applied sciences

MDPI

Article

Self-Supervised Representation Learning for EEG-Based Detection of Neurodegenerative Diseases

Federico Del Pup [1,2], Louis Fabrice Tshimanga [2], Andrea Zanola [2,3], Luca Taffarello [2], Elisa Tentori [2,4] and Manfredo Atzori [2,3,5,*]

[1] Department of Information Engineering, University of Padua, 35131 Padua, Italy; federico.delpup@studenti.unipd.it
[2] Padova Neuroscience Center, University of Padua, 35129 Padua, Italy; louisfabrice.tshimanga@unipd.it (L.F.T.); andrea.zanola@studenti.unipd.it (A.Z.); luca.taffarello@studenti.unipd.it (L.T.); elisa.tentori@unipd.it (E.T.)
[3] Department of Neuroscience, University of Padua, 35121 Padua, Italy
[4] Department of Biomedical Sciences, University of Padua, 35131 Padua, Italy
[5] Information Systems Institute, University of Applied Sciences Western Switzerland (HES-SO Valais), 3960 Sierre, Switzerland
* Correspondence: manfredo.atzori@unipd.it

Abstract

Electroencephalography (EEG) is an important noninvasive diagnostic tool for detecting neurodegenerative disorders. In this context, EEG-based deep learning models show promise due to their ability to capture nonlinear brain dynamics but often suffer from poor generalizability caused by high inter-subject variability. Self-supervised learning (SSL) offers a promising solution by enabling models to learn robust representations from large unlabeled datasets. This study introduces a double-masking representation learning framework for EEG analysis. Using data aggregated from eight multi-center datasets (3156 subjects; 439 h of EEG recordings), a hybrid convolutional-transformer model (TransformEEG) is pretrained to enhance generalization in neurodegenerative disease classification, focusing on Parkinson's and Alzheimer's diseases. This approach combines phase-swap data augmentation, designed to facilitate the learning of EEG phase-amplitude coupling, with a double-masking function that operates at both the signal and the transformer's token levels. The pretrained model was evaluated against a fully supervised baseline using Monte Carlo cross-validation with 100 splits across three public pathology detection datasets. Pretraining led to consistent improvements in both median balanced accuracy and Inter-Quartile Range (IQR) across all fine-tuning datasets. Compared to the fully supervised baseline, the proposed approach increases median balanced accuracy by 2.8% for Parkinson's disease detection and by 4.2% for Alzheimer's disease detection, while also reducing performance variability across 100 Monte Carlo splits. These results demonstrate that SSL can enhance EEG deep learning performance, though achieving robust generalization in clinical applications remains an open research challenge.

Keywords: EEG; self-supervised learning; contrastive learning; deep learning; artificial intelligence; transformer; Alzheimer's disease; Parkinson's disease

check for updates

Academic Editor: Jesús B. Alonso-Hernández

Received: 11 November 2025
Revised: 2 December 2025
Accepted: 12 December 2025
Published: 18 December 2025

Citation: Del Pup, F.; Tshimanga, L.F.; Zanola, A.; Taffarello, L.; Tentori, E.; Atzori, M. Self-Supervised Representation Learning for EEG-Based Detection of Neurodegenerative Diseases. *Appl. Sci.* **2025**, *15*, 13275. https://doi.org/10.3390/app152413275

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Electroencephalography (EEG) is a non-invasive, low-cost technique commonly used to examine neural activity. Due to its accessibility, EEG has become integral to various applications across both the medical and industrial sectors. Common applications includes

emotion recognition [1], sleep staging [2], epilepsy detection [3], and brain-computer interfaces [4].

In clinical settings, EEG is considered an essential diagnostic tool for supporting the diagnosis of neurodegenerative disorders, such as Parkinson's (PD) and Alzheimer's disease (AD). Research has shown that the onset of these diseases is associated with alterations in normal oscillatory brain activity across specific frequency bands. For instance, cognitive impairments in Parkinson's disease correlate with EEG "slowing" characterized by decreased spectral power in the alpha and beta bands and increased spectral power in the delta and theta bands [5]. As a result, detecting alterations in brain dynamics and extracting quantitative EEG measures as biomarkers for disease severity and progression has become a critical focus in supporting early detection of these pathologies.

Analyzing EEG recordings presents significant challenges due to the complexity of neural activity, which is often marked by intricate non-linear patterns. Moreover, brain dynamics can be heavily influenced by both biological and non-biological artifacts (such as eye blinks, muscle activity, and line noise), all of which degrade signal quality and lower the Signal-to-Noise Ratio (SNR) [6]. These factors make it difficult to extract meaningful, disease-specific signatures using traditional statistical methods or conventional machine learning approaches. In this context, deep learning has proven to be a valuable tool, achieving state-of-the-art results in various clinical tasks. This success is largely attributed to the growing availability of diverse datasets, many of which are accessible through platforms like OpenNeuro (https://openneuro.org/, accessed on 10 November 2025) [7].

Despite the impressive accuracies reported in many studies, EEG deep learning models often show poor generalizability when properly evaluated. For instance, the study in [8] showed that models might rely on inherent biometric features of the EEG signal as shortcuts to reduce training loss. However, this approach can negatively affects the model's ability to generalize, resulting in decreased performance and greater variability. This observation is further corroborated by other research, which highlights the need to split data at the subject level to ensure fair model evaluation and assess generalizability [9,10].

To address generalizability issues, efforts have been made to design novel architectures optimized for EEG data, incorporating advanced neural layers such as transformers. These architectures come with an increased parameter count, but they have demonstrated an improved ability to extract richer representations that capture both local and global patterns within the EEG signal. However, disease-specific features can only be extracted effectively if datasets with a sufficiently large number of subjects are used. For example, in [11], a novel transformer model optimized for EEG analysis outperformed seven other established architectures in detecting Parkinson's disease using harmonized data from four different datasets. However, the study also showed that reducing the number of subject from 290 to 81 led to a drastic decrease in median performance and an increase in variability. This result highlights the importance of harmonizing and aggregating data from heterogeneous, multi-center datasets to achieve meaningful results.

Aggregating data from multiple datasets is crucial to training robust models with better generalizability. However, the availability of large labeled datasets depends heavily on the specific target task. In clinical applications, acquiring data from individuals affected by neurodegenerative diseases is a time-consuming process, subject to ethical regulations, and dependent on the availability of patients, technicians, and medical personnel. Moreover, not all diseases are equally represented. For example, there are more open datasets available for Parkinson's disease than for Frontotemporal dementia or other neurodegenerative conditions. This lack of data limits the efficient training of generalizable deep learning models.

To overcome this issue, self-supervised learning has recently emerged as a promising approach to improve model generalizability by learning meaningful representations from large unlabeled datasets [12]. The goal is to train neural backbone encoders to learn general-purpose representations by solving an auxiliary task (the pretext task) that generates supervisory signals directly from the data. The pretrained encoder can then be transferred to a new model with a custom classification or regression head for effective fine-tuning on a small labeled dataset.

Self-supervised learning has been widely adopted to enhance the efficacy of EEG deep learning models, and the fast progress in this field has been detailed in dedicated reviews [13,14]. These reviews highlight that contrastive learning has historically been one of the most widely explored SSL paradigms in EEG research [15,16]. However, recent work has demonstrated the strong potential of regularization-based representation learning approaches. These methods avoid the need for negative samples and instead rely on tailored regularization strategies that help the encoder learn invariant and physiologically meaningful representations. Regularization-based SSL has shown particular promise for EEG applications. For example, Mohammadi Foumani et al. introduced EEG2Rep [17], a self-supervised learning framework that reconstructs masked latent features using Variance-Invariance-Covariance (VICReg) regularization [18]. Such approaches offer a compelling alternative to contrastive learning, especially in EEG scenarios where obtaining diverse negative samples or applying aggressive augmentations can be challenging or physiologically inappropriate.

A common limitation of many published works is the lack of heterogeneity in the datasets used during pretraining. EEG deep learning models are often pretrained using a single dataset, sometimes even the same one used for fine-tuning. In clinical applications, the Temple University Hospital (TUH) EEG corpus is often used [19]. While the TUH EEG corpus is one of the largest available, it still consists of recordings from a single center. Aggregating data from multiple datasets, however, is crucial for capturing variability and improving model generalizability. In [13], it was stated that data aggregation could enhance model performance, even when recordings come from entirely different experimental settings.

Another limitation is how the evaluation of fine-tuned models is conducted. Improvements made by self-supervised methods are often assessed based on mean accuracy across multiple fine-tuning runs. However, the primary goal of self-supervised learning is to improve model generalizability and reduce result variability. Therefore, it is essential to properly evaluate these aspects.

Research is needed to address these limitations and improve the understanding of how self-supervised learning influences the generalizability of EEG deep learning models.

Contribution: this study introduces a double-masking representation learning method for EEG analysis. By aggregating data from eight different multi-center datasets, totaling 3156 subjects and 439 h of EEG recordings, a convolutional transformer model optimized for EEG data is pretrained to improve its generalizability in neurodegenerative disease classification tasks. This approach combines phase-swap data augmentation, designed to facilitate the learning of EEG phase-amplitude coupling, with a double-masking function that operates at both the signal and the transformer's token levels. The pretrained model is compared with its fully supervised counterpart. To provide reliable performance estimates, the evaluation is conducted on three publicly available datasets targeting pathology detection tasks, using a repeated holdout evaluation method based on 100 different splits.

2. Methods

This section describes, in detail, the key methodological aspects, in accordance with the checklist provided in [20]. In particular, it provides descriptions of the following elements:

- the selected datasets for the study.
- how raw data were preprocessed.
- the proposed self-supervised learning method.
- implementation details, including the data partitioning strategy, model architecture, pretraining and fine-tuning hyperparameters, and evaluation criteria.

2.1. Datasets Selection

A set of eight datasets was selected for the experiments. This set comprises EEG recordings from healthy individuals and patients with neurodegenerative diseases, including Alzheimer's disease and Parkinson's disease. Each recording has a sampling rate greater than 125 Hz and uses at least the minimum number of channels recommended by the American Clinical Neurophysiology Society (ACNS): 19 channels plus 2 contralateral reference channels [21].

Table 1 summarizes, for each dataset, the acquisition settings, demographic characteristics, number of samples (time windows), and how those samples are used in the self-supervised learning pipeline. In total, the eight datasets comprise 3156 subjects and 439 h of EEG recordings. The recordings are highly heterogeneous, reflecting data collected across multiple centers and varying in EEG cap configurations (e.g., reference, number of electrodes, sampling rate), experimental conditions, and demographic characteristics. Such heterogeneity is desirable, as it can improve the efficacy of self-supervised pretraining and the generalizability of the learned representations.

Table 1. Summary table of the selected datasets. Datasets downloaded from OpenNeuro are listed with their identification code (ds00xxxx). The condition column lists the diseases represented in the dataset, or "healthy" if it contains only healthy subjects. Acronyms are: PD = Parkinson's disease, AD = Alzheimer's disease, FTD = frontotemporal dementia. "Multiple" denotes a transdiagnostic dataset containing three or more distinct conditions.

Dataset	# Subj	Age (Mean ± SD)	Male %	Conditions	Reference	# Channels	Sampling Rate [Hz]	Duration (Hours)	# Samples *	Usage
ds004148	60	20.0 ± 1.9	46.7	healthy	FCZ	64	500	9.5	2880	pretrain
ds004504	88	66.2 ± 7.4	50.0	AD; FTD	A1-A2	19	500	19.2	5749	pretrain
ds002778	31	63.4 ± 8.8	45.2	PD	CMS/DRL	41	512	1.5	456	pretrain
ds004584	149	69.3 ± 8.0	63.1	PD	PZ	63	500	6.0	1775	pretrain or fine-tune
ds003490	2408	69.5 ± 9.1	64.0	PD	CPZ	64	500	50	8.0	pretrain or fine-tune
BrainLat	4639	62.7 ± 9.5	62.9	multiple	Common	128	512	125	15.5	pretrain or fine-tune
CAUEEG	90,287	70.7 ± 9.9	37.5	multiple	Common	19	200	1379	301.6	pretrain
TDBrain	23,708	38.51 ± 19.3	54.1	multiple	A1-A2	26	500	1274	77.2	pretrain

* Assuming EEG windows of 16 s and 25% of overlap.

A deep learning model pretrained on heterogeneous EEG signals can better handle inter-subject variability and other cross-dataset differences (such as instrumentation). Ignoring these sources of variability during pretraining can degrade fine-tuning performance because the model may struggle to adapt to shifts in the data distribution.

2.2. Data Preprocessing

The selected datasets were initially organized in the BIDS format, which enabled the use of an automated preprocessing and harmonization pipeline implemented in BIDSAlign (v1.0.0) [22], a MATLAB (2024b)-based library built on top of EEGLAB (2024.2.1.) [23].

The pipeline employed in this study builds upon the approach outlined in [6] and follows several crucial steps. Initially, non-EEG channels are excluded, and the first and last 8 s of each recording are discarded to avoid potential signal drift. The direct current (DC) offset is then removed from each EEG channel. To minimize computational demand, the recordings are resampled to 250 Hz, followed by a band-pass filter with passband edges at 1 Hz and 45 Hz.

Subsequently, independent components (ICs) are extracted using the runica algorithm, and noisy ICs are automatically discarded through ICLabel. Specifically, ICs classified as noise (e.g., muscle, eye, heart, line noise, or channel noise) with high confidence (90–10%) are removed, along with those categorized as brain activity with very low confidence (0–10%).

The EEG recordings are then re-referenced to the common average, and the channels are aligned to the 10/10 IFCN template. To meet GPU memory limitations, a subset of 32 channels is selected for further processing. The signals are then downsampled to 125 Hz to improve computational efficiency. Each EEG segment is standardized across channels using z-score normalization, and the data is finally divided into 16-s windows with a 25% overlap for analysis.

Due to the large volume of aggregated data, manual screening of EEG recordings was not feasible. For this reason, all subjects were retained, and only signal-level quality improvements were applied through the automated preprocessing pipeline. Corrupted recordings had already been removed by the dataset authors, and keeping all remaining subjects allowed us to preserve heterogeneity and support the generalizability of the learned representations. The "brainlat" recordings were already preprocessed with a similar intense pipeline. Therefore, only channel alignment, downsampling, standardization, and window extraction were considered in this study.

2.3. Self-Supervised Pretraining

Self-supervised learning is a deep learning approach designed to train neural backbone encoders on large amounts of unlabeled data to learn general-purpose representations that can be used for downstream (real-world) applications. This approach is especially useful when labeled data are scarce, such as in clinical applications. Pretraining is done by solving an auxiliary (pretext) task whose supervision signals come from the data itself. The pretrained encoder's weights can be transferred to a downstream model that adds a supervised head (e.g., for classification or regression) and is trained on the available labeled data.

Different categories of pretext tasks can be used to pretrain the backbone encoder. These categories are extensively described in [13]. Among them, contrastive learning frameworks are largely used in EEG applications. The contrastive objective is to learn representations by bringing augmented views of the same instance into agreement (positive pairs) while pushing apart views from different instances (negative pairs). Positive and negative samples are generated by applying data augmentations that preserve the underlying global structure of the data. These views are typically fed into a Siamese network (weight-sharing networks for two or more inputs) [24], and the representations are compared with carefully designed loss functions, like the *NT-Xent* (Normalized Temperature-Scaled Cross Entropy loss) used in SimCLR [25].

As noted in Section 1, recent work has demonstrated the strong potential of regularization-based representation learning approaches. These methods avoid the need for large batch sizes populated with many negative samples and instead rely on tailored regularization strategies to prevent representational collapse. The following section introduces a novel SSL framework that belongs to this family of regularization-based approaches.

Double-Masking Pretraining

Starting from the SimCLR contrastive learning algorithm, this study proposes a double-masking representation learning framework for EEG transformer models, as illustrated in Figure 1.

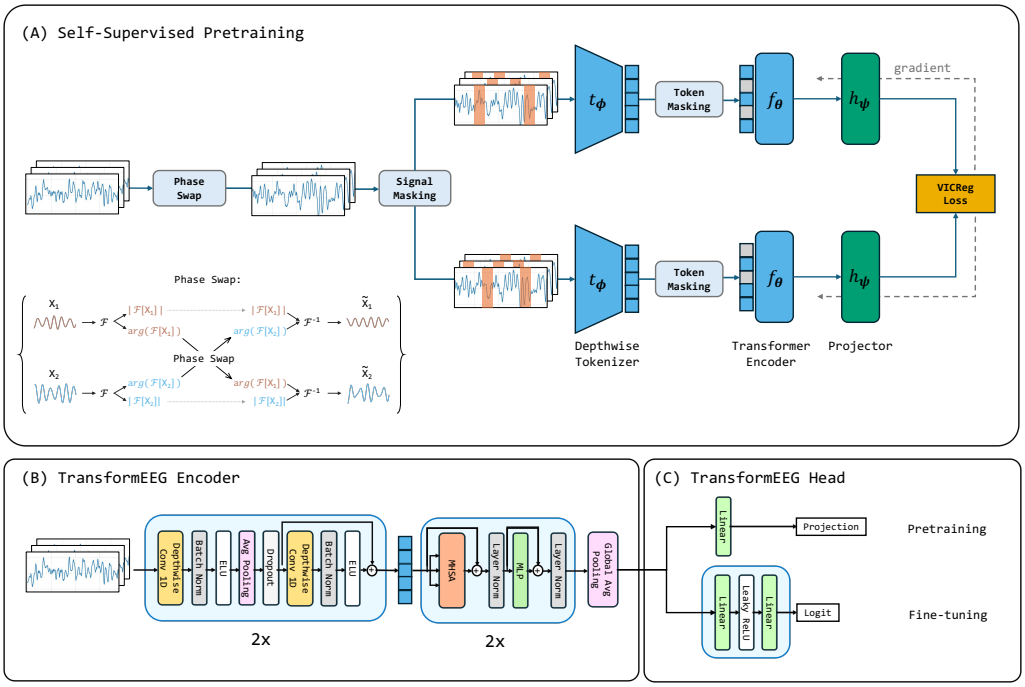


Figure 1. Schematic representation of the proposed double-masking representation learning framework. (**A**) The input batch is first augmented using the phase swap transformation (lower-left panel in (**A**)). Phase swap takes two signals, decomposes them in the frequency domain, and reconstructs them by preserving each signal's amplitude while exchanging their phase components, as formalized in Equation (1). A masking augmentation is then applied, zeroing out portions of the signals. The resulting masked views are passed through a tokenizer to produce sequences of tokens. For each view, a different random subset of tokens is masked using an attention mask. The representations are then mapped into a latent space by a projection head. Finally, the VICReg loss is computed between the two projected views. (**B**) The TransformEEG encoder, composed of a depthwise convolutional tokenizer followed by two stacked transformer encoder layers. (**C**) Heads used in pretraining and fine-tuning: a linear projection head for SSL and a classification head (MLP) for downstream tasks.

The input is a mini-batch $\mathrm{I} \in \mathbb{R}^{N \times C \times L}$ of EEG samples $\mathrm{X} \in \mathbb{R}^{C \times L}$, where:

- N is the batch size,
- C is the number of channels,
- L is the number of time steps, computed as $L = f_s \times S$, with f_s the sampling frequency in Hertz (125 Hz) and S the sequence length in seconds (16 s).

The input batch is first augmented using the phase swap transformation introduced in [26] and implemented in the Python package SelfEEG [27]. This augmentation combines the amplitude and phase components of biosignals from different sources to help the model learn their coupling. Specifically, the mini-batch is randomly divided into $N/2$ pairs. For each pair, the signals are decomposed in the frequency domain and then reconstructed by preserving their amplitude while exchanging their phase components. This operation can be analytically expressed as follows:

$$\tilde{\mathrm{X}}^i = \mathrm{Re}\left(\mathcal{F}^{-1}\left[|\mathcal{F}[\mathrm{X}^i]| \odot e^{\circ j \arg(\mathcal{F}[\mathrm{X}^j])}\right]\right) \quad (1)$$

where:

- $|\mathcal{F}[X^i]|$ is the amplitude of the EEG sample X^i,
- $\arg(\mathcal{F}[X^j])$ is the phase of the EEG sample X^j,
- $\odot$ is the Hadamard product, representing the element-wise product of two matrices.
- $e^{\circ X} = (e^{x_{ij}})$ is the Hadamard exponential, which computes the element-wise exponential of a matrix X.

The augmented batch is subsequently processed with an additional masking transformation that suppresses selected portions of the phase-swapped signals by setting them to zero. Mathematically, let p denote the fraction of samples to be masked and let k represent the number of suppressed temporal blocks. The resulting masked signal is then defined as:

$$\tilde{X} = X \odot (\mathbf{1}_C \mathbf{m}^T) \tag{2}$$

where $\mathbf{1}_C$ is a C-dimensional vector of ones and $\mathbf{m} \in \{0,1\}^L$ is a binary temporal mask. The mask vector $\mathbf{m}$ is obtained by alternating preserved and suppressed blocks:

$$\mathbf{m} = \mathbf{1}_{b_1} \oplus \mathbf{0}_{b_2} \oplus \mathbf{1}_{b_3} \oplus \mathbf{0}_{b_4} \oplus \cdots \oplus \mathbf{0}_{b_{2k}} \oplus \mathbf{1}_{b_{2k+1}} \tag{3}$$

where each b_i denotes the length of the i-th block. The total masked portion of the signal is obtained by summing the lengths of all zero-valued segments and is expressed as:

$$Lp = \sum_{i=2,4,6,\ldots,2k} b_i. \tag{4}$$

Two distinct masked views are generated by randomly selecting the number of masked segments $k \in \{3,4,5\}$ and the masking ratio $p \in [0.2, 0.35]$.

For computational efficiency, a single temporal masking pattern is generated for each view and then applied to all samples in the mini-batch, rather than generating a different mask for each sample.

The resulting masked signals are processed by the two branches of a Siamese network, consisting of the TransformEEG backbone encoder (tokenizer plus transformer) and a projection head.

The tokenizer extracts a sequence of EEG tokens from each masked input. Then, a second masking step is applied at the token level: for each view, a different random subset of 20% of the tokens is selected and masked. This is implemented using an attention mask (a boolean tensor) passed to the transformer encoder, which prevents the model from attending to the selected tokens. Therefore, each view uses a distinct attention mask, ensuring that different token subsets are masked in the two branches of the Siamese network. Unlike typical text-based approaches, the masked tokens are not replaced with learnable special tokens.

The output of the transformer encoder is converted into a batch of vector representations via global average pooling.

Each representation is then passed through the projection head that maps it into a latent space. While projection heads are typically multi-layer perceptrons with one hidden layer, batch normalization, and ReLU activation, this study uses a single linear layer. This architectural choice is based on preliminary experiments assessing embedding quality during pretraining across different projection head configurations.

Finally, the VICReg loss (Variance-Invariance-Covariance Regularization) is computed to compare the two projected views. Given two batches of projections $Z = [z_1, \ldots, z_N]^T \in \mathbb{R}^{N \times d}$ and $Z' = [z'_1, \ldots, z'_N]^T \in \mathbb{R}^{N \times d}$, the VICReg loss is defined as:

$$\mathcal{L} = \lambda \cdot \mathcal{L}_{\text{inv}} + \mu \cdot \mathcal{L}_{\text{var}} + \nu \cdot \mathcal{L}_{\text{cov}} \tag{5}$$

where:

- $\mathcal{L}_{\text{inv}}$ is the invariance term, which encourages projections of the two views of the same sample to be close:

$$\mathcal{L}_{\text{inv}} = \frac{1}{N} \sum_{i=1}^{N} ||z_i - z'_i||_2^2 \tag{6}$$

- $\mathcal{L}_{\text{var}}$ is the variance term, which encourages each dimension of the projection to have non-zero variance to avoid collapse:

$$\mathcal{L}_{\text{var}} = \frac{1}{d} \sum_{i=1}^{d} [\text{ReLU}(\gamma - \sigma_i(Z)) + \text{ReLU}(\gamma - \sigma_i(Z'))] \tag{7}$$

  with $\sigma_j(Z) = \sqrt{\text{Var}(z^j) + \epsilon}$ the standard deviation of the vector z^j composed of each value at dimension j in all vectors in Z (ϵ is added for numerical stability) and $\gamma = 1$ the target lower bound for the standard deviation in each dimension.
- $\mathcal{L}_{\text{cov}}$ is the covariance term, which encourages the dimensions of the projections to be decorrelated. It is defined as:

$$\mathcal{L}_{\text{cov}} = \frac{1}{d} \sum_{i \neq j} ([C_Z]_{i,j}^2 + [C_{Z'}]_{i,j}^2) \tag{8}$$

  with

$$C_Z = \frac{1}{N-1} \tilde{Z}^T \tilde{Z} \in \mathbb{R}^{d \times d} \tag{9}$$

  and $\tilde{Z}$ the zero-mean version of Z.

The use of VICReg places the proposed double-masking pretraining approach within the category of non-contrastive learning methods. As described in the original work, VICReg is a regularization-based approach designed to prevent representational collapse in the absence of negative samples by enforcing variance, invariance, and covariance constraints.

2.4. Implementation Details

The rest of the experiment was conducted within a *Python* environment. Models were trained using *SelfEEG* [27] (v0.2.1), a PyTorch-based [28] open-source library developed to assist researchers in conducting self-supervised learning experiments on EEG data. Figures were generated with *Seaborn* [29]. All experiments were run on the Department of Neuroscience computing node, equipped with 3 NVIDIA A30 GPU (NVIDIA Corporation, Santa Clara, CA, United States) devices (CUDA 12.2). The maximum memory allocation during pretraining was 23.15 GB.

2.4.1. Data Partitioning

To provide a fair evaluation of the proposed method, the datasets were partitioned to prevent bias during both pretraining and fine-tuning. Specifically, EEG samples from all datasets except the downstream-task dataset were used during pretraining. For example, if the model was evaluated on the "brainlat" dataset, all other datasets were used for pretraining, ensuring the exploitation of all the available data. During fine-tuning,

pretrained models were evaluated with a repeated holdout strategy, as schematized in Figure 2. The repeated holdout method, often called Monte Carlo cross-validation, consists of performing the holdout split multiple times at the subject level. At each split, random subsets of subjects are assigned to the training, validation, and test sets, ensuring that samples from the same subject appear in the same set. The model is fine-tuned on the training set, with early stopping based on the validation data, and evaluated on the test set. This analysis uses 100 stratified partitions, assigning 60% of the subjects to the training set, 20% to the validation set, and the remaining 20% to the test set. Stratification ensures that samples from all target classes appear in every set, while the large number of partitions ensures that all subjects appear at least once in the test set.

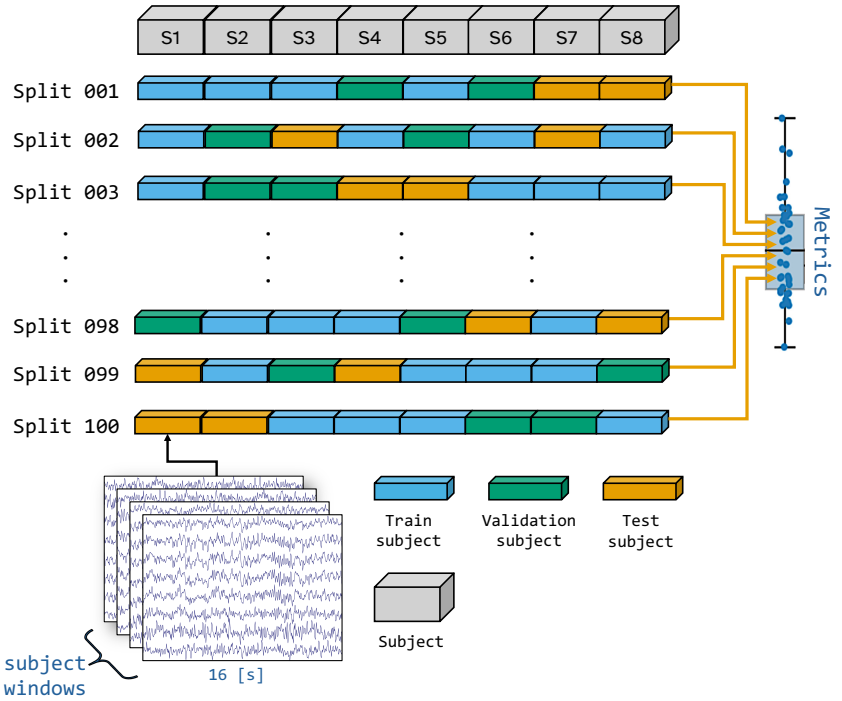


Figure 2. Schematic representation of the repeated holdout method used during fine-tuning. The repeated holdout method consists of performing the holdout split multiple times at the subject level. At each split, random subsets of subjects are assigned to the training, validation, and test sets, ensuring that samples from the same subject appear in the same set. Models are trained on the training set, monitored with early stopping techniques on the validation set, and evaluated on test set. This study uses 100 partitions. The generated ensemble of performances is used to compare deep learning models.

The reasons for using the repeated holdout method are explained below. As discussed in [8], EEG signals show strong inter-subject variability and possess inherent biometric-like properties that can overestimate model accuracy if the partitioning strategy is inappropriate. Windows from the same subject should not appear in both the training and test set, because the model can exploit subject-specific patterns to artificially boost performance. Data should be split at the subject level; however, even then the choice of the partitioning method matters. For example, defining a single holdout partition and training the model multiple times with different initializations is not suitable. As shown in [8], performance can vary dramatically depending on the subjects assigned to the test set. Nested approaches like Nested-Leave-N-Subjects-Out cross-validation are suggested as more rigorous and fair evaluation methods. They allow evaluating not only the effects of subject assignment to the test set, but also those to the validation set, which is typically used for early stopping. However, nested approaches partition the entire dataset at every split. Therefore, they

cannot be used to evaluate how accuracy varies with different amounts of fine-tuning training data, as discussed in Section 3.2. The repeated holdout, by contrast, can evaluate the model's adaptability across a large number of partitions and can be used to randomly create split sets of incrementally increasing size.

2.4.2. Model Architecture

To evaluate the efficacy of the proposed double masking self-supervised learning method, this analysis uses the recently proposed TransformEEG model, extensively described, with a PyTorch implementation, in [11]. The architecture is illustrated in panels B and C of Figure 1, while Table 2 provides a summary of the input shape, output shape, kernel size, number of convolutional groups, and the parameters for each layer.

Table 2. Summary Table of TransformEEG. Depthwise Conv Blocks include a skip connection.

Layer (Type)	Input Shape	Output Shape	Kernel Shape	Groups	Param #
TransformEEG	[N, 32, 2000]	[N, 1]	–	–	–
+Depthwise Tokenizer	[N, 32, 2000]	[N, 128, 498]	–	–	–
| +Depthwise Conv Block	[N, 32, 2000]	[N, 64, 999]	–	–	–
| | +Conv1d	[N, 32, 2000]	[N, 64, 2000]	[5]	32	384
| | +BatchNorm1d	[N, 64, 2000]	[N, 64, 2000]	–	–	128
| | +ELU	[N, 64, 2000]	[N, 64, 2000]	–	–	–
| | +AvgPool1d	[N, 64, 2000]	[N, 64, 999]	[4]	–	–
| | +Dropout1d	[N, 64, 999]	[N, 64, 999]	–	–	–
| | +Conv1d	[N, 64, 999]	[N, 64, 999]	[5]	64	384
| | +BatchNorm1d	[N, 64, 999]	[N, 64, 999]	–	–	128
| | +ELU	[N, 64, 999]	[N, 64, 999]	–	–	–
| +Depthwise Conv Block	[N, 64, 999]	[N, 128, 498]	–	–	2048
+Transformer Encoder	[N, 498, 128]	[N, 498, 128]	–	–	–
| +Permute	[N, 498, 128]	[N, 128, 498]	–	–	–
| +TransformerEncoderLayer	[N, 498, 128]	[N, 498, 128]	–	–	–
| | +MultiheadAttention	[N, 498, 128]	[N, 498, 128]	–	–	66,048
| | +Dropout	[N, 498, 128]	[N, 498, 128]	–	–	–
| | +LayerNorm	[N, 498, 128]	[N, 498, 128]	–	–	256
| | +Linear	[N, 498, 128]	[N, 498, 128]	–	–	16,512
| | +Dropout	[N, 498, 128]	[N, 498, 128]	–	–	–
| | +Linear	[N, 498, 128]	[N, 498, 128]	–	–	16,512
| | +Dropout	[N, 498, 128]	[N, 498, 128]	–	–	–
| | +LayerNorm	[N, 498, 128]	[N, 498, 128]	–	–	256
| +TransformerEncoderLayer	[N, 498, 128]	[N, 498, 128]	–	–	99,584
| +Permute	[N, 498, 128]	[N, 128, 498]	–	–	–
+Classification MLP	[N, 128, 498]	[N, 1]	–	–	–
| +AdaptiveAvgPool1d	[N, 128, 498]	[N, 128, 1]	–	–	–
| +Linear	[N, 128]	[N, 64]	–	–	8256
| +LeakyReLU	[N, 64]	[N, 64]	–	–	–
| +Linear	[N, 64]	[N, 1]	–	–	65

TransformEEG is a hybrid architecture that combines convolutional and transformer layers, consisting of three main modules.

The first module, called the depthwise convolutional tokenizer, transforms an input EEG sample into a sequence of tokens, each representing a local segment of the EEG window with features specific to each channel. This module includes two stacked depthwise convolutional blocks, with the following operations:

1. A depthwise 1D convolution with a depth multiplier of 2, doubling the feature dimension.
2. Batch normalization, followed by ELU activation.
3. Average pooling with a stride of 2, reducing the sequence length by half.
4. Dropout (with a probability of 0.2) for regularization during training.
5. A second depthwise 1D convolution with a depth multiplier of 1.
6. Batch normalization, followed by ELU activation.
7. A residual connection that adds the output after dropout, improving gradient stability.

The second module is the transformer block, which consists of two transformer encoder layers that process the generated tokens using a self-attention mechanism. Each encoder layer uses an MLP with a hidden layer dimension of 128 and applies layer normalization after each skip connection, as described in [30].

The third module is the classification MLP, which generates class predictions based on the output from the transformer encoder. It first applies global average pooling to produce an embedding vector. This vector is then passed through an MLP with one hidden layer of size 64, using Leaky ReLU as the activation function. The final output is converted to class probabilities using a sigmoid or softmax function, depending on the number of target classes.

Considering an input batch of dimension $(N, 32, 2000)$:

1. The depthwise convolutional tokenizer transforms the input into a tensor of size $(N, 128, 498)$, producing 498 tokens, each of size 128.
2. The global average pooling layer compresses the output from the transformer encoder into a tensor of size $(N, 128)$.
3. The classification MLP then generates class probabilities, producing a tensor of size (N, C), where C is the number of classes ($C = 1$ for binary classification).

2.4.3. Training Hyperparameters

Experiments were conducted using a fixed random seed value of 42 to minimize randomness in the code and enhance reproducibility.

Pretraining hyperparameters: the model was pretrained with Adam optimizer ($\beta_1 = 0.75$, $\beta_2 = 0.999$, no weight decay) [31]. The number of epochs was fixed to 50, without early stopping. The batch size was set to 1024. The initial learning rate was $2.5 \cdot 10^{-4}$, coupled with an exponential scheduler with $\gamma = 0.995$. The VICReg loss function was set with the following parameters: variance $\mu = 7.5$, invariance $\lambda = 5$, and covariance $\nu = 1$.

Fine-tuning and full supervision hyperparameters: Training was performed with Adam optimizer ($\beta_1 = 0.75$, $\beta_2 = 0.999$, no weight decay). The batch size was set to 64. The maximum number of epochs was set to 100. Early stopping on the validation set, with a patience of 20 epochs, was included as an additional regularization method. The initial learning rate was $2.5 \cdot 10^{-4}$, coupled with an exponential scheduler with $\gamma = 0.99$. The encoder remained unfrozen throughout the fine-tuning process. Binary cross entropy was used as the loss function.

2.4.4. Performance Evaluation

The results from all splits are aggregated to compare the pretrained TransformEEG with its fully supervised counterpart. Comparison metrics include the median value and IQR of balanced accuracy and F1-score. Balanced accuracy gives equal weight to all classes, regardless of their proportions in the dataset, while F1-score provides an additional balanced view by combining precision and recall.

Two binary classification tasks were considered in the evaluation. The first aims to distinguish Parkinson's subject against healthy controls with the "ds003490" dataset

(50 subjects, 25 with Parkinson's disease) and with the "ds00458" dataset (144 subjects, 100 with Parkinson's disease). The second aims to distinguish Alzheimer's subjects against healthy controls with the "brainlat" dataset (67 subjects, 35 with Alzheimer's disease).

3. Results

This section presents the results of the comparative analysis of the proposed double-masking representation learning method. It is structured as follows:

- Section 3.1 presents a comparison between the embeddings generated by the TransformEEG model before and after the pretraining process.
- Section 3.2 examines performance variations between a fully supervised and a pre-trained version of the TransformEEG model, on three clinical datasets.

3.1. Embedding Analysis

This subsection compares the embeddings generated by TransformEEG before and after the pretraining process. Specifically, a set of Uniform Manifold Approximation and Projection (UMAP) visualizations was created by extracting embeddings (i.e., the encoder's output prior to the linear projection) from samples in the "ds003490" (control vs. PD) and "brainlat" (control vs. AD) datasets. UMAP was applied using the relevant Python package in an unsupervised manner, with a local neighborhood size of 10 and a minimum effective distance between embedded points set to 0.2.

Analyzing the embeddings produced by the pretrained model on the fine-tuning datasets offers insight into its ability to generate representations that inherently encode features discriminative of the pathology. Figure 3 displays the UMAP visualizations of the embeddings before and after pretraining, categorized by target class. From these visualizations, it can be observed that the overlap between control and pathological groups decreases after the double-masking pretraining. However, the two classes remain partially overlapping and cannot be clearly separated, particularly for the "brainlat"dataset. Given that EEG signatures of PD and AD exhibit high inter-subject variability and are often subtle, a complete separation of classes is not expected in a low-dimensional projection. However, Figure 3 shows improved class separability after self-supervised pretraining, demonstrating the effectiveness of the double-masking approach.

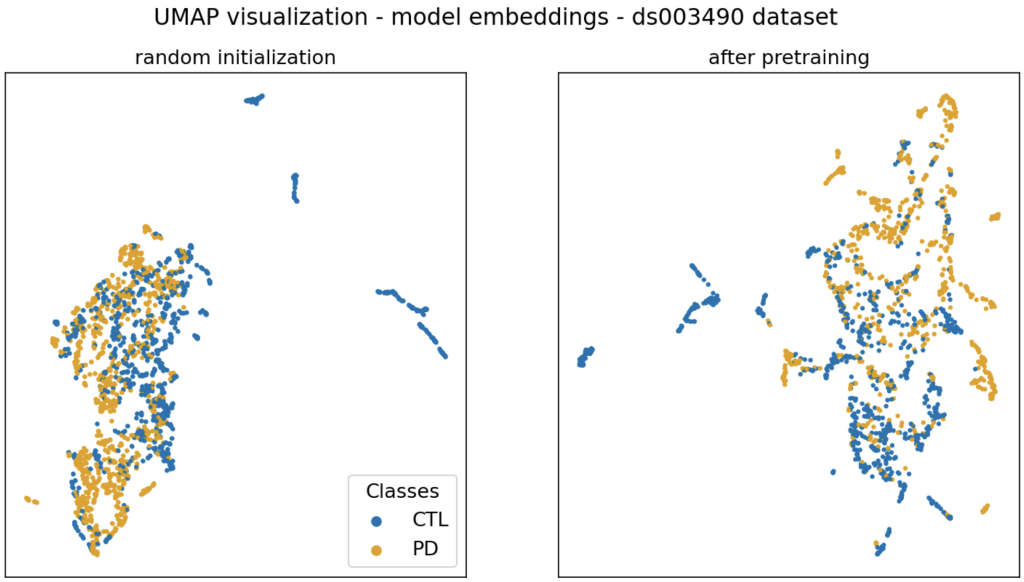


Figure 3. *Cont.*

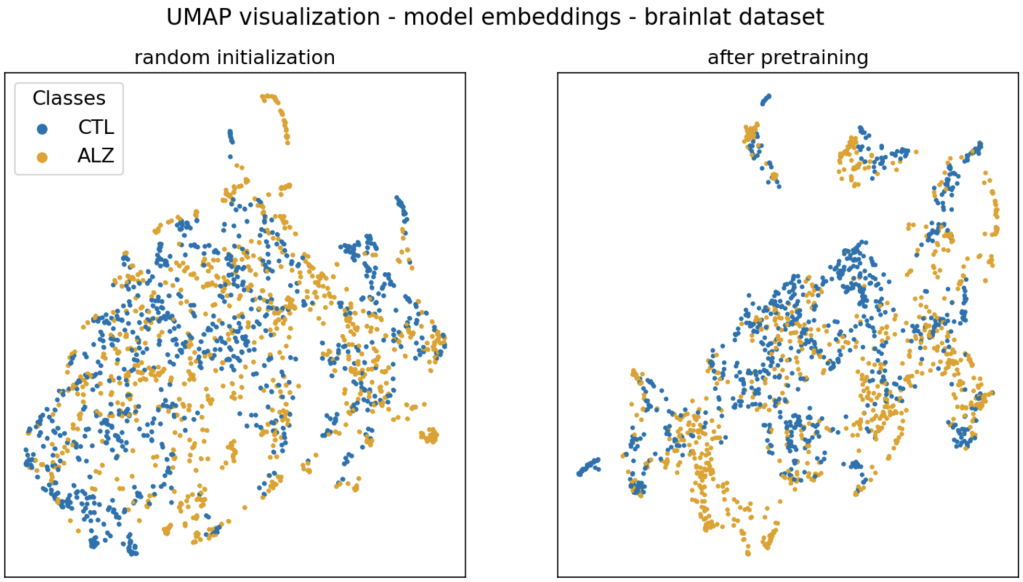


Figure 3. Comparison of embeddings generated before and after pretraining. The figures show UMAP visualizations of encoder outputs (prior to the linear projection) from samples in the "ds003490" dataset (control vs. PD, top) and the "brainlat" dataset (control vs. AD, bottom). Self-supervised pretraining improves cluster separation, demonstrating the effectiveness of the double-masking approach. UMAP embeddings were computed using the *umap-learn* Python library (version 0.5.9.post2).

This finding may explain why the model's performance, as reported in Section 3.2, remains modest even after pretraining. It also highlights the high variability inherent in EEG signals, which cannot be fully captured, even with self-supervised pretraining on thousands of subjects from multiple centers.

3.2. Pretraining Improves Downstream Accuracy

This subsection examines performance variations between a fully supervised and a pretrained version of the TransformEEG model.

The accuracies presented in Tables 3 and 4 show that the double masking pretrained model (TransformEEG) achieved improvements in both the median and IQR across all fine-tuning datasets. In particular, accuracy increased from 74.17% to 75.05% for Parkinson's disease detection in the "ds003490" dataset, along with a notable reduction in IQR from 10.01% to 6.37% (a decrease of 3.64%). Similarly, median accuracy improved in the "brainlat" dataset by 4.22%, with the IQR decreasing from 12.35% to 10.8%. For the "ds004584" dataset, accuracy increased by 2.82%, and the IQR decreased from 10.68% to 8.43%. Comparable improvements in both median and IQR values were also observed in the F1-scores, further supporting the effectiveness of the pretrained model. Table 3. Moreover, a similar trend can be observed when comparing the proposed double-masking pretrained model with a SimCLR-pretrained counterpart. Across all downstream tasks, the double-masking approach provided consistently higher median accuracies and lower IQR values, indicating that it not only improves overall performance but also enhances stability relative to conventional contrastive pretraining.

Additional evaluation of the pretraining efficacy was conducted by analyzing performance variations with different training set sizes. Figure 4 illustrates how model accuracy changes when the repeated holdout method is applied using 25%, 50%, 75%, and 100% of the available training data for fine-tuning. The test set size remained fixed at 20%, as described in Section 2.4.1. Remarkably, the TransformEEG model achieved a median balanced accuracy of 68.5% on the "ds003490" dataset (control vs. PD) using only 25% of the training data. This represents a substantial improvement of 6.85% over the fully

supervised baseline, demonstrating the potential of self-supervised learning to enhance model generalizability.

Table 3. Median and IQR of balanced accuracy and F1-score for different fine-tuning tasks.

Pretraining Type	ds003490 (CTL vs. PD)		brainlat (CTL vs. AD)		ds004584 (CTL vs. PD)	
	Bal. Acc.	F1-Score	Bal. Acc.	F1-Score	Bal. Acc.	F1-Score
No pretraining	74.17 [68.31–78.38]	73.53 [67.92–78.17]	58.54 [52.84–65.19]	56.52 [50.35–63.79]	63.75 [59.09–69.77]	62.61 [57.61–67.05]
SimCLR	73.05 [67.92–77.73]	72.72 [66.42–77.35]	61.61 [56.31–68.16]	56.68 [48.36–61.59]	62.76 [57.21–69.23]	60.83 [56.04–65.21]
Double Masking	75.05 [71.32–77.69]	74.85 [70.52–77.76]	62.76 [57.67–68.47]	61.61 [56.31–68.16]	66.57 [61.79–70.22]	62.92 [58.60–66.63]

Table 4. Median and IQR of per-class accuracy for different fine-tuning tasks.

Pretraining Type	ds003490 (CTL vs. PD)		brainlat (CTL vs. AD)		ds004584 (CTL vs. PD)	
	CTL	PD	CTL	AD	CTL	PD
No pretraining	66.86 [55.26–76.60]	81.63 [72.98–89.20]	70.48 [57.17–81.29]	42.33 [33.03–57.47]	59.99 [46.42–72.74]	68.99 [60.15–78.84]
SimCLR	76.83 [68.26–84.68]	73.93 [60.68–81.05]	67.13 [52.49–82.08]	48.49 [28.07–61.23]	64.08 [49.02–74.36]	64.88 [57.03–75.03]
Double Masking	78.79 [71.49–86.29]	72.91 [61.27–80.96]	70.53 [54.79–84.83]	50.84 [34.16–65.73]	73.07 [64.27–81.59]	60.37 [52.50–68.48]

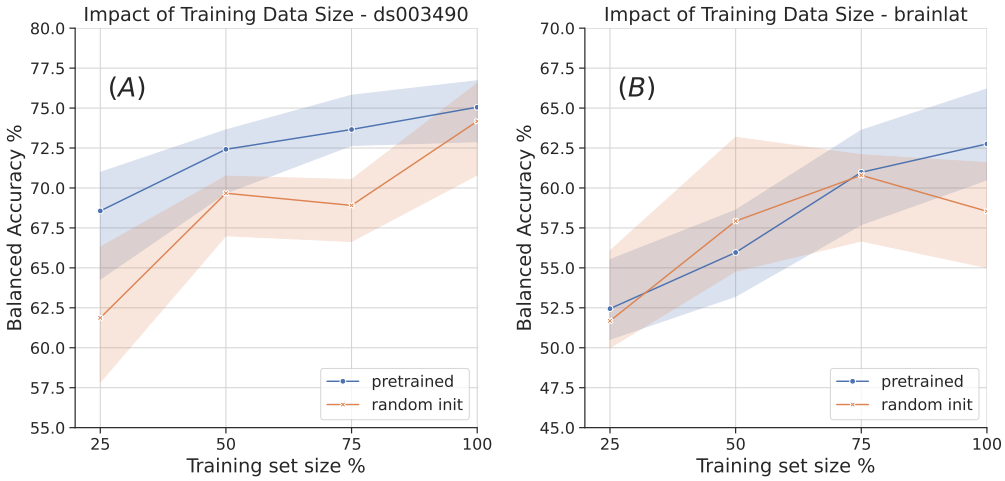


Figure 4. Training set size effects on downstream balanced accuracy for the "ds003490" dataset (CTL vs. PD, Panel (**A**)) and the "brainlat" dataset (CTL vs. AD, Panel (**B**)). Figures are generated by running the repeated holdout method using 25%, 50%, 75%, and 100% of the available training data for fine-tuning. The test set size remained fixed at 20%.

However, similar improvements were not observed for the "brainlat" dataset. In this case, performance remained comparable to that of the fully supervised model until 100% of the training data was used. This result suggests that the benefits of pretraining do not necessarily generalize across all downstream tasks or datasets.

4. Discussion

Model generalizability is a critical issue in deep learning-based EEG analysis. To address this, several strategies can be employed. For example, model architectures can be optimized to extract robust embeddings that capture both global and local EEG signal

patterns. However, as discussed in [11] and in this study, data aggregation is one of the most effective strategies for improving model generalizability.

Self-supervised learning is designed to learn representations from large amounts of unlabeled data, helping to overcome the data availability limitations common in biomedical applications. In this study, EEG recordings from eight different multi-center datasets comprising a total of 3156 subjects were harmonized and used to evaluate the proposed double-masking pretraining method.

As reported in Section 3, the pretrained model (TransformEEG) achieved improvements in both median accuracy and IQR across three fine-tuning tasks. In particular, accuracy in Parkinson's disease detection using the "ds003490" dataset increased from 74.17% to 75.05%, with a reduction in IQR from 10.01% to 6.37% (a decrease of 3.64%). Similarly, median accuracy improved in both the "brainlat"and "ds004584" datasets, increasing by 4.22% (IQR decreasing from 12.35% to 10.8%) and 2.82% (IQR decreasing from 10.68% to 8.43%), respectively. Furthermore, the pretrained model achieved a median balanced accuracy of 68.5% on the "ds003490" dataset (control vs. PD), using only 25% of the available data for fine-tuning. These results represent an improvement over the fully supervised baseline, highlighting the potential of self-supervised learning to enhance model generalizability.

While this analysis confirms previous findings showing that SSL improves downstream accuracy, the median and IQR values reported in Table 3 remain modest, especially for the "brainlat"and "ds004584" datasets. These performance levels suggest that current SSL-based embeddings may insufficiently capture fine-grained neurophysiological differences unless larger quantities of highly task-specific data are available. Furthermore, because the pretraining objective is designed to encourage invariance to augmentations, it may inadvertently obscure clinically relevant oscillatory patterns, thereby constraining downstream performance. In addition, although multi-center data aggregation improves generalizability, it may also introduce confounding factors that degrade model performance in certain scenarios. Differences in acquisition protocols, reference montages, or recording hardware can create artificial variability that the model must learn to ignore. When such variability dominates the underlying physiological signal (especially in smaller datasets) the model's ability to learn meaningful representations diminishes. A more systematic analysis of these confounds, would help clarify the conditions under which performance is most likely to deteriorate.

Future research should aim to address these limitations and propose novel strategies or expand the presented analysis, building on the findings of this study. For example, the current double-masking procedure could be enhanced by incorporating domain-informed priors that promote the learning of cross-channel correlations, facilitating the identification of disease-related patterns within specific regions of interest. In addition, spectral-domain strategies could be integrated into the proposed framework to enable the model to learn richer representations and better distinguish between normal and altered oscillatory patterns.

Although this work aggregates recordings from eight multi-center datasets, the overall number of subjects remains modest compared to other types of medical data (e.g., MRI data in the UK Biobank [32]). Aggregating data from additional large repositories, such as the TUH corpus mentioned in the introduction, could further improve model generalizability. By pretraining on a larger and more diverse pool of subjects, the model can better capture the heterogeneity characteristic of EEG recordings. Moreover, incorporating more datasets would allow evaluation of how the progressive addition of pretraining data affects downstream performance. This type of analysis is crucial to determine whether

pretraining benefits more from aggregating large amounts of heterogeneous data or from smaller datasets that are more specific to the downstream task.

Another potential direction for future research is the integration of multimodal data to enhance the quality of the learned representations. Multimodal approaches can capture complementary information across different data types, enabling the model to learn more comprehensive and robust features. In the context of neurological disorders, combining EEG with structural or functional imaging data could improve the sensitivity and specificity of downstream predictions. Incorporating multimodal or cross-modal pretraining strategies may further boost generalizability and lead to more clinically meaningful representations. However, the availability of large multimodal open datasets combining EEG recordings with other data types remains limited. The release of dedicated multimodal resources will support this line of research.

5. Conclusions

This study introduces a double-masking representation learning method for detecting EEG-based pathologies. This approach combines phase-swap data augmentation, which is designed to facilitate learning EEG phase-amplitude coupling, and a double-masking function operating at the signal and transformer token levels. A convolutional transformer model optimized for EEG recordings (TransformEEG) is pretrained using data aggregated from eight different multi-center datasets totaling 3156 subjects to evaluate potential improvements in its generalizability. We compare the pretrained model with its fully supervised counterpart to assess the quality of the learned embeddings. To provide reliable performance estimates, an evaluation is conducted on three publicly available datasets targeting pathology detection tasks using a repeated holdout method based on 100 different splits. The pretrained model showed an improved median and IQR of balanced accuracy and F1-score across all tasks. However, reported accuracies remain modest, confirming the current challenges of developing deep learning systems that can be reliably used to support clinical practice. These findings demonstrate that self-supervised learning can improve the performance of EEG deep learning models. Nevertheless, the path to solutions that generalize to clinical settings remains an open challenge for the global scientific community.

Author Contributions: Conceptualization, F.D.P.; methodology, F.D.P.; software, F.D.P.; validation, F.D.P.; writing—original draft preparation, F.D.P.; writing—review and editing, L.F.T., A.Z., L.T., E.T. and M.A.; supervision, M.A.; project administration, M.A.; funding acquisition, M.A. All authors have read and agreed to the published version of the manuscript.

Funding: This document is the result of the research project funded by the European Unions Horizon Europe research and innovation programme under Grant agreement no 101137074—HEREDITARY.

Data Availability Statement: All data that support the findings of this study are openly available and can be accessed according to Table 1's specifications. The code used to produce the results is available at https://github.com/MedMaxLab/ssleeg.

Acknowledgments: FDP would like to thank the Padova Neuroscience Center and its members for the support provided during the realization of this study.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Zhang, J.; Yin, Z.; Chen, P.; Nichele, S. Emotion recognition using multi-modal data and machine learning techniques: A tutorial and review. *Inf. Fusion* **2020**, *59*, 103–126. [CrossRef]
2. Loh, H.W.; Ooi, C.P.; Vicnesh, J.; Oh, S.L.; Faust, O.; Gertych, A.; Acharya, U.R. Automated Detection of Sleep Stages Using Deep Learning Techniques: A Systematic Review of the Last Decade (2010–2020). *Appl. Sci.* **2020**, *10*, 8963. [CrossRef]
3. Rasheed, K.; Qayyum, A.; Qadir, J.; Sivathamboo, S.; Kwan, P.; Kuhlmann, L.; O'Brien, T.; Razi, A. Machine Learning for Predicting Epileptic Seizures Using EEG Signals: A Review. *IEEE Rev. Biomed. Eng.* **2021**, *14*, 139–155. [CrossRef]
4. Abiri, R.; Borhani, S.; Sellers, E.W.; Jiang, Y.; Zhao, X. A comprehensive review of EEG-based brain-computer interface paradigms. *J. Neural Eng.* **2019**, *16*, 011001. [CrossRef] [PubMed]
5. Geraedts, V.J.; Boon, L.I.; Marinus, J.; Gouw, A.A.; van Hilten, J.J.; Stam, C.J.; Tannemaat, M.R.; Contarino, M.F. Clinical correlates of quantitative EEG in Parkinson disease: A systematic review. *Neurology* **2018**, *91*, 871–883. [CrossRef]
6. Del Pup, F.; Zanola, A.; Fabrice Tshimanga, L.; Bertoldo, A.; Atzori, M. The More, the Better? Evaluating the Role of EEG Preprocessing for Deep Learning Applications. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2025**, *33*, 1061–1070. [CrossRef]
7. Markiewicz, C.J.; Gorgolewski, K.J.; Feingold, F.; Blair, R.; Halchenko, Y.O.; Miller, E.; Hardcastle, N.; Wexler, J.; Esteban, O.; Goncavles, M.; et al. The OpenNeuro resource for sharing of neuroscience data. *eLife* **2021**, *10*, e71774. [CrossRef]
8. Del Pup, F.; Zanola, A.; Tshimanga, L.F.; Bertoldo, A.; Finos, L.; Atzori, M. The role of data partitioning on the performance of EEG-based deep learning models in supervised cross-subject analysis: A preliminary study. *Comput. Biol. Med.* **2025**, *196*, 110608. [CrossRef]
9. Kamrud, A.; Borghetti, B.; Schubert Kabban, C. The Effects of Individual Differences, Non-Stationarity, and the Importance of Data Partitioning Decisions for Training and Testing of EEG Cross-Participant Models. *Sensors* **2021**, *21*, 3225. [CrossRef]
10. Brookshire, G.; Kasper, J.; Blauch, N.M.; Wu, Y.C.; Glatt, R.; Merrill, D.A.; Gerrol, S.; Yoder, K.J.; Quirk, C.; Lucero, C. Data leakage in deep learning studies of translational EEG. *Front. Neurosci.* **2024**, *18*, 1373515. [CrossRef]
11. Del Pup, F.; Brun, R.; Iotti, F.; Paccagnella, E.; Pezzato, M.; Bertozzo, S.; Zanola, A.; Tshimanga, L.F.; Müller, H.; Atzori, M. TransformEEG: Towards improving model generalizability in deep learning-based EEG Parkinson's disease detection. *Neurocomputing* **2026**, *664*, 132075. [CrossRef]
12. Balestriero, R.; Ibrahim, M.; Sobal, V.; Morcos, A.; Shekhar, S.; Goldstein, T.; Bordes, F.; Bardes, A.; Mialon, G.; Tian, Y.; et al. A cookbook of self-supervised learning. *arXiv* **2023**, arXiv:2304.12210. [CrossRef]
13. Rafiei, M.H.; Gauthier, L.V.; Adeli, H.; Takabi, D. Self-Supervised Learning for Electroencephalography. *IEEE Trans. Neural Netw. Learn. Syst.* **2024**, *35*, 1457–1471. [CrossRef] [PubMed]
14. Liu, Z.; Alavi, A.; Li, M.; Zhang, X. Self-Supervised Contrastive Learning for Medical Time Series: A Systematic Review. *Sensors* **2023**, *23*, 4221. [CrossRef]
15. Mohsenvand, M.N.; Izadi, M.R.; Maes, P. Contrastive Representation Learning for Electroencephalogram Classification. In Proceedings of the Machine Learning for Health NeurIPS Workshop, Virtual Event, 11 December 2020; Volume 136, pp. 238–253.
16. Banville, H.; Chehab, O.; Hyvärinen, A.; Engemann, D.A.; Gramfort, A. Uncovering the structure of clinical EEG signals with self-supervised learning. *J. Neural Eng.* **2021**, *18*, 046020. [CrossRef]
17. Mohammadi Foumani, N.; Mackellar, G.; Ghane, S.; Irtza, S.; Nguyen, N.; Salehi, M. EEG2Rep: Enhancing Self-supervised EEG Representation Through Informative Masked Inputs. In Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Barcelona, Spain, 25–29 August 2024; KDD '24; pp. 5544–5555. [CrossRef]
18. Bardes, A.; Ponce, J.; LeCun, Y. VICReg: Variance-Invariance-Covariance Regularization for Self-Supervised Learning. In Proceedings of the International Conference on Learning Representations, Virtually, 25–29 April 2022.
19. Obeid, I.; Picone, J. The temple university hospital EEG data corpus. *Front. Neurosci.* **2016**, *10*, 196. [CrossRef]
20. Roy, Y.; Banville, H.; Albuquerque, I.; Gramfort, A.; Falk, T.H.; Faubert, J. Deep learning-based electroencephalography analysis: A systematic review. *J. Neural Eng.* **2019**, *16*, 051001. [CrossRef]
21. Sinha, S.R.; Sullivan, L.; Sabau, D.; San-Juan, D.; Dombrowski, K.E.; Halford, J.J.; Hani, A.J.; Drislane, F.W.; Stecker, M.M. American clinical neurophysiology society guideline 1: Minimum technical requirements for performing clinical electroencephalography. *J. Clin. Neurophysiol.* **2016**, *33*, 303–307. [CrossRef]
22. Zanola, A.; Del Pup, F.; Porcaro, C.; Atzori, M. BIDSAlign: A library for automatic merging and preprocessing of multiple EEG repositories. *J. Neural Eng.* **2024**, *21*, 046050. [CrossRef]
23. Delorme, A.; Makeig, S. EEGLAB: An open source toolbox for analysis of single-trial EEG dynamics including independent component analysis. *J. Neurosci. Methods* **2004**, *134*, 9–21. [CrossRef]
24. Koch, G.; Zemel, R.; Salakhutdinov, R. Siamese neural networks for one-shot image recognition. In Proceedings of the ICML Deep Learning Workshop, Lille, France, 10–11 July 2015; Volume 2.
25. Chen, T.; Kornblith, S.; Norouzi, M.; Hinton, G. A Simple Framework for Contrastive Learning of Visual Representations. In Proceedings of the 37th International Conference on Machine Learning, Vienna, Austria, 13–18 July 2020; Volume 119, pp. 1597–1607.

26. Lemkhenter, A.; Favaro, P. Boosting Generalization in Bio-signal Classification by Learning the Phase-Amplitude Coupling. In *Pattern Recognition*; Springer: Cham, Switzerland, 2021; pp. 72–85. [CrossRef]
27. Del Pup, F.; Zanola, A.; Tshimanga, L.F.; Mazzon, P.E.; Atzori, M. SelfEEG: A Python library for Self-Supervised Learning in Electroencephalography. *J. Open Source Softw.* **2024**, *9*, 6224. [CrossRef]
28. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. Pytorch: An imperative style, high-performance deep learning library. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*; Curran Associates Inc.: Red Hook, NY, USA, 2019; p. 721.
29. Waskom, M.L. seaborn: Statistical data visualization. *J. Open Source Softw.* **2021**, *6*, 3021. [CrossRef]
30. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.u.; Polosukhin, I. Attention is All you Need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*; Curran Associates Inc.: Red Hook, NY, USA, 2017; pp. 6000–6010.
31. Kingma, D.; Ba, J. Adam: A Method for Stochastic Optimization. *arXiv* **2015**, arXiv:1412.6980.
32. Bycroft, C.; Freeman, C.; Petkova, D.; Band, G.; Elliott, L.T.; Sharp, K.; Motyer, A.; Vukcevic, D.; Delaneau, O.; O'Connell, J.; et al. The UK Biobank resource with deep phenotyping and genomic data. *Nature* **2018**, *562*, 203–209. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.