# DEFT: Disentanglement-Enhanced Fine-Tuning for EEG Foundation Models

**Yuqiu Deng**[1] **Muyun Jiang**[2] **Zongpeng Zhang**[3] **Mingqing Xiao**[4] **Yansen Wang**[4]
**Peiliang Gong**[2] **Xinliang Zhou**[1] **Yi Ding**[2] **Ziyu Jia**[5] **Cuntai Guan**[2] **Chenyu Liu**[1]
[1]Uploading Inc
[2]College of Computing and Data Science, Nanyang Technological University
[3]Department of Biostatistics, School of Public Health, Peking University
[4]Microsoft Research Asia [5]Institute of Automation, Chinese Academy of Sciences
yuqiudeng@uploading.tech, james.jiang@ntu.edu.sg, zongpeng.zhang98@gmail.com
mingqingxiao@microsoft.com, yansenwang@microsoft.com, plgong@outlook.com
xinliang@uploading.tech, ding.yi@ntu.edu.sg, jia.ziyu@outlook.com
ctguan@ntu.edu.sg, chenyuliu@uploading.tech

## Abstract

EEG foundation models (EFMs) have improved EEG representation learning by transferring knowledge from large-scale multi-subject pretraining to downstream tasks. However, their pretrained embeddings often retain substantial subject-related variation, placing an additional burden on the downstream classifier, especially in cross-subject settings. To address this issue, we propose **D**ISENTANGLED **E**FM **F**INE-**T**UNING (**DEFT**), a plug-and-play adapter for EFM fine-tuning. **DEFT** encourages branch disentanglement of pretrained tokens into task-oriented content and subject-consistent style, then performs downstream prediction from the content branch. It instantiates this factorize-before-classify design with recursive Gaussian refinement, temporal-variance regularization, subject-aware contrastive learning, and reconstruction. Experiments across multiple EEG datasets and four EFM backbones show that **DEFT** improves matched fine-tuning baselines. Further ablations and analyses support the intended content/style branch organization.

## 1 Introduction

Electroencephalography (EEG) provides high-temporal-resolution signals for a wide range of neural decoding tasks, such as emotion recognition [56, 4, 35], motor imagery [51, 48, 32], and sleep staging [22, 64, 62]. Recent EEG foundation models (EFMs) improve scalability by pretraining on large multi-subject corpora and learning reusable representations that transfer across downstream settings [66, 25, 43]. This paradigm has substantially improved the transferability and generalizability of EEG models [25, 43], and is reshaping standard downstream EEG fine-tuning practice.

Probing reveals a consistent pattern across the evaluated settings: ***pretrained EFM embeddings make subject identity more linearly accessible.*** As shown in Figure 1, a probing classifier recovers subject identity more easily from pretrained EFM embeddings than from raw EEG features, while class-label probing improves only modestly after EFM encoding. EFM encoding can therefore make subject-related structure easier to separate in the pretrained token space. We use the compact factorization $\mathbf{Z} \rightarrow (\mathbf{C}, \mathbf{S})$ to summarize this mixed space, where $\mathbf{Z}$ denotes mixed pretrained tokens, $\mathbf{C}$ task-content tokens, and $\mathbf{S}$ subject-style tokens. Raw EEG already carries implicit subject traces, and EFM pretraining can sharpen these directions inside the mixed representation. EEG recordings are influenced by stable individual factors such as anatomy, functional organization, electrode-skin impedance, and recording habits [57, 47, 41, 2]. Current EFM embeddings consequently form mixed representations containing both task and subject structure.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

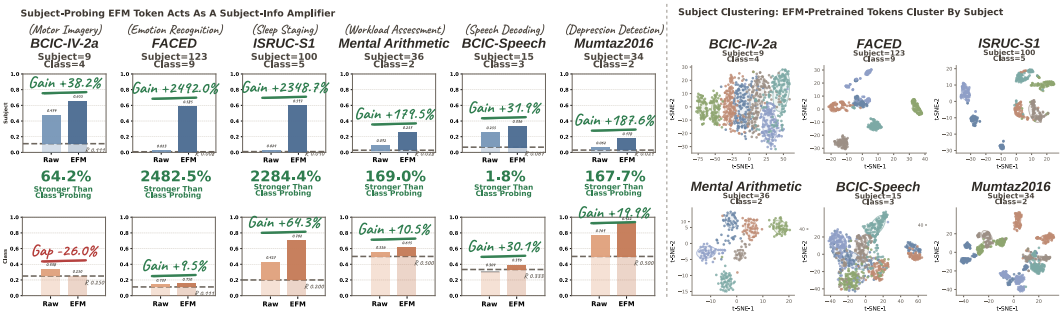



Figure 1: **Pretrained EFM embeddings amplify subject information.** Frozen CBraMod embeddings improve linear subject probing over raw EEG and form subject-dependent clusters. Task probes use subject-disjoint splits; subject probes use held-out trials from known subjects.

This amplification exposes a structural limitation of current EFM fine-tuning: ***the strongest signal in a pretrained token may be the wrong signal for the downstream task.*** Figure 2 (left) illustrates this burden. If pretraining sharpens subject structure more than class structure, the downstream classifier inherits a representation that is rich but misaligned. With only task labels as supervision, the prediction head must first discount easily accessible subject structure and then recover class-relevant content from the remaining mixed representation. This coupling is especially undesirable in ***cross-subject settings***, where subject cues are stable, easy to exploit, and poorly aligned with the desired decision rule. As a result, standard fine-tuning may spend part of its capacity correcting the pretrained representation before learning the classifier, which can obscure class-discriminative cues and weaken robustness in cross-subject EEG decoding [57, 49]. This failure mode raises a critical question:

***How can we prevent amplified subject-related variation in pretrained EFM embeddings from burdening downstream classification?***

To address this problem, we propose **DISENTANGLED EFM FINE-TUNING** (**DEFT**), a post-backbone disentanglement adapter for EFM fine-tuning. It operates on mixed pretrained tokens and forms a subject-style branch together with a task-content branch for downstream classification.

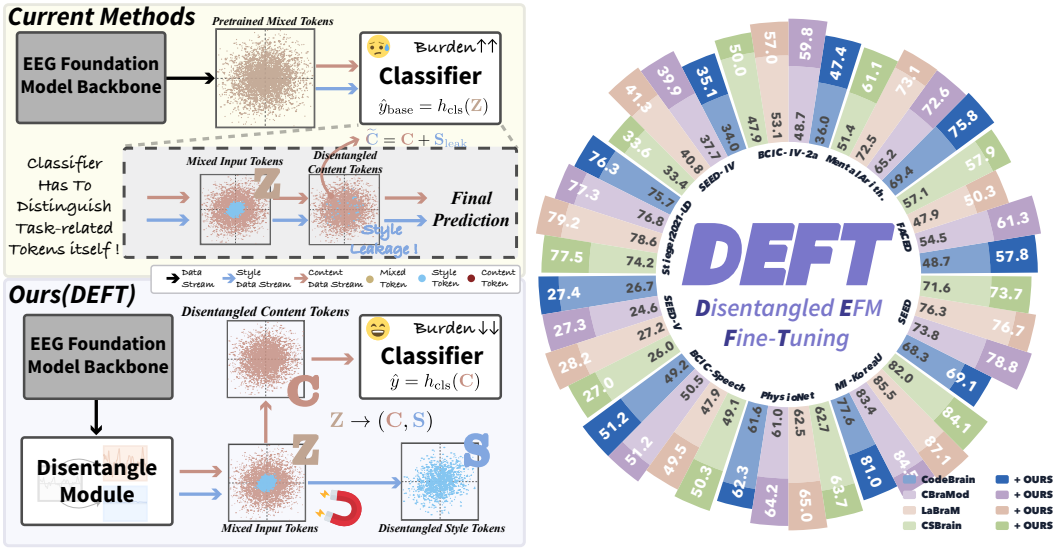



Figure 2: **Overview: motivation and empirical effect of DEFT.** *Left*: standard EFM fine-tuning leaves the classifier to suppress subject-related variation inside mixed pretrained tokens, while **DEFT** models subject-consistent information before classification. *Right*: an illustrative Balanced Accuracy snapshot across representative datasets and foundation models, showing that **DEFT** improves over vanilla fine-tuning in most displayed comparisons.

The key idea is to absorb the stronger subject-related structure earlier in the representation, so that the final head no longer has to resolve the full mixture on its own. The two branches are refined by recursive Gaussian disentanglement, shaped by an intrinsic temporal-variance constraint, and regularized by task classification, subject-aware self-supervised contrast, and reconstruction. In this way, disentanglement is handled at the representation level and the classifier can focus more directly on class discrimination. This design gives the prediction head a cleaner working representation in practice while still preserving the subject branch as an explicit modeling target.

In our main evaluation, **DEFT** improves matched EFM fine-tuning baselines across multiple EEG task families and four foundation backbones. These results support the factorize-before-classify design: encouraging branch disentanglement into task-oriented content and subject-consistent style gives the prediction head a more useful representation for cross-subject EEG decoding. Additional ablations, representation analysis, and probing diagnostics suggest that representation-level adaptation contributes positively to downstream classification and provide analytic evidence consistent with our content/style interpretation.

## 2 Preliminary

### 2.1 EEG Foundation Model

Let $\mathbf{x} \in \mathbb{R}^{T \times N_{\text{ch}}}$ denote an EEG trial with $T$ time steps and $N_{\text{ch}}$ channels. A pretrained EEG foundation model (EFM) backbone $\mathcal{B}(\cdot)$ maps $\mathbf{x}$ to a token tensor

$$\mathbf{Z} = \mathcal{B}(\mathbf{x}) = \{z_{m,w}\}_{m=1,w=1}^{M,W} \in \mathbb{R}^{M \times W \times D}, \tag{1}$$

where $M$ is the latent channel-group dimension, $W$ is the number of temporal windows, and $D$ is the token width. We denote this mixed pretrained representation by $\mathbf{Z}$. EFMs are pretrained on large multi-subject EEG corpora to learn transferable token representations for downstream adaptation.

### 2.2 EFM Fine-Tuning

Let $y \in \mathcal{Y}$ denote the task label and $u \in \mathcal{U}$ denote the subject identity. Standard fine-tuning attaches a downstream classification head $h_{\text{cls}}(\cdot)$ to the pretrained tokens and predicts

$$\hat{y} = h_{\text{cls}}(\mathbf{Z}), \qquad \mathcal{L}_{\text{cls}} = \text{CE}(\hat{y}, y). \tag{2}$$

In practice, $h_{\text{cls}}(\cdot)$ is a lightweight prediction head that operates on pretrained tokens, typically through token aggregation followed by a linear or shallow MLP classifier.

In this work, $\mathbf{Z}$ is treated as a mixed representation space. Each token may encode both class-relevant neural dynamics and subject-related stable patterns, which forces standard fine-tuning to optimize $h_{\text{cls}}(\cdot)$ on a representation where the two factors remain coupled. We therefore use **content** and **style** as functional branch roles in the pretrained token space. Content tokens $\mathbf{C}$ retain label-relevant brain-state structure and its temporal variation for subject-disjoint prediction. Style tokens $\mathbf{S}$ collect trial-stable residual structure organized primarily by subject identity. The prediction head operates on the content branch, while temporal-stability and subject-aware constraints shape the style branch.

## 3 Methodology

This section presents **D**ISENTANGLED **E**FM **F**INE-**T**UNING (**DEFT**), a post-backbone disentanglement adapter for EEG foundation model (EFM) fine-tuning.

### 3.1 Framework Overview

***Intuition:*** Pretrained EFM embeddings mix class-relevant dynamics with subject-specific bias. A direct classifier must learn the task decision while suppressing subject cues in the same token space. **DEFT** inserts an intermediate factorization step before prediction, so subject-consistent structure and task-discriminative variation can be shaped by separate training signals.

In **DEFT**, content and style denote operational branch roles induced by the training objectives. The prediction head is applied only to $\mathbf{C}$, so task gradients pass directly through the content branch.

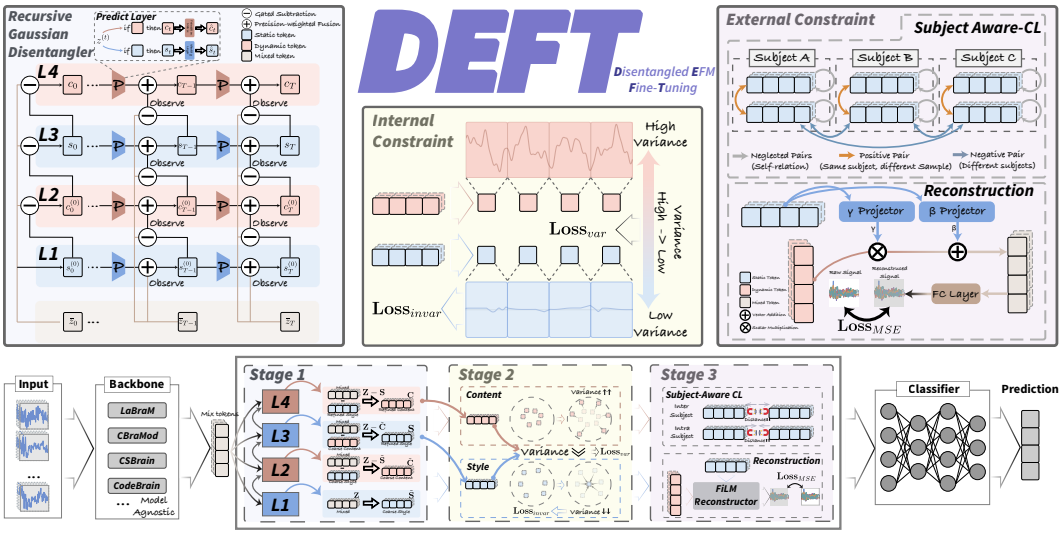



Figure 3: Overall framework of **DEFT**. Mixed pretrained tokens are organized into task-content and subject-style branches, then optimized with temporal-variance regularization, classification, subject-aware contrastive learning, and reconstruction. The downstream classifier uses the content branch for prediction. Detailed processes of each stage are shown above the overall framework.

Subject-aware contrast, temporal stability, and reconstruction shape the style branch, while the content branch serves as the prediction interface. This factorize-before-classify design keeps cross-subject prediction focused on task-oriented content while modeling subject-linked nuisance variation explicitly.

**DEFT** is used as a post-backbone adapter. Starting from the mixed token tensor $\mathbf{Z}$, we summarize it along the latent channel-group dimension and obtain a compact window-level sequence

$$\bar{z}_w = \frac{1}{M}\sum_{m=1}^{M} z_{m,w}, \qquad \bar{\mathbf{Z}} = [\bar{z}_1, \bar{z}_2, \ldots, \bar{z}_W] \in \mathbb{R}^{W\times D}. \tag{3}$$

The framework then follows three stages. The Recursive Gaussian Disentangler first maps the compact mixed sequence into window-level branches, referred to as content and style according to their training roles:

$$(\mathbf{C}^{\text{win}}, \mathbf{S}^{\text{win}}) = \mathcal{G}_{\text{dis}}(\bar{\mathbf{Z}}). \tag{4}$$

These window-level branches are then reinjected into the original token grid to form token-level content $\mathbf{C}$ and style $\mathbf{S}$. The ordering loss $\mathcal{L}_{\text{ord}}$ and style-stability loss $\mathcal{L}_{\text{sty}}$ define the intrinsic constraint; the subject-aware contrastive loss $\mathcal{L}_{\text{con}}$ and reconstruction loss $\mathcal{L}_{\text{rec}}$ provide external anchors. Sections 3.3–3.4 give their full definitions. The compact objective is

$$\mathcal{L}_{\text{int}} = \lambda_{\text{ord}}\mathcal{L}_{\text{ord}} + \lambda_{\text{sty}}\mathcal{L}_{\text{sty}}, \quad \mathcal{L}_{\text{ext}} = \lambda_{\text{con}}\mathcal{L}_{\text{con}} + \lambda_{\text{rec}}\mathcal{L}_{\text{rec}}, \quad \mathcal{L} = \mathcal{L}_{\text{cls}} + \mathcal{L}_{\text{int}} + \mathcal{L}_{\text{ext}}. \tag{5}$$

Together, these components form a task-content **Content Branch** and a subject-style **Style Branch**.

### 3.2 Recursive Gaussian Disentangler

***Intuition:*** The Recursive Gaussian Disentangler performs iterative two-branch inference from counterpart-subtracted observations and temporal priors. It adapts the precision-weighted linear-Gaussian aggregation used in Xi-vector and the prior–observation fusion principle of Kalman filtering to pretrained EFM tokens [31, 26]. Learned diagonal precision controls the contribution of token dimensions whose reliability varies with artifacts, electrode contact, and nonstationary brain activity. Alternating counterpart subtraction and asymmetric static/dynamic transitions then refine content and style estimates.

For each compact window $\bar{z}_w$, we derive observation statistics and remove the current counterpart estimate $r_w$:

$$e_w = f_\mu(\bar{z}_w), \quad \log \Lambda_w = f_\lambda(\bar{z}_w), \quad \tilde{e}_w = e_w - \gamma_w r_w, \tag{6}$$

where $\gamma_w \in (0,1)$ is a learnable gate controlling counterpart removal.

**Observation 3.1.** Let $e_w$ be the observation for window $w$, and let $r_w$ be the current estimate of the counterpart branch. If the branch update uses the counterpart-subtracted observation $\tilde{e}_w = e_w - \gamma_w r_w$ with $\gamma_w \in (0,1)$, then its inner product with the counterpart estimate is reduced by $\gamma_w \|r_w\|_2^2$. Alternating this update for the two branches therefore makes each branch estimate depend on evidence after subtracting the current counterpart estimate. The full Gaussian update equations are given in Appendix D.1.

*Derivation.* For a branch update with counterpart estimate $r_w$, direct substitution gives $\langle \tilde{e}_w, r_w \rangle = \langle e_w, r_w \rangle - \gamma_w \|r_w\|_2^2$. Thus, the observation used by the current branch has a smaller inner product with the counterpart estimate whenever $r_w \neq 0$ and $\gamma_w > 0$. Alternating the updates gives $s_w \leftarrow \mathcal{G}_{\text{sta}}(e_w - \gamma_w c_w)$ and $c_w \leftarrow \mathcal{G}_{\text{dyn}}(e_w - \gamma_w s_w)$, so each branch is estimated after subtracting the current estimate of the other branch. The Gaussian fusion posterior then weights this counterpart-subtracted observation by precision, with $h_w = (\Pi_w + \tilde{\Lambda}_w)^{-1}(\Pi_w T_w h_{w-1} + \tilde{\Lambda}_w \tilde{e}_w)$ as detailed in Appendix D.1. Recursive counterpart subtraction therefore biases each branch update toward lower overlap with its current counterpart estimate before downstream classification.

**Note 3.1.** The key observation is alternating refinement: each branch is updated with access to its current counterpart estimate and with uncertainty-aware fusion.

The two branches share the same Gaussian update but use different transition operators. For the style branch, we use a static transition $T_w = I$. For the content branch, we use a data-dependent transition

$$T_w = G_w = \sum_{n=1}^{N} \alpha_{w,n} G_n, \qquad \boldsymbol{\alpha}_w = \text{Softmax}(g(h_{w-1})), \tag{7}$$

where $\{G_n\}_{n=1}^N$ are learnable transition bases and $g(\cdot)$ is a lightweight gating network that selects their mixture. Starting from the mixed observation sequence, we perform four alternating inference steps:

$$\begin{aligned} \left(s_w^{(0)}, P_w^{(0)}\right) &= \mathcal{G}_{\text{sta}}(e_w, \Lambda_w), & \left(c_w^{(0)}, Q_w^{(0)}\right) &= \mathcal{G}_{\text{dyn}}(e_w, \Lambda_w; s_w^{(0)}, P_w^{(0)}), \\ (s_w, P_w) &= \mathcal{G}_{\text{sta}}(e_w, \Lambda_w; c_w^{(0)}, Q_w^{(0)}), & (c_w, Q_w) &= \mathcal{G}_{\text{dyn}}(e_w, \Lambda_w; s_w, P_w). \end{aligned} \tag{8}$$

This yields window-level factors $\mathbf{C}^{\text{win}} = [c_1, \ldots, c_W]$ and $\mathbf{S}^{\text{win}} = [s_1, \ldots, s_W]$, which are applied through residual reinjection into the original token grid:

$$z_{m,w}^c = (z_{m,w} - \bar{z}_w) + c_w, \qquad z_{m,w}^s = (z_{m,w} - \bar{z}_w) + s_w. \tag{9}$$

The residual term $z_{m,w} - \bar{z}_w$ preserves channel-group information when the shared window summary is replaced. It yields token-level content $\mathbf{C} = [z_{m,w}^c]_{m,w}$ and style $\mathbf{S} = [z_{m,w}^s]_{m,w}$.

### 3.3 Intrinsic Temporal-Variance Constraint

***Intuition:*** For the evaluated cross-subject tasks, we apply a task-dependent soft prior to learned window features: **content is encouraged to vary more across windows**, while **style is encouraged to remain stable** within a trial. This prior targets class-irrelevant subject bias while retaining stable evidence that predicts the task label in the content branch.

**Observation 3.2.** Let $\mathbf{H} = [h_1, \ldots, h_W]$ be a window sequence and let

$$\bar{h} = \frac{1}{W} \sum_{w=1}^{W} h_w, \qquad \text{Var}(\mathbf{H}) = \frac{1}{W} \sum_{w=1}^{W} \left\| h_w - \bar{h} \right\|_2^2. \tag{10}$$

If $\mathbf{C}$ should preserve temporally varying task evidence and $\mathbf{S}$ should absorb trial-stable subject structure, then a variance-ordering loss with a style-stability penalty encourages the intended ordering under the margin constraint and directly reduces within-trial style variation.

*Derivation.* By definition, $\text{Var}(\mathbf{H})$ measures within-trial deviation from the sequence mean. We implement temporal ordering and style stability with

$$\mathcal{L}_{\text{ord}} = \max\left(0, m + \text{Var}(\mathbf{S}) - \text{Var}(\mathbf{C})\right), \qquad \mathcal{L}_{\text{sty}} = \text{Var}(\mathbf{S}). \tag{11}$$

The hinge term is zero exactly when $\mathrm{Var}(\mathbf{C}) \geq \mathrm{Var}(\mathbf{S}) + m$. When it is active, $\partial \mathrm{Var}(\mathbf{H})/\partial h_w = \frac{2}{W}(h_w - \bar{h})$, so gradient descent on $\mathcal{L}_{\mathrm{ord}}$ moves content windows away from their mean and style windows toward their mean. The second term gives the same contraction direction for style by directly penalizing $\mathrm{Var}(\mathbf{S})$. Together, the two losses encode the intended design principle: content is encouraged to retain stronger window-level variation, while style is encouraged to remain stable within a trial.

**Note 3.2.** The intrinsic constraint is a label-free, task-dependent soft prior for settings where subject-linked structure acts as a nuisance factor for cross-subject prediction.

### 3.4 External Constraints

***Intuition:*** The intrinsic variance prior shapes temporal behavior. Two external anchors assign functional roles to the branches: subject-aware contrast organizes style by subject identity, and reconstruction preserves information across the split.

During training, a subject-balanced sampler provides at least one same-subject positive for every anchor in the six main datasets, so the style contrastive loss uses both same-subject positives and cross-subject negatives. Anchors without same-subject positives are excluded from $\mathcal{L}_{\mathrm{con}}$ and retain their classification, temporal-variance, and reconstruction losses.

**Observation 3.3.** Let $\mathbf{C}$ denote the prediction branch, let $\mathbf{S}^{\mathrm{win}} = [s_1, \ldots, s_W]$ denote the window-level style sequence, and let $\bar{z}_w$ denote the mixed window representation. Consider

$$\mathcal{I}_+ = \{i : |P(i)| > 0\}, \qquad \mathcal{L}_{\mathrm{con}} = \sum_{i \in \mathcal{I}_+} \frac{-1}{|P(i)|} \sum_{p \in P(i)} \log \frac{\exp(\mathrm{sim}(\mathbf{s}_i^*, \mathbf{s}_p^*)/\tau)}{\sum_{a \neq i} \exp(\mathrm{sim}(\mathbf{s}_i^*, \mathbf{s}_a^*)/\tau)}, \tag{12}$$

and

$$\mathbf{s}^* = \frac{1}{W} \sum_{w=1}^{W} s_w, \qquad \lambda, b = g_{\mathrm{film}}(\mathbf{s}^*), \qquad \hat{z}_w = \lambda \odot c_w + b, \qquad \mathcal{L}_{\mathrm{rec}} = \frac{1}{W} \sum_{w=1}^{W} \|\hat{z}_w - \bar{z}_w\|_2^2 . \tag{13}$$

Here, $P(i)$ denotes same-subject positives for sample $i$, and anchors without positives are excluded by $\mathcal{I}_+$. Then $\mathcal{L}_{\mathrm{con}}$ organizes $\mathbf{S}$ by subject consistency, while $\mathcal{L}_{\mathrm{rec}}$ discourages information loss in the content-style split.

*Derivation.* For $i \in \mathcal{I}_+$, minimizing $\mathcal{L}_{\mathrm{con}}$ increases $\mathrm{sim}(\mathbf{s}_i^*, \mathbf{s}_p^*)$ for $p \in P(i)$ and decreases its relative value against $\mathrm{sim}(\mathbf{s}_i^*, \mathbf{s}_a^*)$ for $a \neq i$. Thus, same-subject style summaries are pulled together in $\mathbf{S}$. For reconstruction, $\mathbf{s}^*$ is pooled across windows and only supplies sample-level FiLM parameters, while $c_w$ remains the window-varying input to $\hat{z}_w$. Minimizing $\mathcal{L}_{\mathrm{rec}}$ therefore requires the content branch to retain reconstructible window-level information that the style summary alone cannot provide. Thus, contrast anchors the style branch by subject, while reconstruction keeps the split informative.

**Note 3.3.** In the overview, $\mathcal{L}_{\mathrm{ext}}$ contains only the contrastive and reconstruction terms. Thus, the external constraints assist representation learning before the final prediction head. Subject identities are used only during training for $\mathcal{L}_{\mathrm{con}}$ and are not required at inference time. The full training objective is

$$\mathcal{L} = \mathrm{CE}\left(h_{\mathrm{cls}}(\mathbf{C}), y\right) + \lambda_{\mathrm{ord}}\mathcal{L}_{\mathrm{ord}} + \lambda_{\mathrm{sty}}\mathcal{L}_{\mathrm{sty}} + \lambda_{\mathrm{con}}\mathcal{L}_{\mathrm{con}} + \lambda_{\mathrm{rec}}\mathcal{L}_{\mathrm{rec}}, \tag{14}$$

where $\lambda_{\mathrm{ord}}$, $\lambda_{\mathrm{sty}}$, $\lambda_{\mathrm{con}}$, and $\lambda_{\mathrm{rec}}$ balance the intrinsic and external constraints.

## 4 Experiments

### 4.1 Experimental Setup

**Dataset and Backbone Setting:** We evaluate **DEFT** on four representative EEG foundation backbones: CodeBrain [43], CBraMod [53], LaBraM [25], and CSBrain [66]. The main text reports six representative downstream datasets under subject-disjoint evaluation: motor imagery (BCIC-IV-2a [51]), emotion recognition (FACED [4]), sleep staging (ISRUC-S1), workload assessment (MentalArith. [67]), speech decoding (BCIC-Speech), and MDD classification (Mumtaz2016). These datasets

Table 1: Main results on six representative downstream datasets. All metrics report multi-seed mean±std over available seeds. Bold marks the best value in each metric column, and Delta is the absolute improvement of **DEFT** over vanilla fine-tuning. Chance B-ACC is 0.250, 0.111, 0.200, 0.500, 0.333, and 0.500 for BCIC-IV-2a, FACED, ISRUC-S1, MentalArith., BCIC-Speech, and Mumtaz2016, respectively.

| Model | Setting | BCIC-IV-2a | | | FACED | | | ISRUC-S1 | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | B-ACC | Kappa | F1 | B-ACC | Kappa | F1 | B-ACC | Kappa | F1 |
| CodeBrain | Vanilla | 0.3678±0.0220 | 0.1570±0.0293 | 0.2980±0.0369 | 0.5720±0.0047 | 0.5156±0.0055 | 0.5719±0.0048 | 0.7803±0.0071 | 0.7366±0.0084 | 0.7943±0.0059 |
| | **DEFT** | 0.4621±0.0040 | 0.2828±0.0054 | 0.4318±0.0061 | 0.5870±0.0041 | 0.5297±0.0041 | 0.5799±0.0031 | 0.7829±0.0042 | 0.7391±0.0069 | 0.7981±0.0019 |
| | Delta | +0.0943 | +0.1258 | +0.1339 | +0.0150 | +0.0141 | +0.0080 | +0.0026 | +0.0025 | +0.0039 |
| CBraMod | Vanilla | 0.4630±0.0327 | 0.2840±0.0437 | 0.4332±0.0519 | 0.5387±0.0108 | 0.4770±0.0118 | 0.5387±0.0100 | 0.7767±0.0094 | 0.7514±0.0109 | 0.8052±0.0086 |
| | **DEFT** | 0.5579±0.0310 | 0.4105±0.0414 | 0.5522±0.0341 | **0.6005±0.0101** | **0.5467±0.0122** | **0.6002±0.0130** | 0.7818±0.0080 | 0.7285±0.0108 | 0.7906±0.0110 |
| | Delta | +0.0949 | +0.1265 | +0.1191 | +0.0618 | +0.0697 | +0.0615 | +0.0052 | -0.0229 | -0.0145 |
| LaBraM | Vanilla | 0.5434±0.0050 | 0.3912±0.0066 | 0.5349±0.0031 | 0.5336±0.0058 | 0.4727±0.0059 | 0.5351±0.0047 | 0.7730±0.0197 | 0.7350±0.0216 | 0.7773±0.0315 |
| | **DEFT** | **0.5671±0.0032** | **0.4228±0.0043** | **0.5601±0.0064** | 0.5409±0.0086 | 0.4820±0.0101 | 0.5462±0.0096 | **0.8051±0.0086** | **0.7792±0.0072** | **0.8288±0.0149** |
| | Delta | +0.0237 | +0.0316 | +0.0252 | +0.0072 | +0.0093 | +0.0112 | +0.0321 | +0.0442 | +0.0515 |
| CSBrain | Vanilla | 0.5148±0.0233 | 0.3530±0.0311 | 0.4980±0.0256 | 0.4169±0.2162 | 0.3441±0.2433 | 0.4031±0.2598 | 0.7738±0.0048 | 0.7297±0.0121 | 0.7819±0.0064 |
| | **DEFT** | 0.5448±0.0226 | 0.3931±0.0301 | 0.5345±0.0289 | 0.5748±0.0033 | 0.5195±0.0037 | 0.5820±0.0055 | 0.7871±0.0052 | 0.7471±0.0078 | 0.8031±0.0057 |
| | Delta | +0.0301 | +0.0401 | +0.0365 | +0.1579 | +0.1754 | +0.1789 | +0.0133 | +0.0175 | +0.0212 |

| Model | Setting | MentalArith. | | | BCIC-Speech | | | Mumtaz2016 | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | B-ACC | ROC-AUC | PR-AUC | B-ACC | Kappa | F1 | B-ACC | ROC-AUC | PR-AUC |
| CodeBrain | Vanilla | 0.6351±0.0635 | 0.7594±0.0618 | 0.5663±0.0433 | 0.5716±0.0304 | 0.3574±0.0457 | 0.5659±0.0416 | 0.9045±0.0045 | 0.9775±0.0021 | 0.9814±0.0014 |
| | **DEFT** | 0.7267±0.0319 | 0.8080±0.0223 | 0.5785±0.0356 | 0.6188±0.0188 | 0.4281±0.0282 | 0.6158±0.0226 | 0.9090±0.0093 | 0.9701±0.0123 | 0.9770±0.0080 |
| | Delta | +0.0917 | +0.0486 | +0.0122 | +0.0472 | +0.0707 | +0.0499 | +0.0044 | -0.0074 | -0.0045 |
| CBraMod | Vanilla | 0.6316±0.0148 | 0.7224±0.0161 | 0.5378±0.0166 | 0.6662±0.0251 | 0.4993±0.0376 | 0.6661±0.0254 | 0.8531±0.0439 | 0.9764±0.0096 | 0.9779±0.0097 |
| | **DEFT** | **0.7292±0.0059** | **0.8142±0.0190** | **0.6131±0.0503** | **0.6894±0.0172** | **0.5341±0.0258** | **0.6903±0.0187** | **0.9473±0.0142** | **0.9919±0.0018** | **0.9935±0.0018** |
| | Delta | +0.0976 | +0.0918 | +0.0754 | +0.0232 | +0.0348 | +0.0242 | +0.0942 | +0.0155 | +0.0156 |
| LaBraM | Vanilla | 0.7198±0.0507 | 0.7857±0.0574 | 0.6088±0.0932 | 0.5630±0.0321 | 0.3444±0.0481 | 0.5551±0.0414 | 0.9109±0.0096 | 0.9808±0.0054 | 0.9836±0.0042 |
| | **DEFT** | 0.7236±0.0149 | 0.7753±0.0262 | 0.5228±0.0478 | 0.6069±0.0096 | 0.4104±0.0145 | 0.5980±0.0096 | 0.9266±0.0025 | 0.9849±0.0021 | 0.9862±0.0017 |
| | Delta | +0.0038 | -0.0104 | -0.0860 | +0.0440 | +0.0659 | +0.0428 | +0.0157 | +0.0041 | +0.0026 |
| CSBrain | Vanilla | 0.5510±0.0374 | 0.7266±0.0067 | 0.4860±0.0425 | 0.5958±0.0103 | 0.3937±0.0154 | 0.5913±0.0103 | 0.8939±0.0089 | 0.9804±0.0044 | 0.9835±0.0037 |
| | **DEFT** | 0.6319±0.0157 | 0.7252±0.0363 | 0.4201±0.0479 | 0.6299±0.0109 | 0.4448±0.0163 | 0.6300±0.0131 | 0.9220±0.0180 | 0.9869±0.0059 | 0.9884±0.0047 |
| | Delta | +0.0809 | -0.0013 | -0.0659 | +0.0341 | +0.0511 | +0.0387 | +0.0281 | +0.0066 | +0.0049 |

cover distinct EEG task families, and complete results across the full evaluation suite are provided in Appendix E. Preprocessing details are provided in Appendix C.

**Model Setting:** We use the public released checkpoints of CodeBrain, CBraMod, LaBraM, and CSBrain and do not re-train the EFM pretraining stage. Following previous EEG foundation-model fine-tuning protocols, the pretrained EFM backbone, the downstream head, and any added adaptation modules are trainable during downstream training. The vanilla baseline optimizes the backbone and prediction head, while **DEFT** follows the same backbone update policy and jointly optimizes the disentanglement adapter. This keeps the comparison within the same checkpoint, split, and trainable encoder fine-tuning setting. The adapter adds only a small parameter overhead relative to the original fine-tuning model, including the pretrained backbone and downstream prediction head, as summarized in Appendix D.4. Detailed model parameters are described in Appendix D.

**Training and Environment Setting:** Experiments were conducted on NVIDIA RTX 4090 GPUs. Detailed training hyperparameters are provided in Appendix D.3.

**Evaluation Metrics:** Following prior EEG foundation model evaluations [25, 53, 43, 66], we report Balanced Accuracy (B-ACC) for all tasks, ROC-AUC/PR-AUC for binary tasks, and Kappa/weighted F1 for multi-class tasks. In the main table, all metrics report multi-seed mean and standard deviation over available seeds, and the complete records under seed 0 are provided in Appendix E.

### 4.2 Main Results

**Scope of the main comparison.** Table 1 is the primary quantitative comparison: six representative subject-disjoint datasets, four EFM backbones, and multi-seed results over available seeds. Appendix E provides complete seed-0 records for completeness. Across the 24 matched backbone–dataset cells, **DEFT** improves mean B-ACC by 0.0460, with a two-way backbone/dataset clustered-bootstrap 95% confidence interval of $[0.0240, 0.0745]$. The gains appear across CodeBrain, CBraMod, LaBraM, and CSBrain.

Table 2: Multi-seed CBraMod B-ACC ablations on the six representative datasets. Ablations use fixed seeds 0–2; Full is copied from Table 1.

| Dataset | Chance | Full | w/o StyleCL | w/o Invar. | w/o VarRatio | w/o Gaussian | w/o Rec. |
|---|---|---|---|---|---|---|---|
| BCIC-IV-2a | 0.250 | 0.5579±0.0310 | 0.4638±0.0170 | 0.3973±0.0664 | 0.3808±0.0071 | 0.3782±0.0198 | 0.4615±0.0477 |
| FACED | 0.111 | 0.6005±0.0101 | 0.5931±0.0072 | 0.5992±0.0011 | 0.5976±0.0027 | 0.5998±0.0087 | 0.6042±0.0066 |
| ISRUC-S1 | 0.200 | 0.7818±0.0080 | 0.7897±0.0012 | 0.7849±0.0019 | 0.7838±0.0121 | 0.7856±0.0057 | 0.7782±0.0090 |
| MentalArith. | 0.500 | 0.7292±0.0059 | 0.5830±0.0367 | 0.6549±0.0478 | 0.5687±0.0527 | 0.5958±0.0321 | 0.6135±0.0295 |
| BCIC-Speech | 0.333 | 0.6894±0.0172 | 0.4908±0.0034 | 0.4858±0.0082 | 0.4870±0.0109 | 0.4972±0.0076 | 0.4776±0.0120 |
| Mumtaz2016 | 0.500 | 0.9473±0.0142 | 0.8286±0.0678 | 0.9049±0.0475 | 0.7581±0.1772 | 0.9126±0.0114 | 0.8760±0.0329 |

**(a) Improvements span diverse task families.** The largest mean B-ACC gains occur on MentalArith., BCIC-IV-2a, and FACED. Together with the gains on sleep staging, speech decoding, and MDD classification, these results cover distinct EEG prediction settings.

**(b) Complementary metrics support the same overall trend.** Most Kappa, F1, ROC-AUC, and PR-AUC values improve, and Table 1 retains every observed decrease. The confidence intervals for the reviewer-identified secondary-metric differences are reported in Appendix 26.

**(c) Backbone-level gains vary in magnitude.** Averaged over the six representative datasets, B-ACC gains are +0.0628 for CBraMod, +0.0574 for CSBrain, +0.0425 for CodeBrain, and +0.0211 for LaBraM. Appendix F.4 reports additional controls on CBraMod, LaBraM, and REVE.

### 4.3 Ablation Study

We examine each component with fixed seeds 0–2 on CBraMod. Table 2 copies the multi-seed Full values from Table 1 and reports the corresponding multi-seed B-ACC ablations. The component effects are task dependent: the largest reductions appear on BCIC-IV-2a, MentalArith., BCIC-Speech, and Mumtaz2016, while the changes on FACED and ISRUC-S1 are smaller. Appendix F.9 retains the complete seed-0 metric records, and Appendix F.4 provides cross-backbone and objective-combination controls.

**(a) Component sensitivity depends on the task.** BCIC-IV-2a is especially sensitive to the invariance, variance-ratio, and Gaussian components; MentalArith. and BCIC-Speech show substantial reductions across all ablations.

**(b) The objectives provide complementary branch supervision.** The objective-combination control in Appendix F.5 shows task-dependent interactions among the variance, contrastive, and reconstruction terms.

For BCIC-IV-2a, removing invariance changes B-ACC by $-0.1606$, with a Welch 95% confidence interval of $[-0.2999, -0.0213]$ across the three seeds.

### 4.4 Representation Analysis

The main and ablation results show that **DEFT** improves downstream decoding, and we further examine whether the content branch also changes the geometry of the mixed representation in the intended direction. We use a post-hoc *within-subject class separation* score after subject centering and PCA50. Higher values indicate that task-class centroids are farther apart relative to within-class dispersion after subject-level offsets have been controlled. The detailed computation protocol is provided in Appendix D.2.

The top of Figure 4 shows that the content branch improves the class-separation score on five of the six representative datasets: BCIC-IV-2a (0.490 → 0.637, +0.146), FACED (0.520 → 0.783, +0.264), ISRUC-S1 (1.679 → 2.556, +0.877), MentalArith. (0.580 → 1.115, +0.535), and Mumtaz2016 (2.291 → 6.352, +4.061). These increases indicate that, after subject-level offsets are removed, the content branch usually makes class centroids farther apart relative to within-class dispersion.

BCIC-Speech is the only case where this post-hoc geometry score is slightly reduced (0.405 → 0.307), despite metric gains for all four backbones in Table 1. This mismatch suggests that speech-decoding cues may be localized across channels and windows, making them less visible as global PCA clusters.

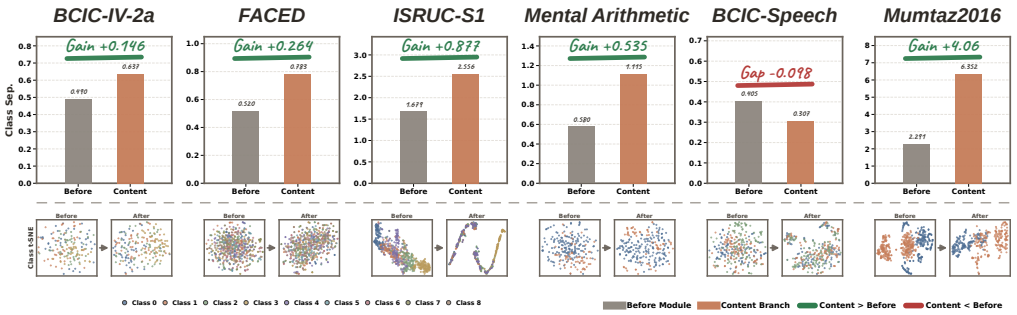



Figure 4: **Representation geometry before and after disentanglement.** Top: subject-controlled within-subject class separation after subject centering, where higher values indicate clearer class structure within each subject. Bottom: class t-SNE visualizations of the same before-module and content-branch representations.

Overall, the content branch usually produces clearer task-content geometry, but the geometry gain can vary with task characteristics.

Direct frozen linear probes provide a complementary measurement in Appendix F.1: content yields higher task-probe B-ACC and style yields higher subject-probe B-ACC in all six evaluated settings. This quantifies the functional specialization of the two branches.

## 5 Related Work

**EEG Foundation Models.** Encoder-based EFMs such as BIOT, EEGPT, LaBraM, CBraMod, CodeBrain, and CSBrain pretrain reusable EEG encoders with masked reconstruction, neural token prediction, or structured spatiotemporal modeling objectives [58, 52, 25, 53, 43, 66]. Decoder-based or language-connected EFMs provide a complementary direction: NeuroLM connects EEG tokens to language-model instruction tuning, and ECHO casts EEG modeling as sequence-to-sequence prediction with support examples for contextual adaptation [24, 34]. For encoder-based EFMs used in downstream classification, the common adaptation recipe remains vanilla fine-tuning: attach a task head and update the pretrained backbone under task supervision. **DEFT** instead studies a tailored fine-tuning adapter for this encoder-based setting, factorizing mixed pretrained tokens before prediction while remaining applicable to EFMs that expose token embeddings.

**Disentanglement Learning.** Disentangled representation learning separates factors such as content and style, stable and dynamic components, or identity and task information, with applications in speech and time-series representations [20, 33, 39, 45]. In EEG, patient-independent seizure detection has used adversarial or patient-invariant objectives to reduce patient-specific interference [16, 60, 61]. Existing approaches learn subject-invariant representations through adversarial, alignment, or sequential factorization objectives. **DEFT** applies factorization after a pretrained EFM backbone and assigns task prediction and subject consistency to separate token branches.

## 6 Conclusion

We introduce **DEFT**, a disentanglement-enhanced fine-tuning framework for EEG foundation models that targets a practical failure mode of EFM adaptation: pretrained embeddings can make subject-related structure highly accessible to the downstream classifier, even when that structure is not the intended target of cross-subject decoding. **DEFT** inserts a post-backbone factorize-before-classify adapter that decomposes the mixed representation into prediction-oriented content and subject-consistent style, then drives downstream prediction from the content branch. Experiments across multiple EEG task families and four EFM backbones show that **DEFT** improves matched fine-tuning baselines, while ablation, representation analysis, and content/style probing support the intended branch organization. Appendix checks cover one additional EFM, one independent MDD dataset, and one MOMENT/UCI HAR setting. Overall, these results support disentangling pretrained EFM embeddings for content-driven cross-subject EEG decoding.

## References


[1] Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra, and Jorge L. Reyes-Ortiz. A public domain dataset for human activity recognition using smartphones. In *European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, 2013.

[2] Geoffrey Brookshire, Jake Kasper, Nicholas M. Blauch, Yunan Charles Wu, Ryan Glatt, David A. Merrill, Spencer Gerrol, Keith J. Yoder, Colin Quirk, and Ché Lucero. Data leakage in deep learning studies of translational EEG. *Frontiers in Neuroscience*, 18:1373515, May 2024.

[3] James F. Cavanagh. EEG: Depression rest. OpenNeuro dataset ds003478, version 1.1.0, 2021.

[4] Jingjing Chen, Xiaobin Wang, Chen Huang, Xin Hu, Xinke Shen, and Dan Zhang. A Large Finer-grained Affective Computing EEG Dataset. *Scientific Data*, 10(1):740, October 2023.

[5] Ricky T. Q. Chen, Xuechen Li, Roger Grosse, and David Duvenaud. Isolating Sources of Disentanglement in Variational Autoencoders, April 2019. arXiv:1802.04942 [cs].

[6] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets, June 2016. arXiv:1606.03657 [cs].

[7] Zhisheng Chen, Yingwei Zhang, Qizhen Lan, Tianyu Liu, Huacan Wang, Yi Ding, Ziyu Jia, Ronghao Chen, Kun Wang, and Xinliang Zhou. Uni-ntfm: A unified foundation model for eeg signal representation learning. In *The Fourteenth International Conference on Learning Representations*, 2026.

[8] Hsiang-Yun Sherry Chien, Hanlin Goh, Christopher Michael Sandino, and Joseph Yitan Cheng. Maeeg: Masked auto-encoder for eeg representation learning. In *NeurIPS 2022 Workshop on Learning from Time Series for Health*, 2022.

[9] Mingzhe Cui, Tao Chen, Yang Jiao, Yiqin Wang, Lei Xie, Yi Pan, and Luca Mainardi. Brainrvq: A high-fidelity eeg foundation model via dual-domain residual quantization and hierarchical autoregression. *arXiv preprint arXiv:2602.16951*, 2026.

[10] Yi Ding, Muyun Jiang, Weibang Jiang, Shuailei Zhang, Xinliang Zhou, Chenyu Liu, Shanglin Li, Yong Li, and Cuntai Guan. Brainpro: Towards large-scale brain state-aware eeg representation learning. *arXiv preprint arXiv:2509.22050*, 2025.

[11] Yi Ding, Yong Li, Hao Sun, Rui Liu, Chengxuan Tong, Chenyu Liu, Xinliang Zhou, and Cuntai Guan. Eeg-deformer: A dense convolutional transformer for brain-computer interfaces. *IEEE Journal of Biomedical and Health Informatics*, 29(3):1909–1918, 2024.

[12] Berkay Döner, Thorir Mar Ingolfsson, Luca Benini, and Yawei Li. Luna: Efficient and topology-agnostic foundation model for eeg signal analysis. In *Advances in Neural Information Processing Systems*, 2025.

[13] Danny Dongyeop Han, Ahhyun Lucy Lee, Taeyang Lee, Yonghyeon Gwon, Sebin Lee, Seongjin Lee, David Keetae Park, Shinjae Yoo, Jiook Cha, and Chun Kee Chung. Diver-0: A fully channel equivariant eeg foundation model. *arXiv e-prints*, pages arXiv–2507, 2025.

[14] 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. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[15] Zitao Fang, CHENXUAN LI, Zhou Hongting, Shuyang Yu, Guodong DU, Ashwaq Qasem, Yang Lu, Jing Li, Junsong Zhang, and Sim Kuan Goh. Neuript: Foundation model for neural interfaces. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[16] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-Adversarial Training of Neural Networks, May 2016. arXiv:1505.07818 [stat].

[17] Mononito Goswami, Konrad Szafer, Arjun Choudhry, Yifu Cai, Shuo Li, and Artur Dubrawski. MOMENT: A family of open time-series foundation models. In *International Conference on Machine Learning*, pages 16115–16152, 2024.

[18] Haokun Gui, Xiucheng Li, and Xinyang Chen. Vector quantization pretraining for eeg time series with random projection and phase alignment. In *ICML*, 2024.

[19] Yiyu Gui, MingZhi Chen, Yuqi Su, Guibo Luo, and Yuchao Yang. Eegmamba: Bidirectional state space model with mixture of experts for eeg multi-task classification. *arXiv preprint arXiv:2407.20254*, 2024.

[20] Wei-Ning Hsu, Yu Zhang, and James Glass. Unsupervised learning of disentangled and interpretable representations from sequential data. In *Advances in Neural Information Processing Systems*, 2017.

[21] Zhihao Jia, Meiyan Xu, Jingyuan Wang, Ziyu Jia, Yong Li, Xin qiu Zhou, Chenyu Liu, Junfeng Yao, and Yi Ding. Sera: Separated coarse-to-fine representation alignment for cross-subject eeg-based emotion recognition. *Proceedings of the 33rd ACM International Conference on Multimedia*, 2025.

[22] Ziyu Jia, Youfang Lin, Jing Wang, Ronghao Zhou, Xiaojun Ning, Yuanlai He, and Yaoshuai Zhao. GraphSleepNet: Adaptive Spatial-Temporal Graph Convolutional Networks for Sleep Stage Classification. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, pages 1324–1330, Yokohama, Japan, July 2020. International Joint Conferences on Artificial Intelligence Organization.

[23] Muyun Jiang, Shuailei Zhang, Zhenjie Yang, Mengjun Wu, Weibang Jiang, Zhiwei Guo, Wei Zhang, Rui Liu, Shangen Zhang, Yong Li, et al. Elastiq: Eeg-language alignment with semantic task instruction and querying. *arXiv preprint arXiv:2509.24302*, 2025.

[24] Weibang Jiang, Yansen Wang, Bao-Liang Lu, and Dongsheng Li. NeuroLM: A Universal Multi-task Foundation Model for Bridging the Gap between Language and EEG Signals. In *International Conference on Learning Representations*, 2025.

[25] Weibang 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*, October 2023.

[26] Rudolf E. Kalman. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1):35–45, 1960.

[27] Hyunjik Kim and Andriy Mnih. Disentangling by Factorising, July 2019. arXiv:1802.05983 [stat].

[28] 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.

[29] Abhishek Kumar, Prasanna Sattigeri, and Avinash Balakrishnan. Variational Inference of Disentangled Latent Concepts from Unlabeled Observations, December 2018. arXiv:1711.00848 [cs].

[30] Vernon J Lawhern, Amelia J Solon, Nicholas R Waytowich, Stephen M Gordon, Chou P Hung, and Brent J Lance. Eegnet: a compact convolutional neural network for eeg-based brain–computer interfaces. *Journal of neural engineering*, 15(5):056013, 2018.

[31] Kong Aik Lee, Qiongqiong Wang, and Takafumi Koshinaka. Xi-Vector embedding for speaker recognition. *IEEE Signal Processing Letters*, 28:1385–1389, 2021.

[32] 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, May 2019.

[33] Yingzhen Li and Stephan Mandt. Disentangled sequential autoencoder. In *International Conference on Machine Learning*, pages 5670–5679, 2018.

[34] Chenyu Liu, Yuqiu Deng, Tianyu Liu, Jinan Zhou, Xinliang Zhou, Ziyu Jia, and Yi Ding. ECHO: Toward Contextual Seq2Seq Paradigms in Large EEG Models, September 2025. arXiv:2509.22556 [cs].

[35] Chenyu Liu, Xinliang Zhou, Yihao Wu, Yi Ding, Liming Zhai, Kun Wang, Ziyu Jia, and Yang Liu. A Comprehensive Survey on EEG-Based Emotion Recognition: A Graph-Based Perspective, August 2024. arXiv:2408.06027 [eess].

[36] Chenyu Liu, Xinliang Zhou, Jiaping Xiao, Zhengri Zhu, Liming Zhai, Ziyu Jia, and Yang Liu. Vsgt: Variational spatial and gaussian temporal graph models for eeg-based emotion recognition. In *IJCAI*, pages 3078–3086, 2024.

[37] Chenyu Liu, Xinliang Zhou, Zhengri Zhu, Liming Zhai, Ziyu Jia, and Yang Liu. Vbh-gnn: Variational bayesian heterogeneous graph neural networks for cross-subject emotion recognition. In *The Twelfth International Conference on Learning Representations*, 2024.

[38] Dingkun Liu, Yuheng Chen, Zhu Chen, Zhenyao Cui, Yaozhi Wen, Jiayu An, Jingwei Luo, and Dongrui Wu. Eeg foundation models: Progresses, benchmarking, and open problems. *arXiv preprint arXiv:2601.17883*, 2026.

[39] Tianchi Liu, Kong Aik Lee, Qiongqiong Wang, and Haizhou Li. Disentangling Voice and Content with Self-Supervision for Speaker Recognition, November 2023. arXiv:2310.01128 [eess].

[40] Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Rätsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging Common Assumptions in the Unsupervised Learning of Disentangled Representations, June 2019. arXiv:1811.12359 [cs].

[41] K.L. Lopez, A.D. Monachino, K.M. Vincent, F.C. Peck, and L.J. Gabard-Durnam. Stability, change, and reliable individual differences in electroencephalography measures: A lifespan perspective on progress and opportunities. *NeuroImage*, 275:120116, July 2023.

[42] Weiheng Lu, Chunfeng Song, Jiamin Wu, Pengyu Zhu, Yuchen Zhou, Weijian Mai, Qihao Zheng, and Wanli Ouyang. Unimind: Unleashing the power of llms for unified multi-task brain decoding. *arXiv preprint arXiv:2506.18962*, 2025.

[43] Jingying Ma, Feng Wu, Qika Lin, Yucheng Xing, Chenyu Liu, Ziyu Jia, and Mengling Feng. CodeBrain: Bridging Decoupled Tokenizer and Multi-Scale Architecture for EEG Foundation Model. In *International Conference on Learning Representations*, October 2025.

[44] Navid Mohammadi Foumani, Geoffrey Mackellar, Soheila Ghane, Saad Irtza, Nam Nguyen, and Mahsa Salehi. Eeg2rep: enhancing self-supervised eeg representation through informative masked inputs. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 5544–5555, 2024.

[45] Khalid Oublal, Saıd Ladjal, David Benhaiem, Emmanuel Le-borgne, and Francois Roueff. Disentangling Time Series Representations via Contrastive Independence-of-Support on l-Variational Inference. 2024.

[46] Jathurshan Pradeepkumar, Xihao Piao, Zheng Chen, and Jimeng Sun. Tokenizing single-channel eeg with time-frequency motif learning. In *The Fourteenth International Conference on Learning Representations*, 2026.

[47] Simanto Saha and Mathias Baumert. Intra- and Inter-subject Variability in EEG-Based Sensorimotor Brain Computer Interface: A Review. *Frontiers in Computational Neuroscience*, 13:87, January 2020.

[48] Gerwin Schalk, Dennis J McFarland, Thilo Hinterberger, Niels Birbaumer, and Jonathan R Wolpaw. EEG Motor Movement/Imagery Dataset, 2009.

[49] Xinke Shen, Xianggen Liu, Xin Hu, Dan Zhang, and Sen Song. Contrastive Learning of Subject-Invariant EEG Representations for Cross-Subject Emotion Recognition. *IEEE Transactions on Affective Computing*, 14(3):2496–2511, July 2023.

[50] Yonghao Song, Qingqing Zheng, Bingchuan Liu, and Xiaorong Gao. Eeg conformer: Convolutional transformer for eeg decoding and visualization. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 31:710–719, 2022.

[51] Michael Tangermann, Klaus-Robert Müller, Ad Aertsen, Niels Birbaumer, Christoph Braun, Clemens Brunner, Robert Leeb, Carsten Mehring, Kai J. Miller, Gernot R. Müller-Putz, Guido Nolte, Gert Pfurtscheller, Hubert Preissl, Gerwin Schalk, Alois Schlögl, Carmen Vidaurre, Stephan Waldert, and Benjamin Blankertz. Review of the BCI Competition IV. *Frontiers in Neuroscience*, 6, 2012.

[52] Guagnyu Wang, Wenchao Liu, Yuhong He, Cong Xu, Lin Ma, and Haifeng Li. EEGPT: Pretrained Transformer for Universal and Reliable Representation of EEG Signals. *Advances in Neural Information Processing Systems*, 37:39249–39280, December 2024.

[53] 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*, October 2024.

[54] Jiquan Wang, Sha Zhao, Zhiling Luo, Yangxuan Zhou, Shijian Li, and Gang Pan. Eegmamba: An eeg foundation model with mamba. *Neural Networks*, page 107816, 2025.

[55] Jiquan Wang, Sha Zhao, Yangxuan Zhou, Yiming Kang, Shijian Li, and Gang Pan. Deeperbrain: A neuro-grounded eeg foundation model towards universal bci. *arXiv preprint arXiv:2601.06134*, 2026.

[56] Wei-Long Zheng and Bao-Liang Lu. Investigating Critical Frequency Bands and Channels for EEG-Based Emotion Recognition with Deep Neural Networks. *IEEE Transactions on Autonomous Mental Development*, 7(3):162–175, September 2015.

[57] Dongrui Wu, Yifan Xu, and Bao-Liang Lu. Transfer Learning for EEG-Based Brain–Computer Interfaces: A Review of Progress Made Since 2016. *IEEE Transactions on Cognitive and Developmental Systems*, 14(1):4–19, March 2022.

[58] Chaoqi Yang, M. Brandon Westover, and Jimeng Sun. BIOT: Cross-data Biosignal Learning in the Wild, May 2023. arXiv:2305.10351 [eess].

[59] Shihao Yang, Xiying Huang, Danilo Bernardo, Jun-En Ding, Andrew Michael, Jingmei Yang, Patrick Kwan, Ashish Raj, and Feng Liu. Foundation and large-scale ai models in neuroscience: A comprehensive review. *arXiv preprint arXiv:2510.16658*, 2025.

[60] Xiang Zhang, Lina Yao, Manqing Dong, Zhe Liu, Yu Zhang, and Yong Li. Adversarial Representation Learning for Robust Patient-Independent Epileptic Seizure Detection, February 2020. arXiv:1909.10868 [eess].

[61] Zongpeng Zhang, Taoyun Ji, Mingqing Xiao, Wen Wang, Guojing Yu, Tong Lin, Yuwu Jiang, Xiaohua Zhou, and Zhouchen Lin. Cross-patient automatic epileptic seizure detection using patient-adversarial neural networks with spatio-temporal EEG augmentation. *Biomedical Signal Processing and Control*, 89:105664, March 2024.

[62] Jianheng Zhou, Chenyu Liu, Jinan Zhou, Yi Ding, Yang Liu, Haoran Luo, Ziyu Jia, and Xinliang Zhou. Introducing Multimodal Paradigm for Learning Sleep Staging PSG via General-Purpose Model, September 2025. arXiv:2509.22810 [eess].

[63] Jianheng Zhou, Chenyu Liu, Jinan Zhou, Yi Ding, Yang Liu, Haoran Luo, Ziyu Jia, and Xinliang Zhou. Introducing multimodal paradigm for learning sleep staging psg via general-purpose model. *arXiv preprint arXiv:2509.22810*, 2025.

[64] Xinliang Zhou, Yuzhe Han, Zhisheng Chen, Chenyu Liu, Yi Ding, Ziyu Jia, and Yang Liu. BiT-MamSleep: Bidirectional Temporal Mamba for EEG Sleep Staging, November 2024. arXiv:2411.01589 [eess].

[65] Xinliang Zhou, Chenyu Liu, Zhisheng Chen, Kun Wang, Yi Ding, Ziyu Jia, and Qingsong Wen. Brain Foundation Models: A Survey on Advancements in Neural Signal Processing and Brain Discovery, 2025.

[66] 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. In *Advances in Neural Information Processing Systems*, October 2025.

[67] Igor Zyma, Sergii Tukaev, Ivan Seleznov, Ken Kiyono, Anton Popov, Mariia Chernykh, and Oleksii Shpenkov. Electroencephalograms during Mental Arithmetic Task Performance. *Data*, 4(1):14, January 2019.

# A Detailed Related Work

## A.1 Scalp EEG Foundation Models

EEG modeling has moved from task-specific supervised decoders toward reusable pretrained backbones. Earlier deep pipelines, including compact CNNs, graph-based sleep staging, convolutional Transformers, dense Transformer variants, and variational spatiotemporal graph models, encode useful task-level inductive biases [30, 22, 50, 11, 36]. Self-supervised EEG representation learning further reduced annotation dependence through contrastive prediction, masked reconstruction, and informative masked modeling [28, 8, 44], setting the stage for larger-scale EEG pretraining.

Recent EEG foundation models (EFMs) scale this paradigm across heterogeneous EEG corpora and task families [65, 59, 38]. Encoder-based EFMs such as BIOT, EEGPT, LaBraM, CBraMod, CodeBrain, and CSBrain learn transferable token representations with cross-data biosignal modeling, masked signal reconstruction, neural token prediction, criss-cross spatial-temporal attention, decoupled tokenization, or cross-scale spatiotemporal modeling [58, 52, 25, 53, 43, 66]. Recent variants broaden this space with state-space backbones and mixture-of-experts routing [54, 19, 7, 15], large-scale or topology-agnostic pretraining [14, 12, 13], neuroscience-inspired priors [10, 55], and richer token interfaces such as vector quantization, time-frequency motifs, and residual quantization [18, 46, 9].

A related branch connects EEG representations with language or multimodal systems, including EEG-language foundation modeling, contextual sequence-to-sequence EEG modeling, instruction-style brain decoding, EEG-language alignment, and general-purpose sleep-staging interfaces [24, 34, 42, 23, 63]. These studies expand EFM capacity and interfaces, while downstream classification still commonly feeds pretrained EFM embeddings directly to a task head and optimizes in the mixed representation space. **DEFT** targets this adaptation stage by inserting a representation-level adapter between the EFM and prediction head, factorizing the mixed representation into content and style before classification.

## A.2 Disentanglement and Subject-Related EEG Variation

Disentangled representation learning organizes latent variables according to factors of variation, with representative methods based on information-theoretic generative modeling, variational objectives, total-correlation penalties, and factorial latent distributions [6, 29, 5, 27]. Later analyses show that useful disentanglement depends on inductive biases, supervision, and downstream utility [40]. In sequential domains, FHVAE and DSAE separate sequence-level factors from segment-level dynamics, and later work isolates content/style factors in speech and time-series representations [20, 33, 39, 45]. **DEFT** extends factorized representation learning to downstream adaptation of pretrained EFM tokens.

Subject-related variation is a central source of difficulty in EEG. Stable individual differences, intra- and inter-subject variability, and leakage from subject-specific patterns can strongly affect apparent generalization [47, 41, 2]. Existing EEG methods often seek invariant or aligned representations through transfer learning, domain-adversarial training, patient-adversarial objectives, subject-invariant contrastive learning, variational graph modeling, or coarse-to-fine cross-subject alignment [57, 16, 60, 61, 49, 37, 21].

**DEFT** operates on pretrained EFM embeddings and treats them as a mixed representation that may already amplify subject-related information. The content branch is optimized for downstream prediction and encouraged to preserve task-relevant temporal variation, while the style branch collects subject-consistent structure that remains stable within a trial. This yields a factorize-before-classify adaptation interface for EFMs.

# B Limitations, Ethics, and Reproducibility

**Limitations.** **DEFT** introduces additional loss weights and an inductive temporal-variance prior, which are selected under the downstream validation protocol. For trait-level or demographic targets, validation selects the temporal-prior strength so stable label-relevant evidence remains in the content branch. A dataset with one sample per training subject provides no same-subject contrastive pairs; the classification, variance, and reconstruction objectives remain applicable. Subject identities support

Table 3: Compute accounting for reproducing the reported experiments. All values are approximate.

| Scope | Training fits | GPU / fit | GPU-hours |
|---|---|---|---|
| Main Table 1 multi-seed reproduction | 120–150 | 1 RTX 4090 GPU | 90–160 |
| Complete seed-0 suite | 96 | 1 RTX 4090 GPU | 55–130 |
| Content/style probing diagnostic | 16 | 1 RTX 4090 GPU | 1–3 |
| Parameter-matched adapter diagnostic | 3 | 1 RTX 4090 GPU | 2–5 |
| GRL subject-confusion diagnostic | 15–30 | 1 RTX 4090 GPU | 3–8 |
| Compute and memory overhead benchmark | 2 one-epoch runs | 1 RTX 4090 GPU | $< 0.1$ |

training-time regularization and are omitted at inference time. Fully unsupervised disentanglement fine-tuning remains an important direction for future work.

**Ethics and reproducibility.** **DEFT** uses subject identities only during training to construct subject-aware contrastive pairs. Experiments use released EEG datasets and subject-disjoint downstream splits. The potential positive impact is more robust cross-subject EEG representation learning for research evaluation. Since EEG and MDD-related data may contain sensitive health-related information, the main risk is inappropriate clinical interpretation or privacy-sensitive use; **DEFT** should not be treated as a clinical diagnostic system without dataset-specific clinical validation. Code and configuration files are available at `https://github.com/wythedee/DEFT` to support reproducibility.

**Compute resources.** The reported experiments use independent single-GPU jobs on NVIDIA RTX 4090 GPUs with 24 GB memory. These jobs can be run in parallel on a multi-GPU machine; GPU-hours therefore measure aggregate compute, not required wall-clock time. Table 3 summarizes the compute needed to reproduce the main reported evaluation, the complete seed-0 evaluation suite, and the diagnostic experiments reported in the appendix.

**Dataset licenses and usage terms.** Table 4 summarizes the source and usage terms for the datasets in the complete evaluation suite. We use each dataset under its original terms, cite the original sources where available, and do not redistribute raw datasets or pretrained backbone checkpoints in the code release. Non-commercial, no-derivatives, and academic-only terms are reported as such and should not be interpreted as open commercial licenses. For datasets where no standard license name could be identified from the official source metadata, we report that status explicitly.

Table 4: Dataset sources and usage terms for the complete evaluation suite.

| Dataset | Source | License / usage terms |
|---|---|---|
| BCIC-IV-2a | BNCI Horizon 2020 dataset 001-2014, BCI Competition IV dataset 2a; `https://bnci-horizon-2020.eu/database/data-sets` | CC BY-ND 4.0. |
| SEED | SJTU BCMI Lab SEED collection; `https://bcmi.sjtu.edu.cn/home/seed/` | Application/signature required; academic research only; no commercial use. |
| SEED-IV | SJTU BCMI Lab SEED-IV collection; `https://bcmi.sjtu.edu.cn/home/seed/` | Application/signature required; academic research only; no commercial use. |
| SEED-V | SJTU BCMI Lab SEED-V collection; `https://bcmi.sjtu.edu.cn/home/seed/` | Application/signature required; academic research only; no commercial use. |
| FACED | FACED dataset, Synapse project syn50614194; `https://www.nature.com/articles/s41597-023-02650-w` | CC-BY metadata; the Scientific Data article is CC BY 4.0, with download terms governed by the Synapse page. |
| MI-KoreaU | Lee2019/OpenBMI, GigaDB DOI 10.5524/100542; `https://moabb.neurotechx.com/docs/generated/moabb.datasets.Lee2019_MI.html` | GPL 3.0 according to MOABB metadata. |
| MentalArith. | PhysioNet EEG During Mental Arithmetic Tasks; `https://physionet.org/content/eegmat/` | Open Data Commons Attribution License v1.0. |

| Dataset | Source | License / usage terms |
|---|---|---|
| PhysioNet-MI | PhysioNet EEG Motor Movement/Imagery Dataset; https://www.physionet.org/content/eegmmidb/ | Open Data Commons Attribution License v1.0. |
| Stieger2021-UD | Stieger et al. continuous SMR BCI learning dataset, up/down task split, figshare DOI 10.6084/m9.figshare.13123148.v1; https://moabb.neurotechx.com/docs/generated/moabb.datasets.Stieger2021.html | CC BY-NC 4.0. |
| Mumtaz2016 | figshare MDD Patients and Healthy Controls EEG Data, DOI 10.6084/m9.figshare.4244171; https://figshare.com/articles/dataset/EEG_Data_New/4244171 | CC BY 4.0. |
| BCIC-Speech | 2020 BCI Competition Track 3 imagined-speech dataset and associated OSF/NeMAR-style metadata; https://brain.korea.ac.kr/bci2020/competition.php | CC BY 4.0 according to NeMAR/EEGDash-style metadata for the matched speech-decoding dataset. |
| ISRUC-S1 | ISRUC-Sleep dataset, Subgroup 1; https://sleeptight.isr.uc.pt/ | Public/open-access dataset; no explicit standard license name identified in the official/source metadata. |

# C Dataset Details

## C.1 Dataset Statistics

Table 5 summarizes each dataset's standardized local sample tensor shape, number of classes, task type, and subject-disjoint split statistics. Shapes are listed before model-specific channel selection or backbone-specific reshaping.

Table 5: Dataset shapes, class counts, task types, and split statistics used in the reported experiments. Each dataset is shown with subject counts and sample counts in separate rows.

| Dataset | Local shape | # Cls. | Task type | Statistic | Train | Val. | Test |
|---|---|---|---|---|---|---|---|
| BCIC-IV-2a | $(22, 4, 200)$ | 4 | Motor imagery (multi.) | Subjects | 5 | 2 | 2 |
| | | | | Samples | 2784 | 1152 | 1152 |
| SEED | $(60, 4, 200)$ | 2 | Emotion (binary) | Subjects | 10 | 2 | 3 |
| | | | | Samples | 2700 | 540 | 810 |
| SEED-IV | $(60, 4, 200)$ | 4 | Emotion (multi.) | Subjects | 10 | 2 | 3 |
| | | | | Samples | 25050 | 5010 | 7515 |
| SEED-V | $(62, 1, 200)$ | 5 | Emotion (multi.) | Subjects | 5 | 5 | 5 |
| | | | | Samples | 36795 | 36795 | 36795 |
| FACED | $(32, 10, 200)$ | 9 | Emotion (multi.) | Subjects | 80 | 20 | 23 |
| | | | | Samples | 6720 | 1680 | 1932 |
| MI-KoreaU | $(62, 4, 200)$ | 2 | Motor imagery (binary) | Subjects | 42 | 5 | 7 |
| | | | | Samples | 16800 | 2000 | 2800 |
| MentalArith. | $(20, 5, 200)$ | 2 | Workload assessment (binary) | Subjects | 26 | 5 | 5 |
| | | | | Samples | 1247 | 240 | 220 |
| PhysioNet-MI | $(64, 4, 200)$ | 4 | Motor imagery (multi.) | Subjects | 70 | 19 | 20 |
| | | | | Samples | 6300 | 1734 | 1803 |
| Stieger2021-UD | $(60, 4, 200)$ | 2 | Motor imagery (binary) | Subjects | 40 | 9 | 9 |
| | | | | Samples | 89982 | 17776 | 19574 |
| Mumtaz2016 | $(19, 5, 200)$ | 2 | MDD classification (binary) | Subjects | 43 | 9 | 10 |
| | | | | Samples | 4891 | 1041 | 1151 |
| BCIC-Speech | $(60, 3, 200)$ | 3 | Speech decoding (multi.) | Subjects | 9 | 3 | 3 |
| | | | | Samples | 4050 | 1350 | 1350 |
| ISRUC-S1 | $(20, 6, 6000)$ | 5 | Sleep staging (multi.) | Subjects | 80 | 10 | 10 |
| | | | | Samples | 3559 | 468 | 435 |

For most datasets, local sample shapes follow $(\text{channels}, \text{temporal segments}, \text{samples per segment})$. ISRUC is stored as sleep-epoch sequences with shape $(\text{epochs}, \text{channels}, \text{samples})$.

### C.2 Backbone-Specific Preprocessing

The downstream experiments reuse the processed EEG inputs from the corresponding foundation-model pipelines without adding a separate raw-signal preprocessing stage for **DEFT**. Across datasets, recordings are converted into fixed-duration EEG tensors with labels, subject identifiers, and split keys. Most samples are resampled to 200 Hz and represented as one-second temporal patches; dataset-appropriate filtering and notch removal follow the original dataset or backbone pipeline when raw signals are available.

Backbone-specific preprocessing is limited to the model input interface. CodeBrain and CBraMod consume the shared LMDB or sequence-folder tensors after common amplitude scaling. CSBrain uses the same processed samples but reshapes them to match its tokenization convention; for BCIC-Speech, it uses the corrected 60-channel, three-window sample format. LaBraM additionally maps dataset channel names to its standard channel-index convention for channel embeddings. Thus, the comparisons use the same signal-level preprocessing whenever possible, while differences are limited to tensor shape, channel metadata, resampling rate, or tokenizer requirements.

## D Implementation Details

We use the public released checkpoints of CodeBrain, CBraMod, LaBraM, and CSBrain and do not re-train the EFM pretraining stage. Since public checkpoint documentation may not expose subject/session-level pretraining metadata for every downstream dataset, our claims focus on matched downstream adaptation comparisons under the same checkpoint, split, and trainable-backbone policy.

### D.1 Details of Recursive Gaussian Inference

Observation 3.1 in the main text focuses on counterpart removal and the resulting branch complementarity. Here we provide the full uncertainty-aware Gaussian update used to implement that residual refinement. The recursive Gaussian disentangler operates on the window-level observation sequence $\{(e_w, \Lambda_w)\}_{w=1}^{W}$ derived from $\bar{\mathbf{Z}}$. To reduce overlap between the two branches, each update uses a counterpart-removed observation in place of the raw mixed window. For a generic branch update, we write

$$\gamma_w = \sigma(a_w), \qquad \tilde{e}_w = e_w - \gamma_w r_w, \qquad \tilde{\Lambda}_w = \left(\Lambda_w^{-1} + (\gamma_w R_w)^{-1}\right)^{-1}, \tag{15}$$

where $r_w$ and $R_w$ denote the counterpart mean and precision from the opposite branch, and $\gamma_w \in (0, 1)$ is a learnable removal gate. This gate controls how much counterpart information is subtracted before the current branch is updated. All precision terms are implemented as positive diagonal vectors. In particular, the observation precision is parameterized from the predicted log-precision as $\Lambda_w = \mathrm{diag}(\exp(f_\lambda(\bar{z}_w)) + \epsilon)$ with a small $\epsilon > 0$, and the same positive-diagonal convention is used for the transition and counterpart precisions. The Gaussian inverses below are therefore computed elementwise and remain well defined.

Given the transition prior

$$h_w \mid h_{w-1} \sim \mathcal{N}(T_w h_{w-1}, \Pi_w^{-1}), \tag{16}$$

and the counterpart-removed observation likelihood

$$\tilde{e}_w \mid h_w \sim \mathcal{N}(h_w, \tilde{\Lambda}_w^{-1}), \tag{17}$$

the posterior is obtained by Gaussian fusion:

$$p(h_w \mid \tilde{e}_{\le w}) \propto \mathcal{N}(h_w; T_w h_{w-1}, \Pi_w^{-1}) \mathcal{N}(\tilde{e}_w; h_w, \tilde{\Lambda}_w^{-1}), \tag{18}$$

which yields

$$\Sigma_w = \left(\Pi_w + \tilde{\Lambda}_w\right)^{-1}, \qquad h_w = \Sigma_w \left(\Pi_w T_w h_{w-1} + \tilde{\Lambda}_w \tilde{e}_w\right). \tag{19}$$

This is the detailed Gaussian-fusion derivation referenced in Observation 3.1 of the main text: the posterior remains Gaussian, and its mean is a precision-weighted combination of temporal prediction and counterpart-removed observation.

The style branch uses a static transition $T_w = I$, while the content branch uses the dynamic transition

$$T_w = G_w = \sum_{n=1}^{N} \alpha_{w,n} G_n, \qquad \boldsymbol{\alpha}_w = \mathrm{Softmax}(g(h_{w-1})), \tag{20}$$

where $\{G_n\}_{n=1}^{N}$ are learnable transition bases. The full recursive procedure alternates the two branches in four steps:

$$\left(s_w^{(0)}, P_w^{(0)}\right) = \mathcal{G}_{\text{sta}}(e_w, \Lambda_w), \tag{21}$$

$$\left(c_w^{(0)}, Q_w^{(0)}\right) = \mathcal{G}_{\text{dyn}}(e_w, \Lambda_w; s_w^{(0)}, P_w^{(0)}), \tag{22}$$

$$(s_w, P_w) = \mathcal{G}_{\text{sta}}(e_w, \Lambda_w; c_w^{(0)}, Q_w^{(0)}), \tag{23}$$

$$(c_w, Q_w) = \mathcal{G}_{\text{dyn}}(e_w, \Lambda_w; s_w, P_w), \tag{24}$$

where the first two steps produce coarse factor estimates and the latter two refine them after counterpart removal. The resulting $\mathbf{C}^{\text{win}} = [c_1, \ldots, c_W]$ and $\mathbf{S}^{\text{win}} = [s_1, \ldots, s_W]$ are then reinjected into the token grid to form $\mathbf{C}$ and $\mathbf{S}$ in the main text.

For the reconstruction term in Section 3.4, we use a lightweight FiLM projection head. Given the sample-level style summary $\mathbf{s}^*$, the FiLM head predicts a per-dimension scale and bias

$$\lambda, b = g_{\text{film}}(\mathbf{s}^*), \tag{25}$$

and reconstructs the mixed window representation through

$$\hat{z}_w = \lambda \odot c_w + b. \tag{26}$$

This implements a simple subject-conditioned linear regression, where both regression parameters are generated from the style embedding. Because the reconstruction head is intentionally lightweight, it encourages a complementary information split: the content token $c_w$ carries the main content information, while the style embedding $\mathbf{s}^*$ modulates reconstruction through the FiLM scale and bias.

### D.2 Representation Geometry Metric

The visualization analysis in Section 4.4 uses a post-hoc within-subject class separation score. This metric is used only for representation analysis and is not part of the training objective. We compute it for the before-module mixed representation and the content-branch representation produced by **DEFT**.

For each dataset and representation type, let $e_i$ denote the sample-level representation of trial $i$, with subject label $s_i$ and task label $y_i$. We first remove the subject-specific mean:

$$\tilde{e}_i = e_i - \frac{1}{|\mathcal{I}_{s_i}|} \sum_{j \in \mathcal{I}_{s_i}} e_j, \tag{27}$$

where $\mathcal{I}_{s_i}$ is the set of samples from subject $s_i$. The centered representations are then standardized over the analysis set and projected with PCA to at most 50 dimensions, using $\min(50, N-1, D)$ components when the sample count or feature dimension is smaller.

For each valid subject with at least two observed task classes, we compute the centroid $\mu_{s,c}$ of each class $c$ and define

$$B_s = \frac{1}{|\mathcal{P}_s|} \sum_{(c_a, c_b) \in \mathcal{P}_s} \|\mu_{s,c_a} - \mu_{s,c_b}\|_2 , \tag{28}$$

where $\mathcal{P}_s$ contains all unordered pairs of classes observed for subject $s$. The within-class dispersion is

$$W_s = \frac{1}{|\mathcal{I}_s|} \sum_{i \in \mathcal{I}_s} \|z_i - \mu_{s,y_i}\|_2 , \tag{29}$$

where $z_i$ is the PCA-projected representation. The subject-level score is

$$R_s = \frac{B_s}{W_s + 10^{-8}}, \tag{30}$$

and the final dataset-level score is the average of $R_s$ over valid subjects. Larger values therefore indicate that task-class centroids are more separated relative to within-class dispersion after subject-level offsets have been removed. The t-SNE panels use the same centered, standardized, and PCA-projected representations, with balanced subsampling for visualization when needed.

Table 6: General hyperparameter settings used for disentangled fine-tuning.

| Hyperparameter | Value / policy |
|---|---|
| Optimizer | AdamW |
| Random seed | 0–2 for Table 1.<br>Complete results are reported under seed 0. |
| Weight decay | 0.05 |
| Dropout | 0.1 |
| Label smoothing | 0.1 |
| Task loss weight | 1.0 |
| Adapter hidden dimension | 200 |
| Gaussian transition components | 16 |
| Remove-gate initialization | -2.0 |
| Contrastive temperature | 0.07 |
| Variance-ratio target | 2.0 |

Table 7: Representative backbone-specific optimization and loss-weight settings. The four $\lambda$ columns correspond directly to Eq. 14.

| Backbone | Learning rate | Batch size | Epochs | $\lambda_{\mathrm{ord}}$ | $\lambda_{\mathrm{sty}}$ | $\lambda_{\mathrm{con}}$ | $\lambda_{\mathrm{rec}}$ |
|---|---|---|---|---|---|---|---|
| CodeBrain | $1 \times 10^{-4}$ | 64 | 20 | 0.005 | 0.001 | 0.02 | 0.01 |
| CBraMod | $1 \times 10^{-4}$ | 64 | 20 | 0.1 | 0.01–0.02 | 0.2 | 0.02 |
| LaBraM | $1 \times 10^{-4}$ | 64 | 20 | 0.1 | 0.02 | 0.2 | 0.02 |
| CSBrain | $1 \times 10^{-4}$ | 64 | 20 | 0.1 | 0.01 | 0.01 | 0.02 |

### D.3 Training Hyperparameters

Hyperparameters and checkpoints are selected only on the validation split associated with each subject-disjoint partition. "Representative" denotes a backbone-level default together with the small dataset-level ranges shown below. For a new EFM, we start from the default configuration and tune the loss weights over the same small validation-only range. The released code includes configuration files for each backbone and dataset. Table 6 and Table 7 summarize the paper-facing hyperparameter settings used in the reported experiments. Across backbones, **DEFT** receives the pretrained token grid with shape $(B, M, W, D)$, performs Gaussian inference on channel-pooled windows with shape $(B, W, D)$, reinjects the resulting content/style corrections into the token grid, and feeds only the content branch to the prediction head.

### D.4 Parameter Overhead

**DEFT** reuses the pretrained backbone and downstream prediction head from the matched fine-tuning baseline, and only adds the post-backbone disentanglement adapter. To quantify the overhead, Table 8 reports both the added adapter parameters and their ratio to the original fine-tuning model size, where the original model includes the pretrained backbone and the downstream prediction head. Across these settings, the active adapter adds 2.07M parameters and accounts for 1.55%–10.81% of the original model size. The percentage varies because the downstream prediction head changes with the task interface and label space, while the CBraMod backbone and the adapter size remain fixed.

### D.5 Compute and Memory Overhead

**DEFT** adds overhead through the post-backbone adapter and auxiliary training objectives. At inference time, subject identities are not required, and prediction is performed through the content branch. Table 9 summarizes the measured parameter, memory, and one-step training FLOP overhead on a representative CBraMod setting.

Table 8: Parameter overhead of **DEFT** on the six representative CBraMod settings. The original model is the matched vanilla fine-tuning model, including the pretrained backbone and downstream prediction head.

| Dataset | Original model | Added adapter | Overhead |
|---|---|---|---|
| BCIC-IV-2a | 19.13M | 2.07M | 10.81% |
| FACED | 133.29M | 2.07M | 1.55% |
| ISRUC-S1 | 26.47M | 2.07M | 7.81% |
| MentalArith. | 25.09M | 2.07M | 8.24% |
| BCIC-Speech | 26.61M | 2.07M | 7.77% |
| Mumtaz2016 | 24.09M | 2.07M | 8.58% |

Table 9: Compute and memory overhead diagnostic for a representative CBraMod setting. FLOPs are measured for one training step.

| Setting | Trainable params | Peak allocated memory | Peak reserved memory | One-step train FLOPs |
|---|---|---|---|---|
| Vanilla | 19.13M | 879.7 MiB | 1216.0 MiB | 85.54G |
| **DEFT** | 21.44M | 970.1 MiB | 1300.0 MiB | 93.86G |
| Overhead | +2.32M (+12.11%) | +90.4 MiB (+10.28%) | +84.0 MiB (+6.91%) | +8.33G (+9.74%) |

# E Complete Main Results

To assess broader coverage, this section reports seed-0 results across the complete evaluation suite. These results are intended as coverage evidence, while Table 1 remains the primary multi-seed quantitative comparison. Tables 10–13 keep one backbone fixed and report all recorded metrics for vanilla fine-tuning and the **DEFT** variant, so the comparison remains within the same pretrained representation.

CodeBrain shows clear improvements on BCIC-IV-2a, FACED, MentalArith., BCIC-Speech, and ISRUC-S1, covering motor imagery, emotion recognition, cognitive-state decoding, speech decoding, and sleep staging. The remaining lower secondary metrics are generally smaller than the gains on the representative datasets.

Table 10: Complete main results for CodeBrain across the evaluation suite and recorded metrics. Positive deltas are shaded orange, negative deltas are shaded blue, and the better value between Vanilla and **DEFT** is bold.

| **Setting** | **BCIC-IV-2a** | | | **SEED** | | | **SEED-IV** | | |
|---|---|---|---|---|---|---|---|---|---|
| | B-ACC | Kappa | F1 | B-ACC | ROC-AUC | PR-AUC | B-ACC | Kappa | F1 |
| Vanilla | 0.3602 | 0.1470 | 0.2815 | 0.6827 | 0.7499 | 0.7559 | 0.3403 | 0.1151 | 0.2982 |
| **DEFT** | **0.4740** | **0.2986** | **0.4523** | **0.6914** | 0.7750 | 0.8068 | **0.3515** | **0.1277** | **0.3025** |
| Δ | +0.1138 | +0.1516 | +0.1708 | +0.0087 | +0.0251 | +0.0508 | +0.0112 | +0.0126 | +0.0043 |

| **Setting** | **SEED-V** | | | **FACED** | | | **MI-KoreaU** | | |
|---|---|---|---|---|---|---|---|---|---|
| | B-ACC | Kappa | F1 | B-ACC | Kappa | F1 | B-ACC | ROC-AUC | PR-AUC |
| Vanilla | 0.2672 | 0.0835 | **0.2651** | 0.4866 | 0.4212 | 0.4864 | 0.7761 | 0.8324 | 0.7926 |
| **DEFT** | **0.2743** | **0.0919** | 0.2618 | **0.5784** | **0.5217** | **0.5761** | **0.8096** | 0.8683 | 0.8707 |
| Δ | +0.0071 | +0.0084 | -0.0033 | +0.0918 | +0.1005 | +0.0897 | +0.0335 | +0.0359 | +0.0781 |

| **Setting** | **MentalArith.** | | | **PhysioNet** | | | **Stieger2021-UD** | | |
|---|---|---|---|---|---|---|---|---|---|
| | B-ACC | ROC-AUC | PR-AUC | B-ACC | ROC-AUC | PR-AUC | B-ACC | Kappa | F1 |
| Vanilla | 0.6938 | 0.7248 | 0.5368 | 0.6157 | 0.4875 | 0.6177 | 0.7566 | **0.8545** | **0.8623** |
| **DEFT** | **0.7583** | **0.8266** | **0.6781** | **0.6228** | **0.4971** | **0.6256** | **0.7632** | 0.8422 | 0.8518 |
| Δ | +0.0645 | +0.1018 | +0.1413 | +0.0071 | +0.0096 | +0.0079 | +0.0066 | -0.0123 | -0.0105 |

| **Setting** | **Mumtaz2016** | | | **BCIC-Speech** | | | **ISRUC-S1** | | |
|---|---|---|---|---|---|---|---|---|---|
| | B-ACC | ROC-AUC | PR-AUC | B-ACC | Kappa | F1 | B-ACC | Kappa | F1 |
| Vanilla | 0.9043 | 0.9732 | **0.9796** | 0.4916 | 0.2373 | 0.4865 | 0.7698 | 0.7175 | 0.7783 |
| **DEFT** | **0.9110** | **0.9742** | 0.9784 | **0.5116** | **0.2673** | **0.5119** | **0.7703** | **0.7437** | **0.7985** |
| Δ | +0.0067 | +0.0010 | -0.0012 | +0.0200 | +0.0300 | +0.0254 | +0.0005 | +0.0262 | +0.0202 |

Table 11: Complete main results for CBraMod across the evaluation suite and recorded metrics. Positive deltas are shaded orange, negative deltas are shaded blue, and the better value between Vanilla and **DEFT** is bold.

| **Setting** | **BCIC-IV-2a** | | | **SEED** | | | **SEED-IV** | | |
|---|---|---|---|---|---|---|---|---|---|
| | B-ACC | Kappa | F1 | B-ACC | ROC-AUC | PR-AUC | B-ACC | Kappa | F1 |
| Vanilla | 0.4870 | 0.3160 | 0.4650 | 0.7383 | 0.8346 | 0.8489 | 0.3773 | 0.1601 | 0.3301 |
| **DEFT** | **0.5981** | **0.4641** | **0.5954** | **0.7877** | **0.8589** | **0.8647** | **0.3989** | **0.1897** | **0.3444** |
| Δ | +0.1111 | +0.1481 | +0.1304 | +0.0494 | +0.0243 | +0.0158 | +0.0216 | +0.0296 | +0.0143 |

| **Setting** | **SEED-V** | | | **FACED** | | | **MI-KoreaU** | | |
|---|---|---|---|---|---|---|---|---|---|
| | B-ACC | Kappa | F1 | B-ACC | Kappa | F1 | B-ACC | ROC-AUC | PR-AUC |
| Vanilla | 0.2465 | 0.0600 | 0.2490 | 0.5451 | 0.4830 | 0.5436 | 0.8343 | 0.9164 | 0.9181 |
| **DEFT** | **0.2734** | **0.0970** | **0.2746** | **0.6126** | **0.5605** | **0.6128** | **0.8450** | **0.9170** | **0.9221** |
| Δ | +0.0269 | +0.0370 | +0.0256 | +0.0675 | +0.0775 | +0.0692 | +0.0107 | +0.0006 | +0.0040 |

| **Setting** | **MentalArith.** | | | **PhysioNet** | | | **Stieger2021-UD** | | |
|---|---|---|---|---|---|---|---|---|---|
| | B-ACC | ROC-AUC | PR-AUC | B-ACC | ROC-AUC | PR-AUC | B-ACC | Kappa | F1 |
| Vanilla | 0.6521 | 0.7408 | 0.5609 | 0.6095 | 0.4794 | 0.6133 | 0.7683 | **0.8591** | **0.8666** |
| **DEFT** | **0.7260** | **0.8376** | **0.6743** | **0.6424** | **0.5230** | **0.6413** | **0.7730** | 0.8575 | 0.8633 |
| Δ | +0.0739 | +0.0968 | +0.1134 | +0.0329 | +0.0436 | +0.0280 | +0.0047 | -0.0016 | -0.0033 |

| **Setting** | **Mumtaz2016** | | | **BCIC-Speech** | | | **ISRUC-S1** | | |
|---|---|---|---|---|---|---|---|---|---|
| | B-ACC | ROC-AUC | PR-AUC | B-ACC | Kappa | F1 | B-ACC | Kappa | F1 |
| Vanilla | 0.9423 | 0.9902 | 0.9902 | 0.5053 | 0.2580 | 0.5060 | 0.7889 | **0.7598** | **0.8123** |
| **DEFT** | **0.9644** | **0.9943** | **0.9958** | **0.5116** | **0.2673** | **0.5124** | **0.7925** | 0.7423 | 0.8049 |
| Δ | +0.0221 | +0.0041 | +0.0056 | +0.0063 | +0.0093 | +0.0064 | +0.0036 | -0.0175 | -0.0074 |

CBraMod shows the strongest breadth pattern in the seed-0 records, with clear gains across several task families and a small number of metric-level regressions.

Table 12: Complete main results for LaBraM across the evaluation suite and recorded metrics. Positive deltas are shaded orange, negative deltas are shaded blue, and the better value between Vanilla and **DEFT** is bold.

| **Setting** | **BCIC-IV-2a** | | | **SEED** | | | **SEED-IV** | | |
|---|---|---|---|---|---|---|---|---|---|
| | B-ACC | Kappa | F1 | B-ACC | ROC-AUC | PR-AUC | B-ACC | Kappa | F1 |
| Vanilla | 0.5312 | 0.3750 | 0.5226 | 0.7630 | 0.8261 | 0.8033 | 0.4078 | 0.2008 | 0.3540 |
| **DEFT** | **0.5703** | **0.4271** | **0.5554** | **0.7667** | **0.8288** | **0.8154** | **0.4132** | **0.2086** | **0.3584** |
| Δ | +0.0391 | +0.0521 | +0.0328 | +0.0037 | +0.0027 | +0.0121 | +0.0054 | +0.0078 | +0.0044 |

| **Setting** | **SEED-V** | | | **FACED** | | | **MI-KoreaU** | | |
|---|---|---|---|---|---|---|---|---|---|
| | B-ACC | Kappa | F1 | B-ACC | Kappa | F1 | B-ACC | ROC-AUC | PR-AUC |
| Vanilla | 0.2716 | 0.0904 | 0.2648 | 0.4787 | 0.4108 | 0.4792 | 0.8546 | 0.9186 | 0.9201 |
| **DEFT** | **0.2825** | **0.1017** | **0.2761** | **0.5030** | **0.4380** | **0.5060** | **0.8707** | **0.9427** | **0.9436** |
| Δ | +0.0109 | +0.0113 | +0.0113 | +0.0243 | +0.0272 | +0.0268 | +0.0161 | +0.0241 | +0.0235 |

| **Setting** | **MentalArith.** | | | **PhysioNet** | | | **Stieger2021-UD** | | |
|---|---|---|---|---|---|---|---|---|---|
| | B-ACC | ROC-AUC | PR-AUC | B-ACC | ROC-AUC | PR-AUC | B-ACC | Kappa | F1 |
| Vanilla | 0.7250 | 0.7611 | 0.6092 | 0.6252 | 0.5001 | 0.6253 | 0.7864 | **0.8732** | **0.8787** |
| **DEFT** | **0.7312** | **0.8191** | **0.6904** | **0.6495** | **0.5326** | **0.6514** | **0.7921** | 0.8725 | 0.8784 |
| Δ | +0.0062 | +0.0580 | +0.0812 | +0.0243 | +0.0325 | +0.0261 | +0.0057 | -0.0007 | -0.0003 |

| **Setting** | **Mumtaz2016** | | | **BCIC-Speech** | | | **ISRUC-S1** | | |
|---|---|---|---|---|---|---|---|---|---|
| | B-ACC | ROC-AUC | PR-AUC | B-ACC | Kappa | F1 | B-ACC | Kappa | F1 |
| Vanilla | 0.9215 | **0.9879** | **0.9892** | 0.4791 | 0.2187 | 0.4680 | 0.7927 | 0.7566 | 0.8088 |
| **DEFT** | **0.9262** | 0.9875 | 0.9890 | **0.4947** | **0.2420** | **0.4893** | **0.8051** | **0.7792** | **0.8288** |
| Δ | +0.0047 | -0.0004 | -0.0002 | +0.0156 | +0.0233 | +0.0213 | +0.0124 | +0.0226 | +0.0200 |

LaBraM gives more conservative gains, which is consistent with its stronger vanilla starting point in the main benchmark. Its gains are most visible on BCIC-IV-2a, FACED, MI-KoreaU, PhysioNet, BCIC-Speech, and ISRUC-S1, while a few probabilistic metrics remain challenging under this backbone.

Table 13: Complete main results for CSBrain across the evaluation suite and recorded metrics. Positive deltas are shaded orange, negative deltas are shaded blue, and the better value between Vanilla and **DEFT** is bold.

| **Setting** | **BCIC-IV-2a** | | | **SEED** | | | **SEED-IV** | | |
|---|---|---|---|---|---|---|---|---|---|
| | B-ACC | Kappa | F1 | B-ACC | ROC-AUC | PR-AUC | B-ACC | Kappa | F1 |
| Vanilla | 0.4792 | 0.3056 | 0.4545 | 0.7160 | 0.7920 | **0.8223** | 0.3341 | 0.1009 | 0.2590 |
| **DEFT** | **0.5000** | **0.3333** | **0.4785** | **0.7370** | **0.8106** | 0.8109 | **0.3357** | **0.1090** | **0.2840** |
| Δ | +0.0208 | +0.0278 | +0.0240 | +0.0210 | +0.0185 | -0.0115 | +0.0016 | +0.0080 | +0.0250 |

| **Setting** | **SEED-V** | | | **FACED** | | | **MI-KoreaU** | | |
|---|---|---|---|---|---|---|---|---|---|
| | B-ACC | Kappa | F1 | B-ACC | Kappa | F1 | B-ACC | ROC-AUC | PR-AUC |
| Vanilla | 0.2605 | 0.0785 | 0.2605 | 0.5706 | 0.5143 | 0.5786 | 0.8200 | 0.9045 | 0.9067 |
| **DEFT** | **0.2699** | **0.0887** | **0.2636** | **0.5790** | **0.5230** | **0.5830** | **0.8407** | **0.9188** | **0.9196** |
| Δ | +0.0094 | +0.0102 | +0.0032 | +0.0085 | +0.0087 | +0.0044 | +0.0207 | +0.0143 | +0.0129 |

| **Setting** | **MentalArith.** | | | **PhysioNet** | | | **Stieger2021-UD** | | |
|---|---|---|---|---|---|---|---|---|---|
| | B-ACC | ROC-AUC | PR-AUC | B-ACC | ROC-AUC | PR-AUC | B-ACC | Kappa | F1 |
| Vanilla | 0.5135 | **0.7292** | 0.4784 | 0.6273 | 0.5030 | 0.6307 | 0.7425 | 0.8510 | 0.8548 |
| **DEFT** | **0.6115** | 0.7229 | **0.4931** | **0.6367** | **0.5156** | **0.6382** | **0.7752** | **0.8637** | **0.8697** |
| Δ | +0.0979 | -0.0062 | +0.0147 | +0.0094 | +0.0126 | +0.0075 | +0.0327 | +0.0127 | +0.0149 |

| **Setting** | **Mumtaz2016** | | | **BCIC-Speech** | | | **ISRUC-S1** | | |
|---|---|---|---|---|---|---|---|---|---|
| | B-ACC | ROC-AUC | PR-AUC | B-ACC | Kappa | F1 | B-ACC | Kappa | F1 |
| Vanilla | **0.8995** | 0.9796 | 0.9825 | 0.4907 | 0.2360 | 0.4905 | 0.7698 | 0.7175 | 0.7783 |
| **DEFT** | 0.8927 | **0.9862** | **0.9873** | **0.5031** | **0.2547** | **0.5039** | **0.7700** | **0.7376** | **0.7947** |
| Δ | -0.0068 | +0.0066 | +0.0048 | +0.0124 | +0.0187 | +0.0134 | +0.0002 | +0.0201 | +0.0164 |

CSBrain also benefits from disentangled fine-tuning across a broad set of datasets, with clear gains on BCIC-IV-2a, MI-KoreaU, MentalArith., BCIC-Speech, and ISRUC-S1. Its remaining lower values are limited to isolated secondary metrics or high-baseline binary settings.

Overall, the complete seed-0 records support the broader applicability of **DEFT** and also reveal that gains vary with dataset, backbone, and metric choice. The remaining decreases are concentrated in isolated high-baseline or score-calibration-sensitive metrics, suggesting broad improvement in task-oriented decoding across backbones.

Figure 5 further compares the training dynamics of vanilla fine-tuning and **DEFT** on CBraMod. For each dataset and method, the recorded training loss is normalized by the first-epoch loss, i.e., $\tilde{\mathcal{L}}_t = \mathcal{L}_t / \mathcal{L}_1$. Under this normalized view, **DEFT** decreases faster and reaches a lower final loss on the four diagnostic datasets. This pattern is consistent with the role of the auxiliary constraints: subject-aware contrast, temporal-variance shaping, and reconstruction provide additional training signals that guide the adapter toward a more structured content branch during fine-tuning.

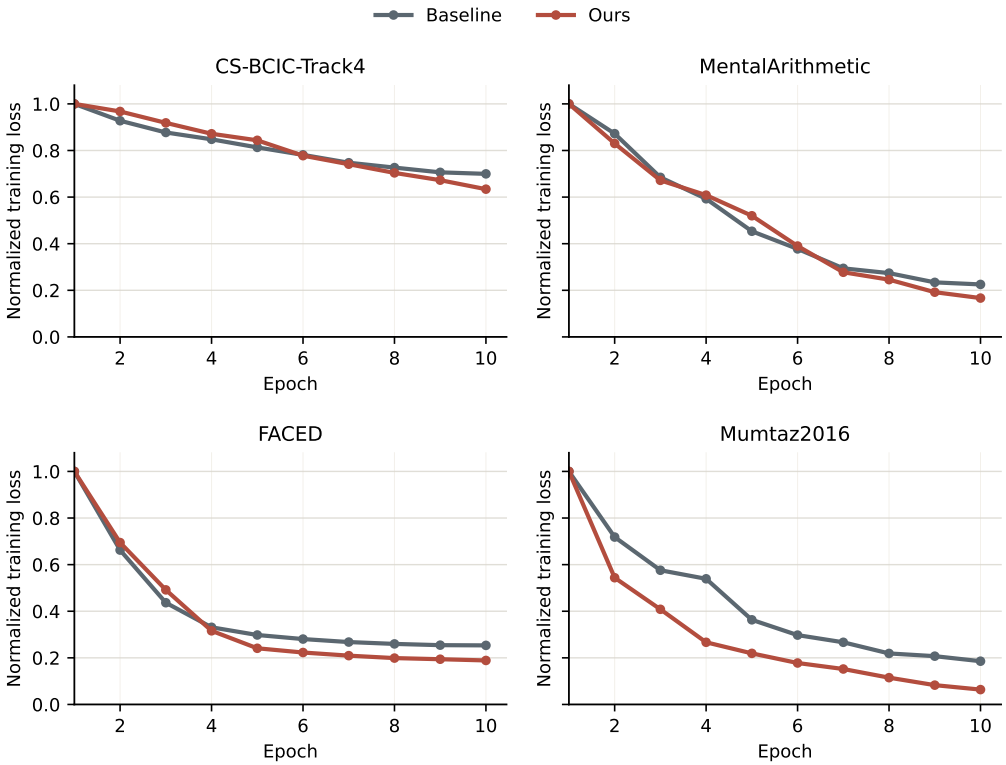



Figure 5: **Normalized training loss curves on CBraMod.** Each curve is divided by its first-epoch loss. The auxiliary constraints in **DEFT** provide additional optimization signals and lead to faster normalized loss reduction on the shown datasets.

Table 14: Frozen linear probes on content and style. All values are B-ACC. Task chance is determined by the dataset class count.

| Backbone | Dataset | Task chance | Content task | Style task | Content subject | Style subject |
|---|---|---|---|---|---|---|
| CBraMod | FACED | 0.1111 | **0.2615** | 0.2458 | 0.9206 | **0.9451** |
| CBraMod | BCIC-IV-2a | 0.2500 | **0.3409** | 0.3183 | 0.8222 | **0.8252** |
| CBraMod | MentalArith. | 0.5000 | **0.6705** | 0.6503 | 0.1734 | **0.1770** |
| CBraMod | BCIC-Speech | 0.3333 | **0.3958** | 0.3849 | 0.3364 | **0.3377** |
| CodeBrain | FACED | 0.1111 | **0.2684** | 0.2595 | 0.9705 | **0.9716** |
| CBraMod | Mumtaz2016 | 0.5000 | **0.9759** | 0.9298 | 0.0633 | **0.1999** |

Table 15: Stable multi-seed capacity controls. Values are test B-ACC mean±std.

| Backbone | Dataset | Generic Adapter | DEFT |
|---|---|---|---|
| CBraMod | BCIC-IV-2a | 0.4806±0.0403 | **0.5579±0.0310** |
| CBraMod | MentalArith. | 0.6597±0.0371 | **0.7292±0.0059** |
| LaBraM | BCIC-IV-2a | 0.5483±0.0067 | **0.5671±0.0032** |
| LaBraM | MentalArith. | 0.6389±0.0620 | **0.7236±0.0149** |

# F Extensive Experiments

## F.1 Content/Style Probing Diagnostic

Table 14 evaluates the branch outputs immediately before the prediction head. We freeze the trained backbone and disentangler, mean-pool the token grid, and fit one-layer linear probes. Task probes follow the subject-disjoint downstream split; subject probes use held-out trials within a closed-set identity space. Probe checkpoints are selected on the corresponding validation split.

Across all six settings, content yields higher task-probe B-ACC and style yields higher subject-probe B-ACC. On Mumtaz2016, the original/content/style task probes are 0.9224/0.9759/0.9298, and the corresponding subject probes are 0.3720/0.0633/0.1999. These results quantify the functional specialization of the branches.

## F.2 Parameter and Sampler Controls

The Generic Adapter is a parameter-matched MLP placed between the backbone and classifier. It retains the task loss and uses the same subject-disjoint split, subject-balanced sampler, and backbone update policy as **DEFT**.

On CBraMod–FACED, Vanilla, a doubled classifier, Vanilla with the subject-balanced sampler, and **DEFT** obtain B-ACC of 0.5387, 0.5363, 0.5441, and 0.6005, respectively. The comparison separates classifier capacity and sampling from the full adapter.

## F.3 Subject-Aware and Sequential Baselines

Gradient Reversal Layer (GRL) [16] adds a subject classifier after the EFM tokens and reverses its gradient to learn a subject-confused representation. FHVAE-token [20] and DSAE-token [33] implement sequential static/dynamic factorization at the same token interface. All methods use the same subject-disjoint splits and samplers.

On BCIC-IV-2a, GRL/FHVAE-token obtain 0.4754±0.0300/0.4465±0.0586 with CBraMod and 0.5139±0.0115/0.5243±0.0270 with LaBraM; the corresponding **DEFT** values are 0.5579±0.0310 and 0.5671±0.0032.

Table 16: Multi-seed subject-aware and sequential baselines on FACED. Values are test B-ACC mean±std.

| Backbone | GRL | FHVAE-token | DSAE-token | DEFT |
|---|---|---|---|---|
| CBraMod | 0.5573±0.0028 | 0.5883±0.0052 | 0.5848±0.0019 | **0.6005±0.0101** |
| LaBraM | 0.5261±0.0151 | 0.4709±0.0070 | 0.4828±0.0031 | **0.5409±0.0086** |

Table 17: Same-loss architecture and recursion-depth controls on FACED. Values are test B-ACC mean±std.

| Backbone | MLP | Attention | 2 steps | 4-step DEFT | 6 steps |
|---|---|---|---|---|---|
| CBraMod | 0.5794±0.0045 | 0.5769±0.0083 | 0.5957±0.0008 | **0.6005±0.0101** | 0.5932±0.0003 |
| LaBraM | 0.5081±0.0098 | 0.5068±0.0085 | 0.5122±0.0060 | **0.5409±0.0086** | 0.5208±0.0058 |

Table 18: Temporal-prior sensitivity on ISRUC-S1 under a fixed multi-seed diagnostic protocol.

| Setting | Default | w/o Invar. | w/o VarRatio | Weight 0.5× | Weight 2× |
|---|---|---|---|---|---|
| Test B-ACC | **0.7922±0.0015** | 0.7849±0.0019 | 0.7838±0.0121 | 0.7809±0.0133 | 0.7815±0.0035 |

Table 19: Objective-combination controls on CBraMod. Values are test B-ACC under the stated diagnostic protocol.

| Dataset | Vanilla | Variance only | Variance + contrast | Variance + rec. | Full |
|---|---|---|---|---|---|
| FACED | 0.5387 | 0.5961 | 0.6040 | 0.5805 | **0.6126** |
| MentalArith. | 0.6521 | 0.5740 | – | – | **0.7260** |

Table 20: LaBraM component controls. Values are test B-ACC.

| Dataset | Full | w/o Invar. | w/o Gaussian | w/o Rec. |
|---|---|---|---|---|
| FACED | **0.5480** | 0.5425 | 0.4936 | 0.5157 |
| MentalArith. | **0.7417** | 0.6729 | 0.6365 | 0.6635 |

### F.4 RGD Isolation and Sensitivity

The same-loss controls retain all four training objectives and replace RGD with a parameter-matched MLP or attention factorizer. The attention and RGD variants contain 1,624,444/1,624,835 parameters for CBraMod and 350,868/351,235 for LaBraM.

The attention control obtains 0.4395/0.5208 on BCIC-IV-2a and 0.6017/0.7156 on MentalArith. with CBraMod/LaBraM, compared with **DEFT** at 0.5579/0.5671 and 0.7292/0.7236. Fixed gate, fixed precision, static transition, and shared transition reduce FACED B-ACC by 0.0104–0.0305 for CBraMod and 0.0226–0.0404 for LaBraM. Four refinement steps give the highest mean B-ACC in the tested settings.

### F.5 Objective and Cross-Backbone Controls

The objective contributions vary by task. Variance-only improves FACED, while the full objective gives the highest result on both datasets.

### F.6 Additional Backbone and Dataset Checks

REVE [14] exposes a post-backbone token sequence compatible with the same adapter interface. Table 21 reports three subject-disjoint compatibility checks.

We also evaluate a trait-level EEG target and one non-EEG time-series setting. OpenNeuro Depression REST uses dataset ds003478 v1.1.0 [3]. The external check uses MOMENT representations [17] on UCI HAR [1] with a subject-disjoint split.

Table 21: REVE compatibility checks. Values are test B-ACC.

| Dataset | REVE Vanilla | REVE + DEFT | Δ |
|---|---|---|---|
| MentalArith. | 0.6531 | **0.7083** | +0.0552 |
| BCIC-Speech | 0.6215 | **0.6393** | +0.0178 |
| BCIC-IV-2a | 0.5712 | **0.5773** | +0.0061 |

Table 22: Additional subject-disjoint checks. Values are mean±std.

| Setting | Method | B-ACC | ROC-AUC | PR-AUC / added params |
|---|---|---|---|---|
| OpenNeuro REST | Vanilla | 0.4968±0.0028 | 0.4150±0.0105 | 0.2712±0.0035 |
| OpenNeuro REST | **DEFT** | **0.5439±0.0278** | **0.6623±0.0137** | **0.4276±0.0227** |
| MOMENT/UCI HAR | Vanilla | 0.5665±0.0078 | – | 0 |
| MOMENT/UCI HAR | Generic MLP | 0.7027±0.0134 | – | 2,965,836 |
| MOMENT/UCI HAR | **DEFT** | **0.7742±0.0249** | – | 2,966,051 |

Table 23: Frozen subject-probe B-ACC across backbones.

| Dataset | Raw EEG | CBraMod | CodeBrain | CSBrain | LaBraM |
|---|---|---|---|---|---|
| FACED | 0.2297 | 0.6381 | 0.7580 | 0.5611 | **0.8598** |
| BCIC-Speech | 0.2434 | 0.3450 | 0.2666 | **0.3464** | 0.2874 |

Table 24: Effective contrastive-anchor coverage and mean positives per anchor.

| Dataset | Random anchors | DEFT sampler | Positives/anchor |
|---|---|---|---|
| BCIC-IV-2a | 99.85% | 100% | 6.9701 |
| ISRUC-S1 | 7.78% | 100% | 1.0736 |
| Mumtaz2016 | 53.17% | 100% | 1.7368 |
| FACED | 16.55% | 100% | 1.1593 |
| MentalArith. | 74.56% | 100% | 2.4592 |
| BCIC-Speech | 99.92% | 100% | 7.8036 |

Table 25: One-second versus full-sample task and subject probes. All values are B-ACC.

| Dataset | Task chance | 1-s task | Full task | Subject chance | 1-s subject | Full subject |
|---|---|---|---|---|---|---|
| MentalArith. | 0.5000 | 0.5585 | **0.6316** | 0.0280 | 0.2104 | **0.2570** |
| BCIC-IV-2a | 0.2500 | 0.3073 | **0.4630** | 0.1111 | 0.6406 | **0.6550** |

These checks cover one additional EFM, one independent MDD dataset, and one pretrained time-series model/benchmark pair.

### F.7 Probe and Sampler Diagnostics

Frozen subject probes show that the accessibility of subject identity depends on both backbone and dataset.

The subject-balanced sampler gives every anchor at least one same-subject positive in the six main datasets. Anchors without positives are excluded from $\mathcal{L}_{\mathrm{con}}$ as defined in Section 3.4.

### F.8 Temporal and Statistical Diagnostics

Full samples accumulate more task evidence than one-second windows in both datasets, while subject information is already accessible from short windows.

At the measured checkpoint, $\mathcal{L}_{\mathrm{ord}}$ and $\mathcal{L}_{\mathrm{sty}}$ have nonzero raw gradient norms of 9.9960 and 0.4042. Their weighted RGD-gradient/task-gradient ratios are 0.0167 and $5.41 \times 10^{-5}$, with task-gradient cosine similarities of 0.2420 and 0.1246.

Table 26: Confidence intervals for the reviewer-identified secondary-metric differences.

| Setting | Metric | Δ | Welch 95% CI |
|---|---|---|---|
| CBraMod–ISRUC-S1 | Kappa | -0.0229 | $[-0.0475, 0.0017]$ |
| CBraMod–ISRUC-S1 | F1 | -0.0145 | $[-0.0374, 0.0084]$ |
| CodeBrain–Mumtaz2016 | ROC-AUC | -0.0074 | $[-0.0368, 0.0220]$ |
| CodeBrain–Mumtaz2016 | PR-AUC | -0.0045 | $[-0.0236, 0.0146]$ |
| LaBraM–FACED | Kappa | +0.0093 | $[-0.0114, 0.0300]$ |
| LaBraM–FACED | F1 | +0.0112 | $[-0.0088, 0.0312]$ |

### F.9 Additional CBraMod Ablation Results

Table 27 retains the seed-0 metric-level CBraMod ablations over the complete evaluation suite in two panels: style-contrast and variance constraints in Panel A, and Gaussian-disentangler and reconstruction ablations in Panel B. Table 2 is the multi-seed B-ACC comparison used for the main conclusions.

Table 27: Complete seed-0 CBraMod ablations, Panel A: style contrast and variance constraints. Orange/blue cells mark positive/negative deltas; bold marks the best value.

| Dataset | Metric | Full | w/o StyleCL | Δ | w/o Invar | Δ | w/o VarRatio | Δ |
|---|---|---|---|---|---|---|---|---|
| BCIC-IV-2a | B-ACC | **0.5981** | 0.4844 | -0.1137 | 0.3403 | -0.2578 | 0.3715 | -0.2266 |
| | Kappa | **0.4641** | 0.3125 | -0.1516 | 0.1204 | -0.3438 | 0.1620 | -0.3021 |
| | F1 | **0.5954** | 0.4673 | -0.1281 | 0.2757 | -0.3197 | 0.3043 | -0.2910 |
| BCIC-Speech | B-ACC | **0.5116** | 0.4880 | -0.0236 | 0.4911 | -0.0204 | 0.4778 | -0.0338 |
| | Kappa | **0.2673** | 0.2320 | -0.0353 | 0.2367 | -0.0307 | 0.2167 | -0.0507 |
| | F1 | **0.5124** | 0.4873 | -0.0251 | 0.4910 | -0.0214 | 0.4758 | -0.0366 |
| FACED | B-ACC | **0.6126** | 0.5837 | -0.0289 | 0.6005 | -0.0121 | 0.5993 | -0.0133 |
| | Kappa | **0.5605** | 0.5302 | -0.0303 | 0.5451 | -0.0154 | 0.5448 | -0.0157 |
| | F1 | **0.6128** | 0.5949 | -0.0179 | 0.5998 | -0.0130 | 0.6019 | -0.0109 |
| Stieger2021-UD | B-ACC | **0.7730** | 0.7653 | -0.0077 | 0.7629 | -0.0101 | 0.7688 | -0.0042 |
| | ROC-AUC | **0.8575** | 0.8566 | -0.0009 | 0.8506 | -0.0069 | 0.8546 | -0.0029 |
| | PR-AUC | 0.8633 | **0.8659** | +0.0026 | 0.8574 | -0.0059 | 0.8637 | +0.0004 |
| MI-KoreaU | B-ACC | **0.8450** | 0.8243 | -0.0207 | 0.8114 | -0.0336 | 0.8275 | -0.0175 |
| | ROC-AUC | **0.9170** | 0.9021 | -0.0150 | 0.8991 | -0.0179 | 0.9053 | -0.0117 |
| | PR-AUC | **0.9221** | 0.9067 | -0.0153 | 0.9057 | -0.0163 | 0.9098 | -0.0122 |
| MentalArith. | B-ACC | **0.7260** | 0.5990 | -0.1271 | 0.7104 | -0.0156 | 0.6281 | -0.0979 |
| | ROC-AUC | **0.8376** | 0.6326 | -0.2050 | 0.8133 | -0.0243 | 0.7009 | -0.1367 |
| | PR-AUC | **0.6743** | 0.4364 | -0.2379 | 0.6101 | -0.0642 | 0.5255 | -0.1488 |
| Mumtaz2016 | B-ACC | 0.9644 | 0.9058 | -0.0586 | **0.9708** | +0.0064 | 0.8999 | -0.0645 |
| | ROC-AUC | **0.9943** | 0.9687 | -0.0256 | 0.9813 | -0.0130 | 0.9687 | -0.0256 |
| | PR-AUC | **0.9958** | 0.9773 | -0.0185 | 0.9865 | -0.0093 | 0.9733 | -0.0225 |
| PhysioNet | B-ACC | **0.6424** | 0.6151 | -0.0272 | 0.6346 | -0.0078 | 0.6263 | -0.0161 |
| | Kappa | **0.5230** | 0.4868 | -0.0362 | 0.5127 | -0.0104 | 0.5016 | -0.0214 |
| | F1 | **0.6413** | 0.6156 | -0.0258 | 0.6338 | -0.0075 | 0.6254 | -0.0159 |
| SEED | B-ACC | **0.7877** | 0.7395 | -0.0481 | 0.7111 | -0.0765 | 0.7444 | -0.0432 |
| | ROC-AUC | **0.8589** | 0.8390 | -0.0199 | 0.7931 | -0.0658 | 0.8334 | -0.0256 |
| | PR-AUC | **0.8647** | 0.8255 | -0.0392 | 0.7869 | -0.0778 | 0.8140 | -0.0507 |
| SEED-IV | B-ACC | **0.3989** | 0.3483 | -0.0505 | 0.3486 | -0.0503 | 0.3583 | -0.0405 |
| | Kappa | **0.1897** | 0.1222 | -0.0675 | 0.1201 | -0.0696 | 0.1347 | -0.0550 |
| | F1 | **0.3444** | 0.2655 | -0.0788 | 0.2629 | -0.0814 | 0.2841 | -0.0603 |
| SEED-V | B-ACC | 0.2734 | **0.2742** | +0.0007 | 0.2734 | 0.0000 | 0.2734 | 0.0000 |
| | Kappa | 0.0970 | **0.0977** | +0.0006 | 0.0970 | 0.0000 | 0.0970 | 0.0000 |
| | F1 | **0.2746** | 0.2720 | -0.0027 | **0.2746** | 0.0000 | **0.2746** | 0.0000 |
| ISRUC-S1 | B-ACC | **0.7925** | 0.7914 | -0.0011 | 0.7843 | -0.0082 | 0.7681 | -0.0244 |
| | Kappa | 0.7423 | **0.7603** | +0.0180 | 0.7398 | -0.0026 | 0.7192 | -0.0231 |
| | F1 | 0.8049 | **0.8135** | +0.0086 | 0.7991 | -0.0058 | 0.7787 | -0.0262 |

*Table 27 continued, Panel B: Gaussian disentangler and reconstruction ablations.*

| Dataset | Metric | Full | w/o Gaussian | Δ | w/o Rec. | Δ |
|---|---|---|---|---|---|---|
| BCIC-IV-2a | B-ACC | **0.5981** | 0.3967 | -0.2014 | 0.5122 | -0.0859 |
| | Kappa | **0.4641** | 0.1956 | -0.2685 | 0.3495 | -0.1146 |
| | F1 | **0.5954** | 0.3560 | -0.2394 | 0.5096 | -0.0858 |
| BCIC-Speech | B-ACC | **0.5116** | 0.4920 | -0.0196 | 0.4773 | -0.0342 |
| | Kappa | **0.2673** | 0.2380 | -0.0293 | 0.2160 | -0.0513 |
| | F1 | **0.5124** | 0.4917 | -0.0207 | 0.4757 | -0.0367 |
| FACED | B-ACC | **0.6126** | 0.5926 | -0.0200 | 0.6040 | -0.0086 |
| | Kappa | **0.5605** | 0.5384 | -0.0221 | 0.5518 | -0.0087 |
| | F1 | **0.6128** | 0.5985 | -0.0143 | 0.6107 | -0.0021 |
| Stieger2021-UD | B-ACC | **0.7730** | 0.7568 | -0.0162 | 0.7563 | -0.0167 |
| | ROC-AUC | **0.8575** | 0.8480 | -0.0095 | 0.8528 | -0.0047 |
| | PR-AUC | 0.8633 | 0.8524 | -0.0109 | 0.8598 | -0.0035 |
| MI-KoreaU | B-ACC | **0.8450** | 0.8207 | -0.0243 | 0.8311 | -0.0139 |
| | ROC-AUC | **0.9170** | 0.9013 | -0.0157 | 0.9138 | -0.0032 |
| | PR-AUC | **0.9221** | 0.9083 | -0.0137 | 0.9200 | -0.0020 |
| MentalArith. | B-ACC | **0.7260** | 0.5521 | -0.1740 | 0.6531 | -0.0729 |
| | ROC-AUC | **0.8376** | 0.7216 | -0.1160 | 0.7507 | -0.0869 |
| | PR-AUC | **0.6743** | 0.5250 | -0.1493 | 0.5037 | -0.1707 |
| Mumtaz2016 | B-ACC | 0.9644 | 0.9272 | -0.0372 | 0.8997 | -0.0647 |
| | ROC-AUC | **0.9943** | 0.9040 | -0.0903 | 0.9013 | -0.0930 |
| | PR-AUC | **0.9958** | 0.9459 | -0.0499 | 0.9463 | -0.0495 |
| PhysioNet | B-ACC | **0.6424** | 0.6152 | -0.0272 | 0.6335 | -0.0089 |
| | Kappa | **0.5230** | 0.4868 | -0.0362 | 0.5112 | -0.0118 |
| | F1 | **0.6413** | 0.6145 | -0.0268 | 0.6322 | -0.0092 |
| SEED | B-ACC | **0.7877** | 0.7481 | -0.0395 | 0.7395 | -0.0481 |
| | ROC-AUC | **0.8589** | 0.8407 | -0.0182 | 0.8101 | -0.0488 |
| | PR-AUC | **0.8647** | 0.8228 | -0.0419 | 0.7849 | -0.0798 |
| SEED-IV | B-ACC | **0.3989** | 0.3752 | -0.0237 | 0.3152 | -0.0836 |
| | Kappa | **0.1897** | 0.1564 | -0.0333 | 0.0774 | -0.1123 |
| | F1 | **0.3444** | 0.3057 | -0.0387 | 0.2161 | -0.1282 |
| SEED-V | B-ACC | 0.2734 | 0.2634 | -0.0100 | 0.2714 | -0.0020 |
| | Kappa | 0.0970 | 0.0831 | -0.0139 | 0.0937 | -0.0033 |
| | F1 | **0.2746** | 0.2659 | -0.0087 | 0.2700 | -0.0046 |
| ISRUC-S1 | B-ACC | **0.7925** | 0.7782 | -0.0143 | 0.7801 | -0.0124 |
| | Kappa | 0.7423 | 0.7395 | -0.0029 | 0.7279 | -0.0145 |
| | F1 | 0.8049 | 0.7974 | -0.0075 | 0.7889 | -0.0160 |

The seed-0 records provide metric-level detail, while the fixed-seed comparison in Table 2 supports the task-dependent component analysis in the main text.

## NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes].

   Justification: The abstract and introduction state the contribution as a post-backbone disentanglement adapter for EFM fine-tuning. The main quantitative scope is defined in Section 4.2, with limitations discussed in Appendix B.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes].

   Justification: Appendix B discusses the use of validation-selected loss weights, the temporal-variance prior, training-time subject identities, and the remaining direction of fully unsupervised disentanglement fine-tuning.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [N/A].

Justification: The paper does not present formal theoretical results. Method derivations used to specify the Gaussian update are provided in Appendix D.1.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes].

Justification: Sections 4.1–4.2 define the evaluation setup, datasets, backbones, metrics, and main comparison. Appendices C–D provide split statistics, preprocessing, hyperparameters, and implementation details, and Appendix B provides the code repository link.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in

some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes].

   Justification: The experiments use released EEG datasets, and code with configuration files is provided at the link in Appendix B.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes].

   Justification: Section 4.1 summarizes the datasets, backbones, update policy, hardware, and metrics. Appendix C reports split statistics, and Appendix D.3 reports optimizer and hyperparameter policies.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes].

   Justification: Table 1 reports mean and standard deviation over available seeds for the main comparison, and Section 4.1 states that the main table uses multi-seed mean and standard deviation.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes].

   Justification: Section 4.1 reports the GPU type, Appendix B reports GPU-hour accounting for reproducing the main evaluation and appendix diagnostics, and Appendix D.5 reports representative parameter, memory, and one-step FLOP overhead.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes].

   Justification: The work uses released datasets, subject-disjoint evaluation and training-time subject IDs only for regularization. Ethical and privacy considerations are discussed in Appendix B.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes].

Justification: Appendix B discusses the research value of more robust cross-subject EEG representation learning and the risk of inappropriate clinical interpretation or privacy-sensitive use of EEG/MDD-related data.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A].

Justification: The paper does not release a high-risk generative model, scraped dataset, or new sensitive dataset. It releases research code and uses already released EEG datasets.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes].

Justification: The paper cites the original datasets and EFM backbones used in the experiments, and Appendix B summarizes dataset sources, license names or usage terms, and cases where no standard license name could be identified. Raw datasets and pretrained checkpoints are not redistributed in the code release.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes].

    Justification: The new released asset is the implementation and configuration files, referenced in Appendix B. No new dataset is introduced.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A].

    Justification: This work does not conduct new crowdsourcing or collect new human-subject data; it evaluates on previously released EEG datasets.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A].

Justification: The paper performs secondary analysis of released EEG datasets and does not collect new data from human participants. The manuscript cites the original dataset sources and reports no new IRB approval for newly collected data.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A].

Justification: The proposed method and experiments do not use LLMs as an important, original, or non-standard component. Any writing or formatting assistance is outside the core methodology and does not require declaration under the checklist wording.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.