# NEUROEVOLVE: LEARNING WHAT EEG TO GENERATE FOR FOUNDATION MODEL ADAPTATION

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

EEG foundation models (FMs) have shown strong transferability across neural decoding tasks, yet their downstream adaptation remains constrained by limited task-specific data. Synthetic EEG generation offers a promising solution, but existing methods primarily optimize for fidelity to real EEG or generic data augmentation, overlooking a critical discrepancy: **high-fidelity EEG is not necessarily model-beneficial EEG**. This raises a fundamental question: *what EEG should be generated for effective foundation model adaptation?* We propose **NeuroEvolve**, a feedback-driven EEG generation paradigm that learns what EEG to generate based on model needs. NeuroEvolve combines **Closed-loop Strategy Discovery**, where planner and evaluator agents collaboratively refine generation strategies through foundation model feedback, with **Successful Experience Internalization**, which distills successful interactions into reusable knowledge for unseen tasks. Extensive experiments across diverse EEG tasks, datasets, and foundation models demonstrate consistent adaptation gains and validate the effectiveness of both mechanisms. NeuroEvolve shifts EEG generation from reproducing what real EEG looks like toward learning what EEG FMs actually need. Code is available at `https://anonymous.4open.science/r/NeuroEvolve-C1D0`.

## 1 INTRODUCTION

Large-scale pre-training has enabled EEG foundation models (FMs) to learn transferable representations from diverse neural datasets (Kuruppu et al., 2026; Lai et al., 2025; Wu et al., 2025). However, adapting these models to downstream tasks remains constrained by limited task-specific data (Roy et al., 2019). EEG acquisition and annotation are costly, while substantial inter-subject and inter-session variations make limited datasets insufficient to cover diverse task-relevant patterns (Wang et al., 2025b). Consequently, adapting EEG FMs solely with available downstream data may be insufficient, motivating the need to expand task-relevant data beyond existing observations.

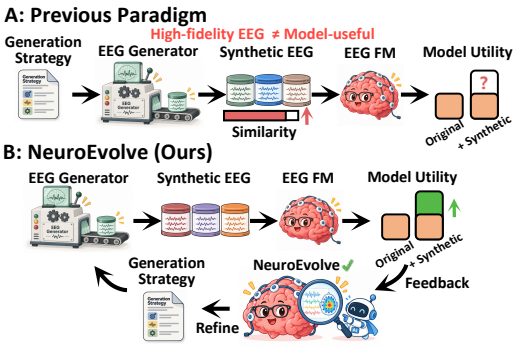


Figure 1: **Motivation of NeuroEvolve.** (A) Existing EEG generation primarily optimizes for fidelity to real EEG, but high-fidelity EEG is not necessarily model-beneficial. (B) NeuroEvolve leverages foundation model feedback to iteratively refine generation strategies, thereby learning what EEG to generate based on model needs.

Synthetic EEG generation provides a natural way to expand limited observations (Wang et al., 2026; Song et al., 2021). Existing methods mainly optimize for fidelity to real EEG or rely on generic data augmentation with predefined generation strategies. **Yet high-fidelity EEG is not necessarily model-beneficial EEG**: synthetic samples that closely resemble real EEG may not improve model adaptation, whereas those targeting underrepresented or task-relevant patterns may provide greater utility. This discrepancy between *fidelity to real EEG* and



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

*utility to the model* raises an important yet underexplored question: **what EEG does a foundation model actually need for effective adaptation?**

To answer this question, we propose **NeuroEvolve** (Figure.1), a **feedback-driven EEG generation paradigm** that learns what EEG to generate based on model needs. NeuroEvolve comprises two complementary mechanisms. **(i) Closed-loop Strategy Discovery** couples a *planner agent* that proposes what EEG to generate with an *evaluator agent* that assesses its model utility, progressively refining generation strategies through foundation model feedback. **(ii) Successful Experience Internalization** distills successful interactions into the planner, turning costly task-specific exploration into reusable generation knowledge for unseen tasks. Together, NeuroEvolve learns *what EEG to generate* from foundation model feedback, shifting EEG generation from pursuing fidelity to real EEG toward serving model needs. The main contributions of this work are summarized as follows:

- We identify a critical gap between **EEG fidelity and model utility**, and propose **NeuroEvolve**, shifting EEG generation from reproducing what real EEG looks like toward learning what EEG FMs actually need.
- We develop **Closed-loop Strategy Discovery** and **Successful Experience Internalization**. A planner–evaluator agent loop discovers model-beneficial generation strategies through foundation model feedback, while experience internalization turns costly task-specific exploration into reusable knowledge for efficient adaptation to unseen tasks.
- Extensive experiments across diverse EEG tasks, datasets, and foundation models demonstrate consistent improvements in downstream adaptation, with comprehensive analyses validating the effectiveness of feedback-driven generation and experience internalization.

## 2 Related Work

**EEG Generation and Data Augmentation.** EEG data augmentation has been widely studied to mitigate limited supervision and improve generalization in neural decoding. Conventional approaches rely on predefined signal transformations, such as temporal cropping, noise perturbation, frequency manipulation, and sample interpolation (Lashgari et al., 2020; Wang et al., 2024b). Recent generative approaches leverage diffusion and flow-based models to learn complex EEG distributions and synthesize high-fidelity samples (Song et al., 2021; Wang et al., 2026). Existing approaches, however, largely emphasize fidelity to real EEG or employ predefined augmentation strategies, without explicitly optimizing what to generate for downstream model adaptation. NeuroEvolve instead shifts the focus from reproducing what real EEG looks like toward learning what EEG to generate for effective foundation model adaptation.

**EEG Foundation Model Adaptation.** Recent EEG foundation models learn transferable neural representations from large-scale EEG corpora and have demonstrated strong generalization across diverse downstream tasks (Abiri et al., 2019; Aggarwal & Chugh, 2022; Lotte et al., 2018; Värbu et al., 2022). Existing studies adapt these models through full fine-tuning, parameter-efficient tuning, and prompting-based approaches (Wang et al., 2025a; Jiang et al., 2024; Wang et al., 2024a; Xiao et al., 2025). Despite their different mechanisms, these methods primarily focus on how to adapt foundation models given the available downstream data. NeuroEvolve takes a complementary data-centric perspective. Rather than changing how the model is adapted, it focuses on what additional EEG should be generated to better support adaptation. This connects data generation with foundation model adaptation, allowing generation decisions to be guided by model needs.

## 3 Method

### 3.1 Problem Formulation and Overview

Given a pre-trained EEG foundation model $F_\theta$ and a downstream dataset $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$, we consider synthetic EEG generation for downstream model adaptation. Existing generation methods primarily emphasize fidelity to real EEG, yet high-fidelity synthetic EEG does not necessarily benefit foundation model adaptation. This motivates a fundamental question: *what EEG should be generated for effective foundation model adaptation?* To address this question, we propose **NeuroEvolve**, a feedback-driven EEG generation paradigm that learns what EEG to generate based

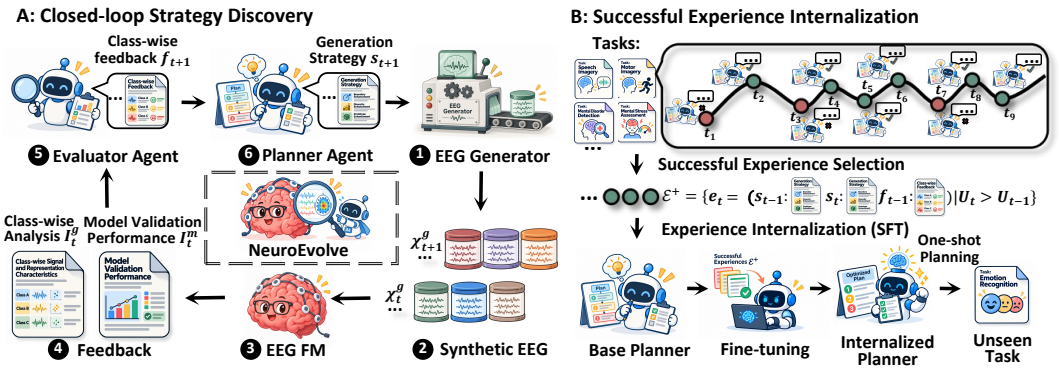


Figure 2: **Overview of NeuroEvolve.**(A) **Closed-loop Strategy Discovery** couples a planner agent with an evaluator agent to progressively refine EEG generation strategies through foundation model feedback. (B) **Successful Experience Internalization** distills successful strategy refinements into the planner, turning costly task-specific exploration into reusable generation knowledge and enabling one-shot planning on unseen tasks.

on model needs. As illustrated in Figure. 2, NeuroEvolve comprises two complementary stages. **Closed-loop Strategy Discovery (CSD)** couples a *planner agent* with an *evaluator agent* to discover model-beneficial generation strategies through foundation model feedback. **Successful Experience Internalization (SEI)** distills successful strategy refinements into the planner, reducing costly task-specific exploration for unseen tasks.

### 3.2 Closed-loop Strategy Discovery

The utility of synthetic EEG depends on how it affects downstream model adaptation and cannot be determined by fidelity alone. We therefore formulate EEG generation as an iterative strategy discovery process, where generation decisions are iteratively refined based on generated-data characteristics and foundation model responses.

**Class-wise Generation Planning.** Given dataset statistics $I_{\mathcal{D}}$, a planner agent $P_\phi$ constructs a class-specific generation strategy: $s_t = \{(a_t^c, \mathcal{R}_t^c, \sigma_t^c)\}_{c=1}^C$, where $a_t^c$ specifies the generation objective, $\mathcal{R}_t^c$ contains the reference EEG that anchors generation, and $\sigma_t^c$ controls the generation variation for class $c$. This formulation defines a structured yet expressive decision space, where the planner is constrained to select meaningful generation behaviors while retaining sufficient flexibility to adapt to diverse downstream tasks. Specifically, we consider three generation objectives: *boundary*, *diversity*, and *prototype enhancement*, targeting ambiguous, underrepresented, and representative patterns, respectively. The planner is implemented using DeepSeek-R1-Distill-Qwen-7B (Guo et al., 2025), which performs reasoning over dataset statistics (Appendix A.5) and produces the initial generation strategy:

$$s_1 = P_\phi(I_{\mathcal{D}}). \tag{1}$$

The resulting strategy conditions a pre-trained EEG generator through the selected references and variation strengths, producing class-specific synthetic EEG $\mathcal{X}_t^g = G(s_t)$ (Detailed planner prompts, strategy outputs, and generator details are provided in Appendix A.6, Appendix A.7, and Appendix A.10).

**Model-grounded Generation Assessment.** We assess each generation round from two complementary views. First, a generated-data analyzer (Appendix A.5) characterizes the synthetic EEG in the foundation model representation space:

$$I_t^g = A(F_\theta, \mathcal{X}_t^g), \tag{2}$$

summarizing class-wise signal and representation characteristics. Second, we evaluate its effect on downstream adaptation by independently adapting the pre-trained foundation model $F_\theta$ at each round:

$$F_t = \mathrm{Adapt}(F_\theta, \mathcal{D}_{\mathrm{train}} \cup \mathcal{X}_t^g), \tag{3}$$

with its validation response summarized as $I_t^m$. Together, $I_t^g$ and $I_t^m$ reveal *what was generated* and *how the model responded*, providing complementary evidence for strategy refinement.

**Feedback-guided Strategy Refinement.** An evaluator agent $E_\psi$ integrates the current strategy with generation and model evidence to produce class-wise feedback:

$$f_t = E_\psi(s_t, I_t^g, I_t^m). \tag{4}$$

For each class, the evaluator diagnoses the generation outcome and recommends whether to *retain*, *adjust*, or *switch the current strategy*. The planner then translates this feedback into the next generation strategy:

$$s_{t+1} = P_\phi(I_\mathcal{D}, s_t, f_t). \tag{5}$$

Through this planner–evaluator loop, NeuroEvolve progressively learns *what EEG to generate* for effective foundation model adaptation (Detailed evaluator prompts and feedback outputs are provided in Appendix A.8 and Appendix A.9).

### 3.3 Successful Experience Internalization

Closed-loop discovery progressively identifies effective generation strategies, but repeating multi-round exploration for every new task is costly. We therefore internalize successful experiences into the planner, enabling discovered generation knowledge to be efficiently reused on unseen tasks.

**Successful Experience Selection.** From each discovery trajectory, we construct an experience: $e_t = (I_\mathcal{D}, f_{t-1}, s_{t-1}, s_t)$, capturing how evaluator feedback $f_{t-1}$ guides the refinement from $s_{t-1}$ to $s_t$. We retain only experiences whose revised strategies improve downstream adaptation. Let $U_t = U(F_t)$ denote the validation performance induced by strategy $s_t$; the successful experience set is

$$\mathcal{E}^+ = \{e_t \mid U_t > U_{t-1}\}. \tag{6}$$

**Experience Internalization.** We internalize successful experiences via supervised fine-tuning (SFT), optimizing the planner to predict the successful next strategy conditioned on the dataset statistics, current strategy, and evaluator feedback:

$$\phi^* = \arg\min_\phi \sum_{e_t \in \mathcal{E}^+} -\log P_\phi\left(s_t \mid I_\mathcal{D}, s_{t-1}, f_{t-1}\right). \tag{7}$$

This distills successful feedback-guided refinements into the planner, allowing discovered generation knowledge to guide future planning.

**One-shot Planning for Unseen Tasks.** For an unseen target dataset $\mathcal{D}_{\text{tar}}$, we obtain $P_{\phi^*}$ by internalizing successful experiences from the remaining datasets. The resulting planner directly produces a generation strategy:

$$s^* = P_{\phi^*}(I_{\mathcal{D}_{\text{tar}}}). \tag{8}$$

This avoids iterative evaluator interaction on the target task, enabling efficient one-shot planning for unseen tasks.

## 4 Experiments

### 4.1 Experimental Setup

**Datasets.** We evaluate NeuroEvolve on nine widely used EEG benchmark datasets spanning six representative neural decoding tasks (Table 1): motor imagery (BCIC-2A (Brunner et al., 2008), PhysioNet-MI (Schalk et al., 2004), and SHU-MI (Ma et al., 2022)), emotion recognition (FACED (Chen et al., 2023) and SEED-V (Liu et al., 2021)), imagined speech (BCIC2020-3 (Jeong et al., 2022)), mental disorder diagnosis (Mumtaz2016 (Wajid, 2016)), stress detection (MentalArithmetic (Zyma et al., 2019)), and event detection (TUEV (Obeid & Picone, 2016)). These datasets exhibit substantial heterogeneity in acquisition protocols, subject populations, and task semantics, providing a diverse and challenging benchmark for evaluating EEG foundation model adaptation. We follow the standard preprocessing protocol adopted by CBraMod and apply a consistent preprocessing pipeline across all datasets. Dataset preprocessing details are provided in Appendix A.1.

Table 1: Overview of downstream BCI tasks and datasets.

| BCI Tasks | Datasets | Rate | # Channels | Duration | # Samples | Label |
|---|---|---|---|---|---|---|
| I. Motor Imagery Classification | BCIC-2A | 250Hz | 22 | 4s | 5,088 | 4-class |
| | PhysioNet-MI | 160Hz | 64 | 4s | 9,837 | 4-class |
| | SHU-MI | 250Hz | 32 | 4s | 11,988 | 2-class |
| II. Emotion Recognition | FACED | 250Hz | 32 | 10s | 10,332 | 9-class |
| | SEED-V | 1000Hz | 62 | 1s | 115,001 | 5-class |
| III. Imagined Speech Classification | BCIC2020-3 | 256Hz | 64 | 3s | 6,000 | 5-class |
| IV. Mental Disorder Diagnosis | Mumtaz2016 | 256Hz | 19 | 5s | 7,143 | 2-class |
| V. Mental Stress Detection | MentalArithmetic | 500Hz | 20 | 5s | 1,707 | 2-class |
| VI. Event Type Classification | TUEV | 250Hz | 16 | 5s | 113,353 | 6-class |

**Baselines.** As NeuroEvolve introduces a feedback-driven EEG generation paradigm for foundation model adaptation, we compare it against two groups of baselines. (i) Pretrained EEG foundation models: we consider representative EEG foundation models, including CBraMod (Wang et al., 2025a), LaBraM (Jiang et al., 2024), EEGPT (Wang et al., 2024a), and BrainOmni (Xiao et al., 2025). All models are adapted to downstream tasks using full fine-tuning under a consistent evaluation protocol, allowing us to assess whether feedback-driven EEG generation provides additional gains beyond strong pretrained representations. (ii) EEG generation methods: we further compare NeuroEvolve with representative generative augmentation approaches, including Vanilla Diffusion (Song et al., 2021), a diffusion-based EEG generator, and JET (Wang et al., 2026), a flow-matching-based EEG generation method. For a controlled comparison, all generation-based methods use the same downstream foundation model and adaptation protocol, such that performance differences primarily reflect the performance of the generated EEG rather than differences in downstream optimization. Additional details on baselines, FM training, and implementation are provided in Appendix A.2, Appendix A.11, and Appendix A.12.

**Backbone Models.** NeuroEvolve is model-agnostic and does not require architectural modifications to the underlying EEG foundation model. To evaluate its generalizability, we instantiate NeuroEvolve with three representative EEG foundation models, including CBraMod (Wang et al., 2025a), LaBraM (Jiang et al., 2024), and EEGPT (Wang et al., 2024a), which differ in model architecture and pretraining strategy. In each case, NeuroEvolve treats the foundation model as the target model and uses its downstream feedback to iteratively guide EEG generation during adaptation. This evaluation examines whether the benefits of NeuroEvolve extend beyond a particular foundation model. Additional backbone details are provided in Appendix A.3.

**Evaluation Metrics.** For binary classification tasks, we report Balanced Accuracy, AUC-PR, and AUROC, with AUROC used as the primary metric for model selection. For multi-class classification tasks, we report Balanced Accuracy, Cohen's Kappa, and weighted F1-score, with Cohen's Kappa used as the primary metric for model selection. Unless otherwise specified, all results are averaged over five random seeds and reported as mean $\pm$ standard deviation (Table 2). Statistical significance is assessed using a two-sided Wilcoxon signed-rank test on paired results across benchmark datasets and evaluation metrics, evaluating whether the improvements of NeuroEvolve over competing methods are statistically significant (Pratt, 1959; Liu et al., 2022; Li et al., 2026). We additionally report $\Delta\mathrm{IMP}$, defined as the performance improvement of NeuroEvolve over the strongest baseline for each dataset and evaluation metric. Details of the evaluation metrics are provided in Appendix A.4.

### 4.2 Effectiveness of Closed-loop Experience Discovery

**Setup.** Unless otherwise specified, all EEG generation baselines (i.e., Vanilla Diffusion and JET) and NeuroEvolve use CBraMod as the backbone foundation model and are evaluated under the same full fine-tuning protocol to ensure a fair and consistent comparison. **Results.** Table 2 reports the performance under the full fine-tuning setting. Across all downstream tasks, NeuroEvolve achieves the best overall performance, consistently outperforming both pretrained EEG foundation models and EEG generation baselines. The improvements are statistically significant under the Wilcoxon signed-rank test ($p < 0.001$), demonstrating that the gains are consistent across diverse downstream tasks. **Analysis.** The results reveal two key observations. (i) Compared with pretrained EEG foundation models, NeuroEvolve consistently improves downstream performance across diverse tasks. This demonstrates that feedback-driven EEG generation can effectively complement limited task-

Table 2: Performance comparison across multiple downstream EEG datasets.

| Methods | BCIC-2A, 4-class | | | PhysioNet-MI, 4-class | | | SHU-MI, 2-class | | |
|---|---|---|---|---|---|---|---|---|---|
| | **Bal.Acc.** | **Kappa** | **F1-w** | **Bal.Acc.** | **Kappa** | **F1-w** | **Bal.Acc.** | **AUC-PR** | **AUROC** |
| EEGPT | 36.11 ± 1.24 | 14.81 ± 1.65 | 31.17 ± 2.70 | 47.35 ± 1.47 | 29.79 ± 1.96 | 47.59 ± 1.54 | 52.65 ± 1.42 | 54.58 ± 0.97 | 54.51 ± 1.38 |
| BrainOmni | 38.84 ± 1.45 | 18.45 ± 1.93 | 37.12 ± 1.75 | 53.29 ± 1.31 | 37.71 ± 1.74 | 53.20 ± 1.16 | 60.43 ± 2.32 | 66.71 ± 1.14 | 67.27 ± 0.72 |
| LaBraM | 44.88 ± 4.29 | 26.50 ± 5.72 | 43.76 ± 4.61 | 60.40 ± 1.21 | 47.20 ± 1.61 | 60.57 ± 1.33 | 60.80 ± 1.61 | 67.36 ± 2.16 | 65.88 ± 2.80 |
| CBraMod | 46.77 ± 1.35 | 29.03 ± 1.80 | 44.83 ± 1.95 | 62.71 ± 0.15 | 50.28 ± 0.21 | 62.76 ± 0.13 | 60.54 ± 1.58 | 66.83 ± 1.81 | 66.53 ± 2.63 |
| Vanilla Diff. | 46.68 ± 1.06 | 28.91 ± 1.41 | 45.04 ± 1.53 | 61.97 ± 0.84 | 49.29 ± 1.12 | 62.00 ± 0.87 | 61.13 ± 1.80 | 67.37 ± 1.76 | 66.63 ± 2.83 |
| JET | 46.65 ± 1.97 | 28.87 ± 2.62 | 43.87 ± 2.11 | 61.92 ± 0.38 | 49.21 ± 0.50 | 61.98 ± 0.38 | 61.10 ± 2.44 | 67.25 ± 2.92 | 66.54 ± 3.94 |
| **NeuroEvolve** | **48.09** ± 1.33 | **30.79** ± 1.77 | **46.44** ± 1.42 | **63.52** ± 0.44 | **51.01** ± 0.59 | **63.55** ± 0.44 | **62.42** ± 0.91 | **70.09** ± 1.36 | **69.58** ± 1.00 |
| ΔIMP | **+2.82%** | **+6.06%** | **+3.11%** | **+1.29%** | **+1.45%** | **+1.26%** | **+2.11%** | **+4.04%** | **+3.43%** |
| **Methods** | **FACED, 9-class** | | | **SEED-V, 5-class** | | | **BCI2020-3, 5-class** | | |
| | **Bal.Acc.** | **Kappa** | **F1-w** | **Bal.Acc.** | **Kappa** | **F1-w** | **Bal.Acc.** | **Kappa** | **F1-w** |
| EEGPT | 34.48 ± 0.65 | 26.04 ± 0.76 | 34.10 ± 0.88 | 23.86 ± 0.52 | 4.91 ± 0.62 | 23.57 ± 0.76 | 22.13 ± 1.33 | 2.67 ± 1.66 | 21.42 ± 1.43 |
| BrainOmni | 22.94 ± 0.82 | 13.24 ± 0.91 | 22.50 ± 0.77 | 24.06 ± 0.51 | 5.32 ± 0.57 | 24.08 ± 0.69 | 27.60 ± 1.51 | 9.50 ± 1.89 | 27.36 ± 1.56 |
| LaBraM | 55.75 ± 1.06 | 50.10 ± 1.21 | 56.53 ± 0.97 | 38.99 ± 0.43 | 23.86 ± 0.62 | 39.42 ± 0.66 | 26.16 ± 1.04 | 7.70 ± 1.30 | 25.66 ± 1.38 |
| CBraMod | 57.09 ± 0.68 | 51.48 ± 0.75 | 57.47 ± 0.67 | 38.68 ± 0.97 | 23.85 ± 1.26 | 39.48 ± 0.99 | 32.51 ± 1.80 | 15.63 ± 2.25 | 32.48 ± 1.84 |
| Vanilla Diff. | 55.98 ± 0.71 | 50.27 ± 0.85 | 56.35 ± 0.72 | 39.02 ± 0.66 | 24.25 ± 0.91 | 39.78 ± 0.73 | 32.75 ± 3.20 | 15.93 ± 4.00 | 32.73 ± 3.23 |
| JET | 57.51 ± 0.77 | 52.03 ± 0.86 | 58.10 ± 0.82 | 39.95 ± 0.95 | 25.41 ± 1.38 | 40.76 ± 1.11 | 33.41 ± 1.95 | 16.77 ± 2.44 | 33.42 ± 1.93 |
| **NeuroEvolve** | **58.71** ± 0.79 | **53.26** ± 0.87 | **59.06** ± 0.75 | **40.73** ± 0.43 | **26.55** ± 0.54 | **41.57** ± 0.48 | **36.64** ± 2.29 | **20.80** ± 2.86 | **36.58** ± 2.33 |
| ΔIMP | **+2.09%** | **+2.36%** | **+1.65%** | **+1.95%** | **+4.49%** | **+1.99%** | **+9.67%** | **+24.03%** | **+9.46%** |
| **Methods** | **Mumtaz2016, 2-class** | | | **MentalArithmetic, 2-class** | | | **TUEV, 6-class** | | |
| | **Bal.Acc.** | **AUC-PR** | **AUROC** | **Bal.Acc.** | **AUC-PR** | **AUROC** | **Bal.Acc.** | **Kappa** | **F1-w** |
| EEGPT | 89.81 ± 1.64 | 97.86 ± 0.71 | 97.84 ± 0.73 | 53.33 ± 2.03 | 37.97 ± 2.82 | 63.86 ± 1.68 | 62.40 ± 2.88 | 63.47 ± 4.43 | 80.67 ± 2.22 |
| BrainOmni | 80.95 ± 0.69 | 88.00 ± 1.58 | 89.58 ± 1.04 | 61.67 ± 4.45 | 50.61 ± 7.48 | 70.76 ± 5.78 | 30.24 ± 2.28 | 23.22 ± 4.52 | 57.59 ± 4.19 |
| LaBraM | 89.53 ± 0.62 | 97.16 ± 0.94 | 96.78 ± 1.31 | 63.06 ± 2.87 | 51.12 ± 4.70 | 73.84 ± 4.40 | 59.66 ± 2.75 | 58.65 ± 4.29 | 78.20 ± 2.17 |
| CBraMod | 88.47 ± 0.89 | 97.35 ± 0.65 | 97.16 ± 0.78 | 59.79 ± 6.01 | 43.70 ± 4.03 | 70.72 ± 2.91 | 61.11 ± 3.14 | 59.50 ± 4.43 | 78.69 ± 2.25 |
| Vanilla Diff. | 89.21 ± 0.71 | 97.43 ± 0.65 | 97.03 ± 0.84 | 57.29 ± 1.79 | 43.50 ± 8.06 | 68.43 ± 4.45 | 59.49 ± 3.26 | 62.15 ± 4.99 | 80.28 ± 2.50 |
| JET | 89.63 ± 1.43 | 97.89 ± 0.94 | 97.68 ± 1.07 | 60.42 ± 4.32 | **54.15** ± 6.18 | 76.07 ± 3.53 | 59.64 ± 1.88 | 59.87 ± 1.76 | 79.20 ± 0.89 |
| **NeuroEvolve** | **90.85** ± 1.23 | **98.56** ± 0.41 | **98.55** ± 0.43 | **63.35** ± 3.23 | 54.02 ± 5.88 | **76.63** ± 2.32 | **62.90** ± 1.47 | **63.68** ± 3.21 | **81.49** ± 1.33 |
| ΔIMP | **+1.16%** | **+0.68%** | **+0.73%** | **+0.45%** | **-0.24%** | **+0.74%** | **+0.80%** | **+0.33%** | **+1.02%** |

Table 3: Ablation study of generation strategy planning in NeuroEvolve.

| Methods | Capabilities | | | SHU-MI, 2-class | | | FACED, 9-class | | |
|---|---|---|---|---|---|---|---|---|---|
| | **Generation** | **Feedback** | **Reasoning** | **Bal.Acc.** | **AUC-PR** | **AUROC** | **Bal.Acc.** | **Kappa** | **F1-w** |
| Random Strategy | ✓ | × | × | 60.71 ↓ 1.71 | 67.48 ↓ 2.61 | 66.40 ↓ 3.18 | 57.92 ↓ 0.79 | 52.38 ↓ 0.88 | 58.23 ↓ 0.83 |
| Static Strategy | ✓ | × | ✓ | 61.84 ↓ 0.58 | 68.12 ↓ 1.97 | 68.66 ↓ 0.92 | 58.52 ↓ 0.19 | 53.01 ↓ 0.25 | 58.68 ↓ 0.38 |
| Rule-based Strategy | ✓ | ✓ | × | 62.33 ↓ 0.09 | 69.46 ↓ 0.63 | 68.90 ↓ 0.68 | 58.52 ↓ 0.19 | 53.03 ↓ 0.23 | 58.72 ↓ 0.34 |
| **NeuroEvolve** | ✓ | ✓ | ✓ | **62.42** | **70.09** | **69.58** | **58.71** | **53.26** | **59.06** |

specific data by generating samples that better support foundation model adaptation. (ii) Compared with EEG generation methods such as Vanilla Diffusion and JET, NeuroEvolve yields further improvements despite these methods being designed to generate realistic EEG. This result supports that high-fidelity EEG is not necessarily model-beneficial EEG.

### 4.3 Ablation Study of Generation Strategy

**Setup.** To evaluate the contribution of different generation strategies, we construct three variants of NeuroEvolve. (i) **Random Strategy** randomly selects reference samples and noise levels for each class without model feedback or reasoning. (ii) **Static Strategy** derives generation strategies solely from dataset statistics without downstream adaptation feedback. (iii) **Rule-based Strategy** incorporates model feedback but updates strategies using predefined rules instead of adaptive reasoning. **Results.** Table 3 shows that NeuroEvolve achieves the best performance. Removing feedback and reasoning causes progressive degradation, with Random Strategy performing worst, followed by Static and Rule-based Strategies. **Analysis.** These results reveal two key insights. (i) Model feedback is critical for effective EEG generation. Compared with Random and Static Strategies, Rule-based Strategy benefits from feedback-driven updates, indicating that generation should be optimized for foundation model adaptation utility rather than data characteristics alone. (ii) Reasoning enables adaptive strategy discovery beyond fixed rules. While Rule-based Strategy exploits feedback, its manually designed updates cannot capture complex relationships between generated EEG and model utility. NeuroEvolve instead uses reasoning to iteratively refine strategies and discover EEG generation policies better aligned with adaptation objectives.

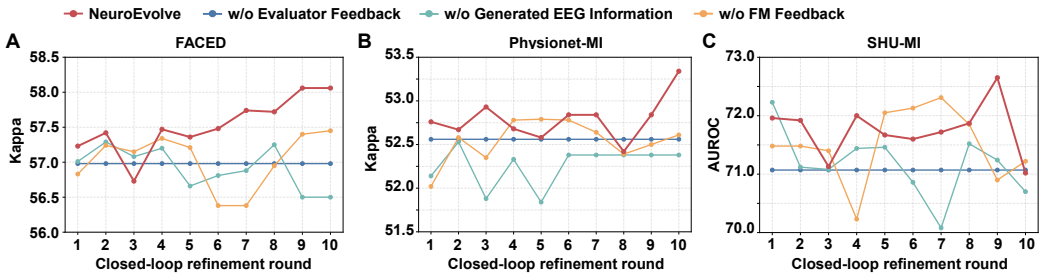


Figure 3: Ablation study of feedback signals in the closed-loop refinement process of NeuroEvolve.

Table 4: Comparison of different data augmentation strategies for EEG foundation model adaptation.

| Methods | Capabilities | | | MentalArithmetic, 2-class | | | BCI2020-3, 5-class | | |
|---|---|---|---|---|---|---|---|---|---|
| | Generation | Feedback | Reasoning | Bal.Acc. | AUC-PR | AUROC | Bal.Acc. | Kappa | F1-w |
| Original | × | × | × | 59.79 | 43.70 | 70.72 | 32.51 | 15.63 | 32.48 |
| Noise Aug. | × | × | × | 61.04 ↑ 1.25 | 53.31 ↑ 9.61 | 75.22 ↑ 4.50 | 30.93 ↓ 1.58 | 13.67 ↓ 1.96 | 30.90 ↓ 1.58 |
| Random mask | × | × | × | 60.21 ↑ 0.42 | 50.16 ↑ 6.46 | 73.02 ↑ 2.30 | 32.72 ↑ 0.21 | 15.90 ↑ 0.27 | 32.59 ↑ 0.11 |
| Vanilla Diff. | ✓ | × | × | 57.29 ↓ 2.50 | 43.50 ↓ 0.20 | 68.43 ↓ 2.29 | 32.75 ↑ 0.24 | 15.93 ↑ 0.30 | 32.73 ↑ 0.25 |
| JET | ✓ | × | × | 60.42 ↑ 0.63 | **54.15** ↑ 10.45 | 76.09 ↑ 5.37 | 33.41 ↑ 0.90 | 16.77 ↑ 1.14 | 33.42 ↑ 0.94 |
| **NeuroEvolve** | ✓ | ✓ | ✓ | **63.35** ↑ 3.56 | 54.02 ↑ 10.32 | **76.63** ↑ 5.91 | **36.64** ↑ 4.13 | **20.80** ↑ 5.17 | **36.58** ↑ 4.10 |

### 4.4 Ablation Study of Feedback Signal

**Setup.** To evaluate the contribution of different feedback signals, we consider three variants: (i) **w/o Evaluator Feedback**, removing evaluator-generated feedback; (ii) **w/o Generated EEG Information**, removing generated EEG descriptions from the evaluator input; and (iii) **w/o FM Feedback**, removing foundation model adaptation feedback. **Results.** Figure. 3 shows that removing any feedback signal degrades the refinement trajectory, while the full feedback loop achieves the best final performance across datasets. **Analysis.** These results demonstrate two key insights. (i) The complete feedback loop enables progressive strategy refinement across iterations rather than isolated improvements. (ii) Generated EEG information and FM feedback provide complementary signals: the former describes the generated samples, while the latter measures their adaptation utility. Together, they allow the planner to discover generation strategies better aligned with foundation model needs.

### 4.5 Ablation Study of Data Augmentation

**Setup.** To evaluate different augmentation strategies for EEG foundation model adaptation, we compare NeuroEvolve with five variants: (i) **Original** without augmentation, (ii) **Noise Aug.** with noise perturbation, (iii) **Random Mask** with random signal masking, (iv) **Vanilla Diff.** using diffusion-based generation without feedback, and (v) **JET** using a representative EEG generator. All augmented samples are combined with the original training data for adaptation. **Results.** Table 4 shows that NeuroEvolve consistently achieves the best performance, outperforming both conventional augmentation methods and existing EEG generation approaches. Generation-based methods generally outperform simple signal perturbations, demonstrating the benefit of synthetic data. **Analysis.** These results reveal two insights. (i) Conventional augmentation relies on predefined transformations and lacks awareness of downstream adaptation requirements, limiting its utility for EEG foundation models. (ii) Although generative augmentation improves diversity, generated samples may not benefit adaptation without proper guidance. NeuroEvolve addresses this issue by learning what data the model needs rather than optimizing generation fidelity alone.

### 4.6 Generalization Across EEG Foundation Models

**Setup.** To evaluate the generalizability of NeuroEvolve across EEG foundation models, we conduct experiments with three representative pretrained backbones: CBraMod, LaBraM, and EEGPT. These models exhibit substantial differences in their architectures and pretraining paradigms. **Results.** Table 5 reports the performance across different backbone models. NeuroEvolve consistently

Table 5: Performance comparison across different EEG foundation model backbones.

| Backbones | PhysioNet-MI, 4-class | | | FACED, 9-class | | | BCIC-2A, 4-class | | |
|---|---|---|---|---|---|---|---|---|---|
| | Bal.Acc. | Kappa | F1-w | Bal.Acc. | Kappa | F1-w | Bal.Acc. | AUC-PR | AUROC |
| EEGPT | 47.35 | 29.79 | 47.59 | 34.48 | 26.04 | 34.10 | 36.11 | 14.81 | 31.17 |
| +Vanilla Diff. | 47.28 ↓0.07 | 29.70 ↓0.09 | 47.53 ↓0.06 | 37.13 ↑2.65 | 29.34 ↑3.30 | 37.06 ↑2.96 | 36.25 ↑0.14 | 15.00 ↑0.19 | 31.82 ↑0.65 |
| +JET | 47.81 ↑0.46 | 30.41 ↑0.62 | 48.18 ↑0.59 | 37.77 ↑3.29 | 29.93 ↑3.89 | 37.63 ↑3.53 | 37.01 ↑0.90 | 16.02 ↑1.21 | 32.03 ↑0.86 |
| **+NeuroEvolve** | **48.65** ↑1.30 | **31.52** ↑1.73 | **48.88** ↑1.29 | **37.95** ↑3.47 | **30.11** ↑4.07 | **37.77** ↑3.67 | **37.19** ↑1.08 | **16.25** ↑1.44 | **33.56** ↑2.39 |
| **Methods** | **PhysioNet-MI, 4-class** | | | **SEED-V, 5-class** | | | **SHU-MI, 2-class** | | |
| | Bal.Acc. | Kappa | F1-w | Bal.Acc. | Kappa | F1-w | Bal.Acc. | AUC-PR | AUROC |
| LaBraM | 60.40 | 47.20 | 60.57 | 38.99 | 23.86 | 39.42 | 60.80 | 67.36 | 65.88 |
| +Vanilla Diff. | 61.76 ↑1.36 | 49.02 ↑1.82 | 61.98 ↑1.41 | 42.11 ↑3.12 | 27.84 ↑3.98 | 42.73 ↑3.31 | 61.11 ↑0.31 | 67.81 ↑0.45 | 65.94 ↑0.06 |
| +JET | 61.06 ↑0.66 | 48.07 ↑0.87 | 61.39 ↑0.82 | 40.96 ↑1.97 | 26.41 ↑2.55 | 41.59 ↑2.17 | 61.30 ↑0.50 | 67.13 ↓0.23 | 65.85 ↓0.03 |
| **+NeuroEvolve** | **62.06** ↑1.66 | **49.40** ↑2.20 | **62.32** ↑1.75 | **42.96** ↑3.97 | **28.99** ↑5.13 | **43.60** ↑4.18 | **62.67** ↑1.87 | **70.16** ↑2.80 | **69.15** ↑3.27 |

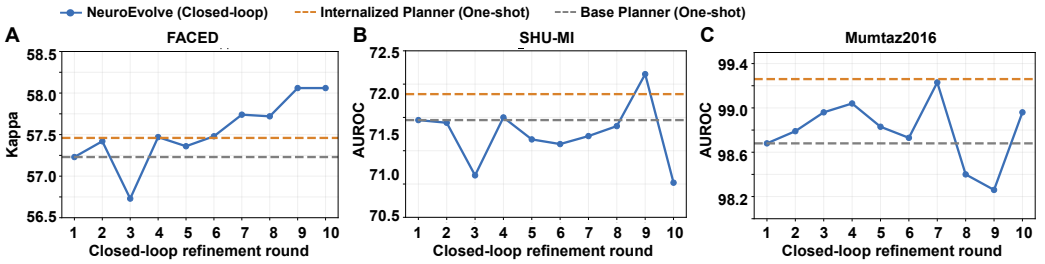


Figure 4: Cross-dataset transfer of internalized experience.

improves downstream performance across all evaluated EEG foundation models and achieves the best performance in most settings. **Analysis.** Despite substantial differences in architecture and pre-training paradigms, NeuroEvolve consistently improves the performance of all evaluated foundation models. This suggests that its effectiveness is not tied to a specific backbone, but rather arises from its feedback-driven generation paradigm, which adapts EEG generation according to the requirements of the underlying model. These results demonstrate the model-agnostic and plug-and-play capability of NeuroEvolve.

### 4.7 Transferability and Efficiency of Experience Internalization

**Setup.** To evaluate whether internalized experience can transfer across datasets, we adopt a cross-dataset transfer setting. For each target dataset, the planner is fine-tuned via supervised fine-tuning (SFT) using only successful experiences collected from the remaining datasets, and then performs one-shot generation planning on the unseen target dataset. We compare the Internalized Planner with the Base Planner under the same one-shot setting, as well as the original closed-loop refinement process. **Results.** As shown in Figure. 4, the Internalized Planner consistently outperforms the Base Planner across unseen datasets and achieves performance close to multi-round closed-loop refinement with only one planning step. This demonstrates that successful experiences can effectively improve one-shot generation planning without requiring iterative exploration. **Analysis.** These results reveal two key advantages of experience internalization. (i) The consistent improvement over the Base Planner indicates that the internalized knowledge is not merely derived from the planner's inherent reasoning capability, but from reusable generation strategies discovered through successful interactions. (ii) By transferring such strategies across datasets, NeuroEvolve converts costly trial-and-error refinement into efficient one-shot adaptation, substantially reducing the number of generation–evaluation cycles required for foundation model adaptation.

### 4.8 Analysis of Generation Fidelity and Model Utility

**Setup.** To investigate whether realistic EEG generation necessarily leads to better foundation model adaptation, we analyze the relationship between generation fidelity and downstream utility. Following prior work Wang et al. (2026), we measure generation fidelity using TS-FID:

$$\text{TS-FID} = |\boldsymbol{\mu}_r - \boldsymbol{\mu}_g|_2^2 + \text{Tr}\left(\boldsymbol{\Sigma}_r + \boldsymbol{\Sigma}_g - 2(\boldsymbol{\Sigma}_r\boldsymbol{\Sigma}_g)^{1/2}\right), \tag{9}$$

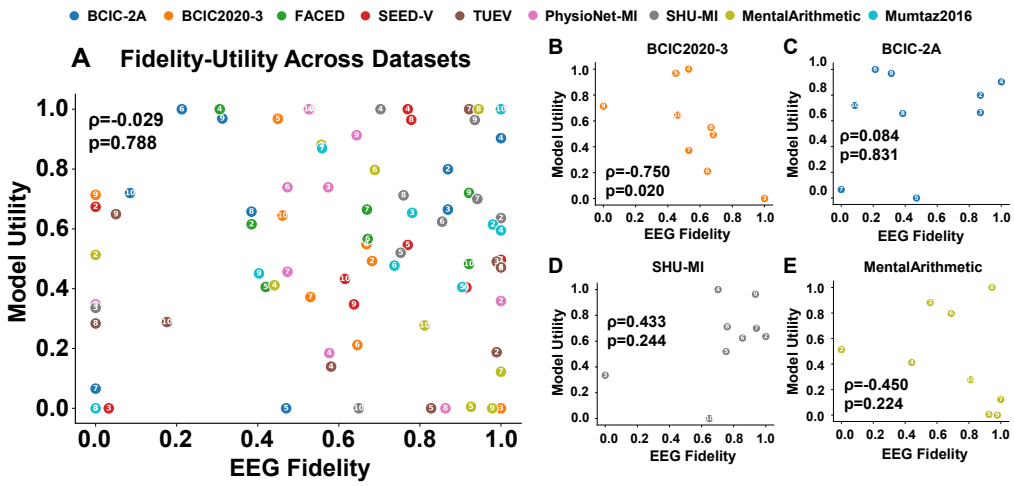


Figure 5: Analysis of the fidelity–utility gap.

where $(\boldsymbol{\mu}_r, \boldsymbol{\Sigma}_r)$ and $(\boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g)$ denote the statistics of real and generated EEG representations, respectively. Since lower TS-FID indicates higher fidelity, we define $\text{Fidelity} = -\text{TS-FID}$. The model utility of generated EEG at round $t$ is measured as:

$$\text{Model Utility}_t = U_t - U_{t-1}, \tag{10}$$

where $U_t$ denotes the downstream performance after incorporating generated EEG from round $t$. For fair comparison across datasets, both fidelity and utility are normalized within each dataset using min–max normalization ( $\hat{x} = (x - \min(x))/(\max(x) - \min(x))$). We then compute Spearman rank correlation between normalized fidelity and utility. **Results.** Figure 5 shows the relationship between EEG fidelity and model utility across datasets. Overall, fidelity exhibits no significant correlation with utility ($\rho = -0.029$, $p = 0.788$), suggesting that higher-fidelity EEG does not consistently translate into more effective foundation model adaptation. Similar patterns are observed within individual datasets, where the correlation varies across tasks and lacks a stable monotonic trend. **Analysis.** These results empirically reveal the fidelity–utility gap in EEG generation. Although fidelity measures how closely generated EEG matches the real-data distribution, it does not indicate whether the generated samples provide useful adaptation signals for foundation models. Therefore, optimizing EEG generation solely for distributional similarity may overlook task-relevant information required by downstream models. This motivates NeuroEvolve to shift the generation objective from producing realistic EEG toward discovering what EEG foundation models need for effective adaptation.

## 5 Conclusion

We present NeuroEvolve, a feedback-driven EEG generation paradigm that learns what EEG to generate based on model needs. NeuroEvolve combines Closed-loop Strategy Discovery, which progressively refines generation strategies through FM feedback, with Successful Experience Internalization, which distills successful refinements into reusable generation knowledge for unseen tasks. Extensive experiments across diverse EEG tasks, datasets, and foundation models demonstrate consistent improvements in downstream adaptation, with further analyses validating the effectiveness of feedback-driven refinement and the transferability of internalized experience. More broadly, our findings highlight the distinction between generation fidelity and model utility, suggesting a shift in EEG generation from reproducing what real EEG looks like toward learning what EEG foundation models actually need. **Limitations and Future Work.** While this work focuses on EEG foundation model adaptation, extending the feedback-driven generation paradigm to other neural modalities and broader adaptation settings remains an important direction for future work.

### AI USE STATEMENT

We did not use generative AI tools to generate scientific claims, conduct experiments, or analyze experimental results. The authors are responsible for the final content of this work.

### ETHICS STATEMENT

This work uses publicly available EEG datasets and does not involve new human subject experiments or data collection.

### REPRODUCIBILITY STATEMENT

To facilitate reproducibility, we provide detailed descriptions of the proposed framework (Appendix A.2, Appendix A.5, Appendix A.4), implementation details (Appendix A.12), and dataset preprocessing procedures (Appendix A.1). The anonymous source code repository is available at: `https://anonymous.4open.science/r/NeuroEvolve-C1D0/NeuroEvolve`.

## REFERENCES

Reza Abiri, Soheil Borhani, Eric W. Sellers, Yang Jiang, and Xiaopeng Zhao. A comprehensive review of eeg-based brain–computer interface paradigms. *Journal of Neural Engineering*, 16, 2019. URL `https://api.semanticscholar.org/CorpusID:54559538`.

Swati Aggarwal and Nupur Chugh. Review of machine learning techniques for eeg based brain computer interface. *Archives of Computational Methods in Engineering*, 29:3001 – 3020, 2022. URL `https://api.semanticscholar.org/CorpusID:255397903`.

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

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, 2023.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Ji-Hoon Jeong, Jeong-Hyun Cho, Young-Eun Lee, Seo-Hyun Lee, Gi-Hwan Shin, Young-Seok Kweon, José del R Millán, Klaus-Robert Müller, and Seong-Whan Lee. 2020 international brain–computer interface competition: A review. *Frontiers in human neuroscience*, 16:898300, 2022.

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

Gayal Kuruppu, Neeraj Wagh, Vaclav Kremen, and Yogatheesan Varatharajah. Eeg foundation models: a critical review of current progress and future directions. *Journal of neural engineering*, 23(2):021001, 2026.

Junhong Lai, Jiyu Wei, Lin Yao, and Yueming Wang. A simple review of eeg foundation models: Datasets, advancements and future perspectives. *arXiv preprint arXiv:2504.20069*, 2025.

Elnaz Lashgari, Dehua Liang, and Uri Maoz. Data augmentation for deep-learning-based electroencephalography. *Journal of neuroscience methods*, 346:108885, 2020.

Weibin Li, Wendu Li, Xihua Yin, Yushan You, Xinke Shen, Zongxiang Tan, and Quanying Liu. Cbp: Learning shared cognitive basis space and connectivity patterns for cross-cognitive-task brain dynamics modeling. *Neurocomputing*, 700:134353, 2026. ISSN 0925-2312. doi: https://doi.org/10.1016/j.neucom.2026.134353. URL `https://www.sciencedirect.com/science/article/pii/S0925231226017510`.

Bo Liu, Changdong Liu, Yanshan Xiao, Laiwang Liu, Weibin Li, and Xiaodong Chen. Adaboost-based transfer learning method for positive and unlabelled learning problem. *Knowledge-Based Systems*, 241:108162, 2022. ISSN 0950-7051. doi: https://doi.org/10.1016/j.knosys.2022.108162. URL https://www.sciencedirect.com/science/article/pii/S0950705122000260.

Wei Liu, Jie-Lin Qiu, Wei-Long Zheng, and Bao-Liang Lu. Comparing recognition performance and robustness of multimodal deep learning models for multimodal emotion recognition. *IEEE Transactions on Cognitive and Developmental Systems*, 14(2):715–729, 2021.

Fabien Lotte, Laurent Bougrain, Andrzej Cichocki, Maureen Clerc, Marco Congedo, Alain Rakotomamonjy, and Florian Yger. A review of classification algorithms for eeg-based brain–computer interfaces: a 10 year update. *Journal of neural engineering*, 15(3):031005, 2018.

Jun Ma, Banghua Yang, Wenzheng Qiu, Yunzhe Li, Shouwei Gao, and Xinxing Xia. A large eeg dataset for studying cross-session variability in motor imagery brain-computer interface. *Scientific Data*, 9(1):531, 2022.

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

John W. Pratt. Remarks on zeros and ties in the wilcoxon signed rank procedures. *Journal of the American Statistical Association*, 54:655–667, 1959. URL https://api.semanticscholar.org/CorpusID:120225998.

Yannick Roy, Hubert J. Banville, Isabela Albuquerque, Alexandre Gramfort, Tiago H. Falk, and Jocelyn Faubert. Deep learning-based electroencephalography analysis: a systematic review. *Journal of Neural Engineering*, 16, 2019. URL https://api.semanticscholar.org/CorpusID:58014179.

Gerwin Schalk, Dennis J McFarland, Thilo Hinterberger, Niels Birbaumer, and Jonathan R Wolpaw. Bci2000: a general-purpose brain-computer interface (bci) system. *IEEE Transactions on biomedical engineering*, 51(6):1034–1043, 2004.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=St1giarCHLP.

Kaido Värbu, Naveed Muhammad, and Yar Muhammad. Past, present, and future of eeg-based bci applications. *Sensors*, 22(9):3331, 2022.

Mumtaz Wajid. Mdd patients and healthy controls eeg data (new). 2016. URL https://api.semanticscholar.org/CorpusID:151858102.

Guangyu Wang, Wenchao Liu, Yuhong He, Cong Xu, Lin Ma, and Haifeng Li. EEGPT: Pretrained transformer for universal and reliable representation of EEG signals. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024a. URL https://openreview.net/forum?id=lvS2b8CjG5.

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 *The Thirteenth International Conference on Learning Representations*, 2025a. URL https://openreview.net/forum?id=NPNUHgHF2w.

Yifan Wang, Yijia Ma, Wen Li, and Chenyu You. Let EEG models learn EEG. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=TP8OuKKmsf.

Ziwei Wang, Siyang Li, Jingwei Luo, Jiajing Liu, and Dongrui Wu. Channel reflection: Knowledge-driven data augmentation for eeg-based brain–computer interfaces. *Neural Networks*, 176:106351, 2024b.

Ziwei Wang, Siyang Li, Xiaoqing Chen, and Dongrui Wu. Time–frequency transform based eeg data augmentation for brain–computer interfaces. *Knowledge-Based Systems*, 311:113074, 2025b.

Jiamin Wu, Zichen Ren, Junyu Wang, Pengyu Zhu, Yonghao Song, Mianxin Liu, Qihao Zheng, Lei Bai, Wanli Ouyang, and Chunfeng Song. Adabrain-bench: Benchmarking brain foundation models for brain-computer interface applications. *arXiv preprint arXiv:2507.09882*, 2025.

Qinfan Xiao, Ziyun Cui, Chi Zhang, SiQi Chen, Wen Wu, Andrew Thwaites, Alexandra Woolgar, Bowen Zhou, and Chao Zhang. Brainomni: A brain foundation model for unified EEG and MEG signals. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=cjHQj0tCy6.

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, 2019.

# A Appendix

## A.1 Additional Details on Datasets and Preprocessing

We provide additional details on the datasets, preprocessing procedures, and evaluation protocols used for the downstream BCI tasks.

**Motor Imagery Classification**

**BCIC-2A** is a widely used motor imagery (MI) EEG benchmark comprising recordings from 9 subjects with 22 EEG channels. It contains four MI classes: left hand, right hand, both feet, and tongue. We segment the EEG recordings into 4-s trials, resulting in 5,088 samples. Each trial is band-pass filtered between 0.3 and 50 Hz and resampled to 200 Hz. Following the evaluation protocol of CBraMod, we adopt a strict subject-independent split, using subjects 1–5 for training, subjects 6–7 for validation, and subjects 8–9 for testing.

**SHU-MI** is an MI EEG dataset comprising recordings from 25 subjects with 32 EEG channels. It contains two MI classes: left-hand and right-hand motor imagery. We segment the EEG recordings into 4-s trials, yielding 11,988 samples, and resample all signals to 200 Hz. We adopt a subject-independent split, with subjects 1–15 for training, subjects 16–20 for validation, and subjects 21–25 for testing.

**PhysioNet-MI** is a large-scale MI EEG dataset comprising recordings from 109 subjects with 64 EEG channels. We consider four MI classes: left fist, right fist, both fists, and both feet. The EEG recordings are segmented into 4-s trials, resulting in 9,837 samples. For preprocessing, we apply common average referencing (CAR), followed by a 0.3-Hz high-pass filter and a 60-Hz notch filter to suppress low-frequency drift and power-line interference, respectively. All signals are subsequently resampled to 200 Hz. We adopt a subject-independent split, using subjects 1–70 for training, subjects 71–89 for validation, and subjects 90–109 for testing.

**Emotion Recognition**

**FACED** is a large-scale, fine-grained emotion recognition EEG dataset comprising recordings from 123 subjects with 32 EEG channels. It covers nine emotion categories: amusement, inspiration, joy, tenderness, anger, fear, disgust, sadness, and neutral. We segment the EEG recordings into 10-s trials, resulting in 10,332 samples, and resample all signals to 200 Hz. We adopt a subject-independent split, using subjects 1–80 for training, subjects 81–100 for validation, and subjects 101–123 for testing.

**SEED-V** is an emotion recognition EEG dataset comprising recordings from 16 subjects with 62 EEG channels and five emotion categories: happiness, sadness, neutrality, disgust, and fear. Each subject participates in three experimental sessions, with 15 trials per session. Following the official dataset protocol, one corrupted session is excluded. For preprocessing, we first remove the task-irrelevant M1, M2, VEO, and HEO channels and extract individual trials according to the provided onset and offset annotations. The EEG signals are then resampled to 200 Hz and band-pass filtered between 0.3 and 75 Hz. Each trial is further divided into non-overlapping 1-s segments, resulting in 115,001 samples in total. Within each session, the 15 trials are evenly divided into training, validation, and test sets with a 5:5:5 split.

**Imagined Speech Classification**

**BCIC2020-3** is an imagined speech EEG classification dataset comprising recordings from 15 subjects with 64 EEG channels, originally sampled at 256 Hz. It contains five imagined speech classes corresponding to the words or phrases 'Hello," 'Help me," 'Stop," 'Thank you," and "Yes." Following the official dataset protocol, each class contains 60 training, 10 validation, and 10 test trials per subject, yielding 6,000 trials in total. Each trial consists of a 3-s EEG recording and is treated as an individual sample. We resample all EEG signals to 200 Hz for subsequent analysis.

**Mental Disorder Diagnosis**

**Mumtaz2016** is a clinical EEG dataset comprising 34 subjects with major depressive disorder (MDD) and 30 normal controls (NC), recorded with 19 EEG channels at a sampling rate of 256 Hz. EEG recordings were collected under three conditions: eyes-open, eyes-closed, and task. Following the setting of CBraMod, we use only the eyes-open and eyes-closed recordings. For preprocessing, the EEG signals are band-pass filtered between 0.3 and 75 Hz and notch filtered at 50 Hz to suppress power-line interference. The signals are then resampled to 200 Hz and segmented into non-overlapping 5-s windows, resulting in 7,143 samples. We adopt a subject-independent split: 21 MDD and 20 NC subjects are used for training, 5 MDD and 4 NC subjects for validation, and 8 MDD and 6 NC subjects for testing.

**Mental Stress Detection**

**MentalArithmetic** is an EEG dataset collected before and during mental arithmetic tasks for cognitive stress recognition. It comprises recordings from 36 subjects, where EEG recorded before the arithmetic task is labeled as *no mental stress*, while EEG recorded during the task is labeled as *mental stress*. The signals were recorded using 20 EEG channels at a sampling rate of 500 Hz. For preprocessing, we apply a 0.5–45 Hz band-pass filter. Following the setting of CBraMod, the signals are resampled to 200 Hz and segmented into non-overlapping 5-s windows, resulting in 1,707 samples. We adopt a subject-independent split, using subjects 1–28 for training, subjects 29–32 for validation, and subjects 33–36 for testing.

**EEG Event Classification**

**TUEV** is a clinical EEG dataset for event classification, comprising six event types: Spike and Sharp Wave (SPSW), Generalized Periodic Epileptiform Discharges (GPED), Periodic Lateralized Epileptiform Discharges (PLED), Eye Movement (EYM), Artifacts (ARTF), and Background Activity (BCKG). The EEG recordings contain 23 channels sampled at 250 Hz. Following the setting of CBraMod, we select 16 bipolar montage channels derived from the international 10–20 system for evaluation. The signals are band-pass filtered between 0.3 and 75 Hz, notch filtered at 60 Hz to suppress power-line interference, and resampled to 200 Hz. We segment the recordings into 5-s EEG trials, resulting in 112,491 samples. We retain the official training/test split and further divide the original training set into training and validation sets with an 8:2 ratio.

### A.2 Additional Details on Baselines

We provide additional details on the EEG foundation models and generative baselines used in our experiments.

**Pre-trained EEG Foundation Models**

**CBraMod** is a large-scale EEG foundation model built upon a Criss-Cross Transformer architecture that separately captures spatial and temporal dependencies in EEG signals. It further introduces asymmetric conditional positional encoding to preserve the positional information of EEG patches.

**LaBraM** is a large-scale brain foundation model designed to learn generalizable representations from heterogeneous EEG recordings. It represents EEG signals as channel-level patches and introduces a neural tokenizer based on vector-quantized neural spectrum prediction, which transforms continuous EEG signals into compact discrete neural representations for large-scale pre-training.

**EEGPT** introduces a spatio-temporal representation alignment strategy for EEG pre-training, where self-supervised learning targets are constructed from high-quality EEG representations rather than directly modeling raw signals. It further employs a hierarchical architecture to separately capture

spatial and temporal characteristics, reducing the computational cost of modeling long EEG sequences.

**BrainOmni** is a brain foundation model jointly pre-trained on EEG and MEG recordings. It introduces BrainTokenizer to transform continuous spatio-temporal brain activity into discrete representations and employs a sensor encoder to capture sensor-specific characteristics, enabling adaptation across heterogeneous recording devices and modalities.

**EEG Generation Methods**

**Vanilla Diffusion** serves as a diffusion-based generative baseline for EEG synthesis. It progressively perturbs real EEG signals with Gaussian noise through a forward diffusion process and learns the corresponding reverse denoising process. Once trained, synthetic EEG samples are generated by iteratively denoising samples initialized from noise.

**JET** is an EEG generation framework based on conditional flow matching. It formulates EEG synthesis as learning continuous probability paths and estimates a smooth vector field that transports samples from a noise distribution toward the real EEG data distribution. During training, JET further incorporates theoretically motivated constraints to preserve salient characteristics of EEG signals.

### A.3 More Details for Backbone Models

To evaluate the generalizability of NeuroEvolve across different EEG foundation models, we consider three representative backbones with distinct architectural designs and pre-training paradigms: CBraMod, LaBraM, and EEGPT.

**CBraMod** is a large-scale EEG foundation model designed to capture the intrinsic spatio-temporal structure of EEG signals. It introduces a Criss-Cross Transformer architecture to separately model spatial dependencies across EEG channels and temporal dependencies within EEG sequences. It further employs asymmetric conditional positional encoding to preserve the positional information of EEG patches.

**LaBraM** is a large-scale brain foundation model designed to learn generalizable representations from heterogeneous EEG recordings. It represents EEG signals as channel-level patches and introduces a vector-quantized neural tokenizer based on neural spectrum prediction, which transforms continuous EEG signals into compact discrete neural tokens for representation learning.

**EEGPT** adopts a spatio-temporal representation alignment strategy for EEG pre-training. Rather than directly modeling raw EEG signals, it constructs self-supervised learning targets from high-quality EEG representations. It further employs a hierarchical architecture to efficiently capture spatial and temporal characteristics while reducing computational complexity.

### A.4 Details of Evaluation Metrics

We provide detailed definitions of the evaluation metrics used for downstream EEG classification and the fidelity–utility analysis.

**Downstream Classification Metrics**

For binary classification tasks, we report Balanced Accuracy, AUC-PR, and AUROC. For multi-class classification tasks, we report Balanced Accuracy, Cohen's Kappa, and Weighted F1-score.

**Balanced Accuracy** computes the average recall across all classes, assigning equal importance to each class regardless of its frequency. Given $C$ classes, it is defined as

$$\text{Bal.Acc.} = \frac{1}{C}\sum_{c=1}^{C}\frac{\text{TP}_c}{\text{TP}_c + \text{FN}_c}, \tag{11}$$

where $\text{TP}_c$ and $\text{FN}_c$ denote the numbers of true positives and false negatives for class $c$, respectively. This metric is robust to class imbalance and is applicable to both binary and multi-class EEG classification tasks.

**AUC-PR** summarizes the trade-off between precision and recall across different classification thresholds. At threshold $\tau$, precision and recall are defined as

$$\text{Precision}(\tau) = \frac{\text{TP}(\tau)}{\text{TP}(\tau) + \text{FP}(\tau)}, \qquad \text{Recall}(\tau) = \frac{\text{TP}(\tau)}{\text{TP}(\tau) + \text{FN}(\tau)}. \tag{12}$$

AUC-PR is then expressed as the area under the precision–recall curve:

$$\text{AUC-PR} = \int_0^1 \text{Precision}(r), dr, \tag{13}$$

where $r$ denotes recall. AUC-PR emphasizes performance on the positive class and is particularly informative for imbalanced binary classification tasks.

**AUROC** evaluates the threshold-independent discriminative ability of a classifier by measuring the trade-off between the true positive rate (TPR) and false positive rate (FPR):

$$\text{TPR}(\tau) = \frac{\text{TP}(\tau)}{\text{TP}(\tau) + \text{FN}(\tau)}, \qquad \text{FPR}(\tau) = \frac{\text{FP}(\tau)}{\text{FP}(\tau) + \text{TN}(\tau)}. \tag{14}$$

The AUROC is defined as

$$\text{AUROC} = \int_0^1 \text{TPR}(f)\,\mathrm{d}f, \tag{15}$$

where $f$ denotes the false positive rate.

**Cohen's Kappa** measures the agreement between predicted and ground-truth labels while correcting for agreement expected by chance:

$$\kappa = \frac{p_o - p_e}{1 - p_e}, \tag{16}$$

where $p_o$ denotes the observed agreement and $p_e$ denotes the expected agreement by chance. Specifically,

$$p_o = \frac{1}{N}\sum_{c=1}^{C} n_{cc}, \qquad p_e = \sum_{c=1}^{C} \frac{n_{c\cdot}}{N}\frac{n_{\cdot c}}{N}, \tag{17}$$

where $n_{cc}$ is the number of correctly classified samples in class $c$, $n_{c\cdot}$ and $n_{\cdot c}$ denote the corresponding ground-truth and predicted class marginals, and $N$ is the total number of samples. Cohen's Kappa is commonly used for EEG classification because it accounts for chance-level agreement and is less sensitive to class imbalance than standard accuracy.

**Weighted F1-score** aggregates class-wise F1 scores according to the number of samples in each class. For class $c$, the F1 score is

$$\text{F1}_c = \frac{2\,\text{Precision}_c\,\text{Recall}_c}{\text{Precision}_c + \text{Recall}_c}. \tag{18}$$

The weighted score is then computed as

$$\text{F1}_{\text{weighted}} = \sum_{c=1}^{C} \frac{N_c}{N}\,\text{F1}_c, \tag{19}$$

where $N_c$ is the number of samples belonging to class $c$, and $N = \sum_{c=1}^{C} N_c$ is the total number of samples. This metric provides an overall measure of precision–recall performance while accounting for the empirical class distribution.

**Fidelity–Utility Analysis Metrics**

To investigate whether generating EEG that more closely resembles real data necessarily leads to more effective foundation model adaptation, we quantify generated EEG from two complementary perspectives: model utility and EEG fidelity.

**Model Utility** measures the downstream benefit of generated EEG to the target foundation model. At generation round $t$, let $U_{t-1}$ denote the downstream validation performance before incorporating

the EEG generated at round $t$, and $U_t$ denote the performance after incorporating these generated samples for model adaptation. We define model utility as the resulting performance change:

$$\text{Utility}_t = U_t - U_{t-1}. \tag{20}$$

Here, $U$ denotes validation performance, measured by AUROC for binary classification and Cohen's Kappa for multi-class classification. A positive utility indicates that the generated EEG improves downstream adaptation, whereas a negative utility indicates degraded model performance. This definition directly quantifies the usefulness of generated EEG from the perspective of the target model.

**EEG Fidelity** quantify the fidelity of generated EEG using Time-Series Fr'echet Inception Distance (TS-FID), following the evaluation protocol adopted by JET . Unlike the conventional FID developed for natural images, TS-FID measures the discrepancy between real and generated EEG distributions in a spectral feature space, thereby capturing their alignment in frequency-domain structure.

Let $\mathbf{z}_r$ and $\mathbf{z}_g$ denote spectral feature representations extracted from real and generated EEG samples, respectively. We approximate their distributions by Gaussians,

$$\mathbf{z}_r \sim \mathcal{N}(\boldsymbol{\mu}_r, \boldsymbol{\Sigma}_r), \qquad \mathbf{z}_g \sim \mathcal{N}(\boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g), \tag{21}$$

where $\boldsymbol{\mu}_r$ and $\boldsymbol{\mu}_g$ are the corresponding feature means, and $\boldsymbol{\Sigma}_r$ and $\boldsymbol{\Sigma}_g$ are their covariance matrices. TS-FID is computed as

$$\text{TS-FID} = \|\boldsymbol{\mu}_r - \boldsymbol{\mu}_g\|_2^2 + \text{Tr}\left(\boldsymbol{\Sigma}_r + \boldsymbol{\Sigma}_g - 2\left(\boldsymbol{\Sigma}_r^{1/2}\boldsymbol{\Sigma}_g\boldsymbol{\Sigma}_r^{1/2}\right)^{1/2}\right). \tag{22}$$

A lower TS-FID indicates greater similarity between the real and generated EEG distributions. For intuitive interpretation and correlation analysis, we define the EEG fidelity score as

$$\text{Fidelity} = -\,\text{TS-FID}, \tag{23}$$

such that a higher value consistently represents higher generation fidelity.

We further quantify the association between EEG fidelity and model utility using Spearman's rank correlation coefficient:

$$\rho_s = \text{corr}\left(\text{rank}(\text{Fidelity}), \text{rank}(\text{Utility})\right). \tag{24}$$

This analysis examines the relationship between EEG generation fidelity and its utility for foundation model adaptation.

### A.5 Details of Data Statistics

The data statistics summarizes the generated EEG from both the signal and representation perspectives. Given an EEG sample $x$ with label $y$, we first extract its signal statistics and obtain its representation $\mathbf{h} = F_\theta(x)$ using the frozen EEG foundation model. We then characterize the generated samples using the following statistics.

**Signal Quality.** For each sample, we compute its standard deviation $\sigma(x)$ and spectral power over five commonly used EEG frequency bands: $\delta$ (0.5–4 Hz), $\theta$ (4–8 Hz), $\alpha$ (8–13 Hz), $\beta$ (13–30 Hz), and $\gamma$ (30–50 Hz). The total spectral power is defined as

$$P(x) = P_\delta(x) + P_\theta(x) + P_\alpha(x) + P_\beta(x) + P_\gamma(x). \tag{25}$$

We characterize signal quality based on the deviation of these statistics from their typical values across the analyzed EEG set. Specifically, for $z \in \{\sigma(x), \log(P(x) + \epsilon)\}$, we define the robust deviation as

$$D(z) = \frac{|z - \text{median}(z)|}{\text{MAD}(z) + \epsilon}, \tag{26}$$

where $\text{MAD}(z) = \text{median}(|z - \text{median}(z)|)$. The signal quality score is then

$$Q(x) = \exp\left[-\frac{1}{2}\left(D(\sigma(x)) + D(\log(P(x) + \epsilon))\right)\right]. \tag{27}$$

A higher $Q(x)$ indicates that the sample exhibits signal statistics closer to the typical range of the analyzed EEG set.

**Class-Prototype Distances:** To characterize the location of each sample in the foundation model representation space, we compute the prototype of class $c$ as

$$\boldsymbol{\mu}_c = \frac{1}{|\mathcal{D}_c|} \sum_{i:y_i=c} \mathbf{h}_i. \tag{28}$$

For sample $i$, its distance to the prototype of its own class is

$$d_i^{\text{own}} = \|\mathbf{h}_i - \boldsymbol{\mu}_{y_i}\|_2, \tag{29}$$

while its distance to the nearest prototype of another class is

$$d_i^{\text{other}} = \min_{c \neq y_i} \|\mathbf{h}_i - \boldsymbol{\mu}_c\|_2. \tag{30}$$

Their difference defines the class margin

$$m_i = d_i^{\text{other}} - d_i^{\text{own}}. \tag{31}$$

A smaller margin indicates that the sample lies closer to the boundary between classes.

**Neighborhood Consistency:** We further characterize the local neighborhood of each sample using its $K$ nearest neighbors in the normalized representation space. Let $\mathcal{N}_K(i)$ denote the $K$ nearest neighbors of sample $i$. The same-label neighbor ratio is defined as

$$R_i = \frac{1}{K} \sum_{j \in \mathcal{N}_K(i)} \mathbb{I}(y_j = y_i), \tag{32}$$

where $\mathbb{I}(\cdot)$ is the indicator function. A larger $R_i$ indicates that the sample lies in a locally class-consistent region.

**Novelty Score:** We measure sample novelty using the distance to its nearest neighbor:

$$N_i = \min_{j \neq i} \left(1 - \frac{\mathbf{h}_i^\top \mathbf{h}_j}{\|\mathbf{h}_i\|_2 \|\mathbf{h}_j\|_2}\right). \tag{33}$$

A larger novelty score indicates that the sample is less similar to existing samples in the representation space.

**Boundary Score:** To identify samples near class boundaries, we combine the class margin with local neighborhood consistency. Let $\widetilde{m}_i \in [0, 1]$ denote the min–max normalized class margin. The boundary score is

$$B_i = \frac{1}{2}(1 - \widetilde{m}_i) + \frac{1}{2}(1 - R_i). \tag{34}$$

A larger $B_i$ indicates a more boundary-like sample: it is relatively close to another class prototype and has lower agreement with nearby samples.

**Class-Wise Clustering:** To characterize the internal structure of each class, we first reduce the representation dimensionality using PCA and then apply $K$-means separately within each class. For cluster $k$ of class $c$, its relative size is

$$r_{c,k} = \frac{|\mathcal{C}_{c,k}|}{|\mathcal{D}_c|}, \tag{35}$$

where $\mathcal{C}_{c,k}$ denotes the samples assigned to that cluster. For each sample, we also record its Euclidean distance to the corresponding cluster center,

$$d_i^{\text{cluster}} = \|\mathbf{z}_i - \boldsymbol{\nu}_{c,k}\|_2, \tag{36}$$

where $\mathbf{z}_i$ is its PCA embedding and $\boldsymbol{\nu}_{c,k}$ is the cluster center.

Finally, the boundary level of each cluster is summarized by

$$B_{c,k} = \frac{1}{|\mathcal{C}_{c,k}|} \sum_{i \in \mathcal{C}_{c,k}} B_i. \tag{37}$$

These statistics provide the planner and evaluator with a compact description of the generated EEG, including signal quality, representation novelty, class-boundary structure, neighborhood consistency, and within-class distributional structure.

### A.6 Details of Planner Prompt

The Planner determines EEG generation strategy for each class based on the statistics of the target dataset. In the initial round, the generation plan is constructed solely from the dataset summary. In subsequent rounds, the previous generation plan and Evaluator feedback are additionally provided, allowing the Planner to refine its strategy according to the observed utility of previously generated EEG. Figure 6 shows a simplified prompt template.

Planner Prompt Template

```
You are an EEG data generation planner.

Your goal is to design class-specific EEG generation plans
that improve downstream model adaptation.

Task:
{task description}

Dataset summary:
{EEG statistics}

Previous generation plan:   # available after the first round
{previous plan}

Evaluator feedback:         # available after the first round
{feedback on generated EEG and model utility}

Step 1: Select one generation strategy for each class.
1. Boundary enhancement
2. Diversity enhancement
3. Prototype enhancement

Step 2: Select class-specific reference samples.
1. Boundary enhancement select from boundary samples
2. Diversity enhancement select from rare samples
3. Prototype enhancement select from representative samples

Step 3: Return a valid JSON object containing:
{
  dataset_diagnosis: str,
  generation_plan: [
    {
      target_class: int,
      strategy: str,
      reference_sample_ids: list,
      noise_strength: float,
      explanation: str
    }
  ]
}
Include every class exactly once and return valid JSON only.
```

Figure 6: Simplified Planner Prompt Template.

### A.7 Details of Planner Output

To provide a concrete example of the generation plan produced by the Planner, we present a simplified output for SHU-MI, a binary motor imagery dataset. Given the dataset summary and class-wise statistics, the Planner first diagnoses the generation needs of each class and then selects a corresponding generation strategy, reference samples, and noise strength. An example output is shown in Fig. 7.

Example Planner Output for SHU-MI

```
dataset diagnosis:
Class 0 exhibits stronger boundary characteristics,
while Class 1 forms more compact and stable clusters.

generation_plan:
[{
  target_class: 0,
  strategy: boundary enhancement,
  reference_sample_ids: [3182, 1119, ...],
  noise_strength: 0.7,
  explanation: Generate around boundary samples
  to strengthen underrepresented decision regions.
},
{
  target_class: 1,
  strategy: prototype enhancement,
  reference_sample_ids: [6395, 160, ...],
  noise_strength: 0.3,
  explanation: Generate around representative samples
  to reinforce stable class characteristics.
}]
```

Figure 7: Example Planner Output for SHU-MI.

### A.8 Details of Evaluator Prompt

The Evaluator assesses whether the generated EEG is beneficial to the target foundation model and provides class-wise feedback for the next generation round. Specifically, it jointly considers the current generation strategy, the characteristics of the generated EEG, and the downstream fine-tuning response. Figure 8 shows a simplified prompt template.

### A.9 Details of Evaluator Output

The Evaluator analyzes the current generation plan together with the generated EEG statistics and downstream foundation model feedback, and provides class-wise suggestions for the next generation round. Its output summarizes which generation strategies should be retained or revised according to their observed utility to the target model. An example output is shown in Fig. 9

### A.10 Details of Generator

The generator is implemented as a lightweight conditional Transformer trained with flow matching. It takes a noisy EEG sample, its class label, and the current timestep as inputs, and predicts a velocity field that progressively transports the noisy signal toward the EEG data distribution.

**Flow-Matching Training:** Let $\mathbf{x}$ denote a real EEG sample and $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ denote Gaussian noise. For each training sample, we uniformly sample $t \sim \mathcal{U}(0, 1)$ and construct an intermediate state along a linear probability path:

$$\mathbf{x}_t = t\mathbf{x} + (1 - t)\boldsymbol{\epsilon}, \tag{38}$$

where $t = 0$ corresponds to Gaussian noise and $t = 1$ corresponds to the real EEG sample. Under this path, the target velocity is

$$\mathbf{v}^* = \mathbf{x} - \boldsymbol{\epsilon}. \tag{39}$$

Given $\mathbf{x}_t$, timestep $t$, and class label $y$, the generator predicts the corresponding velocity $\mathbf{v}_\theta(\mathbf{x}_t, t, y)$. We optimize the generator using the flow-matching objective

$$\mathcal{L}_{\mathrm{FM}} = \mathbb{E}_{\mathbf{x},\boldsymbol{\epsilon},t}\left[\|\mathbf{v}_\theta(\mathbf{x}_t, t, y) - (\mathbf{x} - \boldsymbol{\epsilon})\|_2^2\right]. \tag{40}$$

Evaluator Prompt Template

```
You are an evaluator for a closed-loop EEG foundation
model adaptation.

Task:
{task description}

Current generation plan:
{generation plan}

Generated EEG information:
{generated EEG statistics}

Foundation model feedback:
{fine-tuning performance}

For each class, evaluate:
1. Strategy effectiveness
2. Generation effectiveness
3. Model response

For each class, provide one recommendation:
1. maintain the current strategy
2. increase or reduce its effectiveness
3. switch to another available strategy

Return a valid JSON object:
{
  summary: str,
  feedback: [
    {
      target_class: int,
      issue: str,
      suggestion: str.
    }
  ]
}
Provide exactly one feedback entry for every class.
Base all conclusions only on the provided evidence.
```

Figure 8: Simplified Evaluator Prompt Template.

**EEG Generation:** During generation, the reference EEG selected by the Planner determines the initial state of the sampling trajectory. Given a reference sample $\mathbf{x}^r$ and the planned noise strength $s \in [0, 1]$, we set $t_0 = 1 - s$ and initialize

$$\mathbf{z}_{t_0} = t_0 \mathbf{x}^r + (1 - t_0)\boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}). \tag{41}$$

Thus, the noise strength controls the extent of perturbation around the selected reference EEG. Starting from $\mathbf{z}_{t_0}$, we integrate the learned velocity field from $t_0$ to 1 using Euler updates:

$$\mathbf{z}_{t_{k+1}} = \mathbf{z}_{t_k} + (t_{k+1} - t_k)\mathbf{v}_\theta(\mathbf{z}_{t_k}, t_k, y), \tag{42}$$

where $\{t_k\}_{k=0}^K$ is a uniformly spaced timestep schedule. The final state $\mathbf{z}_1$ is used as the generated EEG sample. Independent Gaussian noise is sampled for each generated instance, allowing multiple EEG samples to be synthesized from the same reference while following the class-specific generation plan.

Example Evaluator Output

```
summary: Class 1 remains underperforming, suggesting that
diversity enhancement may not sufficiently. Boundary enhancement
may better improve class separability.

feedback:
[{
  target_class: 0
  issue: Prototype enhancement maintains stable performance.
  suggestion: Keep prototype enhancement.
},
{
  target_class: 1
  issue: Diversity enhancement provides limited improvement.
  suggestion: Switch to boundary enhancement.
}]
```

Figure 9: Example Evaluator Output on SHU-MI.

### A.11 EEG Foundation Model Training Details

For all downstream experiments, we adopt the training protocol of CBraMod and use a consistent set of optimization hyperparameters across datasets and baselines. The complete configuration is summarized in Table 6.

Table 6: Hyperparameter configuration used for EEG foundation model adaptation.

| **Hyperparameter** | **Setting** |
|---|---|
| Training epochs | 50 |
| Batch size | 64 |
| Optimizer | AdamW |
| Initial learning rate | $1 \times 10^{-4}$ |
| Weight decay | $5 \times 10^{-2}$ |
| Learning rate scheduler | CosineAnnealingLR |
| Cosine annealing period | 50 epochs |
| Minimum learning rate | $1 \times 10^{-6}$ |
| Dropout rate | 0.1 |
| Gradient clipping norm | 1.0 |
| Label smoothing | 0.1 |

### A.12 Implementation Details

All experiments are implemented in Python 3.10.20 with PyTorch 2.5.1 and CUDA 12.1. Experiments are conducted on a Linux server running Ubuntu 20.04.6 LTS, equipped with an AMD EPYC 9554 64-Core Processor and an NVIDIA L40 GPU with 48 GB of memory.