ADAPTIVE MULTI-OBJECTIVE ALIGNMENT IN PRETRAINING TIME-SERIES FOUNDATION MODELS FOR EEG TASKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Time-series foundation models have become a powerful approach for analyzing EEG data, showing strong performance across healthcare and neuroscience. However, most models rely on a single self-supervised pretraining objective, such as masked reconstruction, which may limit their ability to capture diverse features and hinder generalization across downstream tasks. Incorporating multiple learning objectives during pretraining is a promising direction, though disparities in gradient magnitudes can cause certain objectives to dominate the optimization, which reduces the benefits of joint training. In this work, we systematically study this phenomenon and explore the benefits of multi-objective pretraining of time-series foundation models. We introduce MAPE (Multi-objective Alignment in Pretraining time-series foundation models for EEG tasks), a theoretically grounded framework that efficiently aligns multiple objectives within a single training pipeline by analyzing their pretraining losses. Our straightforward, low-cost approach is applicable to any model. Our empirical analysis across three different encoders demonstrates that MAPE leads to improved performance compared to unaligned multi-objective pretraining and single self-supervised pretraining settings, highlighting its effectiveness in learning robust and generalizable EEG representations.

1 INTRODUCTION

Electroencephalography (EEG) is a primary non-invasive neuroimaging modality, providing high-temporal-resolution data critical for clinical diagnostics, such as epilepsy detection (Acharya et al., 2018; Daoud & Bayoumi, 2019), and for the development of closed-loop brain-computer interfaces (Edelman et al., 2019; Appriou et al., 2020). Despite its broad utility, EEG signals are perennially hampered by low signal-to-noise ratios (SNR), inherent temporal nonstationarity, and significant inter-subject variability, which together pose formidable challenges to cross-subject generalization (Sun & Mou, 2023; Lotte et al., 2007). To mitigate these issues, recent research has pivoted toward large-scale foundation models that integrate EEG-specific inductive biases (Guetschel et al., 2024). CBraMod (Wang et al., 2025), employs dual channel-wise and temporal attention to capture the complex covariance of spatio-temporal neural patterns. Similarly, CSBrain (Zhou et al., 2026) utilizes multi-scale windows to aggregate information across disparate frequency bands. While these advancements offer significant improvements over pioneering frameworks such as BIOT (Yang et al., 2023) and LaBraM (Jiang et al., 2024), they remain largely constrained by their reliance on single-objective pretraining.

The myopia of single-objective pretraining. Almost all time-series foundation models depend on one objective in pretraining, in particular masked reconstruction as used in Masked Autoencoders (MAE) (Wang et al., 2025; Zhou et al., 2026; El Ouahidi et al., 2026; Chien et al., 2022; Döner et al., 2026; Goswami et al., 2024). While MAE encourages the model to capture local dependencies by reconstructing masked signal patches, this approach allocates disproportionate representational capacity to fine-grained, stochastic signal fluctuations that often lack global semantic relevance. In contrast, contrastive learning, as used in (Feofanov et al., 2026; Yang et al., 2023; Kostas et al., 2021), enforces global semantic invariance across augmented views and often discards the high-resolution nuances essential for transient waveform and micro-state recognition. Thus, relying on any single

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.

pretraining objective inherently restricts feature learning, limiting the model's capacity to extract the multi-faceted representations required for robust neural decoding.

Potential of diversified pretraining. Integrating diverse learning objectives offers a pathway toward more generalized representations, as the shared encoder is forced to synthesize multifaceted information. For instance, UniTS (Gao et al., 2024) demonstrates that the synergy between generative and predictive objectives can equip models with enhanced capabilities across more downstream datasets. Furthermore, multi-objective pretraining can be modular, offering "plug-and-play" compatibility with various backbones and attention mechanisms without requiring additional data preparation. However, this approach remains underutilized in the EEG domain because designing a unified multi-objective architecture is complex.

A fundamental limitation in multi-objective self-supervised learning is **objective competition**, where disparate gradients from different objectives can pull the shared encoder in different directions, thereby inhibiting convergence (Fig. 2a). Such interference suggests that optimizing for one objective can actively degrade the performance of others, undermining the synergy required for a truly multi-perspective encoder. Consequently, two questions emerge:

1. *Can multi-objective pretraining extract more informative features and yield more robust representations than single-objective approaches?*
2. *Given the challenge of objective competition, how can we align several pretraining objectives to facilitate a universal, generalized encoder for EEG adaptation*?

To address these, we introduce MAPE (**M**ulti-objective **A**lignment in **P**retraining time-series foundation models for **E**EG tasks), a framework featuring a unified pipeline that integrates several self-supervised objectives (Fig. 1). With this framework, we explore gradient competition during pretraining and introduce a theoretically grounded alignment method to dynamically regulate these objectives.

Our contributions are summarized as follows:

- We introduce a **theoretically-grounded adaptive alignment mechanism** based on loss analysis which regulates multiple pretraining objectives (Sec. 3.3). We leverage these findings and propose a multi-objective pretraining pipeline tailored for time-series foundation models.
- We provide an empirical analysis of objective competition during pretraining (Sec. 3.2), and demonstrate how this limits downstream performance (Sec. 4.3).
- We provide comprehensive empirical evidence of the utility of our framework, demonstrating the efficiency of our alignment method across general time-series foundation architecture (MANTIS) and EEG-specific foundation model architectures (CBraMod and EEGPT). Our MAPE enables efficient multi-objective pretraining with competitive/superior performance compared to the standard single-objective pretraining and to unaligned multi-objective pretraining (Sec. 4).

2 Related work

Time-series foundation models. The development of foundation models for EEG tasks represents an evolving frontier in neural decoding, aimed at extracting robust representations from high-dimensional, non-stationary biosignals. BIOT (Yang et al., 2023), a pioneering EEG model with contrastive pretraining, lays the groundwork for cross-task transfer by tokenizing heterogeneous biosignals into unified, sentence-like sequences. Recent advancements have increasingly focused on neuroanatomical inductive biases. CBraMod (Wang et al., 2025) introduces a dual spatio-temporal attention mechanism to capture complex intra-sample dependencies, while CSBrain (Zhou et al., 2026) and LUNA (Döner et al., 2026) partition electrodes into functional cortical regions based on physiological priors. Despite these structural innovations, these EEG-specific models remain heavily tethered to the masked signal reconstruction objective. Advanced general-purpose time-series models such as MANTIS (Feofanov et al., 2026) and MOMENT (Goswami et al., 2024), pretrained on time-series data, also achieve promising performance across EEG benchmarks. However, these architectures also rely on a single pretraining objective: MANTIS uses only contrastive learning, while MOMENT learns via masked reconstruction. We conjecture that such pretraining introduces representational myopia, causing the model to focus on a singular feature space and ultimately hindering universal generalization.

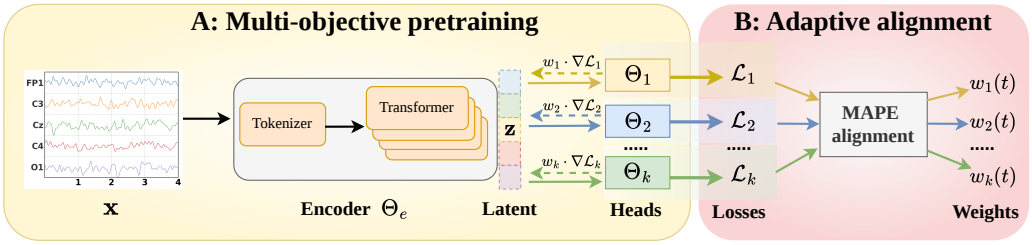


Figure 1: Overall framework of the proposed approach. **(A)** Unified pipeline for pretraining time-series foundation models with multiple objectives. Patched $\mathbf{x}$ is augmented, then encoded to latents $\mathbf{z}$ by the shared encoder $\mathbf{\Theta}_\text{e}$. k objectives operate on the latent $\mathbf{z}$ to jointly update the encoder $\mathbf{\Theta}_\text{e}$ during backpropagation. **(B)** By analysing losses $\mathcal{L}_i$, MAPE updates the objective weight w_i on-the-fly, which systematically regulates gradient flows $\mathbf{g}_i = w_i \cdot \nabla\mathcal{L}_i$ to shared encoder $\mathbf{\Theta}_\text{e}$.

Multi-objective self-supervised learning. While multi-objective pretraining has been applied in natural language processing (e.g., BERT (Devlin et al., 2019)) and computer vision (e.g., iBOT (Zhou et al., 2021)), its use in time-series biosignals remains limited. Specifically, UniTS (Gao et al., 2024) combines predictive and reconstruction objectives for general time-series pretraining; EEGPT (Wang et al., 2024) pretrains an EEG model using masked reconstruction and teacher-student alignment. However, both ignore the inherent objective imbalance, where dominant gradients overwhelm weaker ones. Other multi-objective approaches (Saeed et al., 2019; Nshimyimana et al., 2025; Kim et al., 2024) rely on domain-specific heuristics, such as image-specific augmentations or human kinematic priors, which do not directly apply to stochastic, non-stationary EEG time-series signals. Generally, the literature lacks a systematic study of objective imbalance in time-series pretraining, and none of the mentioned works proposes a model-agnostic solution to alleviate this problem for a well-generalized neural representation, such as MAPE.

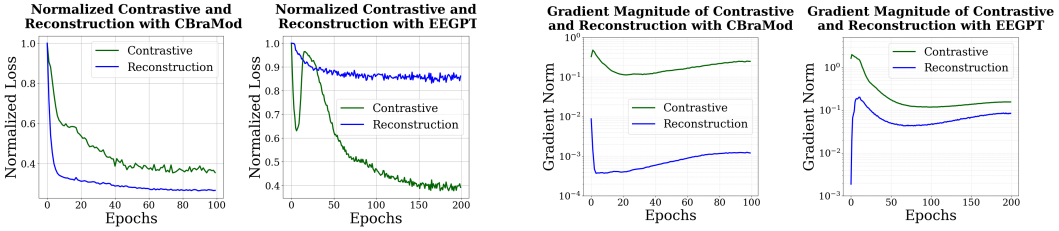
Adaptive alignment. The competition between objectives was previously observed in multimodal supervised learning, where a dominant modality can disproportionately influence gradient updates and inhibit the learning of the others (Wang et al., 2020; Huang et al., 2022). Similar competitive dynamics have been documented in supervised multi-task learning (Crawshaw, 2020; Kendall et al., 2018). To address this competition, loss modulation (Wang et al., 2020) adaptively reweights loss components based on their learning progress. An alternative paradigm is gradient modulation (Li et al., 2023; Peng et al., 2022), which directly intervenes in the backward pass to harmonize conflicting gradients. However, gradient-modulation approaches typically rely on per-iteration gradient projections or manipulations, which, although affordable for lightweight architectures, introduce prohibitive computational overhead during the pretraining of large-scale foundation models. To enable efficient alignment for time-series foundation models, MAPE develops a lightweight loss-modulation approach to mitigate objective competition at near-zero additional computational cost (Sec. 3.3), ensuring scalability without sacrificing the benefits of multi-objective pretraining.

3 MAPE: Model-Agnostic Alignment for Multiple Objectives

3.1 Unified pretraining framework

Pretraining setup: The shared encoder can be pretrained on k objectives, as shown in Fig. 1.A. Raw time-series input with C channels and t time points is patched into sample $\mathbf{x} \in \mathbb{R}^{C\times N\times T}$, with $N = \lceil \frac{t}{T} \rceil$ being the number of patches or sequence length, and predefined T being the patch size or window size. After being augmented, the sequence is tokenized and encoded by the encoder $\mathbf{\Theta}_\text{e}(\cdot)$ into latent $\mathbf{z} = \mathbf{\Theta}_\text{e}(\mathbf{x})$. In self-supervised pretraining, embedding $\mathbf{z}$ is fed into k distinguished heads, corresponding to k pretraining objectives. Denote $\mathcal{L}_i$ as the loss of objective i, $i \in \{1, 2, \ldots, k\}$.

Pretraining objectives: We augment sample $\mathbf{x}$ to support multiple learning paradigms within a single pretraining pipeline. In self-supervised pretraining, masked reconstruction and contrastive learning are the two most common techniques to learn semantic data structure:



a) Normalized loss curves of pretraining with Reconstruction and Contrastive.

b) Gradient magnitudes of contrastive dominates pretraining.

Figure 2: **Objective competition** in multi-objective pretraining with EEGPT and CBraMod encoders. **a)** illustrates instability in the loss curves (e.g., contrastive loss in green), while **b)** shows the significant gradient dominance of specific objective.

- **Reconstruction** is the most common pretraining objective for EEG foundation models (Döner et al., 2026; Wang et al., 2025; Zhou et al., 2026; Wang et al., 2024; Jiang et al., 2024; Yang et al., 2023). This objective is to reconstruct the original sample $\mathbf{x}$ from the embedding $\mathbf{z}$ via a lightweight decoder $\mathbf{\Theta}_{\mathrm{r}}$: $\hat{\mathbf{x}} = \mathbf{\Theta}_{\mathrm{r}}(\mathbf{z}) \in \mathbb{R}^{C\times N\times T}$. The reconstruction loss is defined as:

$$\mathcal{L}_r = ||\mathbf{x} - \hat{\mathbf{x}}||_2^2. \tag{1}$$

- **Contrastive learning** guides the model to distinguish positive samples from negative ones, thereby enabling it to learn global semantic information across multiple time-series samples. In this learning paradigm, the encoder learns to produce similar representations for two random augmentations of the same time-series input $\mathbf{x}$, while producing different representations for augmentations of different inputs. Via linear contrastive head $\mathbf{\Theta}_{\mathrm{c}}(\cdot)$, we project the encoded latents $\mathbf{z}^{(1)},\ \mathbf{z}^{(2)}$ of 2 augmentations into projection space: $\mathbf{h}^{(1)} = \mathbf{\Theta}_{\mathrm{c}}(\mathbf{z}^{(1)})$, $\mathbf{h}^{(2)} = \mathbf{\Theta}_{\mathrm{c}}(\mathbf{z}^{(2)})$. Within one batch of B samples, we update the parameters of $\mathbf{\Theta}_{\mathrm{e}},\ \mathbf{\Theta}_{\mathrm{c}}$ by minimizing the contrastive loss:

$$\mathcal{L}_c = -\frac{1}{B}\sum_{i=1}^{B}\log\frac{\exp\left(\mathrm{sim}(\mathbf{h}_i^{(1)},\mathbf{h}_i^{(2)})/\tau\right)}{\sum_{j=1}^{B}\exp\left(\mathrm{sim}(\mathbf{h}_i^{(1)},\mathbf{h}_j^{(2)})/\tau\right)}, \tag{2}$$

  where $\mathrm{sim}(\cdot,\cdot)$ being the cosine similarity function, τ being the temperature, set to 0.1 as in (Chen et al., 2020).

Detailed implementations of both objectives for each architecture are in Appendix C.2. Note that the framework is **extendable to additional objectives** beyond reconstruction and contrastive learning. We further demonstrate this by applying the time-frequency reconstruction objective to validate the framework in a tri-objective pretraining scenario (Appendix C.1, Sec. 4.2). Our multi-objective pretraining pipeline provides a versatile, **plug-and-play** framework in which complementary objectives facilitate a more generalizable encoder. Nevertheless, our work shows that joint pretraining with multiple objectives can lead to gradient imbalance; thus, the encoder's performance on one objective may decrease as it learns the others.

3.2 Competition in Multi-Objective Pretraining

In multi-modal supervised learning, modality competition is a phenomenon in which a subset of modalities is learned while the model fails to explore others (Huang et al., 2022). Competition has also been observed in multitask supervised learning (Li et al., 2023; Peng et al., 2022), where the shared model is updated by multiple contradicting gradients arising from different classification heads. Consequently, the shared model may be pulled in different directions, hindering learning and degrading convergence. As a result, joint training performs worse than the best uni-task training (Phan et al., 2021).

However, competition in multi-objective self-supervised learning remains unexplored. Thus, generalizing the concept to multi-objective self-supervised learning, we define **objective competition** as the phenomenon in which the model learns certain objectives at the expense of others. Such competition intuitively arises when one objective focuses on local patch-level semantics (e.g., reconstruction),

while another focuses on global semantics (e.g., contrastive). This competition yields unstable loss trajectories, as divergent gradients pull the shared encoder in different directions to satisfy their targets (Fig. 2a).

We also attribute this objective competition to imbalanced gradient scales, where a single dominant gradient suppresses learning of other objectives. For example, contrastive learning dominates the optimization trajectory, as its gradient magnitude is approximately $1000\times$ larger than reconstruction in CBraMod and $2\times$ in EEGPT, as shown in Fig. 2b. Appendix B provides further evidence of objective competition and its impact on representation learning.

To reduce competition between objectives, their gradients $\mathbf{g}_i = \nabla_{\mathbf{\Theta}_e} \mathcal{L}_i^T$ for $i \in \{1, 2, \ldots, k\}$ needed to be modulated. However, extracting and processing multiple gradients at each pretraining iteration adds significant training overhead. Therefore, our approach assigns adaptive weights $w_i(t)\ \forall i \in \{1, 2, \ldots, k\}$ for each loss component and adjusts them based on the model's behavior throughout pretraining. This allows for the composite gradient to consider meaningful contributions from all objectives at hand, rather than explicitly resolving any inherent directional gradient conflicts.

3.3 Model-Agnostic adaptive alignment for multiple objectives

In k-objective pretraining, we denote model parameters at training iteration t as $\mathbf{\Theta}_t$. The total training and validation losses, $\mathcal{L}^T$ and $\mathcal{L}^V$, are defined as:

$$\mathcal{L}^T(\mathbf{\Theta}_t) = \sum_{i=1}^{k} w_i(t) \cdot \mathcal{L}_i^T(\mathbf{\Theta}_t), \quad \mathcal{L}^V(\mathbf{\Theta}_t) = \sum_{i=1}^{k} w_i(t) \cdot \mathcal{L}_i^V(\mathbf{\Theta}_t), \tag{3}$$

where $\mathcal{L}_i^T(\mathbf{\Theta}_t)$ and $\mathcal{L}_i^V(\mathbf{\Theta}_t)$, respectively represent the objective-specific training and validation losses, and $w_i(t)$ is their corresponding weight. While conventional pretraining uses fixed weights ($w_i(t) = w_i$), we mitigate gradient dominance by adaptively updating $w_i(t)$ every N_0 training iterations based on each objective's generalization and overfitting gap.

Overfitting-to-Generalization alignment (OGR). Considering objective i, (Wang et al., 2020) quantify its overfitting as the discrepancy between training and validation loss reductions over N_0 iterations $O_i^{(N_0)} = (\mathcal{L}_i^T(\mathbf{\Theta}_0) - \mathcal{L}_i^T(\mathbf{\Theta}_{N_0})) - (\mathcal{L}_i^V(\mathbf{\Theta}_0) - \mathcal{L}_i^V(\mathbf{\Theta}_{N_0}))$. Correspondingly, the generalization progress $G_i^{(N_0)}$ is defined as the improvement in validation loss, $G_i^{(N_0)} = \mathcal{L}_i^V(\mathbf{\Theta}_0) - \mathcal{L}_i^V(\mathbf{\Theta}_{N_0})$. High $O_i^{(N_0)}$ reflects poor validation performance despite training progress, while low $G_i^{(N_0)}$ indicates stagnant generalization. Defining $O^{(N_0)} = \sum_i^k w_i O_i^{(N_0)}$, $G^{(N_0)} = \sum_i^k w_i G_i^{(N_0)}$ as the total Overfitting and total Generalization, they propose the optimization target as:

$$\text{OGR} = f(\mathbf{w}) = \left| \frac{O^{(N_0)}}{G^{(N_0)}} \right|. \tag{4}$$

To minimize this optimization target, they update objective weight w_i for the next training iteration as:

$$w_i(t+1) = \frac{G_i^{(N_0)}(t)}{\left(O_i^{(N_0)}(t)\right)^2}. \tag{5}$$

Further derivations on this weight selection appear in Appendix F. The straightforward idea of jointly maximizing validation gain $G_i^{(N_0)}$ and minimizing overfitting $O_i^{(N_0)}$ based solely on the objectives' losses makes the approach theoretically applicable to any model. We extend this approach to our multi-objective pretraining pipeline as a baseline: OGR.

However, this approach has *3 problems*. First, the most critical one is that the decoupled update rule for w_i (Eq. 5) does not account for the performance of other objectives. This overlooks objective competition explored in Sec. 3.2. Second, OGR is scale-sensitive because differences in loss magnitudes also drive the optimum of Eq. 5. Finally, the OGR optimization target has a worst-case concern: unexpected negative Generalization $G_i^{(N_0)}$ may satisfy the optimization target in Eq. 4.

Model-Agnostic adaptive alignment (MAPE) addresses these problems by analyzing normalized losses to update the objective weights (Fig. 1.B). We describe **Model-Agnostic alignment** as a

dynamic objective-balancing mechanism compatible with any objectives and any encoder during multi-objective pretraining. Our approach backpropagates gradients to the shared encoder in a scale-invariant manner to mitigate objective imbalance. First, we normalize each training and validation loss by its initial value (constant) to handle the problem of disparities in loss magnitudes: $\overline{\mathcal{L}_i}(t) = \frac{\mathcal{L}_i(t)}{\mathcal{L}_i(0)}$. We then redefine the total Overfitting and total Generalization via these normalized terms to avoid bias weighting due to magnitude disparity:

$$\overline{O}^{(N_0)} := \sum_i^k w_i \overline{O}_i^{(N_0)}, \quad \overline{G}^{(N_0)} := \sum_i^k w_i \overline{G}_i^{(N_0)}. \tag{6}$$

$\overline{O}_i^{(N_0)} = \big(\overline{\mathcal{L}_i^T}(\mathbf{\Theta}_0) - \overline{\mathcal{L}_i^T}(\mathbf{\Theta}_{N_0})\big) - \big(\overline{\mathcal{L}_i^V}(\mathbf{\Theta}_0) - \overline{\mathcal{L}_i^V}(\mathbf{\Theta}_{N_0})\big)$, $\overline{G}_i^{(N_0)} = \overline{\mathcal{L}_i^V}(\mathbf{\Theta}_0) - \overline{\mathcal{L}_i^V}(\mathbf{\Theta}_{N_0})$ are normalized overfitting and normalized generalization gain of objective i. Both terms are computationally efficient and can be determined dynamically during the pretraining process.

Requirements: We seek the optimal weight $\mathbf{w} = [\, w_1 \; w_2 \; \dots w_k]^T \in \mathbb{R}^{k\times 1}$ to: *1) jointly maximize total generalization $\overline{G}^{(N_0)}$ and minimize total overfitting $\overline{O}^{(N_0)}$, 2) bypass trivial solutions that fail to account for inter-objective interference*. A simple optimization approach that satisfies the first requirement would consider minimizing the term $\Big[\Big(\overline{O}^{(N_0)}\Big)^2 + \Big(-\overline{G}^{(N_0)}\Big)\Big]$. However, without a lower bound, this can diverge to negative infinity. Therefore, our approach exponentiates the negative term and includes a weight regularization term (with $\lambda > 0$) to further stabilize the pretraining:

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} \Big[\Big(\overline{O}^{(N_0)}\Big)^2 + \exp\Big(-\overline{G}^{(N_0)}\Big) + \lambda||\mathbf{w} - \mathbf{b}||_2^2\Big], \tag{7}$$

with $\mathbf{b} \in \mathbb{R}^{k\times 1}$ being constant vector to regularize weights $\mathbf{w}$. We set $\mathbf{b} = \mathbf{1} \in \mathbb{R}^{k\times 1}$ to ensure equality in weighting objectives, and anchor our schedule to the contemporary uniform weighting.

Proposition 1 (Optimal Multi-Objective alignment) *Let the objective function with respect to the weight vector* $\mathbf{w}$ *(Eq. 7) be defined as:*

$$f(\mathbf{w}) = (\mathbf{w}^\top \mathbf{O})^2 + \exp(-\mathbf{w}^\top \mathbf{G}) + \lambda\|\mathbf{w} - \mathbf{b}\|_2^2, \tag{8}$$

with $\mathbf{O} = \Big[\overline{O}_1^{(N_0)} \; \overline{O}_2^{(N_0)} \; \dots \; \overline{O}_k^{(N_0)}\Big]^T \in \mathbb{R}^{k\times 1}$, $\mathbf{G} = \Big[\overline{G}_1^{(N_0)} \; \overline{G}_2^{(N_0)} \; \dots \; \overline{G}_k^{(N_0)}\Big]^T \in \mathbb{R}^{k\times 1}$, *regularizer* $\lambda > 0, \mathbf{b} \succ \mathbf{0}$. $f(\mathbf{w})$ *is a strictly convex function with the closed-form optimal minimum* $\mathbf{w}^*$*:*

$$\mathbf{w}^* = \frac{1}{K} W\Big(\frac{K}{2} \cdot \exp\left(-C\right)\Big)\mathbf{A}^{-1}\mathbf{G} + \lambda\mathbf{A}^{-1}\mathbf{b}, \tag{9}$$

where $W(\cdot)$ *is Lambert-W function;* $\mathbf{A} = \mathbf{O}\mathbf{O}^\mathbf{T} + \lambda\mathbf{I}$, $K = \mathbf{G}^T\mathbf{A}^{-1}\mathbf{G}$, $C = \lambda\mathbf{G}^T\mathbf{A}^{-1}\mathbf{b}$ *are computed on-the-fly via losses.*

This optimal weight $\mathbf{w}^*$ solves all three problems of OGR and satisfies our requirements. Details of this closed-form solution and its proof are shown in the Appendix G. Details of MAPE implementation are in Algorithm 1.

4 Experiments

4.1 Experiment Setting

Pretraining setup. We conduct extensive experiments with **one general-purpose time-series foundation model** MANTIS (Feofanov et al., 2026), and **two different EEG foundation models**: CBraMod (Wang et al., 2025) and EEGPT (Wang et al., 2024). CBraMod adopts a full-token pretraining strategy that operates on complete latent representations, similar to CSBrain (Zhou et al., 2026). In contrast, MANTIS and EEGPT employ compact summary tokens during pretraining and use these in downstream tasks rather than the full representations. We pretrain MANTIS for 100 epochs with Cauker synthetic time-series data (Xie et al., 2026), as in their setup. In EEG-specific foundation models, we pretrain CBraMod for 100 epochs, whereas we pretrain EEGPT for 200 epochs, with 20% of TUEG (Obeid & Picone, 2016). More pretraining descritions are in Appendix C.2, Appendix C.3.

Table 1: Balanced Accuracy of **MANTIS** in a **Zero-shot evaluation** using a Random Forest classifier in in-distribution setting (ID) and out-of-distribution setting (OOD). Full metric results are given in Tab. 9 in Appendix I. While single contrastive pretraining remains strong, our aligned multi-objective (**MAPE-CR**) boosts performance on CAP and provides comparable gains on others.

Strategy	Obj.	ID Setting			OOD Setting		
		PCL	CAP	SEDFx	PCL	CAP	SEDFx
Single	C	0.563 ± 0.006	0.640 ± 0.007	0.609 ± 0.002	**0.534 ± 0.006**	0.607 ± 0.002	**0.555 ± 0.005**
	R	0.540 ± 0.005	0.572 ± 0.005	0.530 ± 0.002	0.518 ± 0.007	0.553 ± 0.002	0.428 ± 0.003
Uniform	CR	0.561 ± 0.006	0.663 ± 0.005	0.616 ± 0.002	0.525 ± 0.005	0.638 ± 0.001	0.547 ± 0.005
OGR	CR	0.542 ± 0.009	0.590 ± 0.003	0.551 ± 0.003	0.518 ± 0.004	0.577 ± 0.004	0.458 ± 0.005
MAPE	CR	**0.568 ± 0.008**	**0.669 ± 0.004**	**0.617 ± 0.002**	0.530 ± 0.007	**0.639 ± 0.003**	0.547 ± 0.004

Evaluation. For downstream evaluation, we focus on zero-shot setups for MANTIS, as in the original work (Feofanov et al., 2026), and linear-probe setups for CBraMod and EEGPT. In zero-shot evaluations, we freeze the encoder and fit a RandomForest classifier. We report performance across 5 different seeds on 3 EEG datasets: PCL (Schalk et al., 2004; Cho et al., 2017; Lee et al., 2019), CAP (Terzano et al., 2002), and SEDFx (Kemp et al., 2000), as shown in Tab. 7. In linear probing on EEG models, we freeze the entire encoder, then train the prepended adapter and the linear head, following EEGPT. We apply cross-subject evaluation across 5 downstream datasets: BCIC-IV-2a (Brunner et al., 2008), PhysioNet-MI (Goldberger et al., 2000), KaggleERN (Margaux et al., 2012), Sleep-EDFx (Kemp et al., 2000), and TUAB (Obeid & Picone, 2016), as shown in Tab. 8. Details about downstream setups, datasets, and evaluations are in the Appendix D.

Combinations and configurations. We conduct experiments using two base objectives: *C* (Contrastive), and *R* (Reconstruction). We evaluate Single-objective strategy, and three multi-objective alignment strategies: Uniform, OGR, and MAPE (ours):

- Single: Pretraining with a single objective (*C* or *R*), serving as baselines.
- Uniform: Pretraining with multiple objectives using uniform, static weights $w_i = 1$.
- OGR: Multiple-objective pretraining using Overfitting-to-Generalization (Wang et al., 2020) to dynamically update objective weights. Note that OGR is originally proposed for a supervised setting; we therefore adapt it for self-supervised learning.
- MAPE: Multiple-objective pretraining using Model-Agnostic adaptive alignment (Algorithm 1) to unbiasly update objective weights.

Hence, we have three multi-objective alignments with contrastive and reconstruction: Uniform (*Uni-CR*), *OGR-CR*, and **MAPE-CR**. We further pretrain the tri-combination by adding a time-frequency reconstruction objective (*T*).

4.2 Results

Multi-objective pretraining with MAPE generally performs better than single-objective pretraining. Our experiments, comparing the performance of models pretrained with different single objectives, suggest that the performance of reconstruction (*R*) and contrastive (*C*) pretraining depends on implementation setups (pretraining datasets, model architecture) and downstream scenarios (evaluation). Although single-objective contrastive *(C)* remains a strong baseline in the zero-shot setting (Tab. 1), its gain in linear probing EEG-specific models (Tab. 2) is limited. In contrast, reconstruction pretraining remains performant on EEG-specific models but suffers worse results on MANTIS. Notably, by facilitating both objectives during pretraining, our aligned multi-objective (**MAPE-CR**) aggregates contrastive and reconstruction perspectives to achieve consistently high performance across settings. These results also indicate that joint pretraining with MAPE solves the myopia of single-objective learning.

MAPE alignment facilitates multi-objective pretraining across architectures. In Tab. 1 and Tab. 2, our multi-objective alignment (**MAPE-CR**) consistently provides comparable/superior performance to any single-objective baseline across three models. Moreover, pretraining with **MAPE-CR** is better than multi-objective pretraining with uniform weights (*Uni-CR*) or multi-objective pretraining with

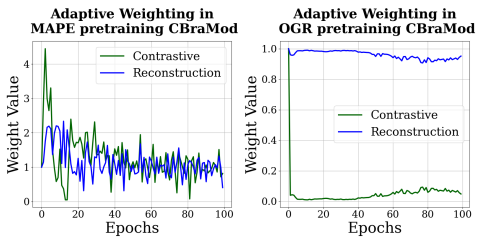
Table 2: Balanced Accuracy of **CBraMod** and **EEGPT** encoders under the **Linear-probe** setup. Full metric results are in Tab. 11, Tab. 12 in Appendix I. While reconstruction pretraining remains strong, our aligned multi-objective (**MAPE-CR**) further boosts performance on BCIC-IV-2a, PhysioNet-MI, and provides comparable gains on other datasets across two architectures.

Model	Strategy	Obj.	BCIC-IV-2a	PhysioNet-MI	KaggleERN	Sleep-EDFx	TUAB
CBraMod	Single	C	0.543 ± 0.085	0.562 ± 0.011	**0.574 ± 0.018**	0.728 ± 0.038	0.812 ± 0.006
		R	0.570 ± 0.083	0.589 ± 0.011	0.531 ± 0.026	0.721 ± 0.035	**0.818 ± 0.004**
	Uniform	CR	0.522 ± 0.083	0.583 ± 0.011	0.571 ± 0.028	**0.733 ± 0.033**	0.813 ± 0.004
	OGR	CR	0.551 ± 0.090	0.586 ± 0.009	0.567 ± 0.020	0.722 ± 0.030	0.807 ± 0.005
	MAPE	CR	**0.578 ± 0.105**	**0.597 ± 0.011**	0.558 ± 0.028	0.726 ± 0.037	0.815 ± 0.006
EEGPT	Single	C	0.427 ± 0.050	0.567 ± 0.007	0.509 ± 0.008	0.681 ± 0.042	0.817 ± 0.006
		R	0.463 ± 0.050	0.568 ± 0.011	**0.577 ± 0.018**	0.688 ± 0.044	0.816 ± 0.005
	Uniform	CR	0.444 ± 0.050	0.573 ± 0.006	0.552 ± 0.027	0.690 ± 0.040	**0.820 ± 0.001**
	OGR	CR	0.477 ± 0.058	**0.578 ± 0.004**	0.557 ± 0.011	0.692 ± 0.044	**0.820 ± 0.007**
	MAPE	CR	**0.485 ± 0.077**	**0.578 ± 0.007**	0.564 ± 0.012	**0.696 ± 0.042**	0.819 ± 0.007

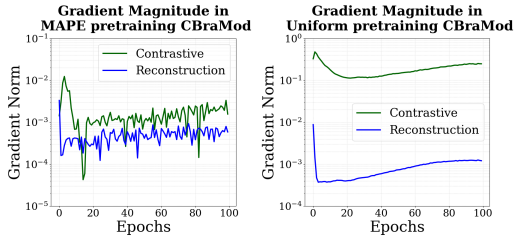
OGR alignment (*OGR-CR*) in almost all comparisons. We further analyze pretraining behaviors of these multi-objective variants to support these findings. As shown in Fig. 3a, OGR alignment biases toward reconstruction throughout pretraining while largely ignoring contrastive, because the large loss-magnitude gap drives the optimum in Eq. 5 toward the smaller-magnitude objective (reconstruction in this case). MAPE, in contrast, adapts weights for both objectives and drives them to one as learning stabilizes in later epochs. Hence, our MAPE nearly balances gradient magnitudes (Fig. 3b), helping it consider both objectives during pretraining instead of ignoring one (as the uniform setting does for reconstruction). These results show that our MAPE is model-agnostic and remains performant across all three diverse architectures. Furthermore, MAPE alignment is proven to account for the objective competition, which remains challenging for other multi-objective pretraining strategies (see more analysis in Appendix H).

MAPE is general and can be extended to any number of objectives. We further validate MAPE alignment in tri-objective scenarios by extending our unified pretraining pipeline to additional Time-frequency reconstruction (*T*), as described in Appendix C.1. Empirical evidence in Tab. 3 again shows that tri-combination **MAPE-CRT** achieves superior or comparable gains to every single baseline in almost all evaluations. Under the tri-objective setting, OGR (*OGR-CRT*) also achieves comparable performance on EEGPT; however, this alignment strategy lacks robustness, as it does not perform well in other scenarios (Tab. 1, Tab. 2). Notably, the slight improvements of tri-objective **MAPE-CRT** over dual-objective **MAPE-CR** show that adding a related objective further boosts performance. These findings demonstrate MAPE's extendability and pave the way for using it with additional objectives.

Limitations. A limitation of our work is the inherently higher cost of pretraining, compared to the single-objective counterparts (Table 4), as multi-objective pretraining requires more objective



a) MAPE and OGR adaptive weighting in pretraining CBraMod.



b) Gradient magnitudes of MAPE and Uniform in pretraining CBraMod.

Figure 3: **MAPE mitigates objective competition** in multi-objective pretraining. **a)** MAPE adapts weights for each objective based on learning behaviors, while OGR biases toward reconstruction. **b)** MAPE eases gradient dominance of contrastive in joint pretraining.

Table 3: Balanced Accuracy of **CBraMod** and **EEGPT** encoders with 3 objectives under the **Linear-probe** setup. Full metric results are in Tab. 11, Tab. 12 in Appendix I. Our aligned tri-objective (**MAPE-CRT**) further improves performance over dual-objective (**MAPE-CR**) in Tab. 2.

Model	Strategy	Obj.	BCIC-IV-2a	PhysioNet-MI	KaggleERN	Sleep-EDFx	TUAB
CBraMod	Single	C	0.543 ± 0.085	0.562 ± 0.011	**0.574 ± 0.018**	0.728 ± 0.038	0.812 ± 0.006
		R	0.570 ± 0.083	0.589 ± 0.011	0.531 ± 0.026	0.721 ± 0.035	**0.818 ± 0.004**
		T	0.542 ± 0.098	0.579 ± 0.009	0.547 ± 0.029	**0.737 ± 0.038**	0.813 ± 0.002
	Uniform	CRT	0.557 ± 0.088	0.568 ± 0.012	0.557 ± 0.032	0.726 ± 0.035	0.810 ± 0.006
	OGR	CRT	0.571 ± 0.093	0.592 ± 0.013	0.567 ± 0.018	0.729 ± 0.031	0.808 ± 0.007
	MAPE	CRT	**0.587 ± 0.105**	**0.604 ± 0.009**	0.567 ± 0.019	0.727 ± 0.035	0.814 ± 0.005
EEGPT	Single	C	0.427 ± 0.050	0.567 ± 0.007	0.509 ± 0.008	0.681 ± 0.042	0.817 ± 0.006
		R	0.463 ± 0.050	0.568 ± 0.011	**0.577 ± 0.018**	0.688 ± 0.044	0.816 ± 0.005
		T	0.472 ± 0.070	0.563 ± 0.014	0.552 ± 0.010	0.687 ± 0.044	0.818 ± 0.006
	Uniform	CRT	0.461 ± 0.054	0.578 ± 0.002	0.528 ± 0.020	0.687 ± 0.039	**0.819 ± 0.004**
	OGR	CRT	**0.517 ± 0.072**	**0.579 ± 0.005**	0.573 ± 0.017	0.688 ± 0.048	0.812 ± 0.003
	MAPE	CRT	0.493 ± 0.071	0.577 ± 0.007	0.564 ± 0.024	**0.692 ± 0.038**	**0.819 ± 0.002**

heads. Note that the computational overhead compared to the single baselines does not affect the downstream evaluation. We find this to be a reasonable trade-off for a well-generalizable encoder.

4.3 Ablations

Closer look: Objective competition hinders multi-objective pretraining. In the linear-probing setup (Tab. 2), it is clear that uniform combination can degrade the performance compared to its corresponding best single-objective variants. Notably, unaligned pretraining (*Uni-CR*) performs similarly to single contrastive pretraining (*C*), validating our observation of contrastive dominance in pretraining CBraMod (Fig. 2b). The objective competition is not harsh on MANTIS (see MANTIS's pretraining curves in Fig. 5); hence, the uniform setting can surpass single counterparts, as shown in zero-shot evaluation in Tab. 1. These results support our hypothesis that competition in a multi-objective pretraining setting suppresses weaker objectives and harms performance.

Table 4: Computational complexity of MAPE-CR, measured on a single NVIDIA A100 GPU.

Model	Objective	Pretraining time (hours)	MFLOPs	#Params (M)
CBraMod	Single-C	13.4	1629	5.3
	Single-R	8.6	857	2.8
	MAPE	15.9	1727	5.6
EEGPT	Single-C	25.1	2067	7.0
	Single-R	15.0	1577	6.6
	MAPE	28.8	2614	10.5

Generalization to other electrophysiological modalities. We further evaluate pretrained MANTIS variants with other electrophysiology data: ECG (MIT-BIH (Moody & Mark, 2001), PTB-XL (Wagner et al., 2020)) and EMG (NinaPro-DB5 (Pizzolato et al., 2017)). As shown in Tab. 10 in Appendix I, our MAPE alignment performs better than or comparably to other baselines across settings.

5 Conclusion

In this work, we focus on multi-objective pretraining time-series foundation models for EEG tasks, particularly studying the phenomenon of competing objective functions. In response, we introduce MAPE (Model-Agnostic adaptive alignment), a theoretically grounded framework (Proposition 1) that efficiently aligns multiple objectives within a single training pipeline by analyzing the pretraining losses of each objective. In addition to our theoretical justifications, we empirically show that MAPE generally improves performance compared to both single-objective pretraining and unaligned multi-objective pretraining strategies. We note that the performance gain of MAPE is only slightly affected by the choice of downstream head and model architecture. As such, at worst, it leads to comparable performance without significant performance degradation. This, in itself, is a benefit, as it can enable easier evaluation and application of novel architectures and/or data, without costly tuning/retraining.

AI USE STATEMENT

In this work, we used generative AI tools to polish text, draft and typeset the textual explanations of our derivations, and assist with coding. We have not used generative AI tools for developing theoretical models or conceptual frameworks, formulating or proving mathematical claims, refining hypotheses, or interpreting results, and the rest of the required disclosure tasks are not applicable to this work. Additionally, we used generative AI tools for plotting images. We have reviewed all AI-assisted work: We checked LLM-generated sentences, including both general text and textual formatting of mathematical proofs; LLM-generated code was verified and tested for correctness by 3 authors. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We provide comprehensive experimental details for reproducibility, including pretraining setups (Appendix C.2); downstream setups, evaluations, and dataset preprocessing (Appendix D). All hyperparameters are fully disclosed. We describe the implementation of our main contribution, MAPE, in Algorithm 1. The complete source code will be released once the paper is accepted.

REFERENCES

U Rajendra Acharya, Shu Lih Oh, Yuki Hagiwara, Jen Hong Tan, and Hojjat Adeli. Deep convolutional neural network for the automated detection and diagnosis of seizure using EEG signals. *Computers in biology and medicine*, 100:270–278, 2018.

Aurelien Appriou, Andrzej Cichocki, and Fabien Lotte. Modern machine-learning algorithms: for classifying cognitive and affective states from electroencephalography signals. *IEEE Systems, Man, and Cybernetics Magazine*, 6(3):29–38, 2020.

Clemens Brunner, Robert Leeb, Gernot Müller-Putz, Alois Schlögl, and Gert Pfurtscheller. BCI competition 2008–graz data set a. *Institute for knowledge discovery (laboratory of brain-computer interfaces), Graz University of Technology*, 16(1-6):1, 2008.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PmLR, 2020.

Hsiang-Yun Sherry Chien, Hanlin Goh, Christopher M Sandino, and Joseph Y Cheng. Maeeg: Masked auto-encoder for EEG representation learning. *arXiv preprint arXiv:2211.02625*, 2022.

Hohyun Cho, Minkyu Ahn, Sangtae Ahn, Moonyoung Kwon, and Sung Chan Jun. EEG datasets for motor imagery brain–computer interface. *GigaScience*, 6(7):gix034, 2017.

Michael Crawshaw. Multi-task learning with deep neural networks: A survey. *arXiv preprint arXiv:2009.09796*, 2020.

Hisham Daoud and Magdy A Bayoumi. Efficient epileptic seizure prediction based on deep learning. *IEEE transactions on biomedical circuits and systems*, 13(5):804–813, 2019.

Federico Del Pup, Andrea Zanola, Louis Fabrice Tshimanga, Alessandra Bertoldo, Livio Finos, and Manfredo Atzori. The role of data partitioning on the performance of EEG-based deep learning models in supervised cross-subject analysis: A preliminary study. *Computers in Biology and Medicine*, 196:110608, 2025.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pp. 4171–4186, 2019.

Berkay Döner, Thorir Mar Ingolfsson, Luca Benini, and Yawei Li. Luna: efficient and topology-agnostic foundation model for EEG signal analysis. *Advances in Neural Information Processing Systems*, 38:70682–70708, 2026.

Bradley J Edelman, Jianjun Meng, Daniel Suma, Claire Zurn, Eric Nagarajan, Bryan S Baxter, Christopher C Cline, and BJSR He. Noninvasive neuroimaging enhances continuous neural tracking for robotic device control. *Science robotics*, 4(31):eaaw6844, 2019.

Yassine El Ouahidi, Jonathan Lys, Philipp Thölke, Nicolas Farrugia, Bastien Pasdeloup, Vincent Gripon, Karim Jerbi, and Giulia Lioi. Reve: A foundation model for EEG-adapting to any setup with large-scale pretraining on 25,000 subjects. *Advances in Neural Information Processing Systems*, 38:22541–22577, 2026.

Vasilii Feofanov, Songkang Wen, Shifeng Xie, Simon Roschmann, Marius Alonso, Hongbo Guo, Romain Ilbert, Malik Tiomoko, Quentin Bouniot, Zeynep Akata, Lujia Pan, Jianfeng Zhang, and Ievgen Redko. Mantis: Lightweight foundation model for time series classification. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=gbJMAjXLZ4.

Jean-Christophe Gagnon-Audet, Kartik Ahuja, Mohammad-Javad Darvishi-Bayazi, Pooneh Mousavi, Guillaume Dumas, and Irina Rish. Woods: Benchmarks for out-of-distribution generalization in time series. *arXiv preprint arXiv:2203.09978*, 2022.

Shanghua Gao, Teddy Koker, Owen Queen, Thomas Hartvigsen, Theodoros Tsiligkaridis, and Marinka Zitnik. Units: A unified multi-task time series model. *Advances in Neural Information Processing Systems*, 37:140589–140631, 2024.

Ary L Goldberger, Luis AN 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: components of a new research resource for complex physiologic signals. *circulation*, 101(23):e215–e220, 2000.

Mononito Goswami, Konrad Szafer, Arjun Choudhry, Yifu Cai, Shuo Li, and Artur Dubrawski. Moment: A family of open time-series foundation models. *arXiv preprint arXiv:2402.03885*, 2024.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284, 2020.

Pierre Guetschel, Sara Ahmadi, and Michael Tangermann. Review of deep representation learning techniques for brain–computer interfaces. *Journal of Neural Engineering*, 21(6):061002, 2024.

Richard N Henson, Daniel G Wakeman, Vladimir Litvak, and Karl J Friston. A parametric empirical bayesian framework for the EEG/MEG inverse problem: generative models for multi-subject and multi-modal integration. *Frontiers in human neuroscience*, 5:76, 2011.

Yu Huang, Junyang Lin, Chang Zhou, Hongxia Yang, and Longbo Huang. Modality competition: What makes joint training of multi-modal network fail in deep learning?(provably). In *International conference on machine learning*, pp. 9226–9259. PMLR, 2022.

Wei-Bang Jiang, Liming Zhao, and Bao-Liang Lu. Large brain model for learning generic representations with tremendous EEG data in BCI. In *International Conference on Learning Representations*, volume 2024, pp. 16405–16426, 2024.

Bob Kemp, Aeilko H Zwinderman, Bert Tuk, Hilbert AC Kamphuisen, and Josefien JL 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.

Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7482–7491, 2018.

Jonghun Kim, Mansu Kim, and Hyunjin Park. Domain aware multi-task pretraining of 3d swin transformer for t1-weighted brain mri. In *Asian Conference on Computer Vision*, pp. 121–141. Springer, 2024.

Demetres Kostas, Stephane Aroca-Ouellette, and Frank Rudzicz. Bendr: Using transformers and a contrastive self-supervised learning task to learn from massive amounts of EEG data. *Frontiers in Human Neuroscience*, 15:653659, 2021.

Min-Ho Lee, O-Yeon Kwon, Yong-Jeong Kim, Hong-Kyung Kim, Young-Eun Lee, John Williamson, Siamac Fazli, and Seong-Whan Lee. EEG dataset and openbmi toolbox for three BCI paradigms: An investigation into BCI illiteracy. *GigaScience*, 8(5):giz002, 2019.

Hong Li, Xingyu Li, Pengbo Hu, Yinuo Lei, Chunxiao Li, and Yi Zhou. Boosting multi-modal model performance with adaptive gradient modulation. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22157–22167. IEEE, 2023.

Fabien Lotte, Marco Congedo, Anatole Lécuyer, Fabrice Lamarche, and Bruno Arnaldi. A review of classification algorithms for EEG-based brain–computer interfaces. *Journal of neural engineering*, 4(2):R1–R13, 2007.

Perrin Margaux, Maby Emmanuel, Daligault Sébastien, Bertrand Olivier, and Mattout Jérémie. Objective and subjective evaluation of online error correction during p300-based spelling. *Advances in Human-Computer Interaction*, 2012(1):578295, 2012. doi: https://doi.org/10.1155/2012/578295. URL https://onlinelibrary.wiley.com/doi/abs/10.1155/2012/578295.

George B Moody and Roger G Mark. The impact of the mit-bih arrhythmia database. *IEEE engineering in medicine and biology magazine*, 20(3):45–50, 2001.

Dominique Nshimyimana, Vitor Fortes Rey, Sungho Suh, Bo Zhou, and Paul Lukowicz. Pim: Physics-informed multi-task pre-training for improving inertial sensor-based human activity recognition. In *2025 International Conference on Activity and Behavior Computing (ABC)*, pp. 1–11. IEEE, 2025.

Iyad Obeid and Joseph Picone. The temple university hospital EEG data corpus. *Frontiers in neuroscience*, 10:196, 2016.

Xiaokang Peng, Yake Wei, Andong Deng, Dong Wang, and Di Hu. Balanced multimodal learning via on-the-fly gradient modulation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8228–8237. IEEE, 2022.

Gert Pfurtscheller, Clemens Brunner, Alois Schlögl, and FH Lopes Da Silva. Mu rhythm (de) synchronization and EEG single-trial classification of different motor imagery tasks. *NeuroImage*, 31(1):153–159, 2006.

Huy Phan, Oliver Y Chén, Minh C Tran, Philipp Koch, Alfred Mertins, and Maarten De Vos. Xsleepnet: Multi-view sequential model for automatic sleep staging. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(9):5903–5915, 2021.

Stefano Pizzolato, Luca Tagliapietra, Matteo Cognolato, Monica Reggiani, Henning Müller, and Manfredo Atzori. Ninapro dataset 5 (double myo armband), October 2017. URL https://doi.org/10.5281/zenodo.1000116.

Mathis Rezzouk, Fabrice Gagnon, Alyson Champagne, Mathieu Roy, Philippe Albouy, Michel-Pierre Coll, and Cem Subakan. Towards generalizable learning models for EEG-based identification of pain perception. In *2025 IEEE 35th International Workshop on Machine Learning for Signal Processing (MLSP)*, pp. 1–6. IEEE, 2025.

Aaqib Saeed, Tanir Ozcelebi, and Johan Lukkien. Multi-task self-supervised learning for human activity detection. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 3(2):1–30, 2019.

Gerwin Schalk, Dennis J McFarland, Thilo Hinterberger, Niels Birbaumer, and Jonathan R Wolpaw. BCI2000: a general-purpose brain-computer interface (BCI) system. *IEEE Transactions on biomedical engineering*, 51(6):1034–1043, 2004.

Congzhong Sun and Chaozhou Mou. Survey on the research direction of EEG-based signal processing. *Frontiers in Neuroscience*, 17:1203059, 2023.

Mario Giovanni Terzano, Liborio Parrino, Arianna Smerieri, Ronald Chervin, Sudhansu Chokroverty, Christian Guilleminault, Max Hirshkowitz, Mark Mahowald, Harvey Moldofsky, Agostino Rosa, et al. Atlas, rules, and recording techniques for the scoring of cyclic alternating pattern (cap) in human sleep. *Sleep medicine*, 3(2):187–199, 2002.

Patrick Wagner, Nils Strodthoff, Ralf-Dieter Bousseljot, Dieter Kreiseler, Fatima I Lunze, Wojciech Samek, and Tobias Schaeffter. Ptb-xl, a large publicly available electrocardiography dataset. *Scientific data*, 7(1):154, 2020.

Guangyu Wang, Wenchao Liu, Yuhong He, Cong Xu, Lin Ma, and Haifeng Li. EEGPT: Pre-trained transformer for universal and reliable representation of EEG signals. *Advances in Neural Information Processing Systems*, 37:39249–39280, 2024.

Jiquan Wang, Sha Zhao, Zhiling Luo, Yangxuan Zhou, Haiteng Jiang, Shijian Li, Tao Li, and Gang Pan. Cbramod: A criss-cross brain foundation model for EEG decoding. In *International conference on learning representations*, volume 2025, pp. 75310–75346, 2025.

Weiyao Wang, Du Tran, and Matt Feiszli. What makes training multi-modal classification networks hard? In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12692–12702. IEEE, 2020.

Shifeng Xie, Vasilii Feofanov, Jianfeng Zhang, Themis Palpanas, and Ievgen Redko. Cauker: Classification time series foundation models can be pretrained on synthetic data. In *International Conference on Learning Representations*, volume 2026, pp. 68606–68634, 2026.

Chaoqi Yang, M Westover, and Jimeng Sun. Biot: Biosignal transformer for cross-data learning in the wild. *Advances in Neural Information Processing Systems*, 36:78240–78260, 2023.

Jin Yin, Aiping Liu, Chang Li, Ruobing Qian, and Xun Chen. A gan guided parallel cnn and transformer network for EEG denoising. *IEEE Journal of Biomedical and Health Informatics*, 29 (6):3930–3941, 2023.

Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. iBot: Image bert pre-training with online tokenizer. *arXiv preprint arXiv:2111.07832*, 2021.

Yuchen Zhou, Jiamin Wu, Zichen Ren, Zhouheng Yao, Weiheng Lu, Kunyu Peng, Qihao Zheng, Chunfeng Song, Wanli Ouyang, and Chao Gou. Csbrain: A cross-scale spatiotemporal brain foundation model for EEG decoding. *Advances in Neural Information Processing Systems*, 38: 87150–87195, 2026.

A MATHS NOTATIONS

Table 5: Summary of mathematical notations used in this paper.

Category	Symbol	Space	Meaning
Parameters	$\mathbf{\Theta}$	–	General model parameters
	$\mathbf{\Theta}_{\text{e}}$	–	Encoder's parameters
	$\mathbf{\Theta}_{\text{r}}$	–	Reconstruction block's parameters
	$\mathbf{\Theta}_{\text{c}}$	–	Contrastive block's parameters
	$\mathbf{\Theta}_{\text{tf}}$	–	Time-frequency decoder's parameters
General	B	$\mathbb{N}$	Size of a mini pretraining batch
	C	$\mathbb{N}$	Number of EEG channels of a sample
	t	$\mathbb{N}$	Number of time points in a raw sample
	N	$\mathbb{N}$	Number of EEG patches (sequence length)
	T	$\mathbb{N}$	Patch size (time points in an EEG patch)
	D	$\mathbb{N}$	Embedding dimension of latents after encoder
	F	$\mathbb{N}$	Number of frequency bins in Wavelet transformation
	k	$\mathbb{N}$	Number of pretraining objectives
Tensors	$\mathbf{x}$	$\mathbb{R}^{C\times N\times T}$	Patched EEG sample
	$\mathbf{z}$	$\mathbb{R}^{N\times D}$	Encoded latent of input $\mathbf{x}$
	$\hat{\mathbf{x}}$	$\mathbb{R}^{C\times N\times T}$	Reconstructed sample from latent $\mathbf{z}$
	$\mathbf{z}^{(1)},\ \mathbf{z}^{(2)}$	$\mathbb{R}^{N\times D}$	Encoded latents of 2 augmentations
	$\mathbf{h}^{(1)},\ \mathbf{h}^{(2)}$	$\mathbb{R}^{N\times D}$	Corresponding contrastive projections of $\mathbf{z}^{(1)},\ \mathbf{z}^{(2)}$
	$\mathbf{x}_{\text{tf}}$	$\mathbb{R}^{C\times N\times F\times T}$	Wavelet transformation of input $\mathbf{x}$
	$\hat{\mathbf{x}}_{\text{tf}}$	$\mathbb{R}^{C\times N\times F\times T}$	Reconstructed time-frequency representation from latent $\mathbf{z}$
	$\mathbf{g}_i$	–	Gradient of objective i w.r.t the shared encoder $\mathbf{\Theta}_{\text{e}}$
Objectives	$\mathcal{L}_r$	$\mathbb{R}_{\geq 0}$	Reconstruction loss
	$\mathcal{L}_c$	$\mathbb{R}_{\geq 0}$	Contrastive loss
	$\mathcal{L}_{\text{tf}}$	$\mathbb{R}_{\geq 0}$	Time-frequency reconstruction loss
	$\mathcal{L}_i^T$	$\mathbb{R}_{\geq 0}$	Training loss of objective i
	$\mathcal{L}_i^V$	$\mathbb{R}_{\geq 0}$	Validation loss of objective i
	$\overline{\mathcal{L}_i^T}$	$\mathbb{R}_{\geq 0}$	Normalized training loss of objective i
	$\overline{\mathcal{L}_i^V}$	$\mathbb{R}_{\geq 0}$	Normalized validation loss of objective i
Alignment	N_0	$\mathbb{N}$	Number of training iterations per update
	M	$\mathbb{N}$	Number of updates
	w_i	$\mathbb{R}_{\geq 0}$	Weight of objective i
	$\mathbf{w}$	$\mathbb{R}^{k\times 1}$	Vector of objectives' weights
	O_i	$\mathbb{R}$	Original Overfitting term of objective i
	G_i	$\mathbb{R}$	Original Generalization term of objective i
	O	$\mathbb{R}$	Original total Overfitting term
	G	$\mathbb{R}$	Original total Generalization term
	$\overline{O}_i$	$\mathbb{R}$	Normalized Overfitting term of objective i
	$\overline{G}_i$	$\mathbb{R}$	Normalized Generalization term of objective i
	$\overline{O}$	$\mathbb{R}$	Normalized total Overfitting term
	$\overline{G}$	$\mathbb{R}$	Normalized total Generalization term
	$\mathbf{O}$	$\mathbb{R}^{k\times 1}$	Vector of normalized Overfitting terms
	$\mathbf{G}$	$\mathbb{R}^{k\times 1}$	Vector of normalized Generalization terms
	λ	$\mathbb{R}_{\geq 0}$	Regularizer of optimization function Eq. 7
	$\mathbf{b}$	$\mathbb{R}_{>0}$	Weight anchor of optimization function Eq. 7
	$\mathbf{A}$	$\mathbb{R}^{k\times k}$	Intermediate constant matrix, defined in Eq. 9
	K	$\mathbb{R}$	Intermediate constant, defined in Eq. 9
	C	$\mathbb{R}$	Intermediate constant, defined in Eq. 9

B OBJECTIVE COMPETITION

In this section, we empirically demonstrate that objective competition happens during joint pretraining of time-series foundation models.

Evidence from original EEGPT. EEGPT is an occasional EEG foundation model using multi-objective pretraining. Unlike other architectures that use only masked autoencoding, EEGPT introduces a unique **alignment loss** specific to its tasking scheme and architecture. This alignment objective uses an EMA encoder and computes the Mean Squared Error between 2-branch latents. Note that this alignment objective differs from the described contrastive objective in architecture, target, and loss function. We advise readers to view EEGPT (Wang et al., 2024) for more details. In this section, we pretrain EEGPT with their datasets, training pipeline, and masking scheme, with the same setup as theirs. As shown in Fig. 4, the loss curves diverge sharply: reconstruction loss decreases slightly, while alignment loss surges at the beginning. We attribute this to a disparity in pretraining targets, such that their divergent gradients pull the encoder in different directions. As such, objective competition arises, inhibiting the pretraining process.



Figure 4: In pretraining EEGPT **(their original pipeline)**, the alignment loss surges in the first epochs, then gradually decreases. This highlights a strong gradient interference between the alignment and reconstruction objectives.

Evidence from UniTS. UniTS (Gao et al., 2024) proposes a unified pretraining pipeline for time-series signals with masked reconstruction and generative self-supervised pretraining. In their downstream experiments on the FaceDetection EEG dataset (Henson et al., 2011), only reconstruction pretraining achieves 70.5% downstream accuracy, whereas pretraining with both reconstruction and generative objectives achieves only 65.4%. This result is consistent with our hypothesis that joint pretraining suffers from objective competition, which inhibits performance.

Evidence from our pretraining pipeline. As shown in Fig. 5a, the multi-objective loss curve is not smooth due to strong competition from different heads, which drives the shared encoder in different directions. Because of large magnitude differences (Fig. 5b) and zero/negative similarities between objectives (Fig. 5c), the dominant objective can strongly drive the encoder while ignoring the others. As such, multi-objective pretraining is inherently difficult and relies on a complex, hidden optimization mechanism. Nevertheless, we show that by adaptively adjusting the objective weights, MAPE eases objective dominance. Details of adaptive weightings are in Appendix H.

C Pretraining pipeline & Setup

C.1 Additional pretraining objective: Time-frequency reconstruction

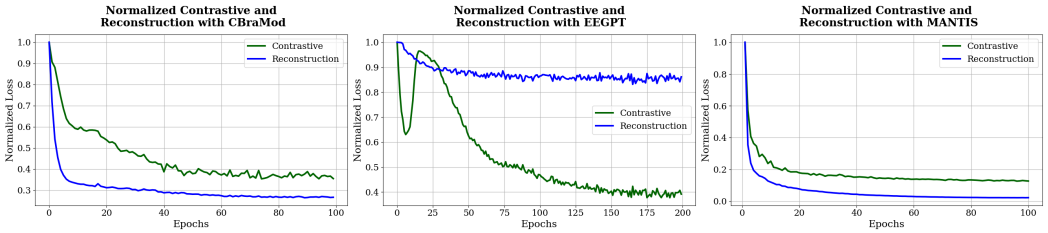
Reconstructing time-frequency representations is mechanistically grounded, since clinical diagnosis and neurophysiological analysis use them to understand spectral dynamics for classification, such as tracking μ-rhythm (8 - 13 Hz) (de)synchronization in motor imagery (Pfurtscheller et al., 2006).

In this objective, we use the Continuous Wavelet transform to convert time-series input into a time-frequency representation: $\mathbf{x}_{\text{tf}} = \text{CWT}(\mathbf{x}) \in \mathbb{R}^{C\times N\times F\times T}$, as the label for reconstruction, with F being the frequency bins for each time point. We then mask a portion of the time-series input $\mathbf{x}$ before encoding this augmentation to latents $\mathbf{z}$, as described in Sec. 3.1. A lightweight convolutional time-frequency reconstructor predicts original Wavelet representation: $\hat{\mathbf{x}}_{\text{tf}} = \mathbf{\Theta}_{\text{tf}}(\mathbf{z})$. The time-frequency reconstruction loss forces the model to learn time-frequency semantic meanings:

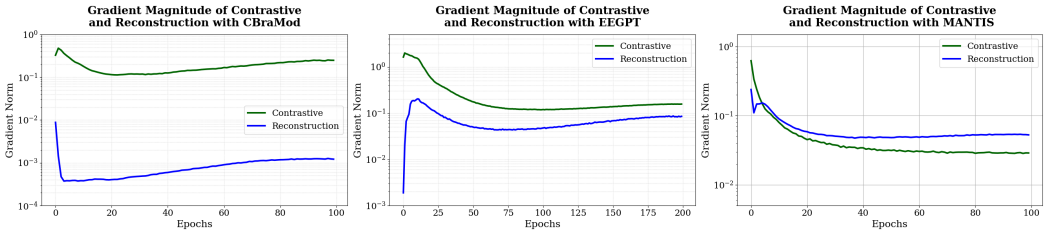
$$\mathcal{L}_{\text{tf}} = ||\mathbf{x}_{\text{tf}} - \hat{\mathbf{x}}_{\text{tf}}||_2^2. \tag{10}$$

C.2 Pretraining setup

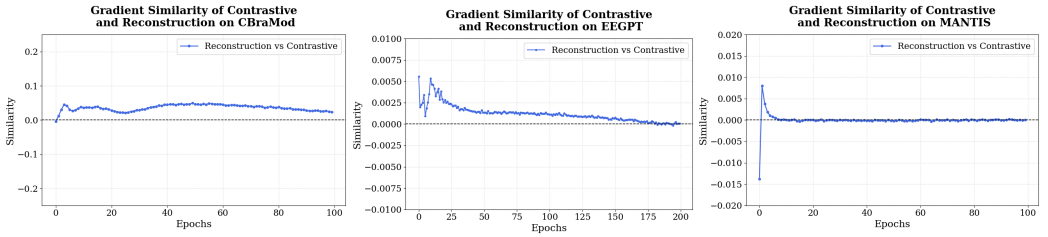
Common settings. We train models with GPU A100, with 12 workers. We select the best epoch for each variant, defined by the lowest pretraining validation loss, for downstreaming. We use the AdamW optimizer for all pretraining. Because the optimal weights of OGR may be negative, which can ruin the pretraining process, we apply weight normalization for aligned pretraining, as indicated in Appendix H.



a) Loss instabilities in uniform multi-objective pretraining of three encoders.



b) Reconstruction gradient is occluded in pretraining CBraMod and EEGPT, while surpassing the contrastive counterpart in pretraining MANTIS.



c) Across three encoders, gradients from different heads are nearly orthogonal.

Figure 5: Analysis of multi-objective pretraining dynamics in uniform settings. (a) Loss instability. (b) Gradient occlusion. (c) Gradient orthogonality.

C.2.1 MANTIS

We strictly follow the pretraining setup of MANTIS (Feofanov et al., 2026). For the contrastive objective, we use random crop augmentation and apply a cross-entropy contrastive loss between two views to update encoder parameters. MANTIS tokenizes a univariate series of length $t{=}512$ into $N{=}32$ non-overlapping patches of 16 samples. Two dilated-free 1D convolutions are applied to the instance-normalized series and its first difference; each feature map is passed through LayerNorm and mean-pooled inside every patch. In parallel, multi-scale scalar encoders (with hidden size 32) encode the per-patch mean and standard deviation. These four descriptors are concatenated and linearly projected to one token per patch. A learnable CLS token is prepended. The encoder is a 6-layer Transformer with 8 heads (head dim 32), MLP width 512, RMSNorm, time-aware RoPE, SwiGLU, and dropout 0.1. This encoder yields latent $\mathbf{z} \in \mathbb{R}^{(N+1)\times D}$. We refer readers to the original MANTIS papers for further architectural details.

We additionally build **masked reconstruction** for MANTIS with a mask ratio of 50%. To construct the reconstruction objective, we use a Transformer-based decoder to reconstruct masked positions. After tokenization, we replace 50% of the N patch tokens with a shared learnable mask token (the CLS token is never masked). The encoder still produces $\mathbf{z} \in \mathbb{R}^{(N+1)\times D}$. A linear map embeds these tokens to dimension 128; we then use a 3-layer decoder Transformer (8 heads, MLP width 256, same RoPE/RMSNorm/SwiGLU block as the encoder) to contextualize visible and masked positions jointly. Dropping the CLS output, a LayerNorm + linear head maps each of the N tokens to a vector of dimension 16 (t/N). We reshape the raw series into patches and apply patch-wise

z-score normalization as the reconstruction target. The reconstruction loss is MSE averaged only over masked patches, following the description in Sec. 3.1.

Other settings: We set the learning rate to 2e-4 in pretraining all variants. In MAPE adaptive alignment for MANTIS, we set regularizer λ to 1e-2.

C.2.2 CBraMod and EEGPT

Pretraining setups. We apply the same augmentation strategy for both EEG-specific encoders. In addition to temporal masking, each sample undergoes one of 4 extra augmentations with equal probability: jittering, interpolation, magnitude scaling, and frequency masking, making the model insensitive to potential data degradation during downstream processing. In detail, jittering helps the model be less affected by noise. Interpolation makes the model robust to interpolated data, which is common in downstream datasets with different sampling rates. Frequency masking helps the model remain robust to downstream sample collapse, while magnitude scaling prevents model collapse or unlearning when it encounters samples from different data domains. Following EEGPT, we extract contrastive views via an EMA (exponential moving average) encoder, which we treat as the target branch. We regard the trainable encoder as the online branch. By anchoring to online projection $\mathbf{h}^{(1)}$, we have contrastive loss as defined in Eq. 2. To avoid bias towards one branch, we use an averaged symmetric contrastive loss:

$$\mathcal{L}_c = -\frac{1}{2B}\sum_{i=1}^{B}\Big(\log\frac{\exp\Big(\mathrm{sim}(\mathbf{h}_i^{(1)},\mathbf{h}_i^{(2)})/\tau\Big)}{\sum_{j=1}^{B}\exp\Big(\mathrm{sim}(\mathbf{h}_i^{(1)},\mathbf{h}_j^{(2)})/\tau\Big)} + \log\frac{\exp\Big(\mathrm{sim}(\mathbf{h}_i^{(2)},\mathbf{h}_i^{(1)})/\tau\Big)}{\sum_{j=1}^{B}\exp\Big(\mathrm{sim}(\mathbf{h}_i^{(2)},\mathbf{h}_j^{(1)})/\tau\Big)}\Big). \tag{11}$$

Details about common settings are indicated in Tab. 6.

Pretraining with CBraMod encoder. Following setup and notation in Sec. 3.1, Appendix C.1, details about the pretraining pipeline of small CBraMod encoder:

- **Augmentation:** We apply 40% temporal masking on online branch and 10% temporal masking on EMA branch as follows (Grill et al., 2020).
- **Encoder:** 6 layers, input embedding size: 200, output embedding size: $D = 200$, forward dimension: 800, number of attention heads: 8.
- **Reconstructor:** Following CBraMod (Wang et al., 2025), we use a simple 2-layer MLP with input dimension, output dimension are 200, hidden dimension is 800.
- **Others:** Contrastive predictor and contrastive projector are 2-layer MLP, following configurations in Tab. 6. Regularizer λ is set to $1e-2$.

Pretraining with EEGPT encoder. EEGPT flavors pretraining with compact summary tokens, rather than full representation, and takes these tokens to downstream tasks ($\mathbf{z} \in \mathbb{R}^{N\times E\times D}$ with N, E, D being the number of patches, number of summary tokens ($E \leq 4$), and the embedding size).Following setup and notation in Sec. 3.1, Appendix C.1, details about the pretraining pipeline of small EEGPT encoder:

- **Augmentation:** We apply 40% temporal masking for all patched EEG samples in both branches.
- **Encoder:** 4 layers, embedding size: $D = 256$, number of attention heads: 8, number of summary token: 1.
- **Predictor:** The built-in EEGPT predictor is kept to facilitate reconstruction and next patch prediction objectives. Predictor's embedding size: $D = 256$, number of layers: 2, number of attention heads: 8.
- **Reconstructor:** The built-in EEGPT reconstructor has 2 transformer-based layers, with an embedding size and number of attention heads equal to those of the encoder.
- **Others:** Contrastive predictor and projector are 2-layer MLP, following configurations in Tab. 6. Regularizer λ is set to $5e-2$.

Table 6: Common hyperparameters for MAPE pretraining on CBraMod and EEGPT.

Block	Component	Hyperparameters	Settings
Input	EEG sample	Channels	19
		Time points	3200 (16s @ 200Hz)
		Patch size	200
		Sequence length	16 patches
	Augmentation	Temporal Masking	40%
		Prob. extra augment.	25%
		Magnitude scaling	Randomly from 0.5 to 2 times
		Jittering	$\mathcal{N}(0, 0.1)$
		Interpolation	20% data points
		Frequency masking	20% frequency
Contrastive	Momentum Encoder	EMA momentum	Cosine ramp $0.996 \rightarrow 1$
	Projector	Input dimension	D (Encoder embedding)
		Hidden dimension	$2D$
		Out dimension	D
		Dropout	0.2
	Predictor	Input dimension	D (Encoder embedding)
		Hidden dimension	$D \div 2$
		Out dimension	D
		Dropout	0.1
Pretraining	Optimization	Batch size	64
		Optimizer	AdamW
		Learning rate	1×10^{-4}
		LR schedule	Warmup + Cosine decay
		Warmup steps	20%
		Weight decay	0.1 (cosine schedule)
		Adam β	(0.9, 0.999)
		Adam ϵ	1×10^{-8}
		Gradient clipping	1.0
		Seed	7

C.3 Pretraining dataset & Preprocessing

C.3.1 MANTIS

Pretraining dataset. As in the original paper, we pretrained all MANTIS variants on 200,000 synthetic samples, generated by Cauker (Xie et al., 2026). CauKer is a causal-kernel data generation pipeline designed for time-series classification tasks with two main objectives: (1) to generate sequences that exhibit common time-series patterns, such as periodicity, seasonality, and trends; and (2) to generate a set of time series with meaningful clustering structures. The pipeline begins by sampling K kernel functions from a kernel bank, $\mathcal{K} = \{\kappa_i(t, t')\}_{i=1}^{n_\mathcal{K}}$. The sampled kernels κ^* are then randomly composed into a single kernel using additive and multiplicative operations: $\kappa^* = \kappa_1(t, t') \star_1 ... \star_{K-1} \kappa_K(t, t')$ where $\{\star_i\}_{i=1}^{K} \sim \{\times, +\}$. This process is repeated M times, and M mean functions are sampled from a mean-function bank, $\mathcal{M} = \{\mu_i(t)\}_{i=1}^{n_\mathcal{M}}$, to construct M root nodes $\{\mathcal{GP}(\mu_i, \kappa_i^*)\}_{i=1}^{M}$ by the Gaussian Process (GP) prior. In the final stage, causal graph propagation is performed. The pipeline randomly generates a Directed Acyclic Graph (DAG), denoted as $(\mathcal{V}, \mathcal{E})$, with $|\mathcal{E}| = E$, where the M root nodes satisfy $M < |\mathcal{V}|$. Each edge is associated with an activation function in set of E activation functions sampled from activation function banks $\mathcal{A} = \{\sigma_i(t)\}_{i=1}^{n_\mathcal{A}}$. Each vertice is associated with 1 chosen activation functions by $\phi : \mathcal{V} \rightarrow \{\sigma_i\}_{i=1}^{E}$. For each of the M root nodes, a time series is sampled from $\mathcal{GP}(\mu_i, \kappa_i^*)$ over a grid of length L. For each non-root node, the values at time step t from all of its parent nodes are concatenated into a vector e. This vector is then passed through the corresponding activation function after applying a linear combination $t_{v_j} = \phi(v_j)(We + b)$ (where $W, b \sim \mathcal{N}(0, 1)$) to obtain the value of the node at time t. Finally, the generated time series are randomly selected from the vertices. The entire generation process is repeated until the desired total number N of sequences is obtained.

Data generation. We use the publicly released CauKer implementation to generate $N = 200,000$ univariate series of length $L = 512$, with the maximum number of kernels per composite GP set to 4, the maximum number of parents per DAG node set to 6, and the maximum number of graph nodes set to 18. All other settings use their defaults, and we apply no additional preprocessing beyond the pipeline's built-in length interpolation.

C.3.2 CBraMod and EEGPT

Pretraining dataset. We use 20% of TUEG (Obeid & Picone, 2016), a large public dataset with a duration of 27,062 hours of EEG recording. The dataset is recorded using 10-20 system. Before processing, we exclude two downstream tasks, TUAB and TUEV (Obeid & Picone, 2016), to avoid data leakage. After preprocessing, the dataset has 1,200 hours in total.

Preprocessing. Our preprocessing steps follow prior works (Wang et al., 2025; Zhou et al., 2026) to eliminate noise and inconsistency in recordings. First, the recording scheme of TUEG may be prone to contamination, especially at the beginning and end of the recording. Therefore, the one-minute beginnings and endings of all recordings are discarded to remove low-quality data. All recordings were recorded with a 10-20 standard system with 40 common channels. Second, to remove noise from degraded channels without missing important data sources, we only selected 19 most common channels (Fp1, Fp2, F7, F3, Fz, F4, F8, T3, C3, Cz, C4, T4, T5, P3, Pz, P4, T6, O1, O2). Third, because EEG signals focus on a small range of frequencies, EEG recordings were then filtered with a band-pass filter (0.3Hz - 75Hz) to remove noise, along with a notch filter at 60Hz to discard the power line noise. Long-duration samples potentially capture the long-term dependency of EEG signals. Therefore, we resampled all recordings to 200Hz and cut them into nonoverlapping 16-second windows to create uniformly long-term samples. Raw EEG is often disturbed by non-brain physiological activities such as ECG, EOG, and EMG (Yin et al., 2023), which can create "excessive pulses" that mislead models. For robust and stable pretraining, we finally eliminated samples with a maximum voltage amplitude exceeding $100\mu V$, before normalizing all samples to $[-1 : 1]$.

D Downstream setup, dataset & evaluation

Common settings. We train models with GPU A100, with 12 workers. For binary tasks, we report Balanced Accuracy, AUROC, and AUC-PR; for multi-class tasks, Balanced Accuracy, Cohen's kappa, and Weighted F1. Our baselines include single-objective, uniform-objective, and multi-objective pretraining with OGR alignment, using the MANTIS, EEGPT, and CBraMod backbones (Sec. 4.1). For a fair comparison, we used the same settings, including scheduler, augmentation, batch size, etc, for all variants.

D.1 MANTIS

Downstream setup. The pretrained encoder is frozen. We linearly resample each input channel to length 512 (MANTIS's input length), then apply per-channel instance standardization. During training, we forward each channel, then concatenate several embeddings for multivariate inputs. We extract embeddings and fit a RandomForest classifier with 200 trees, unlimited depth, and balanced sample weights, following MANTIS.

ID & OOD evaluation. We evaluate both in-distribution (ID) and domain-shift (OOD) splits, matching the MANTIS construction. Under ID, we pool all domains and split them into train, validation, and test at $63.75\%, 11.25\%, 25\%$. Under OOD, train, validation, and test use disjoint domains; the split for each dataset is scripted below. We run every configuration on five fixed seeds $0, 1, 2, 3, 4$ with no seed selection; we report the average and standard deviation over these seeds.

Downstream datasets. Table 7 summarizes the physiological corpora for evaluating MANTIS. EEG datasets follow the WOODS extracts (Gagnon-Audet et al., 2022). We further evaluate with ECG and EMG datasets in the ablation 4.3.

- **PCL** is the WOODS (Gagnon-Audet et al., 2022) motor-imagery dataset formed by aligning three public BCI datasets (Cho17 (Cho et al., 2017), Schalk04 (Schalk et al., 2004), and Lee19 (Lee et al., 2019)) under a common 48-channel 10-10 montage. The dataset keeps only left-hand and right-hand imagery (2-class). Recordings are band-pass filtered at 4-32Hz,

Table 7: Overview of downstream physiological datasets for MANTIS.

Task	Dataset	Modality	Rate	#Channels	Duration	Labels
Motor imagery	PCL	EEG	250 Hz	48	3s	2-class
Sleep staging	CAP	EEG	100 Hz	19	30s	6-class
Sleep staging	SEDFx	EEG	100 Hz	4	30s	6-class
Arrhythmia	MIT-BIH	ECG	360 Hz	2	1s	5-class
Diagnostic ECG	PTB-XL	ECG	100 Hz	12	10s	5-class
Hand gesture	NinaPro-DB5	EMG	200 Hz	16	2s	23-class

resampled to 250Hz, and cropped to a shared 3-second window (750 samples) on the 48 channels. The full set comprises 22,598 recordings from 215 subjects. For OOD evaluation, the three collection protocols are disjoint domains: train on Cho17, validate on Schalk04, and test on Lee19.

- **CAP** (Terzano et al., 2002) is a WOODS sleep-staging dataset with samples collected from 5 machines, measured with 19 shared channels. Experts score each 30-second window into six stages (Awake, Non-REM 1, Non-REM 2, Non-REM 3, Non-REM 4, and REM). Recordings are resampled to 100Hz and band-pass filtered at 0.3-30Hz. The set contains 40,390 recordings from 41 participants and recorded overnight. In ID setting, we use the same pooled stratified split as above. In OOD setting, we train on Machines A, B, and D; validate on Machine C; and tests on Machine E.

- **SEDFx** (Kemp et al., 2000) is the WOODS dataset, created to study age demographic shifts in sleep-staging tasks. This dataset contains four age groups: 20-40, 40-60, 60-80, 80-100. Each 30-second recording at 100Hz is labeled with the same six stages (Awake, Non-REM 1, Non-REM 2, Non-REM 3, Non-REM 4, and REM). We retain 4 common channels (Fpz-Cz, Pz-Oz, horizontal EOG, submental EMG) after a 0.3-30Hz band-pass. The set contains 238,712 recordings from 100 subjects, each was recorded 2 nights. ID follows the pooled stratified protocol; OOD trains on ages 20-60, validates on 60-80, and tests on 80-100.

- **MIT-BIH** (Moody & Mark, 2001) is arrhythmia database was created to support automated arrhythmia detection from ECG signals for reliable cardiac monitoring. It contains 48 half-hour two-channel ECG recordings from 47 subjects, sampled at 360 Hz. The task is heartbeat classification into Normal, Supraventricular, Ventricular, Fusion, and Unknown per record, totaling over 109,000 annotated recordings. Preprocessing includes: (1) R-peak detection via the dataset annotations; (2) segmentation into fixed-length 360 sample-per-second windows centered on each R-peak. For OOD evaluation we follow an inter-patient protocol: training uses AAMI DS1 except four held-out records (102, 104, 107 and 217); validation uses those four DS1 records; and testing uses DS2.

- **PTB-XL** (Wagner et al., 2020) was created to evaluate ECG diagnosis prediction. It contains 21,837 12-lead ECG recordings from 18,885 patients, each 10 seconds long, available at 100 Hz and 500 Hz. The task is 5-label classification into diagnostic categories (Normal, Myocardial Infarction, ST/T Changes, Conduction Disturbance, Hypertrophy). Preprocessing includes: (1) segmentation using fixed 10-second full-lead windows; and (2) resampling 100 Hz with z-score normalization. For OOD evaluation we used: folds 1–8 for training, fold 9 for validation, and fold 10 for testing.

- **NinaPro-DB5** (Pizzolato et al., 2017) supports EMG-based hand gesture recognition for prosthetic control and human-machine interfaces. It contains surface EMG recordings from 10 intact subjects, recorded using two Myo armbands (16 channels total) at a sampling rate of 200 Hz. The task is gesture classification across 52 hand/wrist movements plus a rest state (preprocessed to 23 labels uniquely), each repeated 6 times per subject, grouped into exercises (A: fingers, B: wrist/functional, C: grasps). Preprocessing includes: (1) segmentation by sliding 2-second windows with 1-second overlap) to capture muscle activation patterns; and (2) normalization (per-channel z-score) with raw windows. For OOD evaluation we use the leave-subjects-out split: subjects S1–S7 for training, S8 for validation, and S9–S10 for testing.

D.2 CBraMod and EEGPT

Downstream setups. All downstream experiments use a global batch size of 64 with the seed set to 7. We use AdamW with a maximum learning rate of $1e-3$ for the classification head and $1e-4$ for the patch embedder, with corresponding weight decay of $1e-4$, following a OneCycle schedule with 10% warm-up. Because the channel counts differ between the pretraining dataset and downstream datasets, we create a linear mapping to transfer the dataset's channels to the predefined channels, similar to EEGPT. We employ a 2-layer MLP projection to reduce latent dimensionality for classification. The latent tensor (N, E, D), where N, E, D denote the number of patches, summary tokens, and embedding dimension, is flattened to $(N, E \times D)$ and projected into a $(N, 16)$ hidden space. We then flatten this and map it to the class space via a second MLP. Note that, originally, CBraMod (Wang et al., 2025) flattens the full $N \times C \times D$ tensor before applying a fully-connected MLP; however, this results in an extreme head exceeding 10 million parameters—multiple times the size of the encoder itself. This risks overfitting and may confound the evaluation of the encoder's learned representations; hence, we apply this same 2-MLP protocol to CBraMod.

Subject-wise cross-validation. We observe that subject partitioning during evaluation significantly affects EEG model performance, leading to substantial variation across selections (Del Pup et al., 2025; Rezzouk et al., 2025). We constructed a **subject-wise cross-evaluation** scheme that splits all subjects into *k* folds for the validation or test set to ensure fair assessment (Tab. 8). For instance, we perform *k* fine-tunings, using the remaining folds for training and validation and holding out one fold as the test set. The "best validated checkpoint", defined as the training checkpoint with the highest monitoring metric on the validation set (Cohen's kappa for multiclassification and AUROC for binary classification), is tested once after each fine-tuning phase. The reported performance is the average across all folds. We describe each dataset's train-validation-test splits in detail below.

Table 8: Overview of EEG datasets and their corresponding BCI tasks.

BCI Task	Dataset	Rate	#Channels	Duration	#Folds (k)	Labels
Motor Imagery	BCIC-IV-2a	250 Hz	22	4s	9	4-class
	PhysioNet-MI	160 Hz	64	4s	4	4-class
Event-related Potentials	KaggleERN	200 Hz	56	2s	5	2-class
Sleep Staging	Sleep-EDFx	100 Hz	2	30s	5	5-class
Abnormal EEG Detection	TUAB	250 Hz	16	10s	4	2-class

Downstream BCI Tasks & Preprocessing. As shown in Table 8, we experimented on **4 tasks** using **5 different datasets** to comprehensively evaluate our method. As with the pretraining TUEG dataset, we resample all samples to 200 Hz and scale them to the range $[-1, 1]$ to adapt the pretrained model effectively and develop a universal downstream framework.

- **BCIC-IV-2a:** Data from 9 participants completing four distinct motor imagery tasks form BCIC-IV-2a. The activities involve motor imagery of 4: the mouth (Class 4), feet (Class 3), right hand (Class 2), and left hand (Class 1). Every participant completes two sessions, each with 288 trials, on separate days. We use a 4-second window sample (800 data points), a low-pass filter between 0 and 38 Hz, and a sampling rate of 200 Hz. For evaluation, we use a cross-validation procedure called leave-one-subject-out (LOSO). Each of the nine fine-tunings uses a separate subject as the test set, while the remaining eight subjects are used as the training set.
- **PhysioNet-MI:** PhysioNet-MI is a motor imagery dataset that consists of data from 109 subjects doing trials of 4 different motor imagery tasks. These tasks are motor imagery of the left fist (Class 1), right fist (Class 2), both fists (Class 3), and both feet (Class 4). We apply a band-pass filter with a cut-off frequency of 0.3 Hz, a sampling rate of 200 Hz, and a 4-second window (800 data points). For evaluation, we use PhysioNet-MI's evaluation set as the test set. We divide all subjects into four folds and apply the proposed cross-validation methodology for validation sets. Afterward, we perform four fine-tunings, using the 3 folds of subjects as the training set and the remaining fold as the validation set.

- **KaggleERN:** KaggleERN is an error-related potential dataset, which requires each subject to see letters and numbers (showing 36 possible items on a matrix). Every character item flashes in a different order. Each participant uses a computer interface that generates verbal responses based on their attention. When participants watch whether the system reacts appropriately or incorrectly, we record their EEG. This dataset has two labels: "correct feedback" and "erroneous feedback." We use a 2-second window sample (400 data points) with a sampling rate of 200 Hz. For evaluation, we treat KaggleERN's evaluation set as the test set. We split all subjects into five folds and apply the proposed cross-validation methodology for validation. Afterward, we perform five fine-tunings, using the 4 folds of subjects as the training set and the remaining fold as the validation set.
- **Sleep-EDFx:** Sleep-EDFx is a sleep stage classification dataset, consisting of data from 78 healthy subjects. This dataset contains 5 classes, corresponding to 5 sleep stages: W, N1, N2, N3, REM. We apply a low-pass filter with a 30 Hz cut-off, a sampling rate of 200 Hz, and a 30-second window (6000 data points) to Sleep-EDFx. For evaluation, we adopt the proposed subject-wise cross-validation protocol. We split the total dataset into 5 folds with the same number of subjects. We perform 5 fine-tunings, each using a different fold as the test set, and the remaining 4 folds serve as the training and validation sets. We randomly select training and validation data from these 4 folds, with a val-train ratio of 1:9.
- **TUAB:** TUAB consists of 409,455 10-second samples of subjects annotated as normal or abnormal (2-label classification). We apply a band-pass filter from 0.1 to 75 Hz, a notch filter at 60 Hz, a sampling rate of 200 Hz, and a 10-second window (2000 data points). For evaluation, we use TUAB's evaluation set as the test set. We adopt the proposed cross-validation protocol for validation sets. We split all subjects into 4 folds. We then conduct 4 fine-tunings, each using one fold as the validation set, with the remaining subjects as the training set. Generally, the train-valid-test ratio is 6:2:2.

E METRICS DESCRIPTION

This section describes all metrics used to evaluate the model's performance.

- **Balanced Accuracy:** The performance of unbalanced datasets is typically evaluated using balanced accuracy. It is the average recall across all classes in the dataset.
- **Cohen's kappa:** A statistical measure of inter-rater agreement for categorical items. It evaluates the agreement between predicted labels and ground-truth labels while accounting for the agreement that could occur purely by chance.
- **Weighted F1:** The average of all classes' F1-scores, weighted by the number of true cases in each class, is called weighted F1.
- **AUROC:** Is referred to as the Area Under the Receiver Operating Characteristic curve. AUROC, commonly used for binary classification, evaluates a classifier's ability to discriminate between positive and negative classes.
- **AUC-PR:** Is referred to as Area Under the Precision-Recall Curve. AUC-PR measures the trade-off between recall and precision across thresholds.

F OVERFITTING-GENERALIZATION RATIO (OGR)

Let $\mathcal{L}^T(\mathbf{\Theta}_{N_0})$ denote the total empirical training loss over the fixed training set, and let $\mathcal{L}^V(\mathbf{\Theta}_{N_0})$ be the total validation loss with respect to the target distribution, where N_0 is the training iteration. Overall Overfitting is defined as the:

$$O_{N_0} = \big(\mathcal{L}^T(\mathbf{\Theta}_0) - \mathcal{L}^T(\mathbf{\Theta}_{N_0})\big) - \big(\mathcal{L}^V(\mathbf{\Theta}_0) - \mathcal{L}^V(\mathbf{\Theta}_{N_0})\big).$$

The generalization is given as

$$G_{N_0} = \big(\mathcal{L}^V(\mathbf{\Theta}_0) - \mathcal{L}^V(\mathbf{\Theta}_{N_0})\big).$$

The overfitting-to-generalization ratio (OGR) is defined for as follows.

$$\mathrm{OGR} = \left| \frac{\big(\mathcal{L}^T(\mathbf{\Theta}_0) - \mathcal{L}^T(\mathbf{\Theta}_{N_0})\big) - \big(\mathcal{L}^V(\mathbf{\Theta}_0) - \mathcal{L}^V(\mathbf{\Theta}_{N_0})\big)}{\big(\mathcal{L}^V(\mathbf{\Theta}_0) - \mathcal{L}^V(\mathbf{\Theta}_{N_0})\big)} \right|.$$

The optimization target of both maximizing G_{N_0} and minimizing O_{N_0} is reduced to minimizing the above OGR term. To better capture this, the authors use online Overfitting and Generalization to redefine the OGR between two checkpoints from iteration N_0 to iteration $N_0 + n$:

$$\mathrm{OGR}_{N_0,n} = \left| \frac{\Delta O_{N_0,n}}{\Delta G_{N_0,n}} \right| = \left| \frac{O_{N_0+n} - O_{N_0}}{G_{N_0+n} - G_{N_0}} \right|. \tag{12}$$

Infinitesimal problem for gradient steps.
In (Wang et al., 2020), authors defined an infinitesimal problem: given several estimates of the gradient, blend them to minimize an infinitesimal OGR (or equivalently OGR^2). Let $\hat{g}$ be an estimated gradient step, $\nabla\mathcal{L}^T$ the training gradient, and $\nabla\mathcal{L}^V$ is the validation loss gradient. Using the first-order approximations, the change in overfitting and generalization is given: $\Delta G \approx \left\langle \nabla\mathcal{L}^V(\Theta),\, \hat{g} \right\rangle$ and $\Delta O \approx \left\langle \nabla\mathcal{L}^T(\Theta) - \nabla\mathcal{L}^V(\Theta),\, \hat{g} \right\rangle$. Thus, OGR^2 with respect to the total gradient $\hat{g}$ is:

$$\mathrm{OGR}^2(\hat{g}) = \left(\frac{\langle \nabla\mathcal{L}^T(\Theta) - \nabla\mathcal{L}^V(\Theta), \hat{g} \rangle}{\langle \nabla\mathcal{L}^V(\Theta), \hat{g} \rangle} \right)^2. \tag{13}$$

Closed-form optimal weight using Gradient-blending.
Let $\{\mathcal{L}_i\}_{i=1}^k$ be i-th modality loss and $\nabla\mathcal{L}_i$ gradient estimate at each backpropagation step. The blended gradient is updated as follows:

$$\hat{g} = \sum_{i=1}^{k} w_i \nabla\mathcal{L}_i \tag{14}$$

By minimizing OGR^2 (Eq. 13) and using the first-order approximation, the optimization problem is defined as

$$\min_{\mathbf{w}} \left(\frac{\left\langle \nabla\mathcal{L}^T - \nabla\mathcal{L}^V,\, \sum_{i=1}^k w_i \nabla\mathcal{L}_i \right\rangle}{\left\langle \nabla\mathcal{L}^V,\, \sum_{i=1}^k w_i \nabla\mathcal{L}_i \right\rangle} \right)^2 \tag{15}$$

To solve the above optimization problem Eq. 15, the optimal weight for loss i is given by (Wang et al., 2020) as

$$w_i^* = \frac{\left\langle \nabla\mathcal{L}^V,\, \nabla\mathcal{L}_i \right\rangle}{\mathbb{E}\left[\langle \nabla\mathcal{L}^T - \nabla\mathcal{L}^V,\, \nabla\mathcal{L}_i \rangle^2 \right]}, \tag{16}$$

Assumption. Within this work, the authors use 2 assumptions:

- Normalization: $\left\langle \nabla\mathcal{L}^V,\, \sum_{i=1}^k w_i \nabla\mathcal{L}_i \right\rangle = 1$
- Cross-term overfitting: $\mathbb{E}\Big[\left\langle \nabla\mathcal{L}^T - \nabla\mathcal{L}^V, \nabla\mathcal{L}_i \right\rangle \left\langle \nabla\mathcal{L}^T - \nabla\mathcal{L}^V, \nabla\mathcal{L}_j \right\rangle \Big] = 0 \; (\forall\, i \neq j)$

Implementation approximation. Instead of extracting each loss gradient $\nabla\mathcal{L}_i^V$ or $\nabla\mathcal{L}_i^T$, (Wang et al., 2020) uses approximation, as such:

$$w_i^* = \frac{G_i^{(N_0)}}{\left(O_i^{(N_0)} \right)^2} \quad \forall i \in \{1, 2, \ldots, k\} \tag{17}$$

In addition to these 2 assumptions, OGR uses a non-trivial approximation. Despite gradient-aware analysis, the closed-form solution in Eq. 17 only regulates weight w_i based on losses of objective i.

G Model-Agnostic adaptive alignment (MAPE)

Considering an optimization function addresses 2 problems of **OGR**: unbiased weighting due to the lack of objective normalization, and issues of possible negative Generalization, we propose our optimization function:

$$f(\mathbf{w}) = (\overline{O}^{(N_0)})^2 + \exp\left(-\overline{G}^{(N_0)}\right) + \lambda \cdot ||\mathbf{w} - \mathbf{b}||_2^2 \tag{18}$$

$$= \left[\mathbf{w^T O}\right]^2 + e^{-\mathbf{w^T G}} + \lambda \cdot ||\mathbf{w} - \mathbf{b}||_2^2, \tag{19}$$

Algorithm 1 Model-Agnostic Adaptive Alignment (MAPE)

Input: Input $\mathbf{x}$, model $\mathbf{\Theta}$, k objectives, regularizer $\lambda > 0$, M epochs (M updates).
Initialize $w_i(0) = 1 \; \forall i \in \{1, 2, \ldots, k\}$. Train, validate $\mathbf{\Theta}_{t=0}$, get $\mathcal{L}_i^T(0)$, $\mathcal{L}_i^V(0)$.
Initialize $w_i(1) = 1 \; \forall i \in \{1, 2, \ldots, k\}$.
for epoch $t = 1$ **to** M **do**
  $\{\overline{\mathcal{L}_i^T}(t), \overline{\mathcal{L}_i^V}(t)\}_{i=1}^k = \mathbf{\Theta}_t(\mathbf{x}_t, \mathbf{w}_t)$ {Training 1 epoch}
  Calculate loss: $\overline{\mathcal{L}}(t) = \sum_{i=1}^k w_i(t) \cdot \overline{\mathcal{L}_i}(t)$. Update model $\mathbf{\Theta}_{t+1}$
  Update $\overline{O}_i^{(N_0)}, \overline{G}_i^{(N_0)} \; \forall i$ follow equation 6
  Calculate $\mathbf{A}, K, C$ follow equation 9
  Calculate $\mathbf{w}(t+1)$ follow equation 9
end for

with $\mathbf{O} = [\,\overline{O}_1^{(N_0)}\ \overline{O}_2^{(N_0)}\ \ldots\ \overline{O}_k^{(N_0)}]^T \in \mathbb{R}^{k\times 1}$, $\mathbf{G} = [\,\overline{G}_1^{(N_0)}\ \overline{G}_2^{(N_0)}\ \ldots\ \overline{G}_k^{(N_0)}]^T \in \mathbb{R}^{k\times 1}$.

Derivative of $f(\mathbf{w})$:

$$\nabla_{\mathbf{w}} f(\mathbf{w}) = 2\mathbf{O}(\mathbf{O}^T\mathbf{w}) - \mathbf{G}e^{-\mathbf{G}^T\mathbf{w}} + 2\lambda(\mathbf{w} - \mathbf{b}) \tag{20}$$

Firstly, we prove $f(\mathbf{w})$ **is a convex function** via Hessian matrix $\mathbf{H}$ (second derivative of $f(\mathbf{w})$):

$$\begin{aligned}\mathbf{H} &= \nabla_{\mathbf{w}}\left(2\mathbf{O}(\mathbf{O}^T\mathbf{w}) - \mathbf{G}e^{-\mathbf{G}^T\mathbf{w}} + 2\lambda(\mathbf{w} - \mathbf{b})\right) \\ &= 2\mathbf{O}\mathbf{O}^T - \mathbf{G}\left(-\mathbf{G}^T e^{-\mathbf{w}^T\mathbf{G}}\right) + 2\lambda\mathbf{I} \\ &= 2\mathbf{O}\mathbf{O}^T + e^{-\mathbf{w}^T\mathbf{G}}\mathbf{G}\mathbf{G}^T + 2\lambda\mathbf{I}\end{aligned} \tag{21}$$

For any non-zero vector $\mathbf{v} \in \mathbb{R}^{k\times 1}$, its quadratic form with $\mathbf{H}$:

$$\begin{aligned}\mathbf{v}^T\mathbf{H}\mathbf{v} &= \mathbf{v}^T\left(2\mathbf{O}\mathbf{O}^T + e^{-\mathbf{w}^T\mathbf{G}}\mathbf{G}\mathbf{G}^T + 2\lambda\mathbf{I}\right)\mathbf{v} \\ &= 2\mathbf{v}^T\mathbf{O}\mathbf{O}^T\mathbf{v} + e^{-\mathbf{w}^T\mathbf{G}}\mathbf{v}^T\mathbf{G}\mathbf{G}^T\mathbf{v} + 2\lambda\mathbf{v}^T\mathbf{I}\mathbf{v} \\ &= 2(\mathbf{O}^T\mathbf{v})^2 + e^{-\mathbf{w}^T\mathbf{G}}(\mathbf{G}^T\mathbf{v})^2 + 2\lambda\|\mathbf{v}\|^2 > 0,\end{aligned}$$

because 3 scalars: $2(\mathbf{O}^T\mathbf{v})^2 \geq 0$, $e^{-\mathbf{w}^T\mathbf{G}}(\mathbf{G}^T\mathbf{v})^2 \geq 0$, and $2\lambda\|\mathbf{v}\|^2 > 0$ ($\lambda > 0$ and $\mathbf{v} \neq \mathbf{0}$).

Therefore, the Hessian matrix $\mathbf{H}$ is strictly positive definite, which means $f(\mathbf{w})$ is a convex function.

As such, the optimal solution in Eq. 7 is the solution of setting Eq. 20 to zero, as the solution of below equality:

$$2(\mathbf{O}\mathbf{O}^T + \lambda\mathbf{I})\mathbf{w}^* = \mathbf{G}e^{-\mathbf{w}^{*T}\mathbf{G}} + 2\lambda\mathbf{b} \tag{22}$$

Denote $e^{-\mathbf{w}^T\mathbf{G}} = \alpha$ (positive scalar), $\mathbf{O}\mathbf{O}^T + \lambda\mathbf{I} = \mathbf{A}$ (easily proven to be Positive Definite matrix, similar to the proof for Hessian $\mathbf{H}$). As such, its inversion $\mathbf{A}^{-1}$ always exists.

$$\Rightarrow 2\mathbf{A}\mathbf{w}^* = \alpha\mathbf{G} + 2\lambda\mathbf{b} \tag{23}$$

$$\Rightarrow \mathbf{w}^* = \frac{\alpha}{2}\mathbf{A}^{-1}\mathbf{G} + \lambda\mathbf{A}^{-1}\mathbf{b} \tag{24}$$

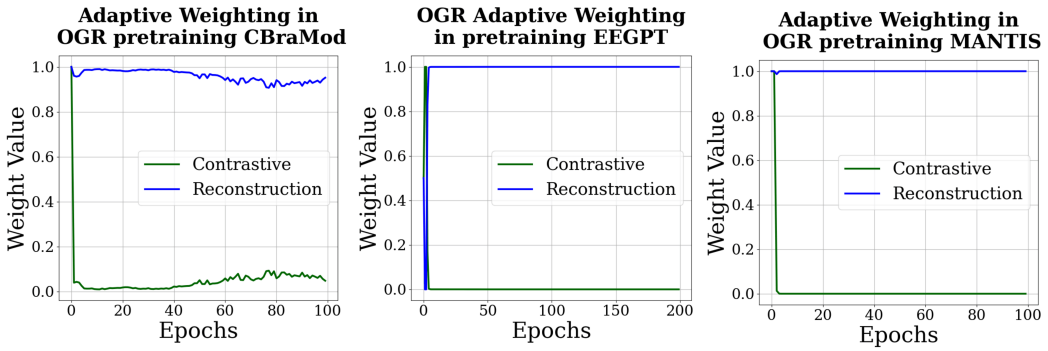
$$\Rightarrow \mathbf{G}^T\mathbf{w}^* = \frac{\alpha}{2}\left(\mathbf{G}^T\mathbf{A}^{-1}\mathbf{G}\right) + \lambda\mathbf{G}^T\mathbf{A}^{-1}\mathbf{b}, \tag{25}$$

with $K = \mathbf{G}^T\mathbf{A}^{-1}\mathbf{G}$ and $C = \lambda\mathbf{G}^T\mathbf{A}^{-1}\mathbf{b}$ being scalars. Noting that scalars K, C, matrices $\mathbf{A}$, and $\mathbf{G}$ can be easily computed via the objectives' losses during training. Following this substitution:

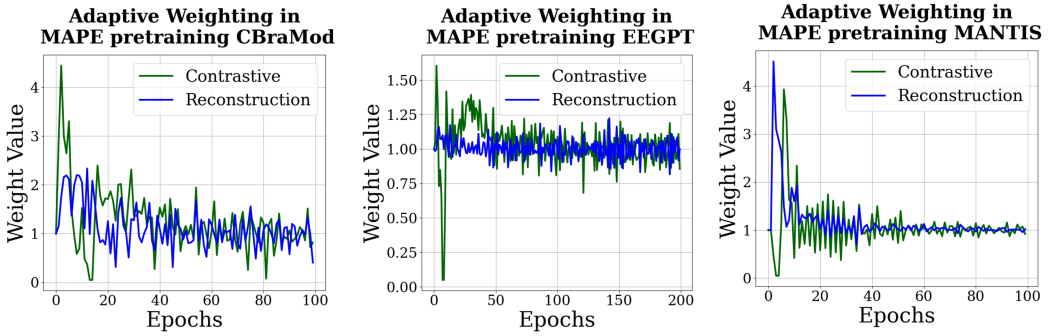
$$\Rightarrow e^{-\mathbf{G}^T\mathbf{w}^*} = e^{-\left(\frac{K}{2}\alpha + C\right)} \tag{26}$$

$$\Rightarrow \alpha = e^{-C}e^{\frac{-K}{2}\alpha} \tag{27}$$

$$\Rightarrow \left(\frac{K}{2}\alpha\right)e^{\frac{K}{2}\alpha} = \frac{K}{2}e^{-C}. \tag{28}$$



a) OGR adaptive weights in multi-objective pretraining with contrastive and reconstruction (*OGR-CR*).



b) MAPE adaptive weights in multi-objective pretraining with contrastive and reconstruction (**MAPE-CR**).

Figure 6: Adaptive weights in multi-objective pretraining with contrastive and reconstruction.

Equation 28 is in the form of $xe^x = y$ (Lambert-W function $W(\cdot)$) so its solution:

$$\frac{K}{2}\alpha = W\Big(\frac{K}{2}e^{-C}\Big) \tag{29}$$

$$\Rightarrow \alpha = \frac{2}{K}W\Big(\frac{K}{2}e^{-C}\Big). \tag{30}$$

Following equation 24, we have optimal $\mathbf{w}^*$:

$$\mathbf{w}^* = \frac{1}{K}W\Big(\frac{K}{2}e^{-C}\Big)\mathbf{A}^{-1}\mathbf{G} + \lambda\mathbf{A}^{-1}\mathbf{b}. \tag{31}$$

Numerical stability. To ensure the robustness of MAPE, we analyze the behavior of the optimal weight $\mathbf{w}^*$ as $K \to 0$. By substituting $x = W(\frac{K}{2}e^{-C})$, which implies $\frac{K}{2}e^{-C} = xe^x$ and $K = 2xe^xe^C$, the edge case:

$$\lim_{K\to 0}\frac{W(\frac{K}{2}e^{-C})}{K} = \lim_{x\to 0}\frac{x}{2xe^xe^C} = \lim_{x\to 0}\frac{1}{2e^xe^C} = \frac{e^{-C}}{2}. \tag{32}$$

As such, optimal $\mathbf{w}^*$ remains well-defined and avoids division-by-zero singularities as the aggregate generalization term vanishes.

Our optimal solution $\mathbf{w}^*$ does not require 2 assumptions and a non-straightforward approximation as OGR (Appendix F). Furthermore, this solution also considers reciprocal interaction between objectives, loss magnitude gap, and numerical stability, which are weaknesses of OGR. Details of MAPE implementation are in Algorithm 1.

H Adaptive weighting of MAPE and OGR

The adaptive weights $\mathbf{w} = \{w_i\}_{i=1}^k$ are the optimal solution to minimize the objective functions of MAPE (Eq. 8) and OGR (Eq. 4), to satisfy two goodness-indicated metrics: Generalization (real

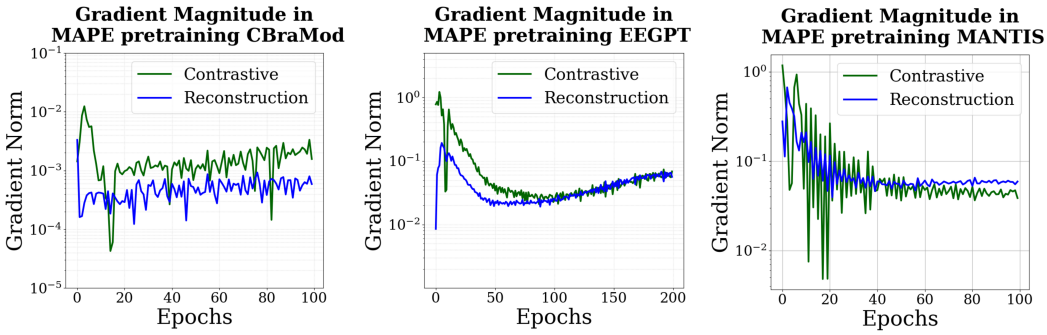
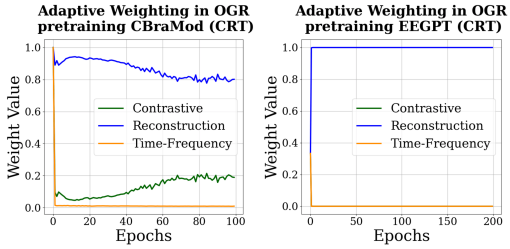
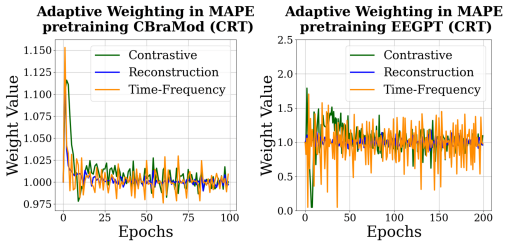


Figure 7: MAPE adaptive weighting mostly balances multiple gradients from divergent unaligned uniform pretrainings in Fig. 5b.



a) OGR adaptive weights in multi-objective pretraining with contrastive, reconstruction, and time-frequency reconstruction (*OGR-CRT*).



b) MAPE adaptive weights in multi-objective pretraining with contrastive, reconstruction, and time-frequency reconstruction (**MAPE-CRT**).

Figure 8: Adaptive weights in tri-objective pretraining.

gain of learning) and Overfitting. These weights will affect the gradient descent backpropagation to regulate the learning amount of the shared encoder via total updating gradient $\mathbf{g} = \sum_i w_i \cdot \mathbf{g}_i$, with $\mathbf{g}_i = \nabla_{\mathbf{\Theta}_e} \mathcal{L}_i$ being the gradient of the objective i w.r.t. the shared encoder $\mathbf{\Theta}_e$. OGR's solutions (Eq. 5) are often negative, which will crash the pretraining process. We further handle this by normalization following (Phan et al., 2021): if the weight w_i is negative, we set it to 0. If all w_i of k objectives are negative, they are set to $\frac{1}{k}$ (sum to 1). Note that this post-hoc normalization is specific to OGR alignment.

We further analyze the alignment behavior of MAPE and OGR, as shown in Fig. 6.a and Fig. 8.a. We empirically show the critical bias of OGR in pretraining weighting, as OGR is strongly affected by divergence in loss magnitudes due to directly affecting the minima of Eq. 4. With these weightings, OGR mostly pretrains the model with only one objective, omitting the benefit of joint pretraining. By eliminating divergence in loss magnitude, MAPE, in contrast, adapts the learning based on normalized Generalization and Overfitting (Fig. 6.b, Fig. 8.b). Our experiments also show that this adaptation nearly solves gradient-magnitude divergence (Fig. 7), compared with non-adaptive uniform settings (Fig. 5b). This involves several objectives in the pretraining process and solves occlusion in the Uniform and OGR settings. MAPE's better performance, as analyzed in Sec. 4, supports our conjecture that objective competition worsens performance.

I FULL RESULTS

Full metric results of MANTIS variants in Tab. 1 are illustrated in Tab. 9. Full metric results of MANTIS variants on ECG and EMG datasets (Ablation 4.3) are shown in Tab. 10.

Full metric results of CBraMod and EEGPT variants in Tab. 2, and Tab. 3 are illustrated in Tab. 11, and Tab. 12.

Table 9: Full evaluation metrics of **MANTIS** under the **Zero-shot learning** setup using a Random Forest classifier on EEG datasets: PCL, CAP, and SEDFx. Results are reported for in-distribution (ID) and out-of-distribution (OOD) settings, averaged over 5 seeds. **Bold** indicates the best performance, and underline indicates the second best.

ID setting										
Strategy	**Obj.**	**PCL**			**CAP**			**SEDFx**		
		BA	**AUROC**	**AUC-PR**	**BA**	**CK**	**F1**	**BA**	**CK**	**F1**
Single	C	0.563 ± 0.006	0.597 ± 0.007	0.595 ± 0.009	0.640 ± 0.007	0.680 ± 0.008	0.741 ± 0.006	0.609 ± 0.002	0.651 ± 0.002	0.728 ± 0.002
	R	0.540 ± 0.005	0.561 ± 0.005	0.558 ± 0.006	0.572 ± 0.005	0.610 ± 0.004	0.677 ± 0.004	0.530 ± 0.002	0.602 ± 0.001	0.684 ± 0.001
Uniform	CR	0.561 ± 0.006	0.593 ± 0.003	0.590 ± 0.002	0.663 ± 0.005	0.699 ± 0.005	0.755 ± 0.004	0.616 ± 0.002	**0.671 ± 0.002**	**0.744 ± 0.002**
OGR	CR	0.542 ± 0.009	0.562 ± 0.008	0.559 ± 0.003	0.590 ± 0.003	0.631 ± 0.003	0.693 ± 0.002	0.551 ± 0.003	0.623 ± 0.003	0.702 ± 0.002
MAPE	CR	**0.568 ± 0.008**	**0.601 ± 0.010**	**0.600 ± 0.011**	**0.669 ± 0.004**	**0.704 ± 0.004**	**0.761 ± 0.003**	**0.617 ± 0.002**	**0.671 ± 0.003**	**0.744 ± 0.002**
OOD setting										
Strategy	**Obj.**	**PCL**			**CAP**			**SEDFx**		
		BA	**AUROC**	**AUC-PR**	**BA**	**CK**	**F1**	**BA**	**CK**	**F1**
Single	C	**0.534 ± 0.006**	**0.551 ± 0.007**	**0.545 ± 0.006**	0.607 ± 0.002	0.665 ± 0.002	0.742 ± 0.001	**0.555 ± 0.005**	0.555 ± 0.001	0.677 ± 0.001
	R	0.518 ± 0.007	0.524 ± 0.006	0.518 ± 0.003	0.553 ± 0.002	0.620 ± 0.002	0.701 ± 0.002	0.428 ± 0.003	0.508 ± 0.001	0.640 ± 0.001
Uniform	CR	0.525 ± 0.005	0.538 ± 0.009	0.531 ± 0.010	0.638 ± 0.001	0.693 ± 0.001	0.761 ± 0.001	0.547 ± 0.005	0.578 ± 0.001	0.689 ± 0.000
OGR	CR	0.518 ± 0.004	0.526 ± 0.005	0.524 ± 0.003	0.577 ± 0.004	0.646 ± 0.002	0.721 ± 0.002	0.458 ± 0.005	0.540 ± 0.001	0.661 ± 0.001
MAPE	CR	0.530 ± 0.007	0.544 ± 0.006	0.540 ± 0.005	**0.639 ± 0.003**	**0.694 ± 0.001**	**0.762 ± 0.001**	0.547 ± 0.004	**0.581 ± 0.002**	**0.691 ± 0.001**

Table 10: Full evaluation metrics of **MANTIS** under the **Zero-shot learning** setup using a Random Forest classifier on ECG datasets (MIT-BIH, PTB-XL), and EMG dataset (NinaPro-DB5). Results are reported for in-distribution (ID) and out-of-distribution (OOD) settings, averaged over 5 seeds. **Bold** indicates the best performance, and underline indicates the second best.

ID setting										
Strategy	**Obj.**	**MIT-BIH**			**PTB-XL**			**NinaPro-DB5**		
		BA	**CK**	**F1**	**BA**	**CK**	**F1**	**BA**	**CK**	**F1**
Single	C	**0.869 ± 0.006**	**0.946 ± 0.003**	**0.983 ± 0.001**	0.447 ± 0.004	0.455 ± 0.004	0.604 ± 0.004	0.293 ± 0.008	0.236 ± 0.006	0.273 ± 0.006
	R	0.806 ± 0.006	0.909 ± 0.002	0.972 ± 0.001	**0.469 ± 0.006**	**0.490 ± 0.007**	**0.628 ± 0.005**	0.241 ± 0.016	0.193 ± 0.016	0.230 ± 0.018
Uniform	CR	0.837 ± 0.006	0.930 ± 0.002	0.978 ± 0.001	0.449 ± 0.007	0.461 ± 0.009	0.609 ± 0.007	0.229 ± 0.007	0.184 ± 0.008	0.225 ± 0.007
OGR	CR	0.813 ± 0.007	0.914 ± 0.002	0.974 ± 0.001	0.450 ± 0.005	0.462 ± 0.006	0.609 ± 0.005	0.279 ± 0.014	0.229 ± 0.012	0.269 ± 0.012
MAPE	CR	0.853 ± 0.005	0.940 ± 0.003	0.981 ± 0.001	0.465 ± 0.008	0.486 ± 0.010	0.626 ± 0.008	**0.368 ± 0.021**	**0.321 ± 0.025**	**0.356 ± 0.023**
OOD setting										
Strategy	**Obj.**	**MIT-BIH**			**PTB-XL**			**NinaPro-DB5**		
		BA	**CK**	**F1**	**BA**	**CK**	**F1**	**BA**	**CK**	**F1**
Single	C	**0.342 ± 0.002**	0.344 ± 0.009	0.864 ± 0.002	0.412 ± 0.003	0.398 ± 0.005	0.560 ± 0.003	**0.210 ± 0.006**	**0.148 ± 0.007**	**0.183 ± 0.006**
	R	0.247 ± 0.000	0.166 ± 0.003	0.837 ± 0.002	0.420 ± 0.004	0.411 ± 0.007	0.576 ± 0.005	0.089 ± 0.006	0.046 ± 0.005	0.084 ± 0.004
Uniform	CR	0.295 ± 0.003	0.367 ± 0.008	0.884 ± 0.001	0.434 ± 0.004	0.433 ± 0.007	0.589 ± 0.005	0.132 ± 0.008	0.087 ± 0.010	0.122 ± 0.009
OGR	CR	0.266 ± 0.003	0.210 ± 0.008	0.846 ± 0.001	0.406 ± 0.002	0.391 ± 0.004	0.559 ± 0.003	0.087 ± 0.007	0.042 ± 0.009	0.084 ± 0.006
MAPE	CR	0.316 ± 0.002	**0.459 ± 0.010**	**0.897 ± 0.001**	**0.449 ± 0.002**	**0.458 ± 0.004**	**0.605 ± 0.003**	0.155 ± 0.004	0.108 ± 0.005	0.142 ± 0.005

Table 11: Full evaluation metrics of **CBraMod, EEGPT** under the **Linear-probe** setup using 2-MLP classifier on EEG datasets: BCIC-IV-2a and PhysioNet-MI. **Bold** indicates best performance, and underline indicates second best per encoder setup.

			BCIC-IV-2a			PhysioNet-MI		
Encoder	**Strategy**	**Obj.**	**BA**	**CK**	**F1**	**BA**	**CK**	**F1**
CBraMod	Single	C	0.543 ± 0.085	0.391 ± 0.114	0.534 ± 0.092	0.562 ± 0.011	0.416 ± 0.015	0.564 ± 0.010
		R	0.570 ± 0.083	0.426 ± 0.111	0.556 ± 0.100	0.589 ± 0.011	0.453 ± 0.015	0.590 ± 0.011
		T	0.542 ± 0.098	0.390 ± 0.130	0.527 ± 0.116	0.579 ± 0.009	0.439 ± 0.012	0.580 ± 0.013
	Uniform	CR	0.522 ± 0.083	0.363 ± 0.111	0.507 ± 0.100	0.583 ± 0.011	0.443 ± 0.014	0.581 ± 0.015
		CRT	0.557 ± 0.088	0.409 ± 0.117	0.540 ± 0.108	0.568 ± 0.012	0.424 ± 0.016	0.570 ± 0.009
	OGR	CR	0.551 ± 0.090	0.401 ± 0.120	0.534 ± 0.108	0.586 ± 0.009	0.448 ± 0.013	0.588 ± 0.010
		CRT	0.571 ± 0.093	0.428 ± 0.124	0.555 ± 0.109	0.592 ± 0.013	0.456 ± 0.018	0.594 ± 0.012
	MAPE	CR	0.578 ± 0.105	0.437 ± 0.141	0.561 ± 0.125	0.597 ± 0.011	0.463 ± 0.015	0.598 ± 0.012
		CRT	**0.587 ± 0.105**	**0.449 ± 0.140**	**0.568 ± 0.131**	**0.604 ± 0.009**	**0.472 ± 0.011**	**0.608 ± 0.008**
EEGPT	Single	C	0.427 ± 0.050	0.237 ± 0.067	0.412 ± 0.056	0.567 ± 0.007	0.423 ± 0.009	0.569 ± 0.008
		R	0.463 ± 0.050	0.284 ± 0.067	0.443 ± 0.066	0.568 ± 0.011	0.425 ± 0.015	0.571 ± 0.013
		T	0.472 ± 0.070	0.296 ± 0.094	0.454 ± 0.091	0.563 ± 0.014	0.417 ± 0.018	0.565 ± 0.015
	Uniform	CR	0.444 ± 0.050	0.258 ± 0.067	0.424 ± 0.057	0.573 ± 0.006	0.430 ± 0.008	0.575 ± 0.005
		CRT	0.461 ± 0.054	0.281 ± 0.072	0.439 ± 0.069	0.578 ± 0.002	0.437 ± 0.003	**0.580 ± 0.003**
	OGR	CR	0.477 ± 0.058	0.303 ± 0.078	0.454 ± 0.079	0.578 ± 0.004	0.438 ± 0.005	**0.580 ± 0.003**
		CRT	**0.517 ± 0.072**	**0.356 ± 0.096**	**0.495 ± 0.097**	**0.579 ± 0.005**	**0.439 ± 0.007**	**0.580 ± 0.006**
	MAPE	CR	0.485 ± 0.077	0.313 ± 0.102	0.464 ± 0.087	0.578 ± 0.007	0.438 ± 0.010	0.578 ± 0.009
		CRT	0.493 ± 0.071	0.324 ± 0.094	0.469 ± 0.092	0.577 ± 0.007	0.436 ± 0.009	0.577 ± 0.007

Table 12: Full evaluation metrics of **CBraMod, EEGPT** under the **Linear-probe** setup using 2-MLP classifier on EEG datasets: KaggleERN, Sleep-EDFx, and TUAB. **Bold** indicates best performance, and underline indicates second best per encoder setup.

		KaggleERN			Sleep-EDFx			TUAB		
Strat.	**Obj.**	**BA**	**AUC**	**PR**	**BA**	**CK**	**F1**	**BA**	**AUC**	**PR**
					CBraMod					
Single	C	**0.574 ± 0.018**	0.622 ± 0.030	0.777 ± 0.021	0.728 ± 0.038	0.694 ± 0.057	0.776 ± 0.040	0.812 ± 0.006	0.886 ± 0.007	0.894 ± 0.006
	R	0.531 ± 0.026	0.568 ± 0.034	0.744 ± 0.016	0.721 ± 0.035	0.683 ± 0.042	0.765 ± 0.030	**0.818 ± 0.004**	**0.896 ± 0.004**	**0.901 ± 0.003**
	T	0.547 ± 0.029	0.601 ± 0.050	0.768 ± 0.032	**0.737 ± 0.038**	**0.699 ± 0.043**	0.777 ± 0.031	0.813 ± 0.002	0.891 ± 0.005	0.897 ± 0.004
Uniform	CR	0.571 ± 0.028	0.640 ± 0.029	0.789 ± 0.022	0.733 ± 0.033	0.698 ± 0.047	**0.780 ± 0.033**	0.813 ± 0.004	0.894 ± 0.003	0.897 ± 0.002
	CRT	0.557 ± 0.032	0.620 ± 0.038	0.777 ± 0.022	0.726 ± 0.035	0.686 ± 0.046	0.772 ± 0.030	0.810 ± 0.006	0.890 ± 0.008	0.895 ± 0.005
OGR	CR	0.567 ± 0.020	0.639 ± 0.022	0.787 ± 0.017	0.722 ± 0.030	0.688 ± 0.057	0.772 ± 0.037	0.807 ± 0.005	0.883 ± 0.006	0.891 ± 0.004
	CRT	0.567 ± 0.018	**0.650 ± 0.026**	**0.795 ± 0.025**	0.729 ± 0.031	0.690 ± 0.050	0.773 ± 0.033	0.808 ± 0.007	0.887 ± 0.008	0.893 ± 0.010
MAPE	CR	0.558 ± 0.028	0.631 ± 0.034	0.781 ± 0.025	0.726 ± 0.037	0.690 ± 0.051	0.773 ± 0.034	0.815 ± 0.006	0.892 ± 0.008	0.900 ± 0.006
	CRT	0.567 ± 0.019	0.646 ± 0.025	0.788 ± 0.019	0.727 ± 0.035	0.691 ± 0.056	0.772 ± 0.040	0.814 ± 0.005	0.892 ± 0.007	0.897 ± 0.008
					EEGPT					
Single	C	0.509 ± 0.008	0.538 ± 0.032	0.733 ± 0.023	0.681 ± 0.042	0.655 ± 0.062	0.737 ± 0.046	0.817 ± 0.006	0.893 ± 0.003	0.898 ± 0.007
	R	**0.577 ± 0.018**	0.647 ± 0.017	0.798 ± 0.011	0.688 ± 0.044	0.655 ± 0.057	0.745 ± 0.041	0.816 ± 0.005	0.891 ± 0.005	0.901 ± 0.004
	T	0.552 ± 0.010	0.617 ± 0.007	0.782 ± 0.005	0.687 ± 0.044	0.660 ± 0.063	0.746 ± 0.044	0.818 ± 0.006	0.891 ± 0.004	0.898 ± 0.002
Uniform	CR	0.552 ± 0.027	0.599 ± 0.034	0.760 ± 0.023	0.690 ± 0.040	0.663 ± 0.051	0.747 ± 0.038	**0.820 ± 0.001**	0.896 ± 0.001	0.902 ± 0.002
	CRT	0.528 ± 0.020	0.561 ± 0.024	0.747 ± 0.013	0.687 ± 0.039	0.669 ± 0.060	0.752 ± 0.041	0.819 ± 0.004	**0.899 ± 0.002**	**0.905 ± 0.004**
OGR	CR	0.557 ± 0.011	0.624 ± 0.037	0.779 ± 0.025	0.692 ± 0.044	0.666 ± 0.062	0.751 ± 0.044	**0.820 ± 0.007**	0.895 ± 0.005	0.902 ± 0.006
	CRT	0.573 ± 0.017	**0.657 ± 0.011**	**0.807 ± 0.008**	0.688 ± 0.048	0.662 ± 0.068	0.747 ± 0.049	0.812 ± 0.003	0.892 ± 0.005	0.902 ± 0.002
MAPE	CR	0.564 ± 0.012	0.635 ± 0.006	0.786 ± 0.009	**0.696 ± 0.042**	**0.671 ± 0.059**	**0.753 ± 0.044**	0.819 ± 0.007	0.897 ± 0.005	0.904 ± 0.006
	CRT	0.564 ± 0.024	0.617 ± 0.045	0.777 ± 0.026	0.692 ± 0.038	0.670 ± 0.050	**0.753 ± 0.034**	0.819 ± 0.002	0.897 ± 0.001	0.902 ± 0.002