OSF	Independent	(.628,.645)	(.003,.017)	(.251,.282)	(.058,.087)
	Naive Alignment	(.601,.617)	(.003,.007)	(.155,.181)	(.039,.065)
	Naive + head fine-tuning	(.603,.619)	(.015,.023)	(.150,.177)	(.088,.127)
	Contract Alignment	(.628,.645)	(.017,.027)	(.220,.246)	(.054,.081)
	Contract + head fine-tuning	(.633,.650)	(.065,.081)	(.215,.242)	(.110,.147)
	Contract + joint fine-tuning	(.665,.681)	(.010,.017)	(.205,.232)	(.084,.112)

Table 10: Macro-F1 gain of Contract + head fine-tuning over Naive + head fine-tuning before expert exchange. Each cell shows the gain followed by its paired 95% patient-cluster bootstrap confidence interval in parentheses (lower, upper). Both systems freeze the same experts and fit four correction heads.

Family	Sleep	Arousal	Breathing	Desaturation
Sleepwalker	+.198 (+.194,+.202)	+.133 (+.126,+.140)	+.072 (+.070,+.075)	+.078 (+.074,+.082)
SleepFM	+.034 (+.033,+.036)	+.057 (+.053,+.061)	+.095 (+.092,+.099)	+.017 (+.015,+.019)
OSF	+.027 (+.025,+.028)	+.027 (+.025,+.030)	+.037 (+.035,+.040)	+.012 (+.010,+.014)

All twelve paired confidence intervals in Table 10 lie above zero, including comparisons in which the fitted naive scores change little from Naive Alignment.

Table 11 gives the evaluation scale for Contract + head fine-tuning. It distinguishes recordings from people and lists scored task intervals rather than input windows. Scored hours are the number of 30-second sleep-stage intervals divided by 120: equivalent sleep-stage coverage, not measured raw PSG duration. OSF's 30-second native event predictions are mapped to the common event

timelines during evaluation; its millions of one-second arousal intervals are therefore not millions of independent native predictions.

Table 11: Common cohorts and scored intervals for Contract + head fine-tuning.

Family	Recordings	Patients	Sleep	Arousal	Breathing	Desaturation	All task intervals	Scored hours
Sleepwalker	1,820	1,418	1,565,796	47,057,249	9,410,018	4,726,032	62,759,095	13,048
SleepFM	1,626	1,278	1,408,460	42,310,723	8,460,799	4,242,540	56,422,522	11,737
OSF	1,785	1,400	1,575,917	47,327,393	9,463,993	4,739,505	63,106,808	13,133

B.2 RATE-ONLY ALIGNMENT

We introduce rate-only alignment as an intermediate control between Naive Alignment and Contract Alignment. Naive Alignment invokes each expert once from a shared input origin and resizes its outputs. Contract Alignment instead runs every expert under its native input and output contract and aligns cross-task predictions by physical time. Rate-only alignment keeps native expert execution but replaces the cross-task timestamp alignment with tensor-length matching. This separates the contribution of executing experts correctly from that of aligning the information they exchange.

For each task, rate-only alignment makes the native expert calls needed to cover that task's own output span and places its logits on that task's timeline. To pass probabilities to another task, it resizes the source sequence to the receiver's number of outputs: nearest-neighbor interpolation when the receiver has more rows, area averaging when it has fewer, and no change when the lengths match. It pairs rows by position without checking their timestamps. For example, the SleepFM breathing and desaturation experts each produce eight outputs, but their first outputs describe seconds $[130, 135)$ and $[110, 120)$ of their respective 300-second inputs (Table 4). Rate-only pairs those first outputs because the tensors have equal length, even though their physical intervals do not overlap. This shape-based rule can approximate alignment when output spans and offsets are similar, but it does not identify which predictions refer to the same time.

Full contract alignment instead makes the native calls needed to cover the common prediction envelope and matches each incoming probability to the receiver's physical intervals (Section 3.1). We fit separate correction heads for the original rate-only and contract-aligned systems while keeping their experts frozen. For zero-shot exchange, each system retains its fitted heads and replaces only the specified expert. Table 12 reports the rate-only scores. Contract-aligned macro-F1 scores are in Tables 1 and 2; Cohen's κ is in Tables 7 and 13. The comparisons use the same recordings within each family, although they may score slightly different intervals.

Table 12: Rate-only alignment. Each cell gives macro-F1 / Cohen's κ; bootstrap intervals are in Tables 14 and 15.

Expert family	System	Sleep	Arousal	Breathing	Desaturation
SleepFM	Rate-only + head fine-tuning	.610 / .524	.563 / .129	.494 / .303	.578 / .159
	Rate-only sleep zero-shot	.783 / .738	.565 / .133	.501 / .318	.580 / .163
	Rate-only breathing zero-shot	.610 / .524	.563 / .129	.601 / .474	.590 / .185
OSF	Rate-only + head fine-tuning	.704 / .642	.537 / .074	.460 / .234	.569 / .143
	Rate-only sleep zero-shot	.785 / .737	.538 / .076	.459 / .234	.570 / .145
	Rate-only breathing zero-shot	.703 / .639	.536 / .072	.700 / .610	.591 / .187

Rate-only and contract-aligned scores are close in most comparisons, and neither has a consistent macro-F1 advantage. This suggests that native expert execution explains much of the gap between Naive Alignment and Contract Alignment in the main paper. The clearest difference is the SleepFM breathing exchange: rate-only gives .601 versus .584 breathing F1 with full contracts, while desaturation F1 is .590 versus .615. These comparisons are somewhat limited, however, because the scored intervals differ slightly. Moreover, full contracts provide an interpretable execution and "proper" alignment compared to the rate-only approximation with comparable scores.

B.3 ADDITIONAL EXCHANGE METRICS

Table 13 gives Cohen's κ for the contract- and naive-aligned exchanges in Table 2. The corresponding unexchanged host scores are in Table 7.

Table 13: Cohen's κ for the exchange systems in Table 2. SleepFM uses 1,626 recordings from 1,278 patients; OSF uses 1,785 from 1,400, except the naive breathing rows with 1,784 from 1,399. Bootstrap intervals are in Table 15.

Expert family	System	Sleep	Arousal	Breathing	Desaturation
SleepFM	Sleep: Contract zero-shot	.738	.133	.317	.164
	Sleep: Contract fine-tuning	.740	.133	.317	.164
	Sleep: Naive zero-shot	.479	.017	.133	.124
	Sleep: Naive fine-tuning	.486	.017	.133	.124
	Breathing: Contract zero-shot	.525	.129	.470	.232
	Breathing: Contract fine-tuning	.529	.130	.621	.225
	Breathing: Naive zero-shot	.484	.017	.219	.189
	Breathing: Naive fine-tuning	.491	.017	.510	.179
OSF	Sleep: Contract zero-shot	.737	.075	.228	.131
	Sleep: Contract fine-tuning	.744	.074	.233	.130
	Sleep: Naive zero-shot	.486	.018	.154	.101
	Sleep: Naive fine-tuning	.494	.021	.165	.108
	Breathing: Contract zero-shot	.638	.068	.609	.185
	Breathing: Contract fine-tuning	.640	.082	.625	.239
	Breathing: Naive zero-shot	.606	.018	.487	.148
	Breathing: Naive fine-tuning	.612	.024	.514	.206

Beyond macro-F1 and κ, zero-shot breathing exchange raises desaturation-positive F1 from .185 to .255 in SleepFM and from .147 to .205 in OSF. After head fine-tuning, apnea/hypopnea F1 reaches .612/.535 for SleepFM and .616/.538 for OSF.

B.4 UNCERTAINTY OF COMPLETE-NIGHT SCORES

The 2,000 patient-cluster bootstrap resamples described in Section 4 give percentile 95% confidence intervals for macro-F1 and Cohen's κ. Paired differences use the same sampled patients for both systems. These intervals quantify test-cohort sampling uncertainty conditional on the fitted models and fixed HSP test set. They do not include variation from retraining, alternative data sets, or model selection, and they should not be used to compare the different family cohorts as though they were paired. All displayed limits are rounded to three decimals.

Tables 8 and 9 give intervals for the six systems in Table 1. Tables 14 and 15 give intervals for the rate-only and exchange systems; the Contract + head fine-tuning rows already appear in the first two interval tables. Tables 10, 16, and 17 report paired differences.

Table 14: 95% patient-cluster bootstrap confidence intervals for macro-F1 in the additional alignment and exchange systems. Row names match Tables 12 and 13.

Expert family	System	Sleep	Arousal	Breathing	Desaturation
SleepFM	Rate-only + head fine-tuning	(.603,.617)	(.558,.567)	(.487,.500)	(.569,.586)
	Rate-only sleep zero-shot	(.778,.789)	(.560,.569)	(.495,.508)	(.571,.588)
	Rate-only breathing zero-shot	(.603,.617)	(.558,.567)	(.593,.609)	(.580,.599)
	Sleep: Contract zero-shot	(.776,.787)	(.560,.569)	(.494,.507)	(.572,.589)
	Sleep: Contract fine-tuning	(.777,.788)	(.560,.570)	(.495,.508)	(.572,.589)
	Breathing: Contract zero-shot	(.602,.617)	(.558,.567)	(.576,.592)	(.604,.626)
	Breathing: Contract fine-tuning	(.607,.621)	(.559,.568)	(.702,.717)	(.601,.621)
	Sleep: Naive zero-shot	(.572,.582)	(.504,.507)	(.395,.403)	(.553,.569)
	Sleep: Naive fine-tuning	(.577,.587)	(.504,.507)	(.394,.403)	(.553,.569)
	Breathing: Naive zero-shot	(.568,.582)	(.504,.507)	(.417,.427)	(.585,.603)
	Breathing: Naive fine-tuning	(.576,.589)	(.505,.507)	(.636,.650)	(.578,.597)
OSF	Rate-only + head fine-tuning	(.697,.710)	(.533,.541)	(.452,.468)	(.559,.579)
	Rate-only sleep zero-shot	(.780,.790)	(.534,.542)	(.452,.467)	(.560,.580)
	Rate-only breathing zero-shot	(.696,.709)	(.532,.540)	(.693,.707)	(.581,.601)
	Sleep: Contract zero-shot	(.779,.790)	(.534,.542)	(.449,.464)	(.553,.572)
	Sleep: Contract fine-tuning	(.781,.791)	(.533,.541)	(.452,.467)	(.552,.571)
	Breathing: Contract zero-shot	(.695,.709)	(.530,.538)	(.692,.706)	(.581,.600)
	Breathing: Contract fine-tuning	(.697,.710)	(.537,.545)	(.705,.719)	(.610,.627)
	Sleep: Naive zero-shot	(.582,.591)	(.507,.510)	(.407,.422)	(.536,.555)
	Sleep: Naive fine-tuning	(.585,.595)	(.508,.512)	(.413,.428)	(.539,.559)
	Breathing: Naive zero-shot	(.667,.680)	(.507,.511)	(.619,.632)	(.560,.581)
	Breathing: Naive fine-tuning	(.673,.686)	(.510,.514)	(.638,.651)	(.592,.611)

Table 15: 95% patient-cluster bootstrap confidence intervals for Cohen's κ in the additional alignment and exchange systems; row definitions match Table 14.

Expert family	System	Sleep	Arousal	Breathing	Desaturation
SleepFM	Rate-only + head fine-tuning	(.514,.532)	(.120,.138)	(.289,.315)	(.142,.175)
	Rate-only sleep zero-shot	(.731,.746)	(.124,.142)	(.305,.330)	(.146,.179)
	Rate-only breathing zero-shot	(.515,.533)	(.120,.138)	(.460,.488)	(.166,.204)
	Sleep: Contract zero-shot	(.731,.745)	(.124,.142)	(.304,.329)	(.148,.181)
	Sleep: Contract fine-tuning	(.733,.748)	(.124,.142)	(.304,.330)	(.147,.180)
	Breathing: Contract zero-shot	(.515,.533)	(.120,.138)	(.458,.482)	(.210,.254)
	Breathing: Contract fine-tuning	(.519,.537)	(.121,.139)	(.611,.631)	(.206,.244)
	Sleep: Naive zero-shot	(.471,.486)	(.015,.019)	(.125,.141)	(.108,.141)
	Sleep: Naive fine-tuning	(.479,.493)	(.014,.019)	(.125,.141)	(.108,.141)
	Breathing: Naive zero-shot	(.475,.493)	(.015,.019)	(.211,.226)	(.171,.206)
	Breathing: Naive fine-tuning	(.482,.500)	(.015,.020)	(.500,.519)	(.160,.198)
OSF	Rate-only + head fine-tuning	(.634,.650)	(.066,.082)	(.220,.248)	(.124,.164)
	Rate-only sleep zero-shot	(.731,.744)	(.068,.084)	(.221,.248)	(.126,.166)
	Rate-only breathing zero-shot	(.631,.647)	(.064,.080)	(.600,.620)	(.167,.206)
	Sleep: Contract zero-shot	(.730,.743)	(.067,.083)	(.215,.242)	(.112,.150)
	Sleep: Contract fine-tuning	(.737,.750)	(.066,.083)	(.220,.247)	(.111,.149)
	Breathing: Contract zero-shot	(.629,.646)	(.061,.076)	(.599,.619)	(.167,.204)
	Breathing: Contract fine-tuning	(.631,.648)	(.074,.091)	(.615,.634)	(.222,.256)
	Sleep: Naive zero-shot	(.479,.492)	(.015,.021)	(.141,.166)	(.083,.120)
	Sleep: Naive fine-tuning	(.487,.500)	(.017,.025)	(.151,.178)	(.090,.128)
	Breathing: Naive zero-shot	(.598,.614)	(.015,.022)	(.477,.496)	(.129,.169)
	Breathing: Naive fine-tuning	(.603,.619)	(.020,.028)	(.505,.522)	(.187,.226)

Table 16: Paired macro-F1 changes after inserting a Sleepwalker expert. The baseline is Contract + head fine-tuning before exchange. The first change compares zero-shot exchange with this baseline; the second compares head fine-tuning after exchange with zero-shot exchange. Parentheses give the lower and upper limits of the 95% patient-cluster bootstrap confidence interval. A dash indicates that no paired fine-tuning change is reported for desaturation.

Expert family	Inserted expert	Scored task	Zero-shot vs. baseline	Head fine-tuning vs. zero-shot
SleepFM	Sleep	Sleep	+.172 (+.166,+.179)	+.001 (+.000,+.002)
	Breathing	Breathing	+.091 (+.084,+.098)	+.125 (+.121,+.130)
	Breathing	Desaturation	+.036 (+.031,+.042)	–
OSF	Sleep	Sleep	+.081 (+.077,+.086)	+.001 (+.001,+.002)
	Breathing	Breathing	+.243 (+.234,+.251)	+.013 (+.011,+.014)
	Breathing	Desaturation	+.029 (+.023,+.036)	–

Table 17: Paired macro-F1 changes after inserting a Sleepwalker expert under naive alignment. The baseline is Naive Alignment before exchange; columns compare zero-shot exchange with this baseline and head fine-tuning after exchange with zero-shot exchange. Parentheses give 95% patient-cluster bootstrap confidence intervals.

Expert family	Inserted expert	Scored task	Zero-shot vs. baseline	Head fine-tuning vs. zero-shot
SleepFM	Sleep	Sleep	+.002 (-.004,+.009)	+.005 (+.004,+.006)
	Breathing	Breathing	+.024 (+.019,+.030)	+.221 (+.214,+.227)
	Breathing	Desaturation	+.032 (+.028,+.036)	-.006 (-.010,-.003)
OSF	Sleep	Sleep	-.090 (-.096,-.085)	+.003 (+.003,+.004)
	Breathing	Breathing	+.206 (+.197,+.214)	+.019 (+.017,+.021)
	Breathing	Desaturation	+.022 (+.015,+.028)	+.031 (+.028,+.035)

Under naive alignment, refitting heads after breathing exchange adds .221 macro-F1 in SleepFM but only .019 in OSF; both adapted scores remain below their contract-aligned counterparts. Sleep-exchange refitting adds only .005/.003 in SleepFM/OSF, and the OSF naive zero-shot swap falls below its unexchanged naive host by .090.

B.5 Which classes account for the macro scores?

Table 18 reports reference-class support and per-class F1 for Contract + head fine-tuning on the respective common cohorts. It exposes the contrast between abundant negative intervals and rare events. Sleepwalker's arousal macro-F1 of 0.640 combines 0.993 F1 for no arousal with 0.286 for arousal. OSF's arousal score of 0.536 combines 0.992 and 0.081, while its desaturation-positive F1 is only 0.147 despite 0.975 for no desaturation. We hypothesize that its coarser native event output and absence of a direct SpO_2 input are the relevant differences here.

Table 18: Reference-class support and F1 for Contract + head fine-tuning. Support counts scored intervals carrying the reference class, not distinct annotated events.

Task	Class	Sleepwalker		SleepFM		OSF	
		Support	F1	Support	F1	Support	F1
Sleep	Wake	222,259	.838	197,447	.651	243,161	.781
	N1	184,441	.562	166,803	.356	184,862	.449
	N2	735,205	.862	662,216	.759	728,482	.807
	N3	208,366	.795	188,238	.688	205,483	.745
	REM	215,525	.871	193,756	.595	213,929	.736
Arousal	No arousal	46,592,686	.993	41,880,837	.983	46,868,627	.992
	Arousal	464,563	.286	429,886	.142	458,766	.081
Breathing	Apnea	240,782	.620	213,371	.343	239,657	.276
	Hypopnea	281,610	.541	249,178	.177	281,185	.127
	Regular breathing	8,887,626	.983	7,998,250	.960	8,943,151	.967
Desaturation	Desaturation	207,163	.366	180,767	.185	204,350	.147
	No desaturation	4,518,869	.977	4,061,773	.973	4,535,155	.975