Master Thesis

Decoding Brains with HFMCA: Self-Supervised Graph Transformers for Robust fMRI Representations

by

Jakub Frąc
(jfr231)

First Supervisor: Shujian Yu
Second Reader: Aneta Lisowska

July 31, 2025

Submitted in partial fulfillment of the requirements for
the VU degree of Master of Science in Artificial Intelligence

Decoding Brains with HFMCA: Self-Supervised Graph Transformers for Robust fMRI Representations

Jakub Frąc[1]

Vrije Universiteit Amsterdam

Abstract. Self-supervised pretraining has significant potential for improving deep learning on functional MRI (fMRI) data. We introduce a self-supervised Graph Transformer encoder pretrained with the Hierarchical Functional Maximal Correlation Algorithm (HFMCA) and evaluate its effectiveness on multiple resting-state fMRI datasets and neuroimaging tasks. HFMCA-pretrained embeddings proved robust and informative for various classification and regression benchmarks, often matching or surpassing strong baselines. The pretrained encoder demonstrated transferability to new datasets, with marked benefits in frozen linear evaluations, although this advantage decreased after fine-tuning and varied across tasks. Importantly, increasing the pretraining dataset size did not consistently enhance downstream performance, with indiscriminate scaling occasionally causing negative transfer. Overall, our results establish HFMCA-based pretraining as a robust and transferable self-supervised strategy for fMRI graph representation learning, while highlighting the need for careful selection of pretraining data to minimize negative transfer. We discuss directions for mitigating such transfer and encourage further development of scalable self-supervised approaches for generalizable brain decoding models.

Keywords: HFMCA · self-supervised learning · graph transformer · fMRI · representation learning

1 Introduction

Functional magnetic resonance imaging (fMRI) has become a cornerstone for non-invasive investigation of human brain dynamics. A central focus in the field is the analysis of resting-state functional connectivity, which assesses statistical relationships in BOLD signal fluctuations across different brain regions. Alterations in resting-state functional connectivity have been observed in various neurological and psychiatric conditions, including autism [13, 18], depression [1, 10], and schizophrenia [15, 20].

Application of deep learning techniques to fMRI data faces several challenges, primarily due to limited dataset sizes and variability in data preprocessing pipelines. Recent developments in foundation models and transfer learning [31] offer potential solutions to these challenges. Several studies have investigated

self-supervised training on temporal BOLD signal data [6, 21, 27], demonstrating improved performance and greater robustness to domain shift. However, the data distribution of high-dimensional temporal fMRI signals is difficult to model, especially given limited sample sizes.

To address these issues, we propose a self-supervised Graph Transformer [19] that models resting-state functional connectivity as graphs, capitalizing on their inherent relational structure and reducing dependence on noisy temporal data. Our encoder is pretrained with the Hierarchical Functional Maximal Correlation Algorithm (HFMCA) [11], a multiview objective designed to align feature representations across diverse graph augmentations and prevent representation collapse, improving generalization even with limited data.

Our research is guided by three core hypotheses:

- $\mathcal{H}_1$: The HFMCA pretrained encoder produces high-quality graph embeddings that effectively capture semantic information relevant for fMRI graph classification tasks.
- $\mathcal{H}_2$: The HFMCA pretrained encoder produces graph embeddings that allow effective transfer learning across different fMRI graph classification tasks.
- $\mathcal{H}_3$: The quality of HFMCA pretrained encoder embeddings improves with the size of the pretraining dataset.

The following sections will provide a detailed introduction to the HFMCA training scheme, describe rs-fMRI modeling with deep neural networks, explain the role of data augmentation, and present the details of the Graph Transformer architecture. Next, we will describe our experiments and present and analyze the results, which will inform the validity of our hypotheses. Finally, we will discuss future directions based on our findings.

2 HFMCA Pretraining for GNNs

Hierarchical **F**unctional **M**aximal **C**orrelation **A**lgorithm utilizes multiview self-supervised learning to investigate the hierarchical relationships between the data and their augmentations.

2.1 FMCA

Let us consider random variables X and Y. Given their probability distributions $p(X)$ and $p(Y)$, we can define their statistical dependence with a formula:

$$\rho(X, Y) := \frac{p(X, Y)}{p(X)p(Y)} \tag{1}$$

The Functional Maximal Correlation Algorithm (FMCA) is built to measure and maximize statistical dependence between two random variables, typically using paired neural networks. [11] presents the theoretical foundation for modeling ρ using neural networks, enabling the training of encoders that encapsulate meaningful semantic information within their embeddings.

It approximates ρ by decomposing it into a sum of eigenvalues and their respective eigenfunctions:

$$\rho(X,Y) = \frac{p(X,Y)}{p(X)p(Y)} \approx \sum_k \sqrt{\sigma_k}\varphi_k(X)\psi(Y) \tag{2}$$

FMCA uses two encoders $f_\theta : \mathcal{X} \to \mathbb{R}^K$ and $g_\omega : \mathcal{Y} \to \mathbb{R}^K$ to approximate the eigenfunctions φ and ψ. The encoders map the original and augmented data into K-dimensional embeddings, and then the autocorrelation and cross-correlation matrices are defined as follows:

$$\begin{aligned} R_F &= \mathbb{E}_X[f_\theta(X) f_\theta^\intercal(X)] \\ R_G &= \mathbb{E}_Y[g_\omega(Y) g_\omega^\intercal(Y)] \\ P_{FG} &= \mathbb{E}_{X,Y}[f_\theta(X) g_\omega^\intercal(Y)] \\ R_{FG} &= \begin{bmatrix} R_F & P_{FG} \\ P_{FG}^\intercal & R_G \end{bmatrix} \end{aligned} \tag{3}$$

FMCA will maximize the log-determinant of the auto-correlations R_F and R_G, while minimizing the log-determinant of the joint autocorrelation R_{FG}:

$$\begin{aligned} \min_{\theta,\omega} \mathcal{L} &:= \log \frac{\det R_{FG}}{\det R_F \det R_G} \\ &:= \log\det R_{FG} - \log\det R_F - \log\det R_G \end{aligned} \tag{4}$$

Eigenvalues of the autocorrelation matrix quantify how much variance in the data is explained by each corresponding eigenvector direction. Knowing that a determinant is a product of all eigenvalues of the matrix, we infer that maximizing log-determinant of auto-correlations R_F and R_G encourages the encoder to create embeddings that explain as much variance in the data as possible. It also encourages the features of the embeddings to be orthogonal.

The joint auto-correlation matrix R_{FG} encapsulates cross-correlation information between embeddings $f_\theta(X)$ and $f_\omega(Y)$. Minimizing its log-determinant will encourage the features from both encoders to be aligned, which is expected since both X and Y represent two views of the same object.

2.2 Hierarchical FMCA for Self-Supervised Learning

Similarly to other SSL methods [2, 3, 7, 30], HFMCA utilizes data augmentations to learn meaningful representations. However, instead of a single pair (X, Y) let us consider a source X and a set of augmentations

$$Y = \{Y_0, Y_1 \dots Y_T\}$$

where $Y_i = \mathcal{T}_i(X)$ and $\mathcal{T}_i$ signifies ith augmentation function.

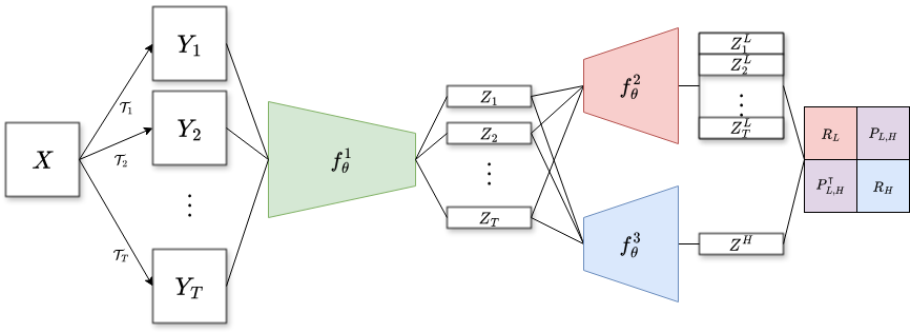

Fig. 1: HFMCA directly measures and maximizes statistical dependence among multiple augmented views of each sample. The backbone f_θ^1 processes each augmentation into low-level features, which are separately projected by f_θ^2 and aggregated by a projection head f_θ^3 into high-level features. HFMCA optimizes a log-determinant cost that ensures these features are orthonormal while capturing shared information across the hierarchy. After pretraining, f_θ^2 and f_θ^3 are discarded and only the backbone f_θ^1 is used for fine-tuning.

We will rely on augmentations that subsample the original data, such as image cropping for computer vision or random walk subsampling for graph data. The main idea is that **each augmentation represents a separate, low-level feature. Aggregations of all the low-level features form higher-level features that summarize all views and semantically represent the original object.**

Instead of modeling dependence ρ between two views of the same object (X, Y), we will model the dependence between hierarchical levels: the set of all augmentation features and their high-level summary. More formally, we will define the dependence $\rho(Z^L, Z^H)$ as:

$$\rho(Z^L, Z^H) = \frac{p(Z^L, Z^H)}{p(Z^L)p(Z^H)} \tag{5}$$

Where:

- $Z^L = \{Z_l^L\}_{t=1}^T$ - features from each augmentation, concatenated together (Low).
- Z^H - a high-level feature ($High$).

To extract features from the augmented images, we will utilize two neural networks f_θ and g_ω. Knowing that the features are hierarchical, we can utilize weight sharing for faster and more robust training.

In the original paper [11], the authors used a backbone encoder f_θ^1 and a projection head f_θ^2. The backbone f_θ^1 is first applied to T augmentations, extracting T low-level features Z_t^L. These features are then concatenated, acting as inputs to the projection head and producing high-level feature Z^H. In the context of the FMCA framework described above, f_θ is defined as f_θ^1, and g_ω is defined as an encoder f_θ^1 followed by the projection head f_θ^2.

A part of our contribution is the introduction of an additional projection head. Instead of using a combination of backbone and a single projection head, we will use a backbone f_θ^1 and two projection heads f_θ^2, f_θ^3 to define the networks f_θ and g_ω. The network $f\theta$ is the backbone f_θ^1 followed by the projection f_θ^2. The network g_ω is the same backbone f_θ^1 followed by the projector f_θ^3. Adding an additional projector head on top of the backbone is a widespread practice in pretraining encoder models [7, 30], and it helps mitigate the noise of data augmentation [2]. This process is visualized in Fig. 1.

Apart from the additional projector, the training protocol remains the same as in [11]. With extracted low-level features Z^L and the high-level feature Z^H, we implement the FMCA protocol:

$$Z^L = \{Z_1^L, Z_2^L \dots Z_T^L\} = \{f_\theta(Y_0), f_\theta(Y_1) \dots f_\theta(Y_T)\}$$

$$Z^H = g_\omega([Z_1^L, Z_2^L \dots Z_T^L])$$

With the low-level features Z^L and the high-level feature Z^H, we compute the autocorrelation and cross-correlation matrices:

$$\begin{aligned} R_L &= \mathbb{E}[Z^L Z^{L\intercal}] \\ R_H &= \mathbb{E}[Z^H Z^{H\intercal}] \\ P_{L,H} &= \frac{1}{T}\sum_{t=1}^{T} \mathbb{E}[Z_t^L Z^{H\intercal}] \\ R_{L,H} &= \begin{bmatrix} R_L & P_{L,H} \\ P_{L,H}^\intercal & R_H \end{bmatrix} \end{aligned} \tag{6}$$

We will minimize the log determinant of the joint auto-correlation matrix $R_{L,H}$ to maximize statistical dependence between two hierarchical levels (Z^L and Z^H), and minimize the log-determinant of the separate auto-correlation matrices R_L and R_H to extract orthogonal features:

$$\min_{\theta,\omega} \mathcal{L} := \log\det R_{L,H} - \log\det R_L - \log\det R_H \tag{7}$$

After pretraining, f_θ^2 and f_θ^3 are discarded and only the backbone f_θ^1 is used for fine-tuning.

2.3 HFMCA Pretraining with rs-fMRI

Preprocessing fMRI Recordings
An rs-fMRI recording is represented as a 4-dimensional tensor of shape (X, Y, Z, T), where (X, Y, Z) denote spatial voxel coordinates and T is the number of time points. For each voxel, a temporal series of brain activity is recorded across T time points. Voxels are assigned to Regions of Interest (ROIs) using anatomical or functional atlases; the time series of voxels within each ROI are then aggregated, typically by averaging, to facilitate analysis.

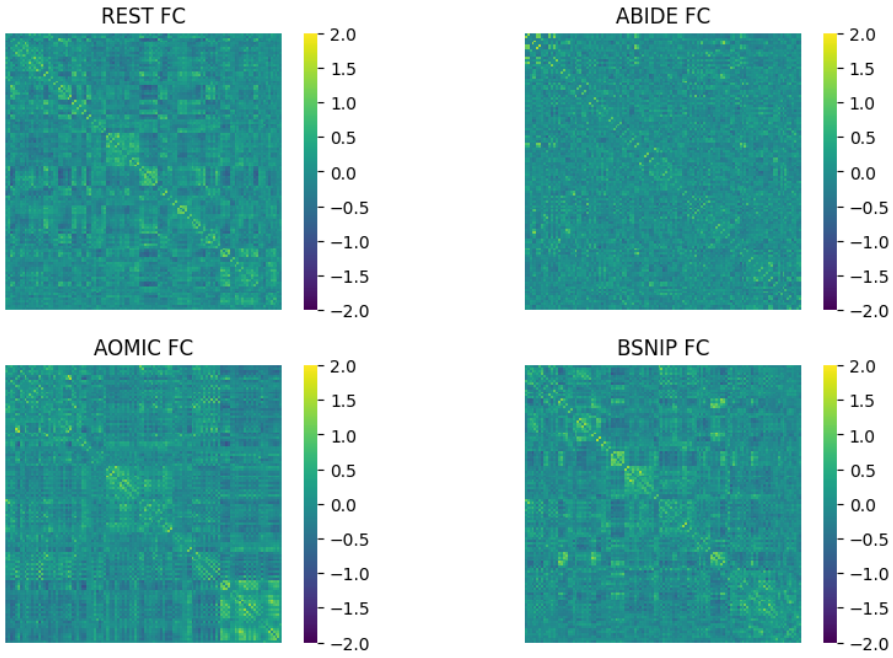

Fig. 2: Visualizations of functional connectivity matrices derived from the REST, ABIDE, AOMIC, and BSNIP datasets illustrate notable differences across datasets. These differences arise primarily from variations in measuring equipment. The most conspicuous differences are seen in the spatial resolution and amplitude of the signals. For instance, the REST sample contains brighter regions and larger clusters of similar colors, indicating higher contrast and lower resolution, compared to the ABIDE sample.

We preprocessed the rs-fMRI data by parcellating each subject's brain into 116 anatomical regions using the Automated Anatomical Labeling (AAL116) atlas [24]. For each subject, we extracted the mean BOLD time series from each ROI, resulting in a matrix of shape (116, T), where T is the number of acquired time points. We then computed the Pearson correlation coefficient for all pairs of ROIs, yielding a symmetric 116×116 functional connectivity matrix per subject. We extract top $\frac{N^2}{400}$ ($N = 116$) coefficients to derive an adjacency matrx representing undirected brain networks. The functional connectivity values themselves can serve as node features (i.e., zero-layer node embeddings). Thus, we obtain a set of graphs suitable for processing by Graph Neural Networks (GNNs).

Figure 2 shows examples of functional connectivity matrices from four distinct datasets. Notable differences are evident, primarily resulting from variations in scanner equipment and acquisition protocols. The most apparent differences are in spatial resolution and signal amplitude.

Data Augmentation
HFMCA relies on augmentations that **subsample the original data**, such that each augmentation represents a separate, low-level feature. We chose three augmentations that will mimic this behavior for graph data:

- *Random Walk Sampling* - generates new training samples or subgraphs by simulating random walks on the network.
- *Node Dropping* - randomly removes certain nodes (and typically their adjacent edges) from the graph.
- *Feature Masking* - randomly masks (zeroes out) a subset of node features for some training instances.
- *Edge Removal* - randomly deletes a portion of the edges in the graph.

The augmentation function is randomly chosen L times with replacement for each data point, where L is a hyperparameter specifying the number of views in the HFMCA algorithm. After we preprocessed and augmented the data, it is ready to be passed to the model.

2.4 Network Architecture

We adopt a Transformer based *GPS* architecture for graph-structured data, as proposed in [19]. *GPS* extends the transformer model to handle graphs by operating over nodes and their neighborhoods, incorporating specialized graph positional encodings to preserve the structural information important for connectivity based domains such as brain networks. The architecture combines local neighborhood information from Message-Passing Graph Neural Networks (`MPNN`) and global information with Transformer-style attention (`ATTN`). As the `MPNN` layer, we will utilize a powerful *GinConv* layer, as defined in [29]. For the `ATTN` operator, we will employ the attention layer from [26].

Random Walk Positional Encodings
To improve the representation power of Graph Transformer layers, Random Walk Positional Encodings (RWPE) can be injected into the node features. RWPEs assign each node a vector that encodes its position within the graph based on random walk landing probabilities, providing a unique embedding that captures global structural relationships between nodes. It can be compared to the positional encodings in text. Original features with concatenated RWPEs are then passed further to the Transformer network.

GPS Graph Transformer
Let us consider n initial node embeddings $x \in \mathbb{R}^n$, which are the rows of the functional connectivity matrix, and the adjacency matrix A constructed from it. We will use x as a 0th layer hidden representation, i.e., $h^0 = x$. At each layer, feature updates are obtained by combining the outputs from the MPNN and global attention components. Both `MPNN` and `ATTN` serve as modular blocks;

specifically, MPNN is a *GinConv* layer [29], and ATTN is a classic fully-connected attention layer [26]. To avoid vanishing/exploding gradient problems, we use skip connections and normalization layers:

$$\begin{aligned} h_M^{\ell+1} &= \texttt{NORM}\left(\texttt{MPNN}_e^{\ell}\left(h^{\ell}, A\right) + h^{\ell}\right) \\ h_T^{\ell+1} &= \texttt{NORM}\left(\texttt{ATTN}^{\ell}\left(h^{\ell}\right) + h^{\ell}\right) \\ h^{\ell+1} &= \texttt{NORM}\left(h_M^{\ell+1} + h_T^{\ell+1} + \texttt{MLP}^{\ell}\left(h_M^{\ell+1} + h_T^{\ell+1}\right)\right) \end{aligned} \tag{8}$$

The node embeddings from the last layer are aggregated using the global mean:

$$h_G = \frac{1}{|V|}\sum_{i=1}^{|V|} \mathbf{h}_i^L.$$

This global representation h_G is then passed to the projector for HFMCA pretraining or an MLP for downstream prediction tasks.

3 Experiments

3.1 HFMCA Pretraining

The GPS encoder was equipped with two projector heads required for the HFMCA scheme. The model was trained for 200 epochs using the Adam optimizer [14] and a cosine decay learning rate schedule [16]. We used a learning rate of 1×10^{-3}, a weight decay of 1×10^{-6}, and a batch size of 256.

The encoders trained with baseline methods (*SimCLR*, *Barlow Twins*, or *VICReg*) and *HFMCA* were pretrained with REST [8] and ABIDE [9] datasets, amounting to the total of 2005 data points. We also include a randomly initialized encoder (*Baseline*) to verify that any sort of pretraining has an effect on the encoder. After each pretraining session, the projectors were discarded. Linear heads were then attached to the encoder outputs, and the models were fine-tuned on their respective classification or regression tasks.

3.2 Fine-tuning

Following the approach of [3, 17, 30], we first train a linear classifier on top of the frozen representations produced by our pretrained backbone across various datasets. To further assess the fine-tuning capabilities of our pretrained models, we repeat the experiments with the backbone unfrozen, allowing its weights to be updated during training.

For experimental setups with a frozen encoder, we trained the output linear layer using the Adam optimizer [14], with a learning rate of 1×10^{-2} and a batch size of 256. For regression tasks, the learning rate was reduced to 1×10^{-3}.

When the encoder was unfrozen, both the encoder and the output layer were trained jointly with Adam (learning rate 1×10^{-3}, batch size 256).

We employed nested cross-validation, using a stratified 5-fold split for each task. For each outer split, we further divided the four training folds into inner training and validation sets. We ran the training algorithm with early stopping (patience of 50 epochs) on the inner split to determine the optimal number of training epochs. Once identified, the model was retrained on all four training folds and evaluated on the held-out fifth fold. We ran this 10 times for each experimental setup, each time with different seeds, and reported means and standard variations.

Model	REST (MDD)	REST (Sex)	ABIDE	BSNIP	AOMIC (Sex)	HCP (Sex)
Majority class	51.6 ± 0.0	61.0 ± 0.0	53.6 ± 0.0	27.1 ± 0.0	51.9 ± 0.0	55.5 ± 0.0
Baseline$_F$	55.5 ± 1.0	64.8 ± 1.1	53.5 ± 1.6	30.7 ± 0.4	56.9 ± 1.3	63.0 ± 2.4
VICReg$_F$	56.6 ± 1.0	65.4 ± 0.5	53.5 ± 1.3	29.5 ± 0.6	60.0 ± 1.5	65.7 ± 1.5
BT$_F$	**57.3 ± 0.6**	64.7 ± 0.7	53.5 ± 1.7	29.5 ± 0.3	**60.9 ± 0.8**	66.0 ± 1.5
SimCLR$_F$	55.9 ± 0.9	**65.9 ± 0.7**	54.0 ± 1.0	30.9 ± 0.3	59.2 ± 1.5	**66.6 ± 1.5**
HFMCA$_F$	**57.4 ± 0.9**	**65.7 ± 1.0**	**54.7 ± 1.5**	**31.4 ± 0.4**	59.4 ± 1.1	66.0 ± 1.9

Table 1: The table presents the mean accuracies of frozen encoders, each paired with a linear classification head, after pretraining with different techniques. Underlined dataset names indicate 0-shot data that was not encountered by the models during pretraining. The model pretrained with HFMCA demonstrates competitive performance with all baseline techniques across both seen and unseen datasets.

3.3 Quality of HFMCA Embeddings

$\mathcal{H}_1$: *The HFMCA pretrained encoder produces high-quality graph embeddings that effectively capture semantic information relevant for fMRI graph classification tasks.*

To assess the quality of graph embeddings generated by our HFMCA pretrained encoder, we trained a single-layer MLP classifier on top of these embeddings using a variety of fMRI datasets, including REST [8], ABIDE [9], BSNIP [23], AOMIC [22], HCP [25], and ADHD200 [4]. The evaluation tasks spanned binary classification (sex, MDD, ASD), multi-class classification (bipolar/schizophrenia), and age prediction (regression).

Table 1 shows that, with a linear classifier trained on frozen embeddings, HFMCA achieves the highest or second-highest accuracy in five out of six classification tasks, consistently outperforming or matching established baselines.

Model	REST	ABIDE	ADHD200	AOMIC
Baseline$_F$	15.3 ± 0.2	7.2 ± 0.1	5.6 ± 0.1	**2.2 ± 0.0**
VICReg$_F$	15.7 ± 0.1	7.4 ± 0.1	**5.4 ± 0.0**	**2.2 ± 0.0**
BT$_F$	16.0 ± 0.1	7.4 ± 0.1	**5.4 ± 0.1**	**2.2 ± 0.0**
SimCLR$_F$	15.0 ± 0.1	**7.1 ± 0.1**	**5.4 ± 0.1**	**2.2 ± 0.0**
HFMCA$_F$	**14.1 ± 0.1**	**7.1 ± 0.1**	**5.3 ± 0.1**	**2.2 ± 0.0**

Table 2: The table presents the mean MAE scores for age prediction of frozen encoders. Each encoder is paired with a linear regression head, after pretraining with different techniques. Underlined dataset names indicate 0-shot tasks that were not encountered by the models during pretraining. The model pretrained with HFMCA demonstrates competitive performance with all baseline techniques across both seen and unseen datasets.

Although improvements are sometimes modest, these results highlight the robustness of HFMCA across diverse datasets. For regression tasks (Table 2), we observe similar trends, with HFMCA equaling or outperforming competing methods.

When the encoder is unfrozen (Table 3), HFMCA maintains strong performance, ranking first or second across all benchmarks, which further underscores the robustness and generalizability of its learned representations.

These findings support $\mathcal{H}_1$, demonstrating that the HFMCA-pretrained encoder generates graph embeddings that are highly informative and transferable for downstream fMRI analysis tasks.

3.4 Transfer Learning

$\mathcal{H}_2$: *The HFMCA pretrained encoder produces graph embeddings that allow effective transfer learning across different fMRI graph classification tasks.*

The BSNIP [23], AOMIC [22], ADHD200 [4], and HCP [25] datasets were not seen by the model during pretraining. To assess whether transfer of knowledge has occurred, we compare the performance of the pretrained HFMCA encoder to a randomly initialized baseline on these datasets (Tables 1, 2, and 3).

Table 1 shows that the pretrained, frozen HFMCA model performs better than the random baseline on all three classification datasets: BSNIP, AOMIC, and HCP. The most significant accuracy improvement is seen on the AOMIC dataset, while only a marginal increase is observed on BSNIP. Notably, the HFMCA model consistently exhibits a smaller standard deviation across ten runs, indicating greater stability in its performance. These results support the hypothesis that pretrained encoder embeddings are superior to random embeddings in linear classification settings.

For the regression task of age prediction on ADHD200 and AOMIC (Table 2), the HFMCA embeddings show a clear advantage on the ADHD200 dataset.

Model	REST (MDD)	REST (Sex)	ABIDE	BSNIP	AOMIC (Sex)	HCP (Sex)
Majority class	51.6 ± 0.0	61.0 ± 0.0	53.6 ± 0.0	27.1 ± 0.0	51.9 ± 0.0	55.5 ± 0.0
Baseline	58.8 ± 1.2	68.3 ± 0.4	54.9 ± 1.8	**30.6 ± 0.8**	62.5 ± 1.8	**71.1 ± 2.2**
VICReg	58.7 ± 1.0	65.5 ± 1.4	52.9 ± 1.7	30.1 ± 0.4	60.5 ± 1.7	66.0 ± 3.2
BT	**59.9 ± 1.4**	65.2 ± 1.5	53.5 ± 1.0	30.0 ± 0.4	60.3 ± 1.4	66.1 ± 1.1
SimCLR	59.3 ± 1.2	68.4 ± 0.9	**56.3 ± 1.3**	**30.5 ± 0.8**	63.7 ± 1.8	68.2 ± 3.1
HFMCA	**59.8 ± 1.2**	**70.3 ± 1.0**	**56.1 ± 1.6**	30.3 ± 0.7	**64.6 ± 1.4**	70.2 ± 0.6

Table 3: The table presents the mean accuracies of unfrozen encoders with linear classification heads, where the encoders were pretrained using various techniques. Underlined dataset names indicate 0-shot data that was not encountered by the models during pretraining. The HFMCA pretrained encoder demonstrates competitive performance relative to the baselines, with particularly notable increases in accuracy observed on the REST and AOMIC datasets.

However, on the AOMIC dataset, all models—including the baseline—achieve similar results, suggesting limited transfer benefit in this case.

When the encoder is unfrozen (Table 3), performance differences become less pronounced. On two out of three classification datasets (BSNIP and HCP), the randomly initialized encoder slightly outperforms the HFMCA pretrained encoder, although the latter's results remain within the standard deviation of the baseline. On the AOMIC dataset, the HFMCA encoder maintains a notable advantage.

Across all evaluations, the HFMCA model demonstrates more consistent results, as reflected in smaller or comparable standard deviations of accuracy over ten runs.

To summarize, the HFMCA pretrained encoder generally produces embeddings that outperform those from random initialization, particularly in frozen linear evaluation. However, its benefit diminishes or becomes dataset-dependent when the encoder is finetuned, suggesting that transferability may vary depending on the downstream task and dataset.

3.5 Data-Performance Scaling

$\mathcal{H}_3$: *The quality of HFMCA pretrained encoder embeddings improves with the size of the pretraining dataset.*

To investigate whether increasing pretraining dataset size improves the quality of HFMCA embeddings, we pretrained four HFMCA encoders using four different datasets:

1. REST (1313 data points)

Model	REST (MDD)	REST (Sex)	ABIDE	BSNIP	AOMIC (Sex)	HCP (Sex)
Majority class	51.6 ± 0.0	61.0 ± 0.0	53.6 ± 0.0	27.1 ± 0.0	51.9 ± 0.0	55.5 ± 0.0
R_F	57.5 ± 1.0	65.3 ± 0.4	51.7 ± 1.2	31.0 ± 0.5	59.0 ± 1.4	65.9 ± 2.2
RA_F	57.4 ± 0.9	65.7 ± 1.0	**54.7 ± 1.5**	**31.4 ± 0.4**	59.4 ± 1.1	66.0 ± 1.9
RAH_F	57.3 ± 0.9	66.0 ± 0.7	54.0 ± 1.1	**31.4 ± 0.7**	57.8 ± 2.2	64.2 ± 2.0
$RAHB_F$	**58.2 ± 0.6**	**66.6 ± 1.0**	53.4 ± 1.9	30.9 ± 0.6	**60.8 ± 2.7**	**66.6 ± 1.8**

Table 4: Linear classifier with frozen encoders pretrained on different volumes of data and fine-tuned for classification tasks. Combination of letters signify different datasets used for pretraining: R - REST, A - ABIDE, H - HCP, B - BSNIP.

Model	ABIDE	ADHD200	AOMIC	REST
R Freeze	**7.1 ± 0.1**	**5.2 ± 0.1**	**2.2 ± 0.0**	14.2 ± 0.1
RA Freeze	**7.1 ± 0.1**	5.3 ± 0.1	**2.2 ± 0.0**	14.1 ± 0.1
RAH Freeze	**7.0 ± 0.1**	5.3 ± 0.1	**2.2 ± 0.0**	14.1 ± 0.1
RAHB Freeze	**7.0 ± 0.1**	**5.1 ± 0.1**	**2.2 ± 0.0**	**13.9 ± 0.1**

Table 5: Linear classifier with frozen encoders pretrained on different volumes of data and fine-tuned for regression tasks. Combination of letters signify different datasets used for pretraining: R - REST, A - ABIDE, H - HCP, B - BSNIP.

2. REST + ABIDE (2005 data points)
3. REST + ABIDE + HCP (2359 data points)
4. REST + ABIDE + HCP + BSNIP (4249 data points)

Then, we replicated the exact experimental setup used for $\mathcal{H}_1$, evaluating each frozen encoder with a single-layer MLP classifier on top. The classifiers were tested on the REST [8], ABIDE [9], BSNIP [23], AOMIC [22], HCP [25], and ADHD200 [4] datasets for binary and multiclass classification, as well as regression tasks.

For each encoder, we plotted its performance across all tasks. To empirically verify $\mathcal{H}_3$, we expected to observe a logarithmic or linear relationship between pretraining dataset size and downstream performance. We anticipated the largest performance increase after including the BSNIP dataset, as its size nearly matches the combined size of the previous datasets. The results are visualized in Figure 3.

Solid lines in the plots represent 0-shot scenarios—where the fine-tuning data was not previously seen by the model during pretraining—while dotted lines represent scenarios where the fine-tuning data was included in the pretraining corpus.

For classification tasks (left panel; higher is better), we do not observe a clear log- or linear scaling of accuracy with respect to data volume. On 4 out of 6 tasks, there is a modest increase in accuracy when incorporating the REST+ABIDE pretraining dataset. However, performance tends to decline after adding the HCP and BSNIP datasets. In the 0-shot scenario (AOMIC), we observe a slight improvement upon including the BSNIP dataset, potentially indicating that larger datasets could produce greater gains.

For regression tasks (right panel; lower is better), we observe small improvements in 3 out of 4 tasks, most noticeably for the REST age prediction task. However, there is no clear monotonic relationship between dataset size and performance.

These results align with recent findings in graph foundation models: in self-supervised learning for GNNs, pretraining schemes borrowed from language or vision domains can lead to a *negative transfer* of knowledge [5, 12, 28]. We attempted to mitigate this issue by using a Graph Transformer as our backbone, which, with its attention blocks, permits deeper architectures without oversmoothing. However, as shown, this modification alone is insufficient.

Notably, [28] demonstrates that while naive scaling of data volume can degrade encoder performance, carefully selecting pretraining samples can lead to better results. Consistent with this, we see performance improvements on the REST, BSNIP, AOMIC, and ABIDE tasks after pretraining on the REST + ABIDE combination.

In summary, our results suggest that simply increasing the size of the pretraining dataset does not guarantee improved performance for HFMCA encoders; the choice and composition of pretraining data appear to play a more critical role.

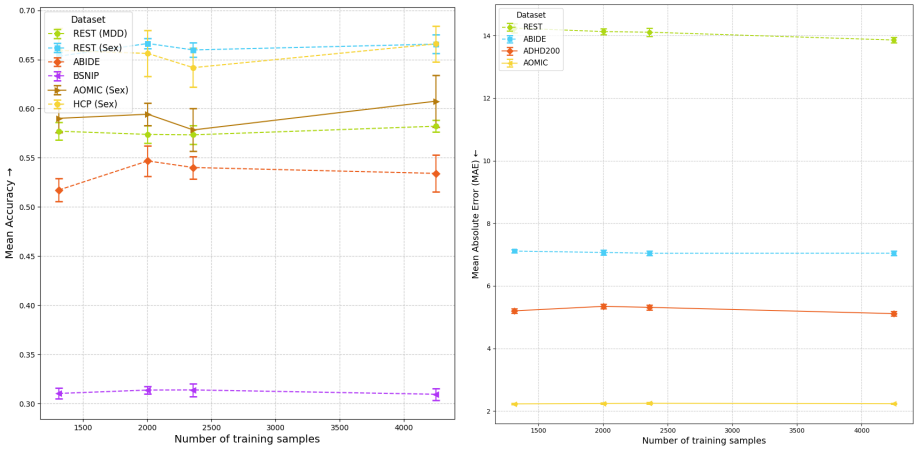

Fig. 3: Encoders were trained with HFMCA using varying amounts of pretraining data and evaluated on both classification and regression tasks. In the plot, dashed lines represent performance on datasets included during pretraining, while solid lines indicate performance in the zero-shot scenario. No clear linear relationship was observed between the amount of pretraining data and downstream performance, which aligns with findings from recent studies. This outcome may be the result of the introduction of noisy data during pretraining, which can reduce the encoder's overall effectiveness.

4 Discussion

4.1 Mutual Information and Comparison to Other Self-Supervised Methods

We can inspect HFMCA through the lens of information theory, where we can observe a clear link between HFMCA pretraining and other self-supervised methods mentioned in this paper.

Let us consider two Gaussian random variables X and Y. We can measure their independence through Mutual Information, which is equal to:

$$I(X;Y) = \frac{1}{2}\log\left(\frac{\det\sigma_X \det\sigma_Y}{\det\sigma_{XY}}\right) \tag{9}$$

Where σ_X and σ_Y are X and Y covariance matrices, and σ_{XY} is their joint covariance matrix. If we parametrize X and Y and choose to maximize the mutual information between them - for example if they represent two views of the same object - this is equivalent to minimizing its negative:

$$
\begin{aligned}
-I(X;Y) &= -\frac{1}{2}\log\left(\frac{\det\sigma_X \det\sigma_Y}{\det\sigma_{XY}}\right) \\
&= -\frac{1}{2}\left(\log\det\sigma_X + \log\det\sigma_Y - \log\det\sigma_{XY}\right) \\
&\propto \log\det\sigma_{XY} - \log\det\sigma_X - \log\det\sigma_Y
\end{aligned}
\tag{10}
$$

Which is very similar to the equation 7. HFMCA protocol relaxes the Gaussian assumption on the random variables, and minimizing its loss 7 is equivalent to maximizing mutual information.

Techniques such as SimCLR [7], Barlow Twins [30], and VICReg [3] all aim to learn informative, non-collapsed self-supervised representations by maximizing shared information between augmented views.

SimCLR does this indirectly through a contrastive InfoNCE loss that maximizes a lower bound on mutual information using positive and negative pairs.

In contrast, Barlow Twins and VICReg explicitly encourage disentangled, diverse features by penalizing redundancy and promoting decorrelation within embeddings—mechanisms closely related to maximizing the log-determinant of covariance matrices, which corresponds to maximizing mutual information under Gaussian assumptions.

The HFMCA loss, like the objectives used in VICReg, SimCLR, and Barlow Twins, is fundamentally connected to maximizing mutual information between augmented views—either directly, as in log-determinant or redundancy-reduction losses, or indirectly through contrastive objectives.

All these self-supervised training approaches are designed to produce informative, diverse representations by encouraging high shared information while preventing feature collapse, aligning their theoretical motivation with the maximization of mutual information between views.

4.2 Data Augmentation for Functional Connectivity Matrices

Figure 2 illustrates the variability of functional connectivity matrices across datasets. Because these matrices are converted into adjacency matrices using a simple thresholding approach, the resulting graphs can have similar structural patterns but at different resolutions.

We hypothesize that downsampling functional connectivity matrices - reducing their resolution - could serve as a useful augmentation method. This approach may facilitate incorporating new datasets into HFMCA training and encourage the model to develop scale-invariance.

Another issue in data collection is that fMRI datasets, even when processed with a consistent atlas, often permute the order of ROIs, which alters the structure of the connectivity matrices. While GNNs are theoretically permutation invariant, this property is broken when the connectivity matrix itself is used as node features (i.e., the input embeddings). Introducing node permutation as a

data augmentation strategy may help the model become robust to such variations, restoring permutation invariance and improving its ability to recognize underlying activity patterns.

In summary, employing downsampling and node permutation as data augmentation techniques could enhance the generalizability and robustness of models trained on functional connectivity data.

5 Conclusions

In this work, we addressed key challenges in deep learning for fMRI data by introducing a self-supervised Graph Transformer encoder, pretrained with the Hierarchical Functional Maximal Correlation Algorithm (HFMCA). Our experiments demonstrate the effectiveness of this approach across diverse resting-state fMRI datasets and downstream neuroimaging tasks.

First, we showed that embeddings from the HFMCA-pretrained encoder are highly informative and robust for various binary and multi-class classification and regression tasks, supporting our first hypothesis ($\mathcal{H}_1$). These embeddings consistently matched or exceeded strong baselines in most benchmark evaluations, underscoring their utility for fMRI graph analysis.

Second, our results support the transferability of the pretrained HFMCA encoder to novel, unseen datasets ($\mathcal{H}_2$). In frozen linear evaluation settings, the pretrained encoder provided more stable and often superior performance compared to random initialization. However, this advantage diminished with encoder fine-tuning and partly depended on the characteristics of the target dataset and task.

Third, regarding the impact of pretraining dataset size ($\mathcal{H}_3$), we did not observe a consistent positive scaling relationship between data volume and downstream performance. While certain combinations yielded modest improvements, indiscriminate scaling sometimes resulted in negative transfer, aligning with recent observations in graph representation learning.

Overall, our findings highlight the potential of HFMCA-based pretraining for self-supervised representation learning in fMRI data. We establish its relationship to existing self-supervised techniques and empirically demonstrate its robust performance relative to alternative methods. These results indicate that HFMCA pretraining enables the learning of transferable, high-quality representations for fMRI graph data. Nevertheless, careful selection of pretraining data and downstream tasks is essential to minimize negative transfer. Future work should investigate strategies to mitigate negative transfer, such as advanced augmentations of functional connectivity matrices. We hope our approach catalyzes further research into scalable self-supervised methods for brain imaging and accelerates the development of more generalizable brain decoding models.

Acknowledgments. I would like to thank Shujian Yu for his regular supervision and invaluable guidance throughout this project. I am grateful to Guido van Wingen for lending his expertise in neuroscience, which supported the project's motivation and

helped define realistic expectations for our research. I also thank Bo Hu, the author of HFMCA, for his support in both the theoretical understanding and implementation of the algorithm. Additionally, I appreciate Qiang Li for providing access to the preprocessed ABIDE, BSNIP, and HCP datasets, which were essential for our experiments and analyses.

Bibliography

[1] Anand, A., Li, Y., Wang, Y., Wu, J., Gao, S., Bukhari, L., Mathews, V.P., Kalnin, A., Lowe, M.J.: Activity and connectivity of brain mood regulating circuit in depression: a functional magnetic resonance study. Biological Psychiatry **57**(10), 1079–1088 (May 15 2005), https://doi.org/10.1016/j.biopsych.2005.02.021

[2] Balestriero, R., Ibrahim, M., Sobal, V., Morcos, A., Shekhar, S., Goldstein, T., Bordes, F., Bardes, A., Mialon, G., Tian, Y., Schwarzschild, A., Wilson, A.G., Geiping, J., Garrido, Q., Fernandez, P., Bar, A., Pirsiavash, H., LeCun, Y., Goldblum, M.: A cookbook of self-supervised learning (2023), URL https://arxiv.org/abs/2304.12210

[3] Bardes, A., Ponce, J., LeCun, Y.: Vicreg: Variance-invariance-covariance regularization for self-supervised learning (2022), URL https://arxiv.org/abs/2105.04906

[4] Bellec, P., Chu, C., Chouinard-Decorte, F., Benhajali, Y., Margulies, D.S., Craddock, R.C.: The neuro bureau adhd-200 preprocessed repository. NeuroImage **144, Part B**, 275–286 (2017), https://doi.org/10.1016/j.neuroimage.2016.06.034

[5] Cao, Y., Xu, J., Yang, C., Wang, J., Zhang, Y., Wang, C., Chen, L., Yang, Y.: When to pre-train graph neural networks? from data generation perspective! (2023), URL https://arxiv.org/abs/2303.16458

[6] Caro, J.O., de Oliveira Fonseca, A.H., Rizvi, S.A., Rosati, M., Averill, C., Cross, J.L., Mittal, P., Zappala, E., Dhodapkar, R.M., Abdallah, C., van Dijk, D.: BrainLM: A foundation model for brain activity recordings. In: The Twelfth International Conference on Learning Representations (2024), URL https://openreview.net/forum?id=RwI7ZEfR27

[7] Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations (2020), URL https://arxiv.org/abs/2002.05709

[8] Chen, X., Lu, B., Li, H.X., Li, X.Y., Wang, Y.W., Castellanos, F.X., Cao, L.P., Chen, N.X., Chen, W., Cheng, Y.Q., Cui, S.X., Deng, Z.Y., Fang, Y.R., Gong, Q.Y., Guo, W.B., Hu, Z.J., Kuang, L., Li, B.J., Li, L., Li, T., Lian, T., Liao, Y.F., Liu, Y.S., Liu, Z.N., Lu, J.P., Luo, Q.H., Meng, H.Q., Peng, D.H., Qiu, J., Shen, Y.D., Si, T.M., Tang, Y.Q., Wang, C.Y., Wang, F., Wang, H.N., Wang, K., Wang, X., Wang, Y., Wang, Z.H., Wu, X.P., Xie, C.M., Xie, G.R., Xie, P., Xu, X.F., Yang, H., Yang, J., Yao, S.Q., Yu, Y.Q., Yuan, Y.G., Zhang, K.R., Zhang, W., Zhang, Z.J., Zhu, J.J., Zuo, X.N., Zhao, J.P., Zang, Y.F., consortium, D., Yan, C.G.: The direct consortium and the rest-meta-mdd project: towards neuroimaging biomarkers of major depressive disorder. Psychoradiology **2**(1), 32–42 (2022), https://doi.org/10.1093/psyrad/kkac005

[9] Craddock, C., Benhajali, Y., Chu, C., Chouinard, F., Evans, A., Jakab, A., Khundrakpam, B., Lewis, J.D., Li, Q., Milham, M., Yan, C., Bellec,

P.: The neuro bureau preprocessing initiative: open sharing of preprocessed neuroimaging data and derivatives. Neuroinformatics (2013)
[10] de Kwaasteniet, B., Ruhe, E., Caan, M., Rive, M., Olabarriaga, S., Groefsema, M., Heesink, L., van Wingen, G., Denys, D.: Relation between structural and functional connectivity in major depressive disorder. Biological Psychiatry **74**(1), 40–47 (2013), ISSN 0006-3223, https://doi.org/https://doi.org/10.1016/j.biopsych.2012.12.024, URL https://www.sciencedirect.com/science/article/pii/S0006322313000401, sources of Treatment Resistance in Depression: Inflammation and Functional Connectivity
[11] Hu, B., Bu, Y., Príncipe, J.: Learning orthonormal features in self-supervised learning using functional maximal correlation. pp. 472–478 (10 2024), https://doi.org/10.1109/ICIP51287.2024.10648197
[12] Hu, W., Liu, B., Gomes, J., Zitnik, M., Liang, P., Pande, V., Leskovec, J.: Strategies for pre-training graph neural networks (2020), URL https://arxiv.org/abs/1905.12265
[13] Kennedy, D.P., Courchesne, E.: The intrinsic functional organization of the brain is altered in autism. NeuroImage **39**(4), 1877–1885 (2008), ISSN 1053-8119, https://doi.org/https://doi.org/10.1016/j.neuroimage.2007.10.052, URL https://www.sciencedirect.com/science/article/pii/S1053811907009950
[14] Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization (2017), URL https://arxiv.org/abs/1412.6980
[15] Liang, M., Zhou, Y., Jiang, T., Liu, Z., Tian, L., Liu, H., Hao, Y.: Widespread functional disconnectivity in schizophrenia with resting-state functional magnetic resonance imaging. Neuroreport **17**(2), 209–213 (Feb 2006), https://doi.org/10.1097/01.wnr.0000198434.06518.b8
[16] Loshchilov, I., Hutter, F.: Sgdr: Stochastic gradient descent with warm restarts (2017), URL https://arxiv.org/abs/1608.03983
[17] Misra, I., van der Maaten, L.: Self-supervised learning of pretext-invariant representations (2019), URL https://arxiv.org/abs/1912.01991
[18] Monk, C.S., Peltier, S.J., Wiggins, J.L., Weng, S.J., Carrasco, M., Risi, S., Lord, C.: Abnormalities of intrinsic functional connectivity in autism spectrum disorders,. NeuroImage **47**(2), 764–772 (2009), ISSN 1053-8119, https://doi.org/https://doi.org/10.1016/j.neuroimage.2009.04.069, URL https://www.sciencedirect.com/science/article/pii/S1053811909004327
[19] Rampášek, L., Galkin, M., Dwivedi, V.P., Luu, A.T., Wolf, G., Beaini, D.: Recipe for a general, powerful, scalable graph transformer (2023), URL https://arxiv.org/abs/2205.12454
[20] Sheffield, J.M., Barch, D.M.: Cognition and resting-state functional connectivity in schizophrenia. Neuroscience & Biobehavioral Reviews **61**, 108–120 (Feb 2016), https://doi.org/10.1016/j.neubiorev.2015.12.007, epub 2015 Dec 14
[21] Shi, C., Wang, Y., Wu, Y., Chen, S., Hu, R., Zhang, M., Qiu, B., Wang, X.: Self-supervised pretraining improves the performance of classification of

task functional magnetic resonance imaging. Frontiers in Neuroscience **17**, 1199312 (Jun 2023), https://doi.org/10.3389/fnins.2023.1199312

[22] Snoek, L., v.d.M.M.M.B.T.V.D.L.A.E.A..S.H.S.: The amsterdam open mri collection, a set of multimodal mri datasets for individual difference analyses. Scientific Data **8**(1), 1–23 (2021), https://doi.org/10.1038/s41597-021-00986-6

[23] Tamminga, C.A., Pearlson, G., Keshavan, M., Sweeney, J., Clementz, B., Thaker, G.: Bipolar and schizophrenia network for intermediate phenotypes: Outcomes across the psychosis continuum. Schizophrenia Bulletin **40**(Suppl2), S131–S137 (02 2014), ISSN 0586-7614, https://doi.org/10.1093/schbul/sbt179, URL https://doi.org/10.1093/schbul/sbt179

[24] Tzourio-Mazoyer, N., Landeau, B., Papathanassiou, D., Crivello, F., Etard, O., Delcroix, N., Mazoyer, B., Joliot, M.: Automated anatomical labeling of activations in spm using a macroscopic anatomical parcellation of the MNI MRI single-subject brain. NeuroImage **15**(1), 273–289 (2002), https://doi.org/10.1006/nimg.2001.0978

[25] Van Essen, D., Ugurbil, K., Auerbach, E., Barch, D., Behrens, T., Bucholz, R., Chang, A., Chen, L., Corbetta, M., Curtiss, S., Della Penna, S., Feinberg, D., Glasser, M., Harel, N., Heath, A., Larson-Prior, L., Marcus, D., Michalareas, G., Moeller, S., Oostenveld, R., Petersen, S., Prior, F., Schlaggar, B., Smith, S., Snyder, A., Xu, J., Yacoub, E.: The human connectome project: A data acquisition perspective. NeuroImage **62**(4), 2222–2231 (2012), ISSN 1053-8119, https://doi.org/https://doi.org/10.1016/j.neuroimage.2012.02.018, URL https://www.sciencedirect.com/science/article/pii/S1053811912001954, connectivity

[26] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need (2023), URL https://arxiv.org/abs/1706.03762

[27] Wang, X., Chu, Y., Wang, Q., Cao, L., Qiao, L., Zhang, L., Liu, M.: Unsupervised contrastive graph learning for resting-state functional mri analysis and brain disorder detection. Human Brain Mapping **44**(17), 5672–5692 (Dec 2023), https://doi.org/10.1002/hbm.26469, epub 2023 Sep 5

[28] Xu, J., Huang, R., Jiang, X., Cao, Y., Yang, C., Wang, C., Yang, Y.: Better with less: A data-active perspective on pre-training graph neural networks (2023), URL https://arxiv.org/abs/2311.01038

[29] Xu, K., Hu, W., Leskovec, J., Jegelka, S.: How powerful are graph neural networks? (2019), URL https://arxiv.org/abs/1810.00826

[30] Zbontar, J., Jing, L., Misra, I., LeCun, Y., Deny, S.: Barlow twins: Self-supervised learning via redundancy reduction (2021), URL https://arxiv.org/abs/2103.03230

[31] Zhuang, F., Qi, Z., Duan, K., Xi, D., Zhu, Y., Zhu, H., Xiong, H., He, Q.: A comprehensive survey on transfer learning. Proceedings of the IEEE **109**(1), 43–76 (2021), https://doi.org/10.1109/JPROC.2020.3004555