MAGE: Towards Generalizable Multi-timescale EEG Representations

Tuan-Kiet Doan[1] **Van-Chien Nguyen**[1] **Trung-Hieu Tran**[1] **Quang-Hung Pham**[1]
Ngoc-Son Vu[2] **Huy Phan**[3] **Phi Le Nguyen**[4]
Duc-Han Le[1] **Nikola Simidjievski**[1] **Van-Tam Nguyen**[1]

[1]LTCI, Télécom Paris, Institut Polytechnique de Paris, France
[2]LIST3N, Université de Technologie de Troyes, France
[3]German Research Center for Artificial Intelligence (DFKI), Germany
[4]Institute for AI Innovation and Societal Impact, Hanoi University of Science and Technology, Vietnam

{tuan.doan,van-chien.nguyen,trung-hieu.tran,quang-hung.pham}@ip-paris.fr
son.vu@utt.fr, huy.phan@dfki.de, lenp@soict.hust.edu.vn
{duc-han.le,nikola.simidjievski,van-tam.nguyen}@telecom-paris.fr

Abstract

Electroencephalography (EEG) data plays an important role in understanding human brain activity, with various major applications. While current approaches, which largely build on EEG foundation models, have demonstrated strong performance across datasets and tasks, they typically rely on a fixed temporal scale. This design choice limits their ability to capture characteristics of neural dynamic behavior, often spanning across multiple timescales, which are naturally captured in EEG recordings. To address this limitation, we propose MAGE (**Multi-timescale Adaptive Generalization for EEG**), a foundation model that dynamically adapts to capture and integrate information across different timescales. Specifically, MAGE combines two key mechanisms: (i) a novel multi-timescale attention mechanism, capable of learning representations across multiple temporal scales and (ii) a trainable aggregation module, that dynamically identifies and adapts to the most informative timescale of the input signal. Experimental results across seven datasets demonstrate that MAGE consistently outperforms existing EEG foundation models while using substantially fewer parameters. Moreover, our results support the hypothesis that different downstream tasks benefit from distinct temporal scales. The code is available at the `MAGE repository`.

1 Introduction

One of science's biggest long-term goals is to understand and decode the human brain that drives applications in brain-computer interface [1, 2], healthcare [3–5], and neuroscience [6, 7]. Electroencephalography (EEG) has emerged as a potential modality for this purpose, because of its accessibility and superior temporal resolution [8, 9]. Various deep learning methods have been proposed for tasks like anomaly detection [10–12], sleep staging [13, 14], and motor imagery [15–17]. However, most of these approaches are typically task-specific and trained on small-scale datasets.

Recognizing these limitations, recent research has shifted toward foundation models [18–22]. These models are able to overcome labeled data scarcity via self-supervised pre-training on large, unlabeled datasets. While existing approaches focus on different aspects, such as spatio-temporal modeling [23, 24] or noise-robust training [25, 26], a common technique across these methods is the adoption of *single-timescale tokenization*, which segments the continuous EEG signal into **fixed-size** windows.

Relying on a fixed window size creates a time-frequency bottleneck: a short window captures high-frequency components but loses low-frequency information, whereas a long window captures

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

low-frequency oscillations but obscures transient details. Consequently, a single fixed scale limits the model's ability to capture the diverse frequency variations required by different tasks. For instance, motor imagery decoding relies on $\mu-$ and $\beta-$desynchronization [27–29], sleep staging distinguishes δ and σ waves [30–32], while emotion recognition targets Frontal α asymmetry and γ oscillations [33–35]. Although recent works use multi-granularity fusion [36, 37] or utilize foundation models with multi-kernel convolutions [38], the former remain task-specific and limited by labeled data, even foundational approaches like [38] struggle to readily adapt to the relevant temporal resolution required by the task at hand. This raises a key research question:

How do we enable multi-timescale learning, generalizable across different task and domains?

To answer this question, we introduce **MAGE** (**M**ulti-timescale **A**daptive **G**eneralization for **E**EG). MAGE first generates multi-timescale tokens to capture spatiotemporal dynamics at each scale level. It then adaptively fuses these representations to produce a specific output, supporting both pretraining and downstream tasks. Specifically, we propose a *Multi-timescale Tokenization* module to segment the raw signal using various patch sizes. Multi-timescale Tokenization enables the model to simultaneously capture fine-grained local patterns and coarse-grained global contexts, ensuring an effective representation of diverse brain activities. To encode these multi-timescale inputs, we propose a *Multi-timescale Attention* module to enrich spatiotemporal representations. This module utilizes temporal attention to capture long-range dependencies within each timescale, while leveraging spatial attention to fuse information from spatially separated channels. To control the aggregation of these scales, we introduce an *Aggregator* module. This module is designed to adaptively assign importance weights to each timescale level by analyzing the intrinsic temporal and spectral properties of the input to determine the importance of each scale. For example, sleep staging may favor coarse low-frequency patterns, whereas seizure detection often requires fine high-frequency transients. To the best of our knowledge, this is the first work to explore dynamic timescale aggregating for EEG analysis, establishing a new paradigm for enhancing the generalizability of EEG foundation models.

Our contributions are:

- We introduce MAGE, a multi-timescale transformer-based foundation model for better generalized EEG decoding, which captures information temporally across different timescales and spatially across different channels.
- We propose a novel Multi-timescale Attention module to extract rich spatiotemporal representations at each timescale, alongside an Aggregator that adaptively weights these scales based on input characteristics.
- The performance across seven EEG decoding tasks shows that MAGE outperforms existing foundation models across various tasks while using fewer parameters. We further analyze the contribution of each timescale, highlighting the necessity of multi-timescale decoding for EEG signals.

2 Related work

EEG Foundation Models. Various works aim to build models that generate generalizable representations of EEG signals by training on a large volume of unlabeled data. To address signal noise, LaBram [39] introduced a trainable neural tokenizer to generate discrete tokens, while EEGPT [40] proposed a dual-loss pipeline combining semantic alignment with detailed reconstruction. Addressing the topological heterogeneity problem of EEG datasets, LuNa [23] proposed a channel-unification module to map diverse channel sets into a unified space. CBraMod [41] proposed a criss-cross attention mechanism, which attends to information across the temporal and spectral dimensions only. However, it is limited by a fixed timescale tokenization, which may have suboptimal representations for diverse downstream tasks. CSBrain [38] extended CBraMod's idea by employing cross-scale spatial-temporal tokenization using multiple kernel sizes. Nevertheless, while CSBrain captures features at different scales, it lacks a mechanism to dynamically prioritize the specific scale most relevant to each individual task. MAGE addresses this problem by adopting a trainable aggregator to dynamically decide the importance of each scale at each specific task.

Multi-scale time series. Multi-scale modeling has proved to be an effective approach to learn the temporal dynamic characteristics of time series signals. Architectures like TimeMixer [42] and TimeMixer++ [43] adopt a sequence of pooling layers to downsample the input, generating a multi-scale representation that captures both global trends and local seasonal patterns. Meanwhile,

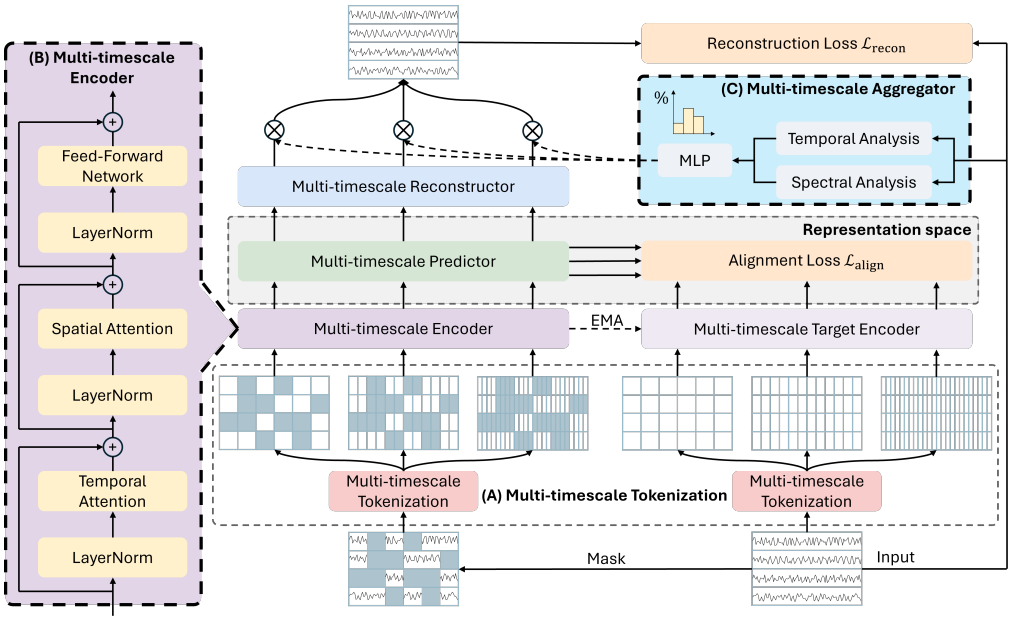


Figure 1: Overview of the MAGE pretraining pipeline. MAGE employs (A) Multi-timescale tokenization and (B) Multi-timescale encoder, optimized with a dual-loss scheme for both semantic and fine-grained learning. (C) Multi-timescale Aggregator dynamically weights each scale for the final reconstruction.

PathFormer [44], MPTSNet [45], and AdaMixT [46] segment the series into varying timescales and aggregate information from these scales within each Transformer block. However, these methods are designed for general time-series forecasting tasks, which involve signals with lower sampling rates and higher signal-to-noise ratios than EEG. While several approaches [37, 36, 47–49] have used multi-timescale modeling of medical signals, they have been designed as a task-specific architecture, limiting their applicability as a universal foundation model. Consequently, the application of multi-scale segmentation for high-frequency, noisy EEG analysis remains largely unexplored. MAGE overcomes many of the limitations of these methods, which typically rely on task-specific training pipelines. By leveraging a pre-training strategy, MAGE dynamically aggregates information across diverse timescales during downstream fine-tuning, successfully circumventing the feature suppression limitations inherent in existing fixed-scale methods.

Mixture of Experts (MoE). MoE is a network that dynamically activates different submodules, or experts, dependent on the input [50]. Recently, MoE has attracted much attention to addressing the difficulties of training in diverse and conflicting tasks [51]. While MoE is traditionally adopted to route data across different subnetworks, MAGE adapts this paradigm to the temporal domain. In our framework, each timescale is treated as an individual expert, with an aggregator dynamically fusing multi-scale information based on the signal's intrinsic properties.

3 MAGE

Figure 1 presents an overview of MAGE, our proposed multi-timescale foundation model for EEG representation. MAGE is composed of three main stages: (i) Multi-timescale Tokenization, which extracts features at different timescales (Sec. 3.1); (ii) Multi-timescale Attention, which captures complex spatiotemporal dependencies in EEG signals (Sec. 3.2); and (iii) Multi-timescale Aggregator, which dynamically weights each timescale based on input-dependent characteristics (Sec. 3.3). The full training pipeline is described in Sec. 3.4.

3.1 Multi-timescale Tokenization

Let $\mathbf{E} \in \mathbb{R}^{C \times T}$ denote EEG signals, with C channels and T timestamps. Due to the complex characteristics of EEG signals, different decoding tasks will depend on different frequencies of interest. However, previous EEG foundation models use fixed patch sizes, limiting generalizability across different decoding tasks. We, therefore, introduce Multi-timescale Tokenization, which segments $\mathbf{E}$ into tokens at multiple patch sizes, capturing both high-frequency transients and low-frequency long-range patterns for a multiscale representation. Let $\mathcal{P} = \{p_1, \ldots, p_G\}$ denote the set of G patch sizes. For each $p_g \in \mathcal{P}$, $\mathbf{E}$ is segmented into non-overlapping patches of length p_g, resulting in $\mathbf{E}_{p_g} \in \mathbb{R}^{C \times n_g \times p_g}$ with $n_g = \lfloor T/p_g \rfloor$ segments per channel.

To effectively capture both local temporal and frequency information within each segment, each patch $\mathbf{e}_{p_g} \in \mathbb{R}^{p_g}$ is processed by a dual-branch spectral-temporal embedding module [41]. The **temporal branch** extracts local temporal features via 1D convolutions, while the **spectral branch** captures frequency information via FFT followed by a linear transformation:

$$\mathbf{x}_{p_g} = \phi_{\text{temp}}(\mathbf{e}_{p_g})) + \phi_{\text{spec}}(|\text{FFT}(\mathbf{e}_{p_g})|) \in \mathbb{R}^D, \tag{1}$$

where D is the feature dimension. After the tokenization module, the raw EEG signal $\mathbf{E}$ is transformed to a set of final embeddings $\mathcal{X}_{\mathcal{P}} = \{\mathbf{X}^{(g)}\}_{g=1}^{G}$, where $\mathbf{X}^{(g)} \in \mathbb{R}^{C \times n_g \times D}$.

3.2 Multi-timescale Attention

Multivariate EEG signals exhibit complex spatiotemporal dynamics, involving both temporal dynamics within each channel and spatial connectivity across them. We decompose representation learning into two stages: the first captures temporal dynamics within each channel, and the second models spatial interactions across channels. This design allows the model to focus attention on the most relevant spatiotemporal regions. We propose the Multi-timescale Attention (MtA) module to process the multi-timescale input $\mathcal{X}_{\mathcal{P}} = \{\mathbf{X}^{(g)}\}_{g=1}^{G}$. MtA consists of two orthogonal attention modules (Temporal Attention module and Spatial Attention module) followed by a Feed-forward Network (Fig. 1).

Temporal Attention module captures sequential dynamics within each channel. For timescale $\mathbf{X}^{(g)}$ and channel $c \in \{1, \ldots, C\}$, the input tensor is sliced into the temporal sequence $\mathbf{X}_c \in \mathbb{R}^{n_g \times D}$.

Query, key, and value projections are computed as $\mathbf{Q}_c = \mathbf{X}_c\mathbf{W}_Q^T$, $\mathbf{K}_c = \mathbf{X}_c\mathbf{W}_K^T$, and $\mathbf{V}_c = \mathbf{X}_c\mathbf{W}_V^T$. Rotary Positional Encoding $\mathcal{R}(\cdot)$ is used to preserve temporal order. The temporal attention output $\mathbf{F}_c \in \mathbb{R}^{n_g \times D}$ is obtained via the output projection $\mathbf{W}_O^T$:

$$\mathbf{F}_c = \left[\text{softmax}\left(\frac{\mathcal{R}(\mathbf{Q}_c)\mathcal{R}(\mathbf{K}_c)^\top}{\sqrt{D}}\right)\mathbf{V}_c\right]\mathbf{W}_O^T + \mathbf{X}_c. \tag{2}$$

Outputs from all channels are then concatenated to form the intermediate feature tensor $\mathbf{F}^{(g)} \in \mathbb{R}^{C \times n_g \times D}$. **Spatial Attention module** captures dependencies across EEG channels. This attention mechanism is applied sequentially after the temporal attention module. To preserve spatial identity, a learnable channel embedding is added to each channel's representation as inspired by [40, 52]. For time step $t \in \{1, \ldots, n_g\}$, let $\mathbf{F}_t \in \mathbb{R}^{C \times D}$ denote features across channels. Spatial projections are $\mathbf{Q}'_t = \mathbf{F}_t\mathbf{W}_Q^S$, $\mathbf{K}'_t = \mathbf{F}_t\mathbf{W}_K^S$, and $\mathbf{V}'_t = \mathbf{F}_t\mathbf{W}_V^S$. The spatial interaction is then computed as:

$$\mathbf{U}_t = \left[\text{softmax}\left(\frac{\mathbf{Q}'_t(\mathbf{K}'_t)^\top}{\sqrt{D}}\right)\mathbf{V}'_t\right]\mathbf{W}_O^S + \mathbf{F}_t. \tag{3}$$

Feed-forward Network (FFN) integrates the features via two linear layers with a GeLU activation. The final representation after the MtA module is: $\mathbf{Y}^{(g)} = \mathbf{U}^{(g)} + \text{FFN}\big(\mathbf{U}^{(g)}\big)$.

3.3 Multi-timescale Aggregator

MtA enables multi-timescale modeling, but distinct EEG decoding tasks rely on specific frequency bands inherent to their temporal characteristics. Increasing timescales naively may add redundancy or noise rather than informative features, and manual tuning the scale is time-consuming. We, therefore, introduce a lightweight Aggregator module to dynamically weight each timescale for better

generalization across tasks. Given raw EEG $\mathbf{E} \in \mathbb{R}^{C \times T}$, the Aggregator uses two parallel temporal and frequency branches to determine the importance of each timescale.

Temporal Analysis. We employ a 1D convolutional network to encode high-level semantic features from the raw sequence. To have a global analysis, a global average pooling layer is adopted to compresses the temporal dimension to capture the overall distribution of features, resulting in a global embedding $\mathbf{a}_{\text{temp}}$ that is robust to temporal shifts.

Frequency Analysis. To capture global periodicities and seasonal information, the signal is transformed via FFT. We process the Magnitude and Phase components separately with two convolutional modules. The resulting representations are concatenated to produce the comprehensive frequency embedding $\mathbf{a}_{\text{freq}}$.

The final decision head, a multi-layer perceptron, concatenates the embeddings $\mathbf{a}_{\text{temp}}$ and $\mathbf{a}_{\text{freq}}$ to form $\mathbf{a}$, and outputs dynamic weight for each timescale:

$$\boldsymbol{\alpha} = \text{softmax}(\text{MLP}(\mathbf{a})) \in \mathbb{R}^G. \tag{4}$$

3.4 Training with MAGE

3.4.1 Pretraining pipeline

To extract both semantic and fine-grained information, we extend the EEGPT framework [40, 25] for multi-timescale training with alignment and reconstruction losses. The raw signal $\mathbf{E}$ is first processed by a Multi-timescale Tokenization block to produce token sequences $\mathcal{X}_{\mathcal{P}} = \{\mathbf{X}^{(g)}\}_{g=1}^{G}$.

Alignment loss. We use a teacher-student architecture to learn robust semantic representations. Let $\mathcal{M} = \{\mathbf{M}^{(g)}\}_{g=1}^{G}$, here $\mathbf{M}^{(g)}$ denote binary masks for tokens $\mathbf{X}^{(g)}$.

The student network processes masked input $\tilde{\mathcal{X}}_{\mathcal{P}} = \{\mathbf{X}^{(g)} \odot \mathbf{M}^{(g)}\}_{g=1}^{G}$. The teacher network has the identical architecture with the student version, and its parameters are updated via Exponential Moving Average (EMA) of student parameters. This module processes unmasked input $\mathcal{X}_{\mathcal{P}}$ to produce stable latent representations:

$$\mathcal{Z}_S = \text{Enc}_S(\tilde{\mathcal{X}}_{\mathcal{P}}), \mathcal{H}_T = \{\mathbf{H}_T^{(g)}\}_{g=1}^{G} = \text{Enc}_T(\mathcal{X}_{\mathcal{P}}), \tag{5}$$

where the indices S and T refer to the student and teacher, respectively. The Predictor maps visible tokens $\mathcal{Z}_S$ to $\hat{\mathcal{H}}_S = \{\hat{\mathbf{H}}_S^{(g)}\}_{g=1}^{G} = \text{Pred}(\mathcal{Z}_S)$. Both Encoder and Predictor are stacks of MtA modules. The alignment loss is computed as:

$$\mathcal{L}_{\text{align}} = \sum_{g=1}^{G} \left(\|\hat{\mathbf{H}}_S^{(g)} - \mathbf{H}_T^{(g)}\| \right). \tag{6}$$

Reconstruction loss. To reconstruct the original signal, latent outputs from all timescales are aggregated. While coarse timescale captures low-frequency components, fine timescale captures high-frequency features. The aggregator is adopted to dynamically weights each timescale. A key challenge when training with aggregator is mode collapse, akin to the "expert collapse" in MoE [51, 53]. The aggregator may collapse by assigning nearly all weight to one timescale, blocking updates to others. We mitigate this by injecting Gaussian noise into the logits before normalization, ensuring gradients are distributed across all timescales.

Formally, stochastic importance weights $\tilde{\boldsymbol{\alpha}}$ are computed by perturbing the decision boundary:

$$\tilde{\boldsymbol{\alpha}} = \text{softmax}\left(\text{MLP}(\mathbf{a}) + \boldsymbol{\epsilon}\right), \text{where } \boldsymbol{\epsilon} \sim \mathcal{N}(0, \sigma^2 \mathbf{I}). \tag{7}$$

The reconstructed signal $\hat{\mathbf{E}}$ is:

$$\hat{\mathbf{E}} = \sum_{g=1}^{G} \tilde{\alpha}_g \cdot \text{Dec}^{(g)}\left(\hat{\mathbf{h}}_S^{(g)}\right), \tag{8}$$

where $\text{Dec}^{(g)}()$ is a linear decoder of scale g. Reconstruction error is minimized via $\mathcal{L}_{\text{recon}} = \|\mathbf{E} - \hat{\mathbf{E}}\|_2^2$. The final loss is:

$$\mathcal{L} = \lambda_{\text{recon}} \mathcal{L}_{\text{recon}} + \lambda_{\text{align}} \mathcal{L}_{\text{align}}, \tag{9}$$

where λ_{recon} and λ_{align} balance the two tasks.

3.4.2 Downstream task adaptation

Similarly to pre-training, during downstream, stochastic noise is added to avoid reliance on a single timescale. The pre-trained encoder is frozen during downstream adaptation to reduce overfitting. The task prediction $\hat{\mathbf{y}}$ is obtained by aggregating the frozen representations via a scale-specific head:

$$\hat{\mathbf{y}} = \sum_{g=1}^{G} \tilde{\alpha}_g \cdot \text{Classifier}^{(g)} \left(\mathbf{Z}_S^{(g)} \right), \tag{10}$$

where $\text{Classifier}^{(g)}$ is the task-specific head for timescale g, and $\tilde{\boldsymbol{\alpha}}$ is the stochastic importance vector derived from the aggregator. Notably, this stochasticity is disabled during evaluation to ensure deterministic predictions. See Appendix C.1 for details.

4 Experimental results

4.1 Pretraining setup

Data processing. We pretrain MAGE on a subset of the Temple University Hospital EEG corpus (TUEG) dataset, with around 500 hours of training data. We opted for this subset size because our preliminary experiments scaling up to 2,500 hours (detailed in the Appendix D.8) demonstrated marginal improvements, suggesting diminishing returns at larger data scales. We follow the prior preprocessing steps [41, 38], detailed is in Appendix B. To avoid amplitude mismatches across devices, we apply z-score normalization per channel to each sample.

Pretraining setting. Following prior work [41, 38], we select 19 standard channels from 10-20 international system to pretrain MAGE. We select three different patch sizes: 50, 100, and 200, corresponding to 250ms, 500ms and 1s, respectively. During pretraining, we randomly mask 70% of the EEG samples and adopt the AdamW optimizer [54]. For the multi-timescale aggregator, a noise with zero mean and 0.3 standard deviation is applied.

4.2 Downstream task setup

Evaluation datasets. We conduct experiments on seven publicly available EEG datasets to evaluate the generalizability of our model. These datasets include BCIC-IV-2a [55], PhysioNet-MI [56, 57], SHU-MI [58], FACED [59], TUAB [60], KaggleERN [61], and Sleep-EDFx [62]. Unlike previous work that uses a fixed split of training, evaluation, and testing sets, in each data set, we conduct experiments many times, with a different set of training, evaluation, and testing subjects. We then report the mean and standard deviation across these different sets to summarize the overall performance. Further details are provided in the Appendix C.2.

Baselines. To evaluate the effectiveness of MAGE, we compare it with several current state-of-the-art foundation models for EEG, which include LaBram [39], BIOT [24], EEGPT [40], CBraMod [41], and CSBrain [38]. Following standard protocol, we keep the pre-trained encoder frozen across all experiments and append a consistent classifier head to assess the quality of the learned representations. Additionally, we compare MAGE with Medformer [37], which is a supervised multi-scale model for medical time-series classification. Following prior work, we use Balanced Accuracy, Cohen's Kappa, and Weighted F1 for multi-class classification problems, while we use Balanced Accuracy, AUC-PR, and AUROC for binary classification. More details about these baselines and metrics are illustrated in the Appendix C.3 and Appendix C.4.

Statistical significance test. The results for EEG signals often have large standard deviation. This is mainly because the substantial inter-subject variability in EEG signals. In our evaluation protocol, each fold uses a different test set, resulting in considerable variation in performance across folds. Therefore, aggregating results over these highly variable folds produces large standard deviations that may obscure the true performance differences. To provide further evidence of performance, we employed a corrected two-sided paired t-test ($p < 0.05$) [63] across all folds and 3 random seeds. The corrected paired t-test is designed to account for the non-independence of cross-validation folds. Demonstrating significance under this strict condition confirms the high reliability of the method.

Table 1: Comparison of MAGE with existing foundation models across various downstream tasks, metrics, and test subjects. Numbers under model names indicate parameter sizes in millions (M). MAGE-froz denotes MAGE where the aggregator is kept frozen during downstream evaluation. MAGE achieves the highest average ranking among all models. **Bold** indicates the best result, and underlined indicates the second best. The symbol ↓ indicates that the baseline model is significantly worse than MAGE, while ↑ indicates that the baseline model is significantly better than MAGE based on the corrected two-sided paired t-test ($p < 0.05$). The average rank is calculated across all metrics.

Dataset	Metrics	Medformer 1.4M	LaBram 5.8M	BIOT 3.2M	EEGPT 25.0M	CBraMod 4.0M	CSBrain 8.9M	MAGE **0.6M**	MAGE-froz **0.6M**
BCIC-IV-2a	bAcc	$52.68_{\pm 6.06}$↓	$49.75_{\pm 8.73}$↓	$49.92_{\pm 11.74}$	$50.49_{\pm 6.38}$↓	$46.09_{\pm 6.63}$↓	$55.11_{\pm 8.63}$	$\underline{56.78}_{\pm 6.08}$	$\mathbf{60.81}_{\pm 10.05}$
	Kappa	$36.91_{\pm 8.08}$↓	$33.00_{\pm 11.64}$↓	$33.23_{\pm 15.65}$	$33.98_{\pm 8.50}$↓	$28.12_{\pm 8.83}$↓	$40.14_{\pm 11.50}$	$\underline{42.38}_{\pm 8.11}$	$\mathbf{47.75}_{\pm 13.40}$
	wF1	$52.08_{\pm 6.34}$↓	$47.64_{\pm 10.12}$↓	$48.06_{\pm 13.36}$	$49.61_{\pm 6.77}$↓	$45.02_{\pm 6.96}$↓	$54.12_{\pm 8.96}$	$\underline{55.61}_{\pm 6.60}$	$\mathbf{59.30}_{\pm 11.72}$
PhysioNet	bAcc	$59.01_{\pm 3.39}$↓	$58.82_{\pm 4.40}$↓	$55.02_{\pm 3.33}$↓	$53.80_{\pm 2.91}$↓	$52.22_{\pm 4.01}$↓	$60.62_{\pm 3.14}$↓	$\underline{62.69}_{\pm 3.59}$	$\mathbf{64.08}_{\pm 3.51}$
	Kappa	$45.36_{\pm 4.52}$↓	$45.11_{\pm 5.86}$↓	$40.05_{\pm 4.43}$↓	$38.41_{\pm 3.88}$↓	$36.30_{\pm 5.35}$↓	$47.52_{\pm 4.18}$↓	$\underline{50.27}_{\pm 4.77}$	$\mathbf{52.12}_{\pm 4.66}$
	wF1	$58.92_{\pm 3.39}$↓	$58.86_{\pm 4.38}$↓	$55.10_{\pm 3.28}$↓	$53.78_{\pm 2.89}$↓	$52.20_{\pm 4.04}$↓	$60.54_{\pm 3.17}$↓	$\underline{62.71}_{\pm 3.54}$	$\mathbf{64.19}_{\pm 3.51}$
FACED	bAcc	$40.61_{\pm 1.40}$↓	$46.20_{\pm 1.80}$↓	$45.04_{\pm 2.13}$↓	$44.12_{\pm 1.80}$↓	$39.88_{\pm 2.06}$↓	$46.80_{\pm 1.59}$↓	$\underline{51.42}_{\pm 2.11}$	$\mathbf{51.44}_{\pm 2.21}$
	Kappa	$32.92_{\pm 1.51}$↓	$39.24_{\pm 1.98}$↓	$37.92_{\pm 2.35}$↓	$36.85_{\pm 2.05}$↓	$32.11_{\pm 2.18}$↓	$39.81_{\pm 1.74}$↓	$\underline{44.97}_{\pm 2.31}$	$\mathbf{44.98}_{\pm 2.56}$
	wF1	$40.25_{\pm 1.30}$↓	$46.04_{\pm 1.74}$↓	$44.77_{\pm 1.94}$↓	$43.81_{\pm 1.79}$↓	$39.61_{\pm 1.91}$↓	$\underline{46.32}_{\pm 1.48}$↓	$\mathbf{50.94}_{\pm 1.97}$	$\mathbf{50.94}_{\pm 2.33}$
Sleep-EDFx	bAcc	$61.92_{\pm 5.75}$↓	$62.89_{\pm 3.72}$↓	$\mathbf{70.64}_{\pm 3.43}$↑	$66.54_{\pm 4.00}$↓	$\underline{70.63}_{\pm 4.11}$	$70.02_{\pm 3.78}$	$70.08_{\pm 4.02}$	$66.36_{\pm 3.21}$
	Kappa	$58.43_{\pm 7.58}$↓	$63.96_{\pm 5.22}$↓	$68.12_{\pm 5.39}$	$65.46_{\pm 5.96}$↓	$\underline{68.79}_{\pm 6.41}$	$\mathbf{68.89}_{\pm 5.56}$	$\mathbf{68.89}_{\pm 5.98}$	$65.86_{\pm 5.70}$
	wF1	$68.69_{\pm 5.87}$↓	$73.30_{\pm 4.34}$↓	$77.02_{\pm 3.96}$	$74.60_{\pm 4.65}$↓	$\mathbf{77.65}_{\pm 4.36}$	$\underline{77.45}_{\pm 4.18}$	$77.33_{\pm 4.47}$	$74.74_{\pm 3.92}$
TUAB	bAcc	$78.72_{\pm 0.53}$	$78.61_{\pm 0.35}$	$\underline{79.15}_{\pm 0.97}$	$78.32_{\pm 0.34}$↓	$77.08_{\pm 0.85}$↓	$77.79_{\pm 0.58}$↓	$\mathbf{79.79}_{\pm 0.89}$	$79.01_{\pm 0.58}$
	PR	$84.40_{\pm 0.91}$↓	$\underline{86.83}_{\pm 0.24}$	$86.21_{\pm 1.59}$	$85.51_{\pm 0.94}$↓	$84.93_{\pm 1.02}$↓	$86.17_{\pm 0.62}$	$\mathbf{86.85}_{\pm 0.88}$	$86.29_{\pm 1.07}$
	ROC	$86.29_{\pm 0.44}$	$\underline{86.90}_{\pm 0.29}$	$86.61_{\pm 0.88}$	$85.94_{\pm 0.39}$↓	$85.82_{\pm 0.83}$	$86.39_{\pm 0.58}$	$\mathbf{87.13}_{\pm 0.80}$	$86.76_{\pm 0.44}$
KaggleERN	bAcc	$59.76_{\pm 1.17}$	$56.15_{\pm 1.46}$↓	$54.18_{\pm 1.65}$↓	$57.32_{\pm 1.59}$↓	$56.38_{\pm 1.26}$↓	$57.56_{\pm 1.39}$↓	$\underline{60.92}_{\pm 1.32}$	$\mathbf{61.10}_{\pm 2.14}$
	PR	$77.77_{\pm 1.29}$	$77.12_{\pm 0.67}$	$75.05_{\pm 1.97}$↓	$77.17_{\pm 1.85}$	$76.96_{\pm 1.19}$	$77.65_{\pm 2.22}$	$\underline{78.40}_{\pm 1.91}$	$\mathbf{79.23}_{\pm 2.05}$
	ROC	$63.29_{\pm 1.32}$	$60.86_{\pm 1.04}$↓	$57.25_{\pm 2.20}$↓	$61.19_{\pm 2.47}$↓	$60.18_{\pm 1.34}$↓	$61.88_{\pm 2.55}$	$\underline{64.68}_{\pm 1.97}$	$\mathbf{65.58}_{\pm 2.30}$
SHU-MI	bAcc	$61.42_{\pm 3.26}$	$\mathbf{63.27}_{\pm 3.65}$	$60.55_{\pm 3.25}$	$60.01_{\pm 2.80}$↓	$62.40_{\pm 3.68}$	$\underline{62.56}_{\pm 3.67}$	$62.22_{\pm 3.49}$	$61.87_{\pm 2.56}$
	PR	$66.67_{\pm 4.71}$	$\mathbf{69.34}_{\pm 5.43}$	$65.86_{\pm 4.72}$	$63.53_{\pm 3.51}$↓	$68.05_{\pm 5.95}$	$\underline{69.03}_{\pm 5.34}$	$68.63_{\pm 4.77}$	$68.24_{\pm 4.00}$
	ROC	$66.59_{\pm 4.52}$	$\mathbf{69.23}_{\pm 5.21}$	$65.73_{\pm 4.73}$	$64.05_{\pm 3.72}$↓	$68.10_{\pm 5.25}$	$\underline{68.80}_{\pm 4.66}$	$67.91_{\pm 4.73}$	$67.66_{\pm 3.80}$
Average Rank		5.43	4.76	5.48	6.14	6.29	3.31	**2.14**	2.45

4.3 Experimental Results

MAGE validates the superiority of multi-timescale method over rigid fixed-patch baselines. As shown in Table 1, MAGE achieves a better or comparable performance compared to other models across various tasks. In BCIC-IV-2a, PhysioNet-MI, FACED, TUAB, and KaggleERN, MAGE outperforms the other models with a gap of $2-3\%$. This consistent gap highlights MAGE's superiority, which stems from its use of multiple patch sizes to effectively capture diverse temporal dynamics. On the Sleep-EDFx and SHU-MI datasets, MAGE performs comparably to the top model, CBraMod. Overall, MAGE earns the best average rank of 2.14. It easily surpasses the runner-up, CSBrain, which ranks at 3.31. These results provide strong empirical evidence for our core hypothesis: effective decoding requires adapting temporal granularity to the specific task. Impressively, MAGE achieves these state-of-the-art results with only 0.6M parameters, a fraction of the size of contemporary foundation models. This demonstrates that dividing signals into multi-timescale tokens significantly enhances modeling performance. When we conduct a statistical significance test, despite the large variance across different evaluation subjects, MAGE still performs significantly better than all baselines on PhysioNet-MI and FACED. On BCIC2a, KaggleERN, and Sleep-EDFx, it significantly outperforms almost other foundation models while for TUAB and SHU-MI, MAGE achieves comparable results across the board. Notably, MAGE demonstrates no statistically significant drop in performance compared to any baseline across all evaluated datasets, except when compared to BIOT on Sleep-EDFx using the Balanced Accuracy metric.

Training the Aggregator at each downstream task yields better performance. Table 1 also compares the effects of freezing versus fine-tuning the aggregator during downstream tasks. Although a frozen aggregator remains reasonably effective on datasets where individual scales inherently perform well, such as PhysioNet-MI or BCIC-IV-2a, it struggles when there is a significant performance disparity between scales. For instance, on the Sleep-EDFx dataset, freezing the aggregator results in a $3-4\%$ performance drop, highlighting the need to fine-tune the aggregator to properly balance diverse scales. While a frozen aggregator already leads to competitive performance, the ability to adaptively adjust weights via a trainable aggregator is a significant structural advantage of MAGE over other baselines. In practical applications, this adaptability can be particularly beneficial, as the optimal aggregation weights for downstream tasks do not necessarily remain the same as those learned during pre-training.

Table 2: Comparison of different timescales performance. MAGE achieves a better performance than any single timescales, while MAGE-router is able to achieve a fair comparison with the best single scale in some tasks. **Bold** indicates the best result, and underlined indicates the second best. The average rank is calculated across all metrics. The result ablation is reported on 1 seed.

Dataset	Metrics	250ms only	500ms only	1s only	MAGE-router	MAGE
FACED	Balanced Acc	$50.93_{\pm 2.82}$	$49.78_{\pm 1.80}$	$46.65_{\pm 1.84}$	$49.87_{\pm 2.45}$	$\mathbf{51.30}_{\pm 3.11}$
	Cohen's Kappa	$44.41_{\pm 3.19}$	$43.14_{\pm 1.97}$	$39.79_{\pm 1.98}$	$43.22_{\pm 2.78}$	$\mathbf{44.89}_{\pm 3.43}$
	Weighted F1	$50.43_{\pm 2.91}$	$49.20_{\pm 1.75}$	$46.36_{\pm 1.53}$	$49.36_{\pm 2.45}$	$\mathbf{50.97}_{\pm 2.96}$
Physionet-MI	Balanced Acc	$60.08_{\pm 0.16}$	$60.65_{\pm 0.98}$	$60.55_{\pm 0.96}$	$58.83_{\pm 1.35}$	$\mathbf{61.15}_{\pm 0.86}$
	Cohen's Kappa	$46.81_{\pm 0.21}$	$47.58_{\pm 1.30}$	$47.41_{\pm 1.28}$	$45.13_{\pm 1.79}$	$\mathbf{48.23}_{\pm 1.15}$
	Weighted F1	$60.17_{\pm 0.35}$	$60.66_{\pm 0.96}$	$60.47_{\pm 1.05}$	$58.83_{\pm 1.42}$	$\mathbf{61.16}_{\pm 0.90}$
Sleep-EDFx	Balanced Acc	$67.25_{\pm 3.57}$	$66.47_{\pm 2.94}$	$67.99_{\pm 3.72}$	$68.69_{\pm 3.67}$	$\mathbf{70.24}_{\pm 4.63}$
	Cohen's Kappa	$63.94_{\pm 6.47}$	$66.81_{\pm 6.05}$	$68.13_{\pm 6.07}$	$65.33_{\pm 5.77}$	$\mathbf{68.91}_{\pm 6.47}$
	Weighted F1	$74.02_{\pm 5.06}$	$75.15_{\pm 4.63}$	$76.34_{\pm 4.35}$	$75.30_{\pm 3.99}$	$\mathbf{77.31}_{\pm 5.00}$
SHU-MI	Balanced Acc	$62.04_{\pm 3.78}$	$62.01_{\pm 3.10}$	$61.74_{\pm 3.75}$	$61.08_{\pm 3.60}$	$\mathbf{62.43}_{\pm 3.18}$
	AUC-PR	$\mathbf{68.84}_{\pm 5.06}$	$68.38_{\pm 4.24}$	$68.28_{\pm 4.43}$	$67.11_{\pm 5.38}$	$68.73_{\pm 5.03}$
	AUROC	$\mathbf{68.14}_{\pm 4.56}$	$67.81_{\pm 4.09}$	$67.46_{\pm 4.29}$	$66.76_{\pm 4.97}$	$\mathbf{68.14}_{\pm 5.06}$
Average Rank		3.00	3.25	3.58	4.00	**1.08**

The importance of each timescale is task-dependent. Figure 2 demonstrates the weight distribution of three timescales across four datasets: FACED, PhysioNet-MI, Sleep-EDFx, and SHU-MI. The results indicate that FACED and SHU-MI rely heavily on the finest timescale (250ms), assigning it weights of 60% and 45% respectively, whereas the coarse timescale (1s) makes a small contribution. Meanwhile, the Sleep-EDFx and PhysioNet-MI prioritize larger patch sizes, evidenced by more than 45% weight for the scale 1s compared to less than 20% for the scale 250ms. These findings suggest that while Sleep-EDFx necessitates long-window contexts, emotion recognition like FACED benefit more from short-patch information, providing valuable insight for selecting task-specific patch sizes. Notably, we observe that the aggregator adapts dynamically to each individual sample rather than outputting a nearly fixed value for each dataset. More detail about how the aggregator adapts to each individual sample is illustrated in the Appendix. D.1.

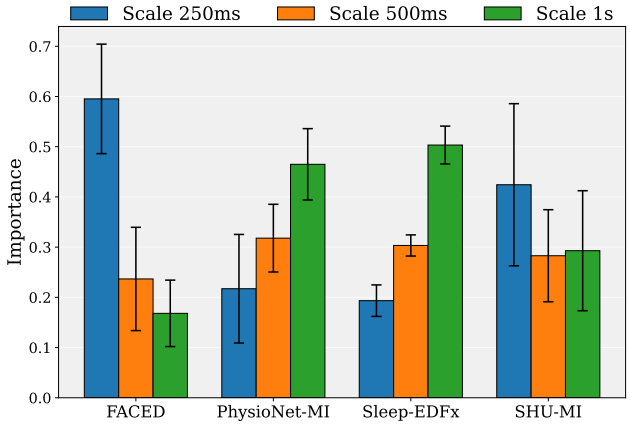


Figure 2: Distribution of importance weights across timescales by dataset. It can be shown that each dataset requires a different scale to capture relevant information.

No single fixed patch size offers a universal solution, as the optimal granularity shifts from task to task. To validate MAGE's architecture, we conducted an ablation study to assess performance when restricting the model to a single timescale. In this setup, only one patch size is active, negating contributions from others. As shown in Table 2, the performance of each timescale varies heavily depending on the tasks. In FACED dataset, the timescale with patch size 250ms only outperforms the scales with patch size 500ms and patch size 1s. Conversely, on the Sleep-EDFx dataset, the patch size of 1s proved superior to the other two. Across almost all datasets, MAGE achieves a fair or better performance compared to any single timescale. This demonstrates that MAGE can effectively aggregate information from all timescales for each specific task.

MAGE-router addresses the trade-off between accuracy and efficiency by dynamically selecting the single most relevant granularity. While MAGE outperforms each single timescale level, it incurs a higher computational cost. To reduce the this, we introduce MAGE-router, an efficient

Table 3: Comparison of MAGE with other multi-timescale methods. MAGE remains competitive with the Inter Attention method, even while utilizing a lightweight aggregator in contrast to a complex attention module. **Bold** indicates the best result, and underlined indicates the second best. The symbol ↓ indicates that the baseline model is significantly worse than MAGE.

Dataset	Metrics	MAGE	Inter Attention	Uniform Weight
FACED	Balanced Acc	**51.42**$_{\pm 2.11}$	50.79$_{\pm 3.04}$	50.85$_{\pm 2.48}$
	Cohen's Kappa	**44.97**$_{\pm 2.31}$	44.27$_{\pm 3.33}$	44.43$_{\pm 2.73}$
	Weighted F1	**50.94**$_{\pm 1.97}$	50.26$_{\pm 2.93}$	50.52$_{\pm 2.38}$
PhysioNet-MI	Balanced Acc	**62.69**$_{\pm 3.59}$	60.48$_{\pm 0.87}$	60.90$_{\pm 1.02}$
	Cohen's Kappa	**50.27**$_{\pm 4.77}$	47.31$_{\pm 1.17}$	47.87$_{\pm 1.35}$
	Weighted F1	**62.71**$_{\pm 3.54}$	60.68$_{\pm 0.87}$	61.09$_{\pm 1.05}$
Sleep-EDFx	Balanced Acc	70.08$_{\pm 4.02}$	**70.14**$_{\pm 3.33}$	69.72$_{\pm 3.62}$
	Cohen's Kappa	**68.89**$_{\pm 5.56}$	66.10$_{\pm 6.14}$↓	65.56$_{\pm 6.51}$↓
	Weighted F1	**77.33**$_{\pm 4.47}$	75.37$_{\pm 4.49}$↓	75.42$_{\pm 4.50}$↓
SHU-MI	Balanced Acc	62.22$_{\pm 3.49}$	**62.95**$_{\pm 3.14}$	62.77$_{\pm 2.92}$
	AUC-PR	68.63$_{\pm 4.77}$	69.59$_{\pm 4.34}$	**69.60**$_{\pm 4.24}$
	AUROC	67.91$_{\pm 4.73}$	**69.01**$_{\pm 4.15}$	68.95$_{\pm 4.17}$

variant designed for diverse downstream tasks. In this variant, after calculating adaptive weights, only the timescale corresponding to the largest weight is activated, while the others are deactivated. While MAGE-router reduces computational overhead, it experiences a performance drop on certain datasets, at times underperforming even the individual single-scale models (Table 2). We hypothesize this is due to the hard-selection mechanism; activating only one scale per sample may disperse gradient updates. Regardless, by automatically routing signals to their optimal scales, MAGE-router eliminates the need for tedious manual hyperparameter tuning. This highly adaptive, resource-optimized approach justifies the performance trade-off, making MAGE-router an ideal solution for low-latency, energy-efficient applications like wearable devices and real-time BCIs.

An aggregator is enough for multi-timescale learning. To validate the Aggregator's effectiveness and efficiency, we benchmark MAGE against two other multi-timescale methods. The first adopts a uniform distribution across all scales. The second applies Medformer's Inter Attention, where the Aggregator is replaced by a Transformer layer inside each encoder block to facilitate cross-scale interaction by attending to different timescales simultaneously. Under identical settings, MAGE performs better than both methods on the FACED, PhysioNet-MI, and Sleep-EDFx datasets (Table. 3). Conversely, Inter Attention is most effective on the SHU-MI dataset. To further validate these findings, we statistically compared MAGE against these approaches. Our analysis shows that MAGE significantly outperforms both baselines on two metrics for the Sleep-EDFx dataset, while maintaining comparable performance across the remaining datasets. Notably, since our aggregator contains only a few thousand parameters, MAGE maintains an efficiency comparable to the Uniform Weight method. Meanwhile, integrating additional attention modules within each block, as in Inter Attention, increases model size, particularly when scaling up. Ultimately, these results confirm that our adaptive aggregator not only guarantees stable baseline performance but also provides a distinct, statistically significant advantage without the computational overhead of alternative multi-scale strategies.

Limitations. Despite these benefits, there are a few limitations that require attention. First, while MAGE demonstrates strong generalizability, it remains dependent on a pre-defined set of patch sizes as a hyperparameter that requires manual optimization. Our work, however, demonstrates that incorporating a diverse range of patch scales is precisely what drives performance gains, representing a significant stride toward a truly generalizable EEG foundation model. Second, the attention mechanism costs quadratically with sequence length, leading to high computational costs when processing multi-timescale inputs simultaneously. Third, as shown in Figure 2, while both SHU-MI and PhysioNet-MI involve motor imagery, their scale preferences differ. We acknowledge that this difference cannot be fully explained at present. Despite sharing a motor imagery paradigm, the two datasets differ in channel configuration, task labels, cue design, and session structure, which are factors that may affect the temporal characteristics of the learned EEG patterns. One possible reason is that longer windows capture more stable mu/beta modulation, whereas shorter windows capture

transient or cue-related activity, and the specific target classes of each dataset dictate which of these temporal features are most critical for classification. Therefore, the learned weights should be viewed as dataset-specific preferences rather than universal properties of motor imagery.

Toward generalizable multi-timescale learning. Lastly, we revisit our core research question: our findings demonstrate that single-scale tokenization is insufficient for building generalized EEG foundation models. By decomposing signals to learn features across distinct timescales and dynamically weighting them, we effectively reduce the architectural complexity required for feature extraction. Consequently, this approach ensures robust generalization across diverse domains while requiring a remarkably small number of parameters.

5 Conclusion

In this work, we demonstrate the effectiveness of a multi-timescale architecture in building generalizable EEG foundation models. Our proposed **MAGE** effectively captures multi-timescale information by segmenting input signals by varying patch sizes. Crucially, MAGE employs a dynamic weighting mechanism, allowing it to adaptively prioritize different timescales based on the specific characteristics of the input signal. Experimental results show that our model achieves superior performance across different datasets spanning different frequency bands.

We believe that multi-timescale architectures for EEG signals remain a promising yet underexplored direction. Currently, while MAGE demonstrates strong generalizability, its reliance on a manually optimized, pre-defined list of patch sizes remains a limitation. Our future work will primarily address this by seeking ways to dynamically generate or select patch scales, removing the need for a fixed list. Additionally, we plan to reduce the computational cost of our attention mechanisms. By doing so, the model will be able to efficiently process a broader, continuous range of temporal information, paving the way for a more robust EEG foundation model. Furthermore, we aim to apply MAGE in a multi-modal context, where EEG can be aligned with other sources of signal like ECG, EMG, EOG [64], speech [65], or fMRI [66].

Acknowledgement

Part of this work was funded by Hi!PARIS Center on Data Analytics and Artificial Intelligence, by the European Union's HORIZON Research and Innovation Programme under grant agreement No 101120657, project ENFIELD (European Lighthouse to Manifest Trustworthy and Green AI) and by French National Research Agency (ANR-22-PEFT-0003 and ANR-22-PEFT-0007) as part of France 2030, the NF-NAI project and NF-FITNESS project.

References

[1] Jonathan R Wolpaw and Dennis J McFarland. Control of a two-dimensional movement signal by a noninvasive brain-computer interface in humans. *Proceedings of the national academy of sciences*, 101(51):17849–17854, 2004.

[2] Jonathan R Wolpaw. Brain–computer interfaces. In *Handbook of clinical neurology*, volume 110, pages 67–74. Elsevier, 2013.

[3] Kai Keng Ang and Cuntai Guan. Brain-computer interface in stroke rehabilitation. *Journal of Computing Science and Engineering*, 7(2):139–146, 2013.

[4] Janis J Daly and Jonathan R Wolpaw. Brain–computer interfaces in neurological rehabilitation. *The Lancet Neurology*, 7(11):1032–1043, 2008.

[5] Sander Koelstra, Christian Muhl, Mohammad Soleymani, Jong-Seok Lee, Ashkan Yazdani, Touradj Ebrahimi, Thierry Pun, Anton Nijholt, and Ioannis Patras. Deap: A database for emotion analysis; using physiological signals. *IEEE transactions on affective computing*, 3(1): 18–31, 2011.

[6] Yukiyasu Kamitani and Frank Tong. Decoding the visual and subjective contents of the human brain. *Nature neuroscience*, 8(5):679–685, 2005.

[7] Tomoyasu Horikawa, Masako Tamaki, Yoichi Miyawaki, and Yukiyasu Kamitani. Neural decoding of visual imagery during sleep. *Science*, 340(6132):639–642, 2013.

[8] Soraia M Alarcao and Manuel J Fonseca. Emotions recognition using eeg signals: A survey. *IEEE transactions on affective computing*, 10(3):374–393, 2017.

[9] Sharlene N Flesher, John E Downey, Jeffrey M Weiss, Christopher L Hughes, Angelica J Herrera, Elizabeth C Tyler-Kabara, Michael L Boninger, Jennifer L Collinger, and Robert A Gaunt. A brain-computer interface that evokes tactile sensations improves robotic arm control. *Science*, 372(6544):831–836, 2021.

[10] Subhrajit Roy, Isabell Kiral-Kornek, and Stefan Harrer. Chrononet: A deep recurrent neural network for abnormal eeg identification. In *Conference on artificial intelligence in medicine in Europe*, pages 47–56. Springer, 2019.

[11] Özal Yıldırım, Ulas Baran Baloglu, and U Rajendra Acharya. A deep convolutional neural network model for automated identification of abnormal eeg signals. *Neural Computing and Applications*, 32(20):15857–15868, 2020.

[12] KG Van Leeuwen, H Sun, M Tabaeizadeh, AF Struck, MJAM Van Putten, and MB Westover. Detecting abnormal electroencephalograms using deep convolutional networks. *Clinical neurophysiology*, 130(1):77–84, 2019.

[13] Caihong Zhao, Jinbao Li, and Yahong Guo. Sleepcontextnet: A temporal context network for automatic sleep staging based single-channel eeg. *Computer Methods and Programs in Biomedicine*, 220:106806, 2022.

[14] Huy Phan and Kaare Mikkelsen. Automatic sleep staging of eeg signals: recent development, challenges, and future directions. *Physiological Measurement*, 43(4):04TR01, 2022.

[15] Wei-Yen Hsu and Ya-Wen Cheng. Eeg-channel-temporal-spectral-attention correlation for motor imagery eeg classification. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 31:1659–1669, 2023.

[16] Hamdi Altaheri, Ghulam Muhammad, Mansour Alsulaiman, Syed Umar Amin, Ghadir Ali Altuwaijri, Wadood Abdul, Mohamed A Bencherif, and Mohammed Faisal. Deep learning techniques for classification of electroencephalogram (eeg) motor imagery (mi) signals: A review. *Neural Computing and Applications*, 35(20):14681–14722, 2023.

[17] Jamal F Hwaidi and Thomas M Chen. Classification of motor imagery eeg signals based on deep autoencoder and convolutional neural network approach. *IEEE access*, 10:48071–48081, 2022.

[18] 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.

[19] Wei-Bang 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. *arXiv preprint arXiv:2409.00101*, 2024.

[20] Daoze Zhang, Zhizhang Yuan, Yang Yang, Junru Chen, Jingjing Wang, and Yafeng Li. Brant: Foundation model for intracranial neural signal. *Advances in Neural Information Processing Systems*, 36:26304–26321, 2023.

[21] Wenhui Cui, Woojae Jeong, Philipp Thölke, Takfarinas Medani, Karim Jerbi, Anand A Joshi, and Richard M Leahy. Neuro-gpt: Towards a foundation model for eeg. In *2024 IEEE International Symposium on Biomedical Imaging (ISBI)*, pages 1–5. IEEE, 2024.

[22] 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. *arXiv preprint arXiv:2510.21585*, 2025.

[23] Berkay Döner, Thorir Mar Ingolfsson, Luca Benini, and Yawei Li. Luna: Efficient and topology-agnostic foundation model for eeg signal analysis. *arXiv preprint arXiv:2510.22257*, 2025.

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

[25] Van-Chien Nguyen, Trung-Hieu Tran, Tuan-Kiet Doan, Quang Hung Pham, Ngoc-Son Vu, Duc Han Le, Huy Phan, Phi Le Nguyen, Nikola Simidjievski, Samuel Tardieu, et al. Balanced latent semantics and signal fidelity for eeg representation learning. In *1st ICLR Workshop on Time Series in the Age of Large Models*, 2026.

[26] Mostafa Neo Mohsenvand, Mohammad Rasool Izadi, and Pattie Maes. Contrastive representation learning for electroencephalogram classification. In *Machine learning for health*, pages 238–253. PMLR, 2020.

[27] Gert Pfurtscheller and FH Lopes Da Silva. Event-related eeg/meg synchronization and desynchronization: basic principles. *Clinical neurophysiology*, 110(11):1842–1857, 1999.

[28] Dennis J McFarland, Laurie A Miner, Theresa M Vaughan, and Jonathan R Wolpaw. Mu and beta rhythm topographies during motor imagery and actual movements. *Brain topography*, 12 (3):177–186, 2000.

[29] Christa Neuper, Michael Wörtz, and Gert Pfurtscheller. Erd/ers patterns reflecting sensorimotor activation and deactivation. *Progress in brain research*, 159:211–222, 2006.

[30] P Achermann and AA Borbély. Low-frequency (< 1 hz) oscillations in the human sleep electroencephalogram. *Neuroscience*, 81(1):213–222, 1997.

[31] Richard B Berry, Rita Brooks, Charlene E Gamaldo, Susan M Harding, Carole Marcus, Bradley V Vaughn, et al. The aasm manual for the scoring of sleep and associated events. *Rules, Terminology and Technical Specifications, Darien, Illinois, American Academy of Sleep Medicine*, 176(2012):7, 2012.

[32] Mircea Steriade. Grouping of brain rhythms in corticothalamic systems. *Neuroscience*, 137(4): 1087–1106, 2006.

[33] Richard J Davidson. Anterior cerebral asymmetry and the nature of emotion. *Brain and cognition*, 20(1):125–151, 1992.

[34] Matthias M Müller, Andreas Keil, Thomas Gruber, and Thomas Elbert. Processing of affective pictures modulates right-hemispheric gamma band eeg activity. *Clinical Neurophysiology*, 110 (11):1913–1920, 1999.

[35] Kai Yang, Li Tong, Jun Shu, Ning Zhuang, Bin Yan, and Ying Zeng. High gamma band eeg closely related to emotion: evidence from functional network. *Frontiers in human neuroscience*, 14:89, 2020.

[36] Yihe Wang, Nadia Mammone, Darina Petrovsky, Alexandros T Tzallas, Francesco C Morabito, and Xiang Zhang. Adformer: A multi-granularity transformer for eeg-based alzheimer's disease assessment. *arXiv preprint arXiv:2409.00032*, 2024.

[37] Yihe Wang, Nan Huang, Taida Li, Yujun Yan, and Xiang Zhang. Medformer: A multi-granularity patching transformer for medical time-series classification. *Advances in Neural Information Processing Systems*, 37:36314–36341, 2024.

[38] 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. *arXiv preprint arXiv:2506.23075*, 2025.

[39] Wei-Bang Jiang, Li-Ming Zhao, and Bao-Liang Lu. Large brain model for learning generic representations with tremendous eeg data in bci. *arXiv preprint arXiv:2405.18765*, 2024.

[40] Guangyu 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, 2024.

[41] 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. *arXiv preprint arXiv:2412.07236*, 2024.

[42] Shiyu Wang, Haixu Wu, Xiaoming Shi, Tengge Hu, Huakun Luo, Lintao Ma, James Y Zhang, and Jun Zhou. Timemixer: Decomposable multiscale mixing for time series forecasting. *arXiv preprint arXiv:2405.14616*, 2024.

[43] Shiyu Wang, Jiawei Li, Xiaoming Shi, Zhou Ye, Baichuan Mo, Wenze Lin, Shengtong Ju, Zhixuan Chu, and Ming Jin. Timemixer++: A general time series pattern machine for universal predictive analysis. *arXiv preprint arXiv:2410.16032*, 2024.

[44] Peng Chen, Yingying Zhang, Yunyao Cheng, Yang Shu, Yihang Wang, Qingsong Wen, Bin Yang, and Chenjuan Guo. Pathformer: Multi-scale transformers with adaptive pathways for time series forecasting. *arXiv preprint arXiv:2402.05956*, 2024.

[45] Yang Mu, Muhammad Shahzad, and Xiao Xiang Zhu. Mptsnet: Integrating multiscale periodic local patterns and global dependencies for multivariate time series classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 19572–19580, 2025.

[46] Huanyao Zhang, Jiaye Lin, Wentao Zhang, Haitao Yuan, and Guoliang Li. Adamixt: Adaptive weighted mixture of multi-scale expert transformers for time series forecasting. *arXiv preprint arXiv:2509.18107*, 2025.

[47] Lu Li, Cunhang Fan, Hongyu Zhang, Jingjing Zhang, Xiaoke Yang, Jian Zhou, and Zhao Lv. Mhanet: Multi-scale hybrid attention network for auditory attention detection. *arXiv preprint arXiv:2505.15364*, 2025.

[48] Zehui Feng, Dian Zhu, Junxuan Li, Yang Bai, and Ting Han. Msanet: Multi-scale semantic aggregation network for brain-assisted speech enhancement in multi-speaker conditions. In *ICASSP 2026-2026 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6846–6850. IEEE, 2026.

[49] Cunhang Fan, Sheng Zhang, Jingjing Zhang, Enrui Liu, Xinhui Li, Gangming Zhao, and Zhao Lv. Dmf2mel: A dynamic multiscale fusion network for eeg-driven mel spectrogram reconstruction. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pages 6977–6985, 2025.

[50] Siyuan Mu and Sen Lin. A comprehensive survey of mixture-of-experts: Algorithms, theory, and applications. *arXiv preprint arXiv:2503.07137*, 2025.

[51] Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*, 2024.

[52] Liuyin Yang, Qiang Sun, Ang Li, and Marc M Van Hulle. Are eeg foundation models worth it? comparative evaluation with traditional decoders in diverse bci tasks. In *The Fourteenth International Conference on Learning Representations*, 2026.

[53] Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538*, 2017.

[54] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[55] 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):34, 2008.

[56] Ary L Goldberger, Luis AN Amaral, Leon Glass, Jeffrey M Hausdorff, Plamen Ch Ivanov, Roger G Mark, Joseph E Mietus, George B Moody, Chung-Kang Peng, and H Eugene Stanley. Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals. *circulation*, 101(23):e215–e220, 2000.

[57] 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.

[58] 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.

[59] 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.

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

[61] Perrin Margaux, Maby Emmanuel, Daligault Sébastien, Bertrand Olivier, and Mattout Jérémie. Objective and subjective evaluation of online error correction during p300-based spelling. *Advances in Human-Computer Interaction*, 2012(1):578295, 2012.

[62] Bob Kemp, Aeilko H Zwinderman, Bert Tuk, Hilbert AC Kamphuisen, and Josefien JL Oberye. Analysis of a sleep-dependent neuronal feedback loop: the slow-wave microcontinuity of the eeg. *IEEE Transactions on Biomedical Engineering*, 47(9):1185–1194, 2000.

[63] Claude Nadeau and Yoshua Bengio. Inference for the generalization error. *Advances in neural information processing systems*, 12, 1999.

[64] Wei-Bang Jiang, Xi Fu, Yi Ding, and Cuntai Guan. Towards robust multimodal physiological foundation models: Handling arbitrary missing modalities. *arXiv preprint arXiv:2504.19596*, 2025.

[65] Cunhang Fan, Ying Chen, Jian Zhou, Zexu Pan, Jingjing Zhang, Youdian Gao, Xiaoke Yang, Zhengqi Wen, and Zhao Lv. M3anet: Multi-scale and multi-modal alignment network for brain-assisted target speaker extraction. *arXiv preprint arXiv:2506.00466*, 2025.

[66] Yamin Li, Ange Lou, Ziyuan Xu, Shengchao Zhang, Shiyu Wang, Dario J Englot, Soheil Kolouri, Daniel Moyer, Roza G Bayrak, and Catie Chang. Neurobolt: Resting-state eeg-to-fmri synthesis with multi-dimensional feature mapping. *Advances in neural information processing systems*, 37:23378–23405, 2024.

[67] Jin Yin, Aiping Liu, Chang Li, Ruobing Qian, and Xun Chen. A gan guided parallel cnn and transformer network for eeg denoising. *IEEE Journal of Biomedical and Health Informatics*, 2023.

Appendix Table of Contents

A Notations **15**

B Additional Pretraining setup **16**

C Additional downstream setup **17**

C.1 Downstream pipeline 17

C.2 Datasets 19

C.3 Baseline model 21

C.4 Evaluation metrics 22

D Additional results **23**

D.1 Sample-level analysis of dynamic routing 23

D.2 Ablation on adding noise during pretraining 23

D.3 Ablation on different timescale levels 25

D.4 Ablation on Aggregator usage during pretraining 25

D.5 Ablation on changing the ratio of alignment loss and reconstruction loss 26

D.6 Ablation on changing the mask ratio 27

D.7 Ablation on using RoPE for channel information 27

D.8 Ablation on increasing the volume of pretraining hours 27

A Notations

Table 4: Table of notations.

Symbol	Space	Meaning
C	$\mathbb{N}$	Number of channels
T	$\mathbb{N}$	Number of timestamps
G	$\mathbb{N}$	Number of patch sizes
p_g	$\mathbb{N}$	Patch size at index g
n_g	$\mathbb{N}$	Number of segments at index g
D	$\mathbb{N}$	Feature dimension
N_{class}	$\mathbb{N}$	Number of target classes for downstream task
$\mathbf{E}$	$\mathbb{R}^{C\times T}$	Input signal
$\hat{\mathbf{E}}$	$\mathbb{R}^{C\times T}$	Reconstructed signal
$\mathbf{E}_{p_g}$	$\mathbb{R}^{C\times n_g\times p_g}$	Signal after patching with patch size p_g
$\mathbf{e}_{p_g}$	$\mathbb{R}^{p_g}$	One patch at patch size p_g
$\mathbf{x}_{p_g}$	$\mathbb{R}^{D}$	One token at patch size p_g
$\mathbf{a}_{\text{temp}}$	$\mathbb{R}^{D_h}$	Aggregator temporal vector
$\mathbf{a}_{\text{spec}}$	$\mathbb{R}^{D_h}$	Aggregator spectral vector
$\mathbf{a}$	$\mathbb{R}^{2D_h}$	Aggregator features vector
α	$\mathbb{R}^{G}$	Deterministic output of aggregator
$\tilde{\alpha}$	$\mathbb{R}^{G}$	Stochastic output of aggregator

Continued on next page...

Table 4 – Continued from previous page

Symbol	Space	Meaning
ϵ	$\mathbb{R}^G$	Gaussian noise vector
σ	$\mathbb{R}$	Standard deviation of Gaussian noise
$\mathbf{X}^{(g)}$	$\mathbb{R}^{C\times n_g\times D}$	Tokens at timescale g
$\mathbf{M}^{(g)}$	$\mathbb{R}^{C\times n_g\times D}$	Masks at timescale g
$\mathbf{X}_c$	$\mathbb{R}^{n_g\times D}$	Token at channel c
$\mathbf{W}_Q^T$	$\mathbb{R}^{D\times D}$	Query projection for temporal attention
$\mathbf{W}_K^T$	$\mathbb{R}^{D\times D}$	Key projection for temporal attention
$\mathbf{W}_V^T$	$\mathbb{R}^{D\times D}$	Value projection for temporal attention
$\mathbf{W}_O^T$	$\mathbb{R}^{D\times D}$	Output projection for temporal attention
$\mathbf{Q}_c$	$\mathbb{R}^{n_g\times D}$	Query matrix for temporal attention
$\mathbf{K}_c$	$\mathbb{R}^{n_g\times D}$	Key matrix for temporal attention
$\mathbf{V}_c$	$\mathbb{R}^{n_g\times D}$	Value matrix for temporal attention
$\mathbf{F}_c$	$\mathbb{R}^{n_g\times D}$	Output for temporal attention at channel c
$\mathbf{F}^{(g)}$	$\mathbb{R}^{C\times n_g\times D}$	Output for temporal attention
$\mathbf{F}_t$	$\mathbb{R}^{C\times D}$	Token at time t
$\mathbf{W}_Q^S$	$\mathbb{R}^{D\times D}$	Query projection for spatial attention
$\mathbf{W}_K^S$	$\mathbb{R}^{D\times D}$	Key projection for spatial attention
$\mathbf{W}_V^S$	$\mathbb{R}^{D\times D}$	Value projection for spatial attention
$\mathbf{W}_O^S$	$\mathbb{R}^{D\times D}$	Output projection for spatial attention
$\mathbf{Q}'_t$	$\mathbb{R}^{C\times D}$	Query matrix for spatial attention
$\mathbf{K}'_t$	$\mathbb{R}^{C\times D}$	Key matrix for spatial attention
$\mathbf{V}'_t$	$\mathbb{R}^{C\times D}$	Value matrix for spatial attention
$\mathbf{U}_t$	$\mathbb{R}^{C\times D}$	Output for spatial attention at time t
$\mathbf{U}^{(g)}$	$\mathbb{R}^{C\times n_g\times D}$	Output for spatial attention
$\mathbf{Z}_S^{(g)}$	$\mathbb{R}^{C\times n_g\times D}$	Student latent representation at timescale g
$\mathbf{H}_T^{(g)}$	$\mathbb{R}^{C\times n_g\times D}$	Teacher target representation at timescale g
$\hat{\mathbf{H}}_S^{(g)}$	$\mathbb{R}^{C\times n_g\times D}$	Student predicted representation at timescale g
$\hat{\mathbf{y}}$	$\mathbb{R}^{N_{\text{class}}}$	Final task prediction vector
$\mathcal{L}_{\text{align}}$	$\mathbb{R}$	Alignment loss
$\mathcal{L}_{\text{recon}}$	$\mathbb{R}$	Signal reconstruction loss
$\mathcal{L}$	$\mathbb{R}$	Total objective function
λ_{recon}	$\mathbb{R}$	Weight hyperparameter for reconstruction loss
λ_{align}	$\mathbb{R}$	Weight hyperparameter for alignment loss
$\mathcal{P}$	Set	Set of patch sizes
$\mathcal{X}_{\mathcal{P}}$	Set	Set of input tokens
$\tilde{\mathcal{X}}_{\mathcal{P}}$	Set	Set of masked input tokens
$\mathcal{Z}_S$	Set	Set of student latent representations
$\mathcal{H}_T$	Set	Set of teacher target representations
$\hat{\mathcal{H}}_S$	Set	Set of student predicted representations

B Additional Pretraining setup

Pretraining dataset. MAGE is pretrained on a subset of 500 hours of TUEG. TUEG is recorded using a standard 10-20 system, with most of the data sampled at 256 Hz. To avoid data leaking, the TUAB dataset is rigorously excluded from the pretraining subset. The TUEG dataset suffers from significant data contamination, including substantial unannotated noise, artifacts, and channel mismatch. To address this, we utilize an efficient preprocessing strategy designed to filter noise from the pre-training data.

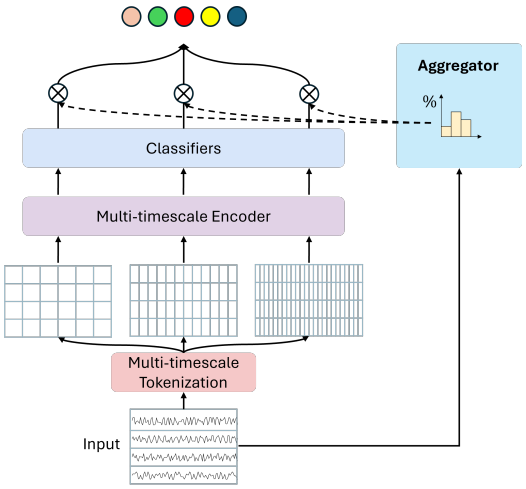


Figure 3: Downstream pipeline of MAGE. Aggregator is adopted to dynamically weight each time scale before adding them together for final prediction.

Preprocessing. Firstly, the beginning and ending of each TUEG recording are contaminated. Therefore, in each recording, the first and last one minute are removed to avoid low-quality data. Secondly, to exclude degraded channels while preserving crucial information, we restricted our dataset to the 19 most standard electrode positions: Fp1, Fp2, F7, F3, Fz, F4, F8, T3, C3, Cz, C4, T4, T5, P3, Pz, P4, T6, O1, and O2. After that, we apply a band-pass filter (0.3 Hz - 75 Hz) to remove noise, along with a notch filter at 60 Hz to discard the power line noise. All recordings are then resampled to 200 Hz, and cut into 30-second windows to create uniformly long-term samples. Raw EEG signals are frequently contaminated by non-brain physiological artifacts, such as ECG, EOG, and EMG [67]. These artifacts can introduce high-amplitude pulses that mimic abnormal brain activity, potentially misleading the model. We filtered out any samples with voltage amplitudes exceeding 100 μV because such high-magnitude peaks deviate from standard physiological brain activity and typically represent non-cerebral artifacts. Finally, we normalized each sample by z-score normalization per channel to standardize the data.

Pretraining settings. Our experiments are implemented with Pytorch 2.1.2 and CUDA 11.8. During each training iteration, we select a set of patch sizes: 50, 100, and 200 (representing 250ms, 500ms, and 1s) to encourage the encoder to learn robust representations across multiple timescales. 70% of patches are randomly masked during the pretraining phase. We adopt the AdamW optimizer with a learning rate $1e-4$, batch size 64, and 50 training epochs. Additional hyperparameters for pretraining MAGE are described in Tab. 5. The pretraining process for 500 hours of EEG data in MAGE requires approximately 8 hours to complete.

C Additional downstream setup

C.1 Downstream pipeline

Figure 3 illustrates the adaptation of MAGE for downstream tasks. When downstreaming, only the encoder is retained for evaluation, while the predictor and decoder are discarded. The input signal first passes through a multi-timescale module, which tokenizes the signal into patches of varying sizes. These tokens are processed by the Multi-timescale Encoder to capture information across both timescales and spatial dimensions. An aggregator then determines an importance score for each scale based on temporal and spectral analysis. Similar to the pre-training phase, stochastic noise is introduced during downstream adaptation to prevent the model from over-relying on any single timescale.

Following the evaluation protocol established by EEGPT, a convolutional layer is employed to project the variable number of input channels from the downstream dataset into a fixed 19-channel space. We then freeze the parameters of the pre-trained encoder and train the tokenization module and the classifiers. This strategy mitigates the risk of overfitting on smaller downstream datasets and enables

Table 5: Hyperparameters for MAGE pre-training on a single NVIDIA A100 40GB.

Component	**Hyperparameters**	**Settings**
EEG sample	Channels	19
	Time points	6000 (30s @ 200Hz)
	Patch sizes	50 100 200 (250ms 500ms 1s)
	Mask ratio	0.7
	Mask token	Constant
Encoder	Layers (depth)	6
	Hidden dimension	64
	Attention heads	4
	MLP ratio	4.0
	qkv bias	True
	Init std	0.02
	Norm ϵ	1×10^{-6}
	Attention pattern	Multi-timescale Attention
Predictor	Depth	3
	Attention	Multi-timescale Attention
Aggregator	Architecture	Dual-Stream (Temporal + Frequency)
	Temporal Dim	32
	Frequency Dim	16
	Temporal Extractor	$2 \times$ Conv1D($k = \{7, 5\}, s = 4$), BatchNorm, GELU, AdaptiveAvgPool
	Magnitude Extractor	2×Conv1D ($k = \{5, 3\}, s = 2$), BN, GELU, AdaptiveAvgPool
	Phase Extractor	1×Conv1D ($k = 5, s = 2$), BN, GELU, AdaptiveAvgPool
	Fusion	Cross-domain Linear Projection
Reconstructor	Head	Linear (hidden dimension, patch dimension)
Momentum Encoder	EMA momentum	Cosine ramp $0.996 \rightarrow 0.9995$
Pre-training	Epochs	50
	Batch size	64 (global)
	Optimizer	AdamW
	Learning rate	1×10^{-4}
	LR schedule	Warmup + Cosine decay
	Warmup steps	$\max(500, 0.1T)$
	Weight decay	5×10^{-2} (cosine schedule)
	Adam β	(0.9, 0.999)
	Adam ϵ	1×10^{-8}
	Gradient clipping	1.0
	Loss weights	$\lambda_{\text{rec}} = 5.0$, $\lambda_{\text{align}} = 1.0$
	Num worker	32

a direct evaluation of the quality of the representations learned during pre-training. For each timescale g, the classification head $\text{Classifier}^{(g)}$ consists of a projection layer followed by a linear classifier. The tokens after the encoder are first flattened across their channel and embedding dimensions. They are subsequently projected to a lower dimension with dropout and max-norm constraints, then concatenated across time steps before passing through the final linear layer. Crucially, this same evaluation head is applied across all baseline models to ensure a fair comparison.

The final task prediction $\hat{\mathbf{y}}$ of MAGE is computed by aggregating these representations via a Classifier:

Table 6: Overview of evaluation EEG datasets.

Dataset	Rate	Number of Channels	Duration	Labels
BCIC-IV-2A	250 Hz	22	4s	4-class
PhysioNet-MI	160 Hz	64	4s	4-class
FACED	250 Hz	32	10s	9-class
Sleep-EDFx	100 Hz	2	30s	5-class
TUAB	250 Hz	16	10s	2-class
KaggleERN	200 Hz	56	2s	2-class
SHU-MI	250 Hz	32	4s	2-class

$$\hat{\mathbf{y}} = \sum_{g=1}^{G} \tilde{\alpha}_g \cdot \text{Classifier}^{(g)} \left(\mathbf{Z}_S^{(g)} \right), \tag{11}$$

where $\text{Classifier}^{(g)}$ denotes the task-specific classification head for timescale g, and $\tilde{\boldsymbol{\alpha}}$ represents the stochastic importance vector derived from the aggregator. Notably, this stochasticity is disabled during evaluation to ensure deterministic predictions.

C.2 Datasets

When evaluating the EEG model, performance is strongly affected by the partition of the subject, which makes the result remarkably different between different selections. Relying on a fixed division of training, validation and testing sets prevents the foundation models from revealing their actual potential. Therefore, we adopt a subject-wise cross-validation evaluation scheme, in which, for each dataset, we evaluate each dataset N folds with different partitions of train, val, and test sets. The reported performance is the average across all folds. Detailed information about the train-validation-test split is illustrated in each dataset below.

C.2.1 BCIC-IV-2a

License: The dataset is sourced from the BNCI Horizon 2020 repository and is utilized under the Creative Commons Attribution No Derivatives (CC BY-ND 4.0) license.

Dataset Overview and Preprocessing. BCIC-IV-2a consists of recordings from nine subjects, each subject performs four different motor imagery tasks: left hand, right hand, feet, and tongue. Each subject conducted two different sessions on different days. Each session contains 288 trials. The data is preprocessed with a 0-38Hz band-pass filter and resampled to 200Hz. Finally, for each 4-second trial, the z-score normalization per channel is applied to standardize the data.

Evaluation. We utilize a leave-one-subject-out cross-validation method. We evaluate each model nine times, with each subject being a test subject each time.

C.2.2 PhysioNet-MI

License: The dataset is made available by PhysioNet under the Open Data Commons Attribution License (ODC-By 1.0).

Dataset Overview and Preprocessing. PhysioNet-MI contains motor imagery recordings from 109 subjects. The task includes four different classes: left fist, right fist, both fists, and both feet. The data is preprocessed with a 0.3Hz cutoff frequency high-pass filter, resampled to 200Hz, and segmented to 4-second samples. We then apply z-score normalization per channel to each sample to standardize the data.

Evaluation. We consider the evaluation set of PhysioNet-MI as the test set. For its train set, we split it into five folds with different evaluation set each fold. The models will then be evaluated five times with different train and evaluation sets.

C.2.3 FACED

License: The dataset is published in Scientific Data and is used under the Creative Commons Attribution 4.0 International (CC BY 4.0) license.

Dataset Overview and Preprocessing. The FACED dataset comprises 32-channel EEG recordings from 123 subjects watching 28 video clips. The stimuli elicit nine fine-grained emotion classes, including amusement, fear, and neutral. We resample each trial to 6,000 points and segment it into three non-overlapping windows, resulting in samples of 2,000 data points (32×2000). Finally, we apply z-score normalization per channel to standardize the data.

Evaluation. We perform subject-independent 4-fold cross-validation. In each fold, one partition is held out for testing. The remaining subjects are further split into training and validation sets to monitor performance during training.

C.2.4 Sleep-EDFx

License: The dataset is obtained from the PhysioNet repository and is used under the Open Data Commons Attribution License (ODC-By) v1.0.

Dataset Overview and Preprocessing. The Sleep-EDFx dataset comprises sleep recordings from 78 healthy subjects, categorized into five distinct sleep stages: Wake (W), N1, N2, N3, and REM. We apply a 30 Hz low-pass filter, resample the data to 200 Hz, and segment it into 30-second samples. After that, to standardize the signal, the z-score normalization per channel is applied.

Evaluation. We employ a subject-wise 5-fold cross-validation strategy. The subjects are equally divided into five folds. In each test, one fold serves as the test set, while the remaining four folds are used for training and validation. These training subjects are further split with a 9:1 ratio for training and validation.

C.2.5 TUAB

License: The dataset is provided by the Neural Engineering Data Consortium (NEDC) and is used under the NEDC Community Resource Free Academic Use License.

Dataset Overview and Preprocessing. The TUAB dataset contains 409,455 10-second EEG samples, labeled for binary classification as either normal or abnormal. We apply a 0.1–75 Hz band-pass filter and a 50 Hz notch filter to remove line noise. The data is resampled to 200 Hz and applied a z-score normalization per channel to standardize the signal.

Evaluation. We utilize the official evaluation set provided by TUAB as our held-out test set. For training and validation, we adopt a 4-fold cross-validation protocol on the remaining subjects. In each run, one fold serves as the validation set while the other three form the training set, resulting in an overall train-validation-test ratio of approximately 6:2:2.

C.2.6 KaggleERN

License: The dataset was originally released for the BCI Challenge @ NER 2015 and is used in accordance with the Kaggle Competition Rules for academic and non-commercial research.

Dataset Overview and Preprocessing. KaggleERN captures error-related potentials (ErrP) from subjects using a P300 speller. This is a binary classification task indicating whether the system's feedback matched the user's intent (Correct vs. Erroneous). We resample the signals to 200 Hz and extract 2-second windows per sample. In each sample, we apply the z-score normalization to standardize the signal.

Evaluation. We utilize the Kaggle evaluation set as the test set. For the training data, we employ a subject-wise 5-fold cross-validation method, where one fold serves as a validation set and the rest as a training set.

C.2.7 SHU-MI

License: The dataset is used under the Creative Commons Attribution 4.0 International (CC BY 4.0) license.

Dataset Overview and Preprocessing. The SHU-MI dataset contains EEG recordings from 55 subjects performing two distinct motor imagery tasks: left-hand and right-hand movement. We resample each trial to 800 data points (corresponding to a 4-second window at 200 Hz). We then apply z-score normalization per channel to standardize the signal distribution.

Evaluation. We employ a subject-independent 5-fold cross-validation strategy. In each iteration, 20% of the subjects are reserved for testing. The remaining 80% of subjects are further partitioned into a training and validation set with a ratio of 9:1.

C.3 Baseline model

Table 7: The sampling rate and number of parameters of each model.

Models	Training Paradigm	Sampling rate (Hz)	Params. (M)
Medformer	supervised	200	1.4
LaBraM	self-supervised	200	5.8
BIOT	self-supervised	200	3.2
EEGPT	self-supervised	256	25.0
CBraMod	self-supervised	200	4.0
CSBrain	self-supervised	200	8.9
MAGE	self-supervised	200	0.6

C.3.1 Medformer

Medformer is a multi-timescale patching transformer model specialized for medical time series data. Medformer learns complex temporal and spatial dependencies by segmenting the input into various patch sizes via cross-channel patching. This multi-scale approach effectively simulates different frequency bands and captures both fine-grained local patterns and coarser global trends. To aggregate this information, the model adopts a two-stage transformer layer: it first applies intra-scale attention to capture features within each scale, and then utilizes inter-scale attention to fuse information across the different patch sizes.

In our experiments, we adopt Medformer with three scales: 250ms, 500ms, 1s, similar to MAGE, and the feature size of the model is 64, resulting in a total of 1.4 M parameters.

C.3.2 LaBraM

LaBraM is a large brain foundation model for the EEG signal. It learns EEG generic representations by predicting the discrete neural tokens of masked EEG patches based on a full-attention transformer. LaBraM adopts a novel vector-quantized neural spectrum prediction (VQ-NSP) framework. It first applies short-time Fourier transforms (STFT) to transform the signal into frequency-domain spectrograms and discretizes this signal into neural codes. LaBraM is pretrained on 2500 hours of EEG data with masked reconstruction tasks.

LaBraM has three variants: LaBraM-Base, LaBraM-Large, LaBraM-Huge, with the number of parameters being 5.8M, 46M, and 369M, respectively. Since only the pretrained checkpoint for LaBraM-Base is available, we adopt this variant for our experiments. To ensure a fair comparison with other models, in downstream evaluation, we freeze all encoder layers of LaBraM, while still fine-tuning the patch embedding layer initially to help it adapt to a new model's data domain. Furthermore, all data are resampled to 200 Hz when training with LaBraM to match the pretraining data.

C.3.3 BIOT

BIOT is a pretrained foundation model for EEG signal. It proposes a novel biosignal tokenization module that segments each channel separately into fixed-length tokens and flattens them to form unified "biosignal sentences". BIOT is pre-trained on multiple large-scale datasets with up to 10 million EEG samples in total.

In our experiments, we adopt the BIOT version with 3.2M parameters and the available pretrained checkpoint from 6 datasets with 18 EEG channels. A convolution layer is applied to the input into 18 channels. Similar to other models, we freeze the encoder of BIOT while unfreezing other modules during downstreaming to evaluate the encoded representation of the model. The trained modules include channel convolution, patch embedding, channel tokens, and a classifier head. All data are resampled to 200 Hz to train with BIOT to match the pretraining data.

C.3.4 EEGPT

EEGPT is a pretrained transformer model for universal EEG feature extraction. It proposes a dual self-supervised learning method with learning by both alignment task and reconstruction task during pretraining. While the reconstruction task helps them to learn detailed features, the alignment task support learn rich semantic information. EEGPT is trained on multiple open-source datasets with more than 200 subjects in total.

EEGPT adopts two variants: EEGPT and EEGPT-Tiny, with the number of parameters is 25M and 4.7M, respectively. In our experiments, we adopt the EEGPT variant with available pretrained checkpoint. Similar to other models, we freeze the encoder of EEGPT while unfreezing other modules during downstreaming to evaluate the encoded representation of the model. The trained modules include channel convolution, patch embeddingto , and a classifier head. Notably, while EEGPT is trained on 256Hz data, all other models are trained on 200Hz data. We therefore preprocess to resample all datasets to 256 Hz when evaluating EEGPT.

C.3.5 CBraMod

CBraMod is a robust EEG foundation model designed to tackle the structural heterogeneity of brain signals across different datasets. It addresses the full EEG modeling of prior foundation models by proposing a criss-cross attention. In this attention mechanism, CBraMod models spatial and temporal dependencies separately by two parallel attention modules. By doing this, CBraMod can avoid attending information from irrelevant tokens. CBraMod is pretrained by the masked reconstruction task on 9000 hours from the TUEG dataset.

In our experiments, we adopt CBraMod with 4.0M parameters and an available pretrained checkpoint. Similar to other models, we freeze the encoder of CBraMod while finetuning other modules during downstream evaluation to evaluate the encoded representation of the model. The trained modules include patch embedding and a classifier head. All inputs were resampled to 200 Hz to match CBraMod's pre-training specifications.

C.3.6 CSBrain

CSBrain is a foundation model designed to advance general-purpose EEG decoding by capturing the multi-scale characteristics of brain signals. CSBrain introduces a cross-scale spatial-temporal transformer to model the complex spatial correlations between brain regions and dynamic temporal characteristics. CSBrain is pretrained by a masked reconstruction task on 9000 hours of data from the TUEG dataset.

In our experiments, we utilize the published pretrained checkpoint of CSBrain with 8.9M parameters. We freeze the encoder of CSBrain and fine-tune other modules to evaluate the representation from the encoder. The trained modules include patch embedding and a classifier head. All data were resampled to 200 Hz to align with the pre-training sampling rate of CSBrain.

C.4 Evaluation metrics

This section introduces all metrics we employed to evaluate the model's performance.

Balanced Accuracy is defined as the mean of recall for each class in the dataset. Balanced accuracy ensures that the model's performance is not biased towards the majority class, which is a common issue in imbalanced datasets.

Cohen's Kappa is a statistical metric used to measure the level of agreement between two classifiers. It accounts for the agreement occurring by chance. In the context of the experiments, one classifier is the true label of the sample, and the other is the predicted label. It is used for multiclass classification problems.

Weighted F1 score is the average of the F1-scores for all classes, where each class's score is weighted by its number of true instances. It is used for multiclass classification problems.

AUROC (Area Under the Receiver Operating Characteristic Curve) measures the trade-off between the true positive rate and the false positive rate across all possible classification thresholds. It is used for binary classification problems.

AUC-PR (Area Under the Precision-Recall Curve) measures the trade-off between precision and recall across different thresholds. It is used for binary classification problems.

D Additional results

D.1 Sample-level analysis of dynamic routing

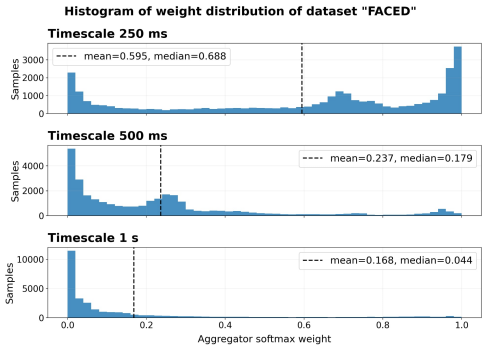
In this section, we will discuss how the aggregator vary at the individual sample level within a single task. These results confirm that the aggregator dynamically adapts to individual signal characteristics rather than relying on a fixed per-task bias. The significant variance observed within a single fold proves that the routing weights cannot be predicted by dataset identity alone. If the aggregator were simply a task-level lookup, intra-task variance would be near zero. Instead, this dynamic adjustment to individual inputs confirms that MAGE performs routing based on signal content, rather than learning a fixed per-task bias (Figure 4).

D.2 Ablation on adding noise during pretraining

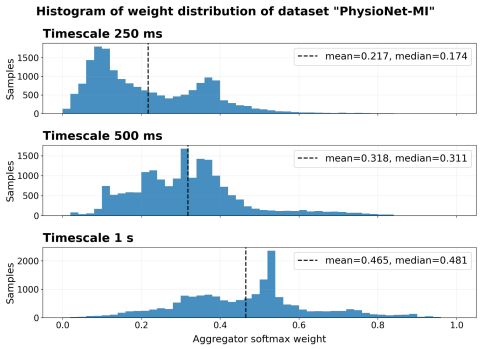
In this ablation, we study the effectiveness of adding noise during pretraining. Without noise, the model can collapse to using only one timescale, leaving the other unexplored. This typically occurs because an expert selected early on (due to random initialization) minimizes the loss faster than others. As shown in Table 8, incorporating noise during the pretraining phase prevents this collapse and leads to better results. Specifically, this noise injection improves performance in three of the four evaluated datasets: FACED, Sleep-EDFx, and SHU-MI.

Table 8: Effectiveness of Noise Injection on Preventing Granularity Collapse. MAGE pretrained with noise achieves a better performance. **Bold** indicates the best result. The result is reported on 1 seed.

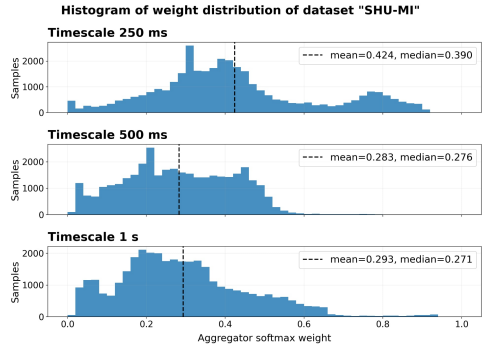
Dataset	Metrics	MAGE	MAGE wo noise
FACED	Balanced Acc	**51.30**$_{\pm 3.11}$	50.35$_{\pm 2.62}$
	Cohen's Kappa	**44.89**$_{\pm 3.43}$	43.80$_{\pm 3.02}$
	Weighted F1	**50.97**$_{\pm 2.96}$	49.84$_{\pm 2.80}$
PhysioNet-MI	Balanced Acc	61.15$_{\pm 0.86}$	**61.57**$_{\pm 1.15}$
	Cohen's Kappa	48.23$_{\pm 1.15}$	**48.78**$_{\pm 1.51}$
	Weighted F1	61.16$_{\pm 0.90}$	**61.66**$_{\pm 1.14}$
Sleep-EDFx	Balanced Acc	**70.24**$_{\pm 4.63}$	68.79$_{\pm 4.24}$
	Cohen's Kappa	**68.91**$_{\pm 6.47}$	66.01$_{\pm 7.53}$
	Weighted F1	**77.31**$_{\pm 5.00}$	75.59$_{\pm 5.72}$
SHU-MI	Balanced Acc	**62.43**$_{\pm 3.18}$	62.24$_{\pm 4.24}$
	AUC-PR	**68.73**$_{\pm 5.03}$	68.54$_{\pm 6.17}$
	AUROC	68.14$_{\pm 5.06}$	**68.45**$_{\pm 5.53}$



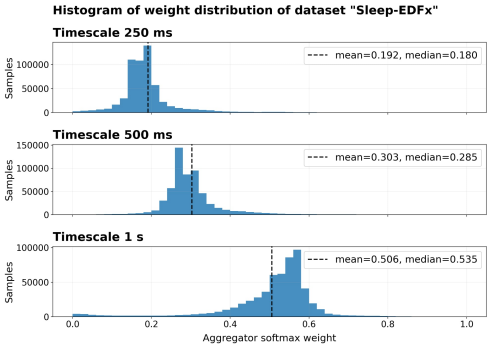
(a) FACED



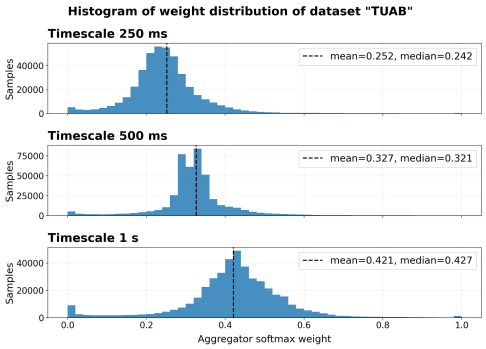
(b) PhysioNet-MI



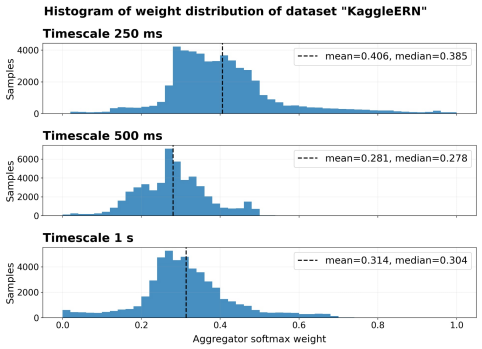
(c) SHU-MI



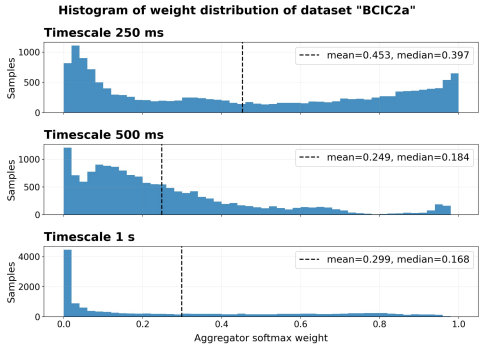
(d) Sleep-EDFx



(e) TUAB



(f) KaggleERN



(g) BCIC2a

Figure 4: Histogram of weight distribution of different datasets.

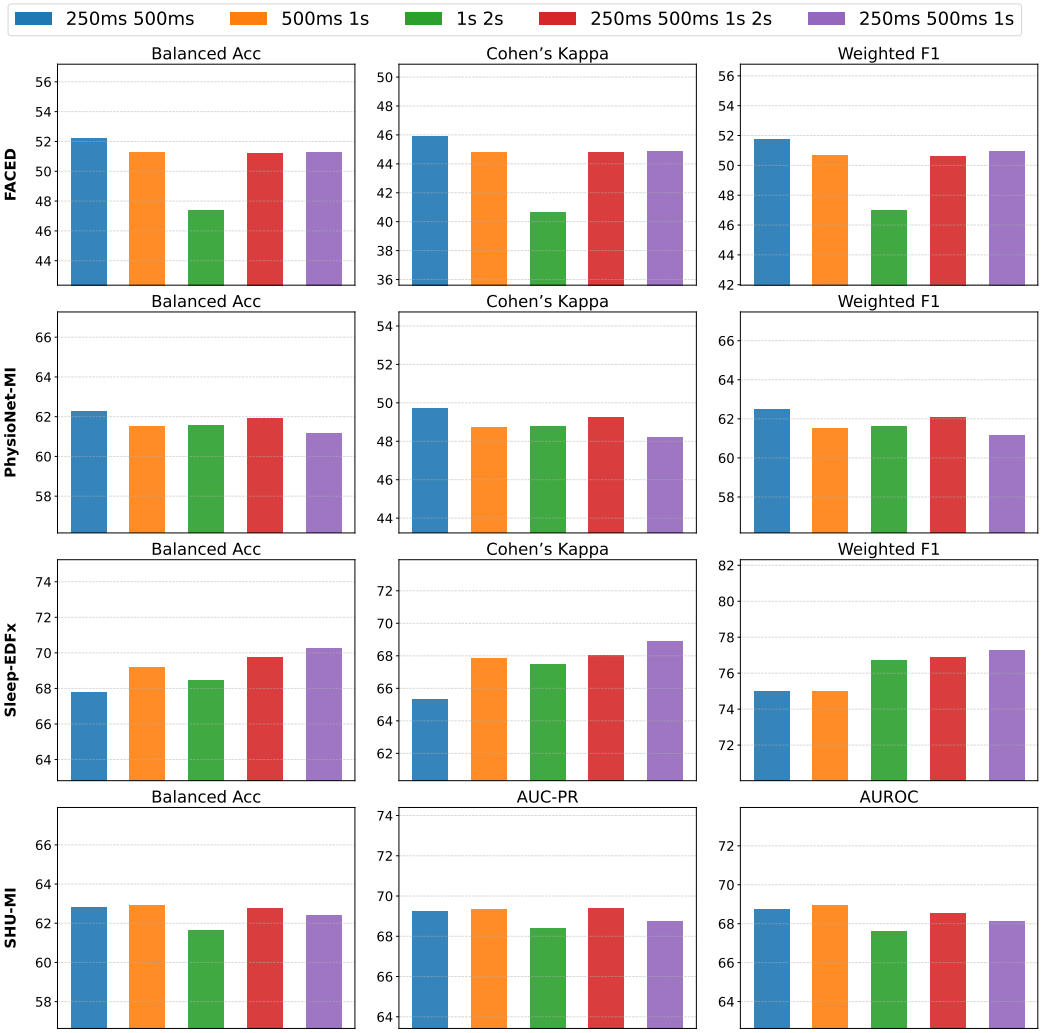


Figure 5: Impact of multi-timescale patch configurations on representation performance across different datasets. Modifying the patch size configurations results in varying performance levels across multiple datasets. The result is reported on 1 seed.

D.3 Ablation on different timescale levels

To investigate the impact of timescale on representation performance, we evaluated MAGE variants with patch configurations of [250ms, 500ms], [500ms, 1s], [1s, 2s], and [250ms, 500ms, 1s, 2s]. As illustrated in Fig. 5, increasing the diversity of timescales results in a more generalizable representation, ensuring robust performance across all metrics and datasets. Notably, different patch sizes possess distinct inductive biases: while the [250ms, 500ms] configuration excels on FACED due to its reliance on short temporal windows, its performance on Sleep-EDFx is significantly degraded. Conversely, the coarser [1s, 2s] configuration achieves better performance on Sleep-EDFx than the [250ms, 500ms] configuration, but fails to generalize effectively to other datasets. In contrast, when employing configurations with three or four timescales, the performance remains consistently strong across all tasks, despite minor fluctuations on individual datasets. This demonstrates that a [250ms, 500ms, 1s] configuration is already sufficiently robust to achieve high generalizability.

D.4 Ablation on Aggregator usage during pretraining

This ablation investigates whether the Aggregator should be optimized during pretraining or introduced only during downstream adaptation. In the standard MAGE setting, the Aggregator is trained

Table 9: MAGE variant where the Aggregator is used only during downstream adaptation, while pretraining is performed with uniform training. The result is reported on 1 seed.

Dataset	Metrics	MAGE	MAGE-uniform pretrain
PhysioNet-MI	Balanced Acc	$61.15_{\pm 0.86}$	$61.15_{\pm 0.27}$
	Cohen's Kappa	$48.23_{\pm 1.15}$	$48.23_{\pm 0.37}$
	Weighted F1	$61.16_{\pm 0.90}$	$61.31_{\pm 0.25}$
FACED	Balanced Acc	$51.30_{\pm 3.11}$	$46.92_{\pm 2.13}$
	Cohen's Kappa	$44.89_{\pm 3.43}$	$40.19_{\pm 2.39}$
	Weighted F1	$50.97_{\pm 2.96}$	$46.88_{\pm 2.11}$
Sleep-EDFx	Balanced Acc	$70.24_{\pm 4.63}$	$63.07_{\pm 4.84}$
	Cohen's Kappa	$68.91_{\pm 6.47}$	$63.49_{\pm 7.62}$
	Weighted F1	$77.31_{\pm 5.00}$	$72.61_{\pm 5.70}$
SHU-MI	Balanced Acc	$62.43_{\pm 3.18}$	$63.49_{\pm 4.19}$
	AUC-PR	$68.73_{\pm 5.03}$	$69.68_{\pm 5.84}$
	AUROC	$68.14_{\pm 5.06}$	$69.41_{\pm 5.63}$

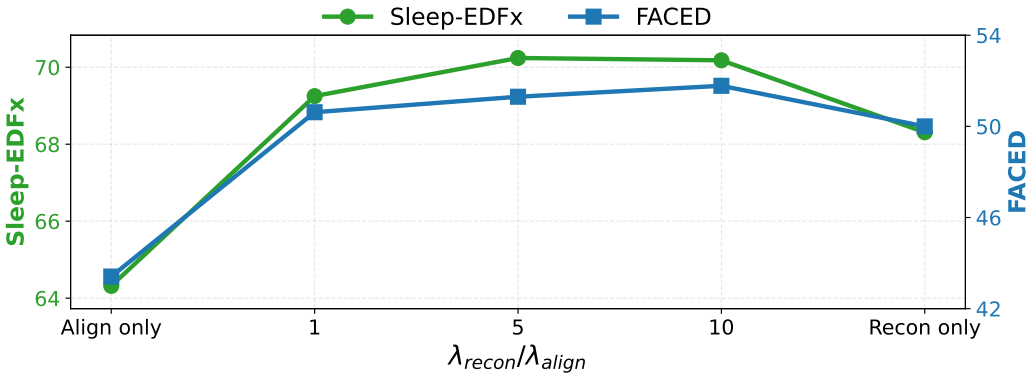


Figure 6: Balance accuracy of MAGE on two datasets when changing the ratio of $\lambda_{\text{recon}}/\lambda_{\text{align}}$. Using both types of loss achieves better performance than a single loss. The result is reported on 1 seed.

together with the multi-scale encoder during pretraining, allowing the model to learn task-agnostic scale fusion from the pretraining data. In contrast, the ablated variant performs uniform pretraining across all scales and initializes the Aggregator only during downstream fine-tuning.

As shown in Table 9, using the Aggregator during pretraining leads to clear improvements on FACED and Sleep-EDFx. These results suggest that optimizing the Aggregator during pretraining helps MAGE learn more transferable multi-scale representations, especially for downstream tasks involving heterogeneous temporal and frequency patterns, such as emotion recognition and sleep staging.

D.5 Ablation on changing the ratio of alignment loss and reconstruction loss

In this section, we investigate the individual contributions of the two objective functions used during the pre-training phase: alignment loss and reconstruction loss. Specifically, we conduct ablation studies by pre-training the model using only the alignment loss or only the reconstruction loss. Furthermore, we explore the effect of varying the corresponding hyperparameters, λ_{recon} and λ_{align}, on the overall pre-training performance. The balanced accuracy for different ratios of these values on Sleep-EDFx and FACED is illustrated in Fig. 6. The results demonstrate that combining both loss functions consistently yields superior performance compared to relying on a single loss type. Crucially, optimizing with only the alignment loss leads to poor results, dropping to approximately 64% balanced accuracy on Sleep-EDFx and 42% on FACED. Furthermore, when both losses are utilized simultaneously, the performance differences across the various weight ratios remain relatively marginal.

Table 10: Balanced accuracy on FACED of different pretraining masking ratio. The overall differences are marginal across different masking ratios. The result is reported on 1 seed.

Dataset	Metrics	Mask 50%	Mask 60%	Mask 70%	Mask 80%
FACED	Balanced Acc	$50.83_{\pm 3.84}$	$51.22_{\pm 3.44}$	$51.30_{\pm 3.11}$	$51.30_{\pm 2.21}$
	Cohen's Kappa	$44.31_{\pm 4.24}$	$44.83_{\pm 3.77}$	$44.89_{\pm 3.43}$	$44.80_{\pm 2.49}$
	Weighted F1	$50.41_{\pm 3.76}$	$50.79_{\pm 3.24}$	$50.97_{\pm 2.96}$	$50.61_{\pm 2.20}$

Table 11: Comparison of using learnable channel embedding versus RoPE for channel positional embedding. The result is reported on 1 seed.

Dataset	Metrics	Learnable Channel Embedding	RoPE
FACED	Balanced Acc	$51.30_{\pm 3.11}$	$50.99_{\pm 3.00}$
	Cohen's Kappa	$44.89_{\pm 3.43}$	$44.43_{\pm 3.26}$
	Weighted F1	$50.97_{\pm 2.96}$	$50.34_{\pm 2.85}$

D.6 Ablation on changing the mask ratio

We run a small ablation study to verify the choice of the masking ratio. As shown in Table 10, although downstream performance fluctuates slightly across different masking ratios, the overall differences are marginal, indicating that MAGE is highly robust to this hyperparameter. Consequently, we adopted a 70% masking ratio as our default setting, consistent with some prior research.

D.7 Ablation on using RoPE for channel information

Unlike temporal data, EEG channels lack a natural linear order, making sequential bias from RoPE inappropriate. Instead, we employ trainable positional tokens (following BIOT and EEGPT) to enable the model to learn each electrode's unique neurophysiological role without assuming an artificial topology. Our small ablation study on FACED (Table 11) confirms this hypothesis: learnable tokens is consistently better than RoPE across all metrics, validating that specific channel identities are more informative than arbitrary sequences.

D.8 Ablation on increasing the volume of pretraining hours

In this section, we evaluate the impact of scaling up MAGE's pre-training data volume. We increased the dataset size fivefold to 2,500 hours, but this expansion yielded only minor improvements on certain tasks as shown in Table. 12. We believe that scaling the dataset further does not justify the

Table 12: MAGE when pretrained on different training hours of EEG signal. The result is reported on 1 seed.

Dataset	Metrics	MAGE-500h	MAGE-2500h
PhysioNet-MI	Balanced Acc	$61.15_{\pm 0.86}$	$\mathbf{62.57}_{\pm 0.73}$
	Cohen's Kappa	$48.23_{\pm 1.15}$	$\mathbf{50.12}_{\pm 0.98}$
	Weighted F1	$61.16_{\pm 0.90}$	$\mathbf{62.62}_{\pm 0.75}$
FACED	Balanced Acc	$51.30_{\pm 3.11}$	$\mathbf{52.02}_{\pm 2.22}$
	Cohen's Kappa	$44.89_{\pm 3.43}$	$\mathbf{45.68}_{\pm 2.40}$
	Weighted F1	$50.97_{\pm 2.96}$	$\mathbf{51.62}_{\pm 1.94}$
Sleep-EDFx	Balanced Acc	$\mathbf{70.24}_{\pm 4.63}$	$68.03_{\pm 4.20}$
	Cohen's Kappa	$\mathbf{68.91}_{\pm 6.47}$	$64.78_{\pm 7.56}$
	Weighted F1	$\mathbf{77.31}_{\pm 5.00}$	$75.13_{\pm 5.48}$
SHU-MI	Balanced Acc	$62.43_{\pm 3.18}$	$\mathbf{63.56}_{\pm 4.11}$
	AUC-PR	$68.73_{\pm 5.03}$	$\mathbf{68.99}_{\pm 6.68}$
	AUROC	$68.14_{\pm 5.06}$	$\mathbf{69.10}_{\pm 5.80}$

massive increase in computational cost. This suggests that raw data volume is not the sole determinant of success; rather, the model's structural inductive bias is equally, if not more, critical. We, therefore, decide to pretrain on our subset of 500 hours for better trade-off between efficiency and performance.

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 primary contribution of this paper is the adoption of a multi-scale architecture for EEG signal analysis, together with an aggregator that adaptively weights timescale importance for various downstream datasets. This contribution is accurately reflected in the abstract and introduction.

   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: The Limitation is discussed in Section 4 of the paper.

   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 provide any new theory assumptions or proof, it focuses on the architecture to enhance the generalizability of EEG foundation models.

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: All the experimental results can be reproduced by the information we provide in Section 4 and the Appendix. We additionally release the code if the paper is accepted.

   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: We provide open access to our code, along with comprehensive instructions to reproduce our results. However, because our evaluation relies on open-source medical datasets containing sensitive health information, we are unable to distribute the raw data directly. Researchers wishing to use these datasets must independently apply for access through the original dataset providers.

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: We specify all training and test details in the Appendix, together with the ratio for splitting and the number of folds for each downstream dataset.

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: We include error bars for our reported experiments. Because EEG signals inherently exhibit high inter-subject variability, our resulting error margins are correspondingly wide.

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: We provide the information about the number of workers, as well as the GPU types we use during the experiments.

   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 research conducted conforms to the NeurIPS Code of Ethics. We use only established, open-source EEG datasets and strictly adhere to their respective licensing and access agreements by not redistributing sensitive medical data. Furthermore, we ensure transparency and reproducibility by providing open access to our code and openly discussing the limitations of our study.

   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: Although this work represents an early, foundational step toward generalizable EEG modeling, we explicitly discuss its potential societal impacts in the future. We highlight the positive impacts of advancing clinical healthcare and BCI applications, as well as potential negative impacts, such as privacy concerns and the risks associated with the deployment of these models in real-world settings.

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: Our model is a lightweight (0.6M parameter) feature extractor for EEG signals, rather than a generative model. Therefore, the paper does not describe safeguards.

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: We cite the original creators of all baseline models and open-source EEG datasets evaluated in our work. Furthermore, we strictly respect the licenses and terms of use of these medical datasets.

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: We will make the codebase open access when the paper is accepted. We additionally provide a README, which includes step-by-step instructions on how to set up the environment, run the models, and fully reproduce the experimental results.

    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 paper does not involve any new crowdsourcing experiments or primary data collection with human subjects.

    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: We do not conduct experiments on new study participants.

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: LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigor, or originality of the research.

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.