EEG Foundation Model for Heterogeneous Sensor Layouts Trained with Task-Switch Contrastive Learning

Leopold Ehrlich
Queen's University
leopold.ehrlich@email.com

Sebastian Medrea
Queen's University
20sm131@queensu.ca

Andrew Crix
Queen's University
d.crix@queensu.ca

Jinpeng Deng
Queen's University
22ss117@email.com

***Abstract*—This paper presents a general-purpose foundation model for electroencephalography data using self-supervised learning. A family of techniques grounded in neuroscience research is used to identify and label cognitively distinct segments of brain activity across large, unlabeled EEG datasets. These pseudo-labels are used to train a contrastive learning objective alongside a graph neural network encoder that, unlike existing approaches, handles arbitrary electrode configurations without discarding any channels. The resulting foundation model can serve as a starting point for downstream tasks, reducing the amount of labeled data required to build accurate, task-specific EEG models. Evaluated on a motor imagery benchmark, our approach produces statistically significant improvements over the baseline while allowing for flexible electrode configurations. This has significant implications for personalized clinical diagnostics and adaptive brain-computer interfaces.**

I. Introduction

Electroencephalography (EEG) is a non-invasive neuroimaging technique that measures the electrical potentials generated by clusters of cortical neurons, recorded via electrodes placed on the scalp [1]. EEG data is a valuable diagnostic tool and has many applications in adaptive Brain-Computer Interface (BCI) technology due to its noninvasiveness, low cost, and portability. These oscillations produced in the imaging are characterized into known frequencies: delta (0.5-4 Hz) is prevalent in deep sleep and disrupted by disorders that affect consciousness; theta (4-8 Hz) is associated with working memory and attention; alpha (8-13 Hz) reflects idling; beta (13-30 Hz) is linked to motor planning and active cognitive processing; and gamma (30-100 Hz) is relevant in memory consolidation [2]. The shifting of these spectral bands can give key information about cognitive states and has the potential to assist with diagnosis. For example, disorders such as Alzheimer's disease are typically associated with a reduction in alpha frequencies and a slowing of theta frequencies [3]. These EEG biomarkers are increasingly studied in the context of depression, autism spectrum disorder, and attention-deficit hyperactivity disorder, where they may offer quantitative data to complement typical behavioural assessments [4]. Furthermore, EEG can be used in individuals with physical impairments, allowing the control of computers and adaptive devices for communication and mobility, greatly improving autonomy [5].

Current challenges in applying EEG to these problems center around data interpretation, where training an effective model for patient diagnosis becomes problematic due to the high cost of acquiring effective new labeled data and how specific data can be to the exact device. The variability of the signals presented from EEG data also depends on tasks performed, electrode placements, skull thickness, differences in neural anatomy, and different EEG hardware configurations, which can range from 4 to 256 electrode channels, presenting even more complications in standardizing the data [6]. The creation of models that can accurately interpret this is challenging, and often requires specialized trials requiring new setups for every task. The large amounts of data necessary for the creation of robust models can be costly, which limits progress.

In the context of adaptive technology, highly advanced EEG devices with many channels can be cost prohibitive and bulky. As such, wearable EEGs often have few channels, which can be located in positions other than those in a clinical device. For instance, configurations where electrodes are located in earpieces or headbands are very common in commercially available devices, compared to the clinical devices that often cover the whole head [7], [8]. As such, models that are trained to expect a layout based on the clinical standard of 10-20 will not be able to apply to these more comfortable EEG devices. As such, in order to create a model to interpret one of these devices, new data would have to be acquired and manually labeled, a significant barrier to the use of these more ergonomic devices in adaptive technology.

In medicine, personalized models recognize that while broad patterns exist in pathology, not every body behaves the same way [9]. This has tremendous implications when machine learning approaches average out effects and features across populations [9]. This means that individuals with atypical presentations are not likely to be strongly represented in the training data and are likely to be poorly served by models [9]. While this has seen some uptake in pharmacology research [9], because of the challenge of creating models from limited data, personalized models are not common in EEG.

Self-supervised learning (SSL) is a technique where unlabeled data is used to train foundation models that contain broad insights on a class of data, which can then be refined into task specific models via fine-tuning on labeled datasets [10]. This has many benefits because it reduces the amount of labeled data one needs to procure. In the context of EEG data, this looks like pooling data across many existing recordings, tasks, and datasets, then training on unsupervised pseudo-labels of the data. This forms a solid foundation that can be tuned based on the smaller amount of specialized labeled data available. Often, this data is collected by recruiting participants, running experimental protocols to generate relevant brain states, and applying many hours of clinical expertise to annotate the datasets [11]. The specific approach to performing SSL on EEG data that this paper will focus on is BENDR.

BENDR is an architecture presented in 2021 that implements SSL on EEG data using a modified wave2vec setup [12]. Standard wav2vec is a speech recognition architecture that works by encoding waveforms into dimension reduced slices, then jumbling and learning to reconstruct the sequences [13]. It does this with a learned encoder block composed of five convolutional layers, then an attention based context network. The main difference in the BENDR paper's approach is that the convolutional encoder block uses 1D convolutions to get the channels into one before the convolutions then the wav2vec contextualizer. Since the wav2vec architecture only works on 1d timeseries data, this approach was used to allow compatibility without introducing extreme memory challenges from extending wav2vec into multiple dimensions. Kostas et. al report two main limitations of their approach, rather limited generalization ability to novel EEG data, and loss of data due to channel reduction [12]. We have three approaches to tackle these problems.

The first BENDR limitation is how in the preprocessing setup, all datasets are truncated to 20 channels [12]. This means that during training, information is lost. Furthermore, the resultant model is fixed to this configuration. While the chosen 20 channels are common to most setups, this means that the model will not be effective on datasets that use smaller EEG setups, or that are localized to specific scalp regions and lack any of the 20 channels. This severely limits applicability to wearable adaptive BCI devices, as it necessitates more expensive and bulky hardware. Furthermore, in the case of larger static clinical sensor layouts that may have upwards of 64 electrodes, a significant amount of information is lost in the truncated channels. An ideal approach would natively support heterogeneous sensor configurations without information loss.

The second limitation is how the loss is set up based on priors from natural language that may not hold [12]. BENDR does contrastive learning like wav2vec, trying to unmix permutations of EEG time series sequences [14]. This approach is very sensible in the context of language processing as spoken sentences are known to be based on highly regular grammars with relatively few valid permutations relative to invalid ones along with having definite start and ends. In the context of EEG data, this might necessarily not be the case. While the wav2vec loss approach has been shown to be fairly effective in [12], it may prove beneficial to incorporate loss based on more strongly validated priors in EEG.

We instead train by unsupervised clustering of segments based on models of task-switching, as indicated through existing neuroscience work. The logic being that when a change in focus is detected, this should bound states of consistent focus or mental state. Sampling from states inside a consistent mental state should produce information vectors closer to each other than those outside of this consistent block of focus. This can be used to form clusters based on a conceptual mental state, which can be trained on.

The third approach that we take to improve generalizability is to pre-train on more modalities of EEG data. The original BENDR paper trained on 3 combined datasets from the Temple University corpus, which is all clinical EEG data, mostly epilepsy monitoring [12], [15]. EEG is used in many contexts, which may each elicit different patterns of brain activity than a screening for epilepsy in a hospital. These include tasks like motor imagery where the subject is directed to think of a motor action, event related potentials, where mental activity that is synchronized with certain activity is recognized, and assorted oddball tasks. Motor imagery classification is one of the key adaptive uses of BCI. By adding more variety of data, we seek to improve generalizability across modalities.

The goal of this project is to investigate the viability of applying prior information from neuroscience to improve an existing framework for performing self-supervised learning on EEG data.

II. Methods

Where BENDR directly maps SSL methodologies from NLP, our approach modifies it so that it is more concretely grounded in the electrophysiological and cognitive priors of the data. We do this in two ways, firstly with a more robust reduction methodology that maps arbitrary sensor counts and configurations and secondly with a contrastive learning method more strongly rooted in cognitive neuroscience.

A. Pre-Training

The pre-training is the most important part of this, and is where all of our contributions are located. There are 3 broad phases for the data during pre-training. First, preprocessing, where the data is filtered and the data is resampled to 256 Hz. Next, this clean data is transformed. In parallel,

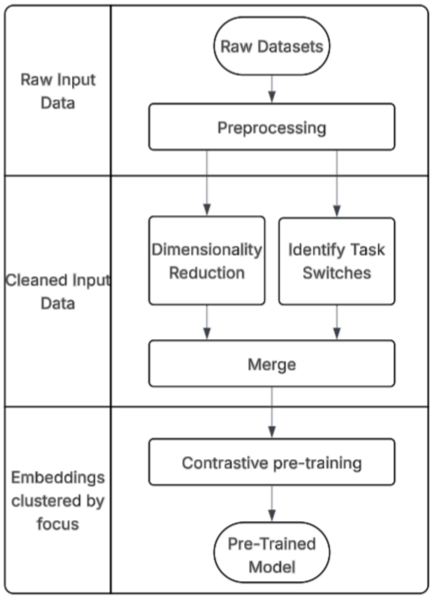

Fig. 1. Flowchart of the whole pre-training process

encoders perform dimensionality reduction and task labeling identifies probable times where task switching occurred. Finally, a contrastive loss is used to pre-train the model on our transformed data.

B. Data

Since this project is aimed at integrating across many datasets, we chose to source our data from OpenNeuro repositories. This has the advantage of the unified BIDS format, making metadata simple to automatically parse. We filtered datasets for EEG channels, which were selectively extracted. Then, we made note of the sensor configuration, grouping datasets according to their hardware layouts. We had two corpora of data, aggregating across many datasets one of which was unlabeled, the other was labeled. We run our pre-training on the datasets of the label-agnostic data, and we run supervised fine-tuning on the labelled data.

There are many possible approaches to filtering and cleaning the data. Based on research which shows that excessive preprocessing is not beneficial to model performance, we chose a minimal scheme [16], [17]. First, we resample to a unified 256 Hz, then we apply a high-pass filter at 0.3 Hz and low-pass at 120 Hz.

The BIDS format [18] allows the use of scripts to automatically generate configurations for the training so that new datasets can be incorporated without having to manually specify new details or clean incompatible data. This approach allowed the very efficient collection of many pre-training datasets.

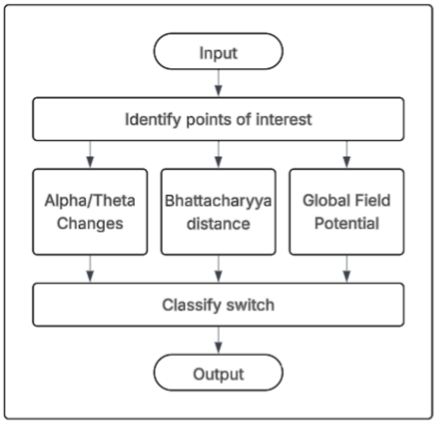

Fig. 2. Flowchart of the task-switch detection process

Currently, we do pre-training on the SRM Resting-state EEG dataset [19], and do downstream evaluation on the motor imagery eegmmidb dataset [20]. Future work will increase the quantity of data used, but this submission was limited by compute and training time.

C. Task-Switch Detection

A task switch refers to a transition between distinct cognitive task sets, during which attentional control, working memory, and response mappings are reorganized. Accurate localization of these change points is challenging due to noise contamination, inter-subject variability, and the multi-scale nature of neural dynamics. To address these challenges, we propose a multi-stage detection pipeline integrating statistical divergence, spectral features, and microstate analysis.

1) Initial Candidate Selection via Spectral Divergence: The first stage of our algorithm focuses on detecting abrupt shifts in the frequency spectrum, a critical feature for identifying task switches in EEG signals. Following the methodology proposed by Chen et al. [21], we quantify the statistical divergence between adjacent signal segments using the Bhattacharyya distance (D_B), which measures the degree of overlap between two probability distributions. This metric has been demonstrated to outperform other similarity measures in capturing EEG state transitions [21]. We define a 500-ms sliding window and compute the Power Spectral Density (PSD) for two contiguous segments. For two normalized spectral distributions P and Q, the distance is calculated as follows:

$$D_B(P,Q) = -\ln\left(\sum_i \sqrt{P_i Q_i}\right)$$

A candidate switch point is identified at time t if D_B exceeds a dynamic threshold:

$$Threshold_t = \mu_{10s} + 3\sigma_{10s}$$

Here, μ_{10s} and σ_{10s} represent the mean and standard deviation of the distance scores over the preceding 10-second baseline, ensuring the detection of statistically significant deviations from the steady-state spectral profile.

2) Validation of Oscillatory Power Dynamics via Dual-Window Search: To ensure that the candidate points are not artifacts, the algorithm validates each candidate against the power dynamics of the alpha (8-13 Hz) and theta (4-8 Hz) bands. These sub-bands are fundamental for capturing task-related neural oscillations and cognitive state transitions [21]. For each band, we first compute the squared Hilbert envelope to represent the energy trend. Specifically, the EEG signal is band-pass filtered, and the analytic signal is obtained via the Hilbert transform. To obtain a robust global estimate, the envelopes are averaged across all N channels before being squared:

$$E^2_{band}(t) = \left(\frac{1}{N} \sum_{i=1}^{N} |H(V_{i,band}(t))| \right)^2$$

Here, $H(\cdot)$ denotes the Hilbert transform. The verification process employs a sliding dual-window comparison within a local search range of ±800 ms centered at the initial candidate point, excluding boundary regions shorter than $2W$ to ensure valid pre and post-transition windows. We utilize an observation window W of 200 ms and a sliding step of 40 ms. At each step t, the average power in the pre-transition window $\bar{E}^2_{pre}$ (defined as $[t, t+W]$) and the post-transition window $\bar{E}^2_{post}$ (defined as $[t+W, t+2W]$) is calculated to determine the relative power change ratio:

$$\Delta\text{Ratio} = \frac{\bar{E}^2_{post} - \bar{E}^2_{pre}}{\bar{E}^2_{pre} + \epsilon}$$

To ensure numerical stability during ratio calculation, a small constant $\epsilon = 10^{-10}$ is introduced to the denominator. A candidate transition point then must meet two further criteria:

1) The relative power shifts must meet the magnitude requirements $|\Delta Ratio_\alpha| \geq 30\%$ and $|\Delta Ratio_\theta| \geq 20\%$.
2) Second, cognitive shifts require anti-correlated oscillations, a requirement formalized by the polarity constraint: $\Delta Ratio_\alpha \cdot \Delta Ratio_\theta < 0$.

The algorithm employs a greedy search strategy within the local range around the candidate time point, terminating immediately upon detecting the first window pair that fulfills these criteria. This approach prioritizes the earliest detectable onset of the transition. The corresponding pre-transition window, defined by the interval $[t, t+W]$, is then designated as the coarse estimate of the task-switch period.

3) Temporal Refinement via Global Field Power (GFP): The final localization of task-switching points is refined using the Global Field Power (GFP), which quantifies the spatial dispersion of scalp potentials across N electrodes at time t:

$$GFP(t) = \sqrt{\frac{1}{N} \sum_{i=1}^{N} \left(V_i(t) - \bar{V}(t) \right)^2}.$$

To suppress high-frequency artifacts and transient spikes, the raw GFP sequence is smoothed using a moving-average filter with a 30-ms window prior to further analysis.

According to microstate theory, EEG topographies remain quasi-stable for durations of approximately 60-120 ms and reach maximal signal-to-noise ratios at GFP peaks [22]. Transitions between microstates typically occur at local GFP minima, which are frequently located within the lowest 15% of the GFP distribution [22], [23]. These minima correspond to rapid large-scale neural reconfigurations [23].

Previous studies indicate that micro-task transitions are strongly associated with transient GFP reductions. [22] We hypothesize that large-scale task switching, occurring on a temporal scale of seconds, represents an aggregation of multiple microstate transitions. Consequently, such macro-level transitions are expected to exhibit more pronounced and spatially coherent GFP modulations. Based on this scaling hypothesis, we assume that genuine task switches should satisfy the following properties:

1) they should coincide with prominent GFP minima within a broad temporal context;
2) they should be preceded by a sustained decline in global synchronization;
3) they should exhibit a short-term decay phase consistent with microstates changing quickly in a short period of time.

For each candidate segment $[t_{\text{start}}, t_{\text{end}}]$, the algorithm first identifies the global GFP minimum within an extended temporal window:

$$t_{\min} = \text{argmin}(GFP(t))$$

$$t \in [t_{\text{start}} - 1500 \text{ ms}, \, t_{\text{end}} + 300 \text{ ms}]$$

The candidate time point is retained only if this minimum lies within a focused verification window:

$$t_{\min} \in [t_{\text{start}} - 300 \text{ ms}, \, t_{\text{end}} + 100 \text{ ms}]$$

This criterion ensures that the detected minimum represents a dominant transition-related event rather than an isolated local fluctuation.

To evaluate whether the transition is preceded by sustained neural desynchronization, the mean GFP within a focused window is compared to that of a broader baseline:

$$\text{mean}GFP(t), \, t \in [t\text{start} - 350, t_{\text{start}}] < \text{mean}GFP(t)$$

$$t \in [t\text{start} - 1500, t_{\text{start}} - 350]$$

A candidate time point is thus only accepted when mean$GFP(t)_{\text{near}}$ is less than $GFP(t)_{\text{far}}$

This condition reflects the assumption that large-scale task switching should not be preceded by increasing global synchronization.

To capture the expected microstate disengagement dynamics preceding a macro-level task switch, the algorithm examines the short-term GFP evolution within a restricted temporal window defined as $[t_{\text{start}} - 300 \text{ ms}, t_{\text{end}}]$.

Within this interval, the GFP sequence is analyzed using first-order temporal differences:

$$\Delta GFP(t_k) = GFP(t_{k+1}) - GFP(t_k).$$

A candidate transition is considered valid only if there exists a contiguous subsequence of duration at least 25 ms in which the GFP does not increase:

$$\Delta GFP(t_k) \leq 0, \quad \forall k \in [1, L],$$

where L corresponds to 25 ms expressed in samples. This criterion allows short plateau phases while enforcing an overall decay pattern, thereby reflecting the gradual disengagement of dominant microstates prior to large-scale cognitive reconfiguration. It further suppresses transient oscillatory artifacts that may otherwise produce spurious minima. Only candidate segments satisfying all three conditions are retained as validated task-switching intervals.

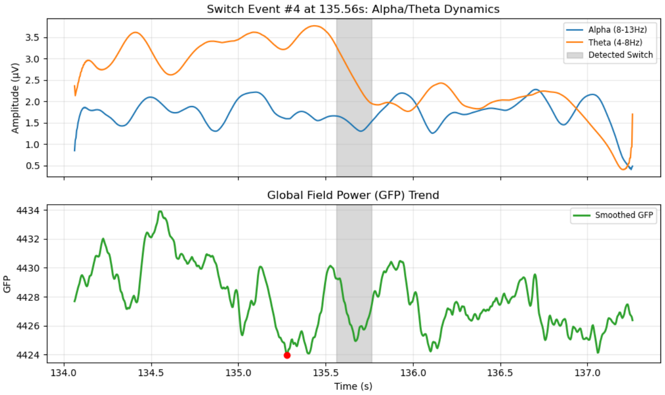

Fig. 3. Example of a detected task-switch event

Figure 3 illustrates a representative task-switch event detected by the proposed framework. During the validated transition interval, the theta-band power exhibits a pronounced decrease, whereas the alpha-band power shows a slight decline followed by a delayed rebound. This is a key indication of task switching. Meanwhile, the GFP signal presents a transient reduction and rapid recovery, indicating large-scale neural desynchronization and subsequent re-stabilization. The GFP minimum aligns closely with the detected transition window, supporting our microstate-based refinement strategy.

D. Graph and Convolution-based Encoder

The encoder portion of this network creates the embeddings from the filtered dataset. These embeddings are the direct subject of optimization during pre-training, so that they are highly informative regarding the underlying brain activity. The encoder block works by first performing a number of dimensionality reductions, then applying learned kernels, and finally a transformer layer which completes the embedding.

Our principal change from BENDR in this section is to use a more universal architecture for the dimensionality reduction, with encoders based on [24]. This allows us to apply our encoder block to many different electrode configurations without truncating any data. The way this works is by constructing graph neural networks for each configuration, then applying a layer at the end that aligns each network's embeddings into the same latent space. Ultimately, this approach is more robust to varying sensor layouts, whereas the BENDR format assumes a fixed layout for each dataset.

The reduction itself has three phase. First, the time series of each electrode is reduced by 1D convolutions. Next, a graph where each vertex is an electrode and each edge links spatially adjacent electrodes is constructed. Finally, a trained graph kernel is applied over the adjacency matrix. A high level overview of the convolutional and graph-based encoder process is shown in Fig. 4

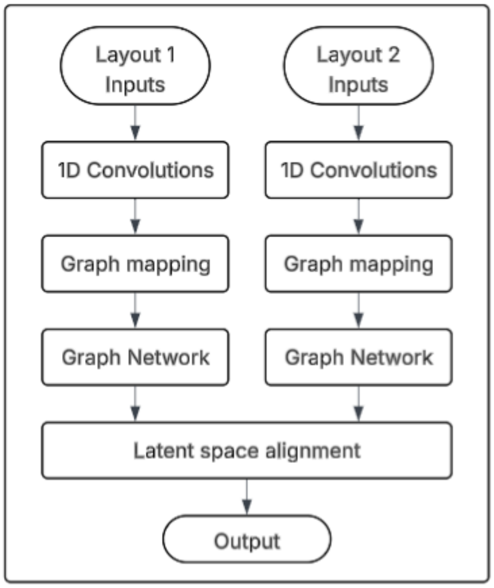

Fig. 4. High level overview of the convolutional and graph based encoder

1) Architecture: The encoder consists of two main components: a convolutional neural network that extracts temporal features, and a graph neural network to gather spatial information while improving robustness across datasets. The raw EEG files from the filtered datasets are first past through a series of CNN blocks with large kernel sizes and pooling layers for dimensionality reduction. Each CNN block consisted of a 1d convolutional layer followed by batch normalization, PReLu activation layer, a 1d average pooling layer, and a dropout layer. The chosen sequence was modeled based on the structure of EEGnet [25]. However, our CNN omitted depthwise convolutions and extracted only temporal features.

The node-feature matrix is then passed to the graph neural network, which consists of neighborhood based adjacency matrix, graph convolutions, batch normalization, PRelu activation, and SAGPooling. The adjacency matrix was constructed by assigning a binary value of 1 to electrode pairs in close spatial proximity and 0 otherwise, such that spatially neighboring nodes contribute most strongly to the feature updates of each node. SAGPooling was incorporated to further reduce dimensionality by scoring nodes according to their learned importance and retaining the most informative nodes for downstream processing [26]. The overall architecture is shown in Fig. 5

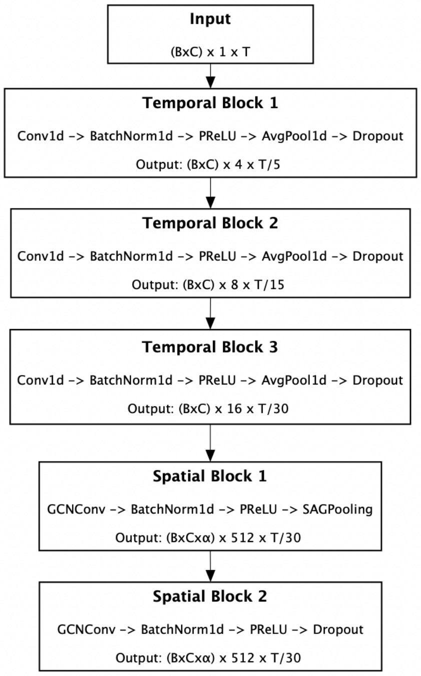
Input
(BxC) x 1 x T
Temporal Block 1
Conv1d -> BatchNorm1d -> PReLU -> AvgPool1d -> Dropout
Output: (BxC) x 4 x T/5
Temporal Block 2
Conv1d -> BatchNorm1d -> PReLU -> AvgPool1d -> Dropout
Output: (BxC) x 8 x T/15
Temporal Block 3
Conv1d -> BatchNorm1d -> PReLU -> AvgPool1d -> Dropout
Output: (BxC) x 16 x T/30
Spatial Block 1
GCNConv -> BatchNorm1d -> PReLU -> SAGPooling
Output: (BxCxα) x 512 x T/30
Spatial Block 2
GCNConv -> BatchNorm1d -> PReLU -> Dropout
Output: (BxCxα) x 512 x T/30

Fig. 5. The flowchart illustrates the architecture for the GNN approach. B is the batch size, C is the number of electrodes, T is the temporal dimension, and α is the parameter that determines the percentage of nodes which are kept after the SAGPooling layer.

E. Loss

The contrastive loss has the task of taking a sampling of embeddings from N different clusters, then reconstructing both the clusters and the order of the clusters. Our approach differs from that in BENDR by including this clustering alongside the ordering, rather than focusing solely on the ordering.

Our total loss combines three terms:

$$\mathcal{L} = \alpha_{\text{InfoNCE}}\,\mathcal{L}_{\text{InfoNCE}} + \alpha_{\text{cluster}}\,\mathcal{L}_{\text{cluster}} + \beta\,\|Z\|_2^2 \tag{1}$$

By default, we use the weights $\alpha_{\text{InfoNCE}} = 1.0$, $\alpha_{\text{cluster}} = 0.5$, $\beta = 1.0$.

The first term is the BENDR contrastive objective inherited from wav2vec 2.0; the second encourages embeddings from the same cognitive state to cluster together; and the third is an L2 penalty on the raw latent features to prevent norm collapse, a common failure in contrastive learning where encoders learn to produce arbitrarily large-magnitude embeddings instead of geometrically meaningful ones [13].

F. InfoNCE

InfoNCE is a contrastive loss that frames each masked time step as a classification problem. To understand it concretely: suppose a 10-second EEG segment from an eyes-open resting recording is fed into the model. The encoder f_θ produces a latent sequence $Z \in \mathbb{R}^{d \times T_e}$ one d-dimensional vector per time step. A random binary mask (rate 6.5%, span 10) replaces some of those latent vectors with a learned placeholder token [27]. The Transformer contextualiser g_ϕ then processes the full sequence (including masked positions) and produces context vectors $C \in \mathbb{R}^{d \times T_e}$. The model's task at each masked position t is: given the context c_t at that position, retrieve the correct original latent z_t from a pool of candidates.

$$\mathcal{L}_{\text{InfoNCE}} = -\mathbb{E}\left[\log \frac{\exp(\cos(c_t, z_t)/\kappa)}{\sum_j \exp(\cos(c_t, z_j)/\kappa)}\right] \tag{2}$$

The denominator sums over the correct z_t plus 20 randomly sampled distractors drawn from other positions in the same batch. Each candidate is scored by its cosine similarity to c_t, scaled by temperature $\kappa = 0.1$. Cosine similarity measures the angle between two vectors regardless of their magnitude, so the model must encode direction rather than magnitude. The temperature sharpens the distribution: at $\kappa = 0.1$, a candidate that is 10% more aligned with c_t than a distractor receives exponentially more probability mass, forcing the encoder to make more precise distinctions. This formulation follows wav2vec 2.0 and is equivalent in practice to a 21-way cross-entropy problem, computed with PyTorch's nn.CrossEntropyLoss [13].

Intuitively, if a participant transitions from eyes-open to eyes-closed during a recording, InfoNCE encourages the latent at any eyes-closed time step to be geometrically close to context vectors from neighbouring eyes-closed time steps, but not from eyes-open steps drawn as distractors. However, InfoNCE enforces this only locally in time: the 20 distractors are drawn from the same recording, so the model learns temporal coherence but not necessarily any broader structure about what kinds of brain states are similar [12].

G. Cluster Contrastive Loss

The cluster loss extends the contrastive objective across time by using the task-switch labels derived in Section 2.3. Where InfoNCE asks the model to identify a specific latent from nearby time steps, the cluster loss is a more abstract approach. It attempts to identify whether two windows, possibly far apart in the recording or even from different recordings, belong to the same cognitive state.

Concretely, consider a recording with three detected stable segments, say, a pre-task rest, an arithmetic block, and a post-task rest. Each stable segment receives a cluster label (0, 1, 2). Windows inside transition periods receive $y_i = -1$ and are excluded. For any window i with label $y_i \geq 0$, the model summarizes its latent sequence by means of a pooling and normalization of L2: $p_i = \text{mean}_t(Z_i)/\| \cdot \|_2$. This collapses the temporal dimension into a single unit-norm vector representing that window's overall cognitive state.

To define what it means for two windows to be similar, we maintain a per cluster memory bank $\mathcal{M}_k \in \mathbb{R}^{d \times M}$ ($M = 1000$ embeddings per cluster) implemented as a circular buffer that stores p_i vectors from previous batches [28]. The loss for window i is:

$$\mathcal{L}_{\text{cluster}} = -\frac{1}{N_+} \sum_{i:y_i \geq 0} \log \frac{\exp(\bar{s}_i^+/\tau_c)}{\exp(\bar{s}_i^+/\tau_c) + \sum_j \exp(s_{ij}^-/\tau_c)} \tag{3}$$

where $\bar{s}_i^+ = \text{mean}(p_i^\top \mathcal{M}_{y_i})$ is the average cosine similarity between window i and all stored embeddings from the same cluster y_i, and s_{ij}^- are cosine similarities to stored embeddings from every other cluster. The cluster temperature $\tau_c = 0.5$ is set higher than the InfoNCE temperature to allow softer gradients across the within-cluster distribution, reflecting the fact that two windows in the same cognitive state are not identical, a resting brain state at minute 1 and minute 5 of the same recording share structure but are not point-wise the same. The asymmetry in the temperature between the 2 losses mirrors findings in the supervised contrastive learning, where broader semantic similarity benefits from higher temperatures than specific local similarity [29].

The loss pushes p_i toward the centroid of $\mathcal{M}_{y_i}$ while pulling it away from all other clusters' banks. Returning to the arithmetic example: the embedding of a window mid-way through the arithmetic block will be pulled toward stored embeddings of other arithmetic windows (same cluster), and pushed away from stored embeddings of both rest segments (different clusters). Over time, the latent space organizes so that windows from matching cognitive states are grouped together, regardless of when they occur in the recording.

EEG recordings are highly uneven in states: a 10-minute recording might contain 8 minutes of rest and only 2 minutes of a target task. In any given mini-batch, a rare cognitive state may appear in only one or two windows. Without a memory bank, the loss would have no meaningful positives or negatives for that state. By accumulating embeddings from the last $M = 1000$ windows of each state across all previous batches, the loss remains defined even when a state is sparsely represented in the current batch [28].

Together, the two losses operate at different scales of similarity. InfoNCE enforces temporal consistency; nearby time steps should have similar representations. The cluster loss enforces cognitive consistency; windows from the same task block should cluster together across temporal distance. The L2 regularization term $\beta\|Z\|_2^2$ prevents the encoder from satisfying both objectives by inflating embedding norms and overfitting, ensuring the geometry of the latent space carries meaningful information.

H. Downstream

Downstream training we take several annotated datasets and use our trained foundation as a base for supervised learning. We have a standard supervised learning setup, with hyperparameters and dropout as specified in [12]. We then use 5-fold cross validation to aquire robust metrics of performance, identifying both accuracy and class balanced accuracy (BAC). In this draft for the 28th February submission, we only trained on one downstream dataset, the physionet motor imagery dataset.

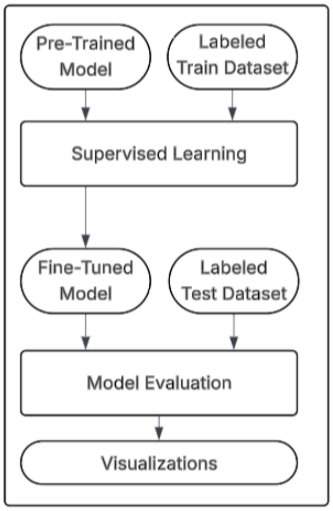

Fig. 6. Flowchart of the process for downstream training

III. Results

We trained four models on our downstream data and measured bac and accuracy for each. First we trained with the baseline BENDR layout, then the BENDR layout using our custom GNN based contextualizer, then we trained one model for each of these incorporating our novel loss based on task-switching.

We used paired t-tests to assess that the differences each pair of models are statistically significant, with the exception of adding cluster labels to the GNN having no statistically significant impact on both BAC and accuracy.

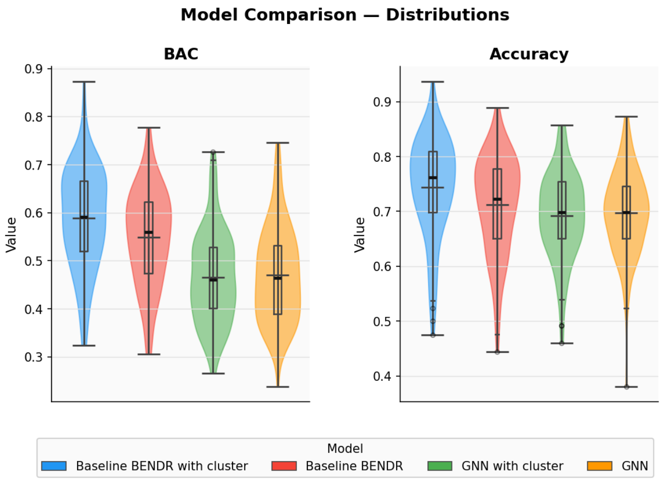

Fig. 7. BAC and accuraccy for all four models

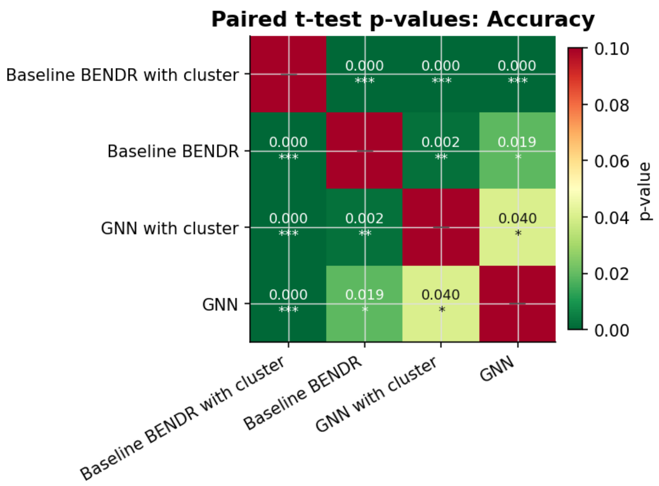

Fig. 9. Heatmap of statistical significance of differences in accuracy

TABLE I
MODEL COMPARISON: MEAN ± STD

Model	BAC	Accuracy
Baseline BENDR with cluster	**0.5887 ± 0.1046**	**0.7441 ± 0.0906**
Baseline BENDR	0.5490 ± 0.0995	0.7120 ± 0.0894
GNN with cluster	0.4658 ± 0.0953	0.6915 ± 0.0780
GNN	0.4702 ± 0.0982	0.6970 ± 0.0762

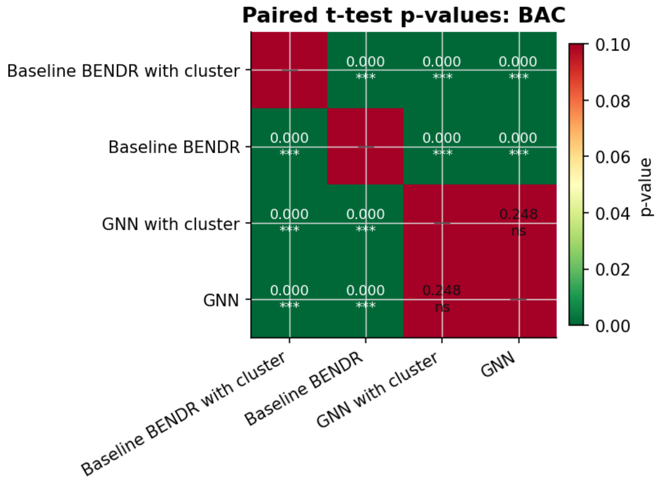

Fig. 8. Heatmap of statistical significance of differences in BAC

IV. DISCUSSION

Our approach has demonstrated that incorporating task switching priors to the loss utilized in pre-training for EEG models improves generalizability and performance. Adding the cluster labels both improved balanced accuracy and raw accuracy for the baseline BENDR model. This is a meaningful result as the cluster loss operates entirely at the pre-training stage, where the downstream architecture and fine-tuning are identical across conditions, reflecting an improved latent space learned during pre-training. Moreover, the violin plot for baseline BENDR with cluster loss not only achieves the highest mean BAC (0.5887), but also shows a relatively tight distribution compared to plain BENDR, suggesting the cluster loss reduces variance across subjects in addition to improving average performance. A model that is consistently good across individuals is more deployable than one that performs well on average but fails on a subset of users. This subject-level variance is a known bottleneck in EEG systems, and any pre-training strategy that reduces it without access to subject-specific labels is worth noting [30].

This has significant implications in novel applications of EEG, and improves on the existing approach by further improving the ability to learn on limited downstream data. This translates directly to clinical and BCI contexts where large amounts of labelled data may be inaccessible or impractical to work with. A model that has organized its latent space around cognitive state boundaries requires fewer labelled examples to learn a new downstream task, as the relevant structure is already implicit in the representation. This opens the door to more few-shot applications for the diagnosis of rare neurological conditions or BCI calibration. This pre-training advantage is likely to compound over time, as the model is not just starting from better weights, it is starting from a representation space where the target signal is partially separated. In theory, a neurologist would be able to use any BCI configuration regardless of electrode channel layout, and gain further insight into prominent frequencies of the patient to improve diagnosis.

We posit several reasons for why our GNN method has lower accuracy for the same amount of training. Firstly, given that we had to limit our pre-train to a smaller set, we only trained on the standard 10-20 layout. As such, while the raw

BENDR removed channels in order to train on a familiar layout, the GNN had to generate new parameters dedicated to collapsing the spatial dimensions of this specific sensor layout. Since part of the encoder has to be retrained for every new sensor configuration. Furthermore, given how our novel approach does not reject any channels, it is attempting to learn from a more complicated signal than the BENDR, which may make it more prone to underfitting. Finally, GNNs are quite sensitive to hyperparameters, and we used the same learning rate and dropout for all models [31]. While performance is lower, it remains capable of mapping a constant contextualizer head to novel dimensionalities, which may be a worthwhile tradeoff in applications where training data for the desired sensor configuration is highly limited. For instance, our GNN model can apply the pre-trained knowledge to headband or earpiece wearable BCI devices, which the BENDR format cannot do.

During training, we also note that the baseline BENDR architecture very quickly converges to a training accuracy of around 99%, whereas all the other models have training accuracy more in line with testing. We believe that this suggests a higher level of overfitting, which may correspond to less room to grow when the network size and training set is scaled up in future work.

While there was an increase in performance on the GNN approach to adding the cluster loss, one additional observation from the heatmaps is that the GNN with cluster and the plain GNN are not statistically significant on BAC (p = 0.208), whereas the same comparison in accuracy is significant (p = 0.040). This is likely due to the GNN encoder not benefiting as much from the pre-training, so the change to the pre-training has less of an effect. BAC accounts for class imbalance by averaging sensitivity and specificity across classes, so it is a stricter measure of whether the model has genuinely learned to distinguish all motor imagery classes or is exploiting class frequency. The fact that the cluster loss improves accuracy but not BAC in the GNN condition suggests it may be helping the model classify the majority class more reliably, without improving discrimination of the harder minority classes. This is consistent with the hypothesis that the GNN is underfitting from insufficient pre-training data, as the cluster signal improves surface-level performance but cannot fully restructure the latent space without more diverse pre-training examples to anchor each cognitive cluster. This can be further tested by training on a downstream dataset that allows for spatial parameters to be reused from training.

In future work, we intend to increase the pre-train size and open the downstream testing across many more modalities and sensor configurations. This would allow for significantly more robust models, and a more thorough validation. We would also thoroughly validate all three potential reasons for GNN accuracy being lower, by comparing to results when a pre-trained encoder can be reused, training on a sensor configuration with only 19 channels, and running hyperparameter optimization.

Furthermore, testing directly on the novel applications that were discussed would allow for a more concrete link to clinical relevance. Directly training a few-shot personalized model would alow for detailed discussion of the generalizability in the few shot context and how this can be applied in a clinical setting. Training a model for data on highly limited sensor configurations would likewise allow for a more detailed discussion of the adaptive implications of our approach.

V. Conclusion

Altogether, this paper investigated whether incorporating neuroscience-grounded priors into the pre-training objective of an EEG foundation model improves downstream performance. We extended the BENDR framework in two directions: a graph-based encoder that accepts arbitrary electrode configurations without discarding channels, and a cluster contrastive loss that uses automatically detected cognitive state boundaries as pseudo-labels during pre-training. Our results on the Motor Imagery benchmark show that the cluster loss produces a statistically significant improvement in both accuracy and BAC for the baseline BENDR architecture, while the GNN encoder demonstrates a proof of concept for hardware-agnostic pre-training that BENDR cannot support by design.

The practical implications of these findings extend beyond the benchmarks. EEG-based systems face a recurring problem: every new device configuration, clinical context, or user population requires new labelled data, which is expensive and time-consuming to collect. A foundation model that learns better-structured representations during pre-training reduces how much of that labelled data is needed downstream. In clinical settings, this could lower the barrier to deploying EEG diagnostic tools for rare neurological conditions. In adaptive BCI, it could enable wearable devices with non-standard electrode layouts to benefit from pre-training on large clinical datasets they would otherwise be incompatible with.

It is worth comparing these contributions in the broader context of foundation models for biomedical signals. Large language models derive much of their power from the fact that language has strong statistical regularities that a pre-training objective can exploit without labels such as grammar and semantic coherence implicit in raw text. EEG lacks an equivalent universal structure; the signal is task-dependent, hardware-dependent, and individual. Our choice of prior, that EEG segments belonging to the same cognitive state should have similar representations, improved downstream performance, suggesting that the path forward for EEG foundation models is not just collecting more data, but being deliberate about what domain knowledge gets built into the training process itself. This work is a step in that direction, and we hope it serves as a foundation for future models that more fully close the gap between what neuroscience knows about the brain and what machine learning can learn from it.

References

[1] R. Mahajan and B. I. Morshed, "Unsupervised eye blink artifact denoising of EEG data with modified multiscale sample entropy, kurtosis, and wavelet-ICA," *IEEE Journal of Biomedical and Health Informatics*, vol. 19, no. 1, pp. 158–165, 1 2015. [Online]. Available: https://ieeexplore.ieee.org/document/6843355/citations#citations

[2] E. Başar, C. Başar-Eroglu, S. Karakaş, and M. Schürmann, "Gamma, alpha, delta, and theta oscillations govern cognitive processes," *International Journal of Psychophysiology*, vol. 39, no. 2-3, pp. 241–248, 1 2001. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/11163901/

[3] J. Jeong, "EEG dynamics in patients with Alzheimer's disease," *Clinical Neurophysiology*, vol. 115, no. 7, pp. 1490–1505, 7 2004. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/15203050/

[4] J. J. Newson and T. C. Thiagarajan, "EEG Frequency Bands in Psychiatric Disorders: A Review of Resting State Studies," *Frontiers in human neuroscience*, vol. 12, 1 2019. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/30687041/

[5] N. Jamil, A. N. Belkacem, S. Ouhbi, and A. Lakas, "Noninvasive Electroencephalography Equipment for Assistive, Adaptive, and Rehabilitative Brain–Computer Interfaces: A Systematic Literature Review," *Sensors 2021, Vol. 21,*, vol. 21, no. 14, 7 2021. [Online]. Available: https://www.mdpi.com/1424-8220/21/14/4754

[6] J. A. Urigüen and B. Garcia-Zapirain, "EEG artifact removal-state-of-the-art and guidelines," *Journal of neural engineering*, vol. 12, no. 3, 6 2015. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/25834104/

[7] V. Mihajlovic, B. Grundlehner, R. Vullers, and J. Penders, "Wearable, wireless EEG solutions in daily life applications: What are we missing?" *IEEE Journal of Biomedical and Health Informatics*, vol. 19, no. 1, pp. 6–21, 1 2015. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/6824740

[8] J. Choi, N. Kaongoen, H. Choi, a. , A. Meiser, A. Lena Knoll, M. G. Bleichner, C. Tremmel, D. J. Krusienski, m. schraefel, J. Woo Choi, H. Kwon, C. Hwang, G. Hwang, B. Hyung Kim, and S. Jo, "The future of wearable EEG: a review of ear-EEG technology and its applications," *Journal of Neural Engineering*, vol. 20, no. 5, p. 051002, 10 2023. [Online]. Available: https://iopscience.iop.org/article/10.1088/1741-2552/acfcdahttps://iopscience.iop.org/article/10.1088/1741-2552/acfcda/meta

[9] Y. Wu, L. Li, B. Xin, Q. Hu, X. Dong, and Z. Li, "Application of machine learning in personalized medicine," *Intelligent Pharmacy*, vol. 1, no. 3, pp. 152–156, 10 2023. [Online]. Available: https://doi.org/10.1001/jamaoncol.2021.6818

[10] A. Jaiswal, A. R. Babu, M. Z. Zadeh, D. Banerjee, and F. Makedon, "A Survey on Contrastive Self-Supervised Learning," *Technologies 2021, Vol. 9,*, vol. 9, no. 1, 12 2020. [Online]. Available: https://www.mdpi.com/2227-7080/9/1/2

[11] F. Lotte, L. Bougrain, A. Cichocki, M. Clerc, M. Congedo, A. Rakotomamonjy, and F. Yger, "A review of classification algorithms for EEG-based brain-computer interfaces: a 10 year update," *Journal of neural engineering*, vol. 15, no. 3, 4 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/29488902/

[12] D. Kostas, S. Aroca-Ouellette, and F. Rudzicz, "BENDR: Using Transformers and a Contrastive Self-Supervised Learning Task to Learn From Massive Amounts of EEG Data," *Frontiers in Human Neuroscience*, vol. 15, p. 653659, 6 2021. [Online]. Available: www.frontiersin.org

[13] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations," *Advances in Neural Information Processing Systems*, vol. 33, pp. 12 449–12 460, 2020. [Online]. Available: https://github.com/pytorch/fairseq

[14] D. Kostas, S. Aroca-Ouellette, and F. Rudzicz, "BENDR: Using Transformers and a Contrastive Self-Supervised Learning Task to Learn From Massive Amounts of EEG Data," *Frontiers in Human Neuroscience*, vol. 15, p. 653659, 6 2021. [Online]. Available: www.frontiersin.org

[15] I. Obeid and J. Picone, "The Temple University Hospital EEG Data Corpus," *Frontiers in Neuroscience*, vol. 10, no. MAY, p. 196, 2016. [Online]. Available: https://pmc.ncbi.nlm.nih.gov/articles/PMC4865520/

[16] A. Delorme, "EEG is better left alone," *Scientific Reports 2023 13:1*, vol. 13, no. 1, pp. 2372–, 2 2023. [Online]. Available: https://www.nature.com/articles/s41598-023-27528-0

[17] R. Kessler, A. Enge, and M. A. Skeide, "How EEG preprocessing shapes decoding performance," *Communications Biology 2025 8:1*, vol. 8, no. 1, pp. 1039–, 7 2025. [Online]. Available: https://www.nature.com/articles/s42003-025-08464-3

[18] C. R. Pernet, S. Appelhoff, K. J. Gorgolewski, G. Flandin, C. Phillips, A. Delorme, and R. Oostenveld, "EEG-BIDS, an extension to the brain imaging data structure for electroencephalography," *Scientific Data 2019 6:1*, vol. 6, no. 1, pp. 103–, 6 2019. [Online]. Available: https://www.nature.com/articles/s41597-019-0104-8

[19] C. Hatlestad-Hall, T. W. Rygvold, and S. Andersson, ""srm resting-state eeg"," 2022.

[20] G. Schalk, D. McFarland, T. Hinterberger, N. Birbaumer, and J. Wolpaw, "Bci2000: a general-purpose brain-computer interface (bci) system," *IEEE Transactions on Biomedical Engineering*, vol. 51, no. 6, pp. 1034–1043, 2004.

[21] G. Chen, G. Lu, W. Shang, and Z. Xie, "Automated Change-Point Detection of EEG Signals Based on Structural Time-Series Analysis," *IEEE Access*, vol. 7, pp. 180 168–180 180, 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8918063

[22] A. Mishra, B. Englitz, and M. X. Cohen, "EEG microstates as a continuous phenomenon," *NeuroImage*, vol. 208, 3 2020. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/31841679/

[23] D. Haydock, S. Kadir, R. Leech, C. L. Nehaniv, and E. Antonova, "EEG microstate syntax analysis: A review of methodological challenges and advances," *NeuroImage*, vol. 309, no. 1, p. 121090, 4 2025. [Online]. Available: https://doi.org/10.1007/s10548-020-00805-1

[24] U. Talukdar, S. M. Hazarika, J. Q. Gan, N. Robinson, M. Ramasubba Reddy, J. Han, X. Wei, and A. Aldo Faisal, "EEG decoding for datasets with heterogenous electrode configurations using transfer learning graph neural networks," *Journal of Neural Engineering*, vol. 20, no. 6, p. 066027, 12 2023. [Online]. Available: https://iopscience.iop.org/article/10.1088/1741-2552/ad09ffhttps://iopscience.iop.org/article/10.1088/1741-2552/ad09ff/meta

[25] M. Zuo, B. Yu, L. Sui, M. Eder, J. Xu, M. Grosse-Wentrup, V. J. Lawhern, A. J. Solon, N. R. Waytowich, S. M. Gordon, C. P. Hung, and B. J. Lance, "EEGNet: a compact convolutional neural network for EEG-based brain–computer interfaces," *Journal of Neural Engineering*, vol. 15, no. 5, p. 056013, 7 2018. [Online]. Available: https://iopscience.iop.org/article/10.1088/1741-2552/aace8chttps://iopscience.iop.org/article/10.1088/1741-2552/aace8c/meta

[26] J. Lee, I. Lee, and J. Kang, "Self-Attention Graph Pooling," *36th International Conference on Machine Learning, ICML 2019*, vol. 2019-June, pp. 6661–6670, 6 2019. [Online]. Available: http://arxiv.org/abs/1904.08082

[27] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation Learning with Contrastive Predictive Coding," 1 2019. [Online]. Available: http://arxiv.org/abs/1807.03748

[28] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum Contrast for Unsupervised Visual Representation Learning," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 9726–9735, 3 2020. [Online]. Available: http://arxiv.org/abs/1911.05722

[29] P. Khosla, P. Teterwak, C. Wang, A. Sarna, G. Research, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan, "Supervised Contrastive Learning," *Advances in Neural Information Processing Systems*, vol. 33, pp. 18 661–18 673, 2020. [Online]. Available: https://t.ly/supcon

[30] N. Padfield, J. Zabalza, H. Zhao, V. Masero, and J. Ren, "EEG-Based Brain-Computer Interfaces Using Motor-Imagery: Techniques and Challenges," *Sensors (Basel, Switzerland)*, vol. 19, no. 6, 3 2019. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/30909489/

[31] Y. Liu, J. Liu, and Y. Li, "Automatic search of architecture and hyperparameters of graph convolutional networks for node classification," *Applied Intelligence 2022 53:9*, vol. 53, no. 9, pp. 11 104–11 119, 8 2022. [Online]. Available: https://link.springer.com/article/10.1007/s10489-022-04096-w