Neurocomputing 664 (2026) 132075

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

TransformEEG: Towards improving model generalizability in deep learning-based EEG Parkinson's disease detection☆

Federico Del Pup[a,b,c,*], Riccardo Brun[a], Filippo Iotti[a], Edoardo Paccagnella[a], Mattia Pezzato[a], Sabrina Bertozzo[a], Andrea Zanola[b,c], Louis Fabrice Tshimanga[a,b,c], Henning Müller[d], Manfredo Atzori[b,c,d]

[a] Department of Information Engineering, University of Padua, Padua, 35131, Italy
[b] Department of Neuroscience, University of Padua, Padua, 35121, Italy
[c] Padova Neuroscience Center, University of Padua, Padua, 35129, Italy
[d] Information Systems Institute, University of Applied Sciences Western Switzerland (HES-SO Valais), Sierre, 3960, Switzerland

HIGHLIGHTS

- This work presents TransformEEG, a novel hybrid convolutional-transformer model.
- TransformEEG reduces performance variability in Parkinson's Disease detection.
- TransformEEG was evaluated on four public datasets comprising 290 subjects.
- Data augmentation further enhances TransformEEG performance.
- Classification threshold correction further enhances TransformEEG performance.

ARTICLE INFO

Communicated by K. Liu

Keywords:
EEG
Deep learning
Generalizability
Transformer
Nested-leave-N-subjects-out
Parkinson's disease

ABSTRACT

Electroencephalography (EEG) is establishing itself as an important, low-cost, noninvasive diagnostic tool for the early detection of Parkinson's Disease (PD). In this context, EEG-based Deep Learning (DL) models have shown promising results due to their ability to discover highly nonlinear patterns within the signal. However, current state-of-the-art DL models suffer from poor generalizability caused by high inter-subject variability. This high variability underscores the need for enhancing model generalizability by developing new architectures better tailored to EEG data. This paper introduces TransformEEG, a hybrid Convolutional-Transformer designed for Parkinson's disease detection using EEG data. Unlike transformer models based on the EEGNet structure, TransformEEG incorporates a depthwise convolutional tokenizer. This tokenizer is specialized in generating tokens composed of channel-specific features, which enables more effective feature mixing within the self-attention layers of the transformer encoder. To evaluate the proposed model, four public datasets comprising 290 subjects (140 PD patients, 150 healthy controls) were harmonized and aggregated. A 10-outer, 10-inner Nested-Leave-N-Subjects-Out (N-LNSO) cross-validation was performed to provide an unbiased comparison against seven other consolidated EEG deep learning models. TransformEEG achieved the highest balanced accuracy's median (78.45 %) as well as the lowest interquartile range (6.37 %) across all the N-LNSO partitions. When combined with data augmentation and threshold correction, median accuracy increased to 80.10 %, with an interquartile range of 5.74 %. In conclusion, TransformEEG produces more consistent and less skewed results. It demonstrates a substantial reduction in variability and more reliable PD detection using EEG data compared to the other investigated models.

☆ This document is the results of the research project by the European Union's Horizon Europe research and innovation programme under Grant agreement no 101137074—HEREDITARY.

* Corresponding author at: Department of Information Engineering, University of Padua, Padua, 35131, Italy.
Email addresses: federico.delpup@studenti.unipd.it (F. Del Pup), riccardo.brun.1@studenti.unipd.it (R. Brun), filippo.iotti@studenti.unipd.it (F. Iotti), edoardo.paccagnella.1@studenti.unipd.it (E. Paccagnella), mattia.pezzato.3@studenti.unipd.it (M. Pezzato), sabrina.bertozzo@studenti.unipd.it (S. Bertozzo), andrea.zanola@studenti.unipd.it (A. Zanola), louisfabrice.tshimanga@unipd.it (L.F. Tshimanga), henning.mueller@hevs.ch (H. Müller), manfredo.atzori@unipd.it (M. Atzori).

https://doi.org/10.1016/j.neucom.2025.132075
Received 10 July 2025; Received in revised form 10 October 2025; Accepted 8 November 2025
Available online 10 November 2025

1. Introduction

Parkinson's Disease (PD) is the second most common chronic neurodegenerative disorder. It is a progressive disease that primarily affects individuals over the age of 65, with both incidence and prevalence steadily increasing with age [1]. The disease results from the death of dopaminergic neurons in the substantia nigra [2]. Symptoms can vary among individuals and include motor symptoms, such as tremors, bradykinesia (slowness of voluntary movement initiation), akinesia (absence of normal unconscious movements), and hypokinesia (reduction in movement amplitude), as well as non-motor symptoms like sleep behavior disorders, constipation, and anxiety [3,4]. PD symptoms can significantly reduce patients' quality of life, especially since no disease-modifying pharmacological treatments are currently available [5]. Therefore, early diagnosis of PD is of clinical significance, as it can help improve patients' quality of life by potentially slowing disease progression.

Electroencephalography (EEG) is establishing itself as an important, low-cost, noninvasive diagnostic tool for the early detection of PD. This success stems from the potential use of quantitative EEG measures as biomarkers of disease severity and progression [6]. In [7], the analysis of 36 different studies confirmed that both global and domain-specific cognitive impairments correlate with EEG "slowing" [8]. EEG slowing is associated with alterations in the normal oscillatory brain activity, characterized by decreased spectral power in alpha and beta bands and increased spectral power in delta and theta bands.

To identify differences between healthy and PD individuals in EEG recordings, several automatic classification approaches have been investigated [9]. Among these, methods based on deep learning (DL) have shown promising results due to the ability of neural networks to discover highly nonlinear patterns embedded within the signal [10]. Various architectures have been explored, including Convolutional Neural Networks (CNNs) [11], Recurrent Neural Networks (RNNs) [12], and, more recently, transformer models [13].

However, current state-of-the-art EEG deep learning models suffer from poor generalizability, underscored by their high sensitivity to variations in the experimental setting. In [14], it was found that variations in the preprocessing pipeline can lead to significant fluctuations in model accuracy. Focusing on the PD detection results reported in this study, when data are preprocessed with only a minimal filtering, median accuracy reaches 66 %. Introducing independent component rejection [15] increases the median accuracy to 75 %, but adding more advanced artifact removal techniques (e.g., artifact subspace reconstruction [16]) causes the metric to drop back down to 67 %. These results show that certain EEG artifacts can improve the accuracy of deep learning models but can also alter the quality of the learned features, ultimately reducing their generalizability.

Similarly, the way EEG data are partitioned can greatly overestimate the model's performance and generalizability. In [17], it was demonstrated that EEG deep learning models can reach almost perfect accuracy (with little variability) if segments from the same EEG recordings are assigned to both the training and test sets. However, when a cross-subject analysis is performed, accuracy drops by over 20 %, accompanied by an alarming increase in result variability. This variability suggests that inherent characteristics of the EEG signal, such as biometric features and potential correlations across consecutive segments, can be exploited by the model to solve the task at hand without necessarily learning features that are truly representative of the underlying pathology.

Recently proposed EEG-DL models incorporate attention layers to better identify and capture long-range patterns within the signal [18–20]. However, these models often use EEGNet-like convolutional encoders to generate the sequence of tokens that are fed into the transformer blocks, which limits the capabilities of the attention layers [21]. While the addition of transformer layers on top of an EEGNet-like convolutional tokenizer can improve model performance, it does not address the generalizability issue reported in [17] using the same family of models, which might stem from how tokens are generated.

When evaluated with unbiased cross-validation methods such as the Nested-Leave-N-Subjects-Out (N-LNSO), EEG deep learning models often show reduced generalizability, reflected in lower performance and higher variability [17]. This variability not only makes it difficult to compare models fairly, but it also underscores the need for enhancing model generalizability by developing new architectures that are better tailored to EEGs. Improving generalizability is essential for ensuring the reliability of EEG deep learning systems and supporting their application in clinical scenarios.

Despite this, recent studies introducing novel EEG deep learning models have predominantly focused on evaluating mean classification accuracy, overlooking another critical aspect of performance: variability across subjects and datasets. In EEG, where inter-subject variability is high and clinical reliability is essential, generalizability is as important as accuracy. The absence of systematic investigations into variability and generalization therefore constitutes a significant gap in current research.

Contributions: This study introduces TransformEEG, a novel hybrid Convolutional-Transformer model designed for PD detection using EEG data. Unlike transformer models based on the EEGNet structure, TransformEEG incorporates a carefully designed depthwise convolutional tokenizer. This tokenizer specializes in generating tokens composed of channel-specific features, enabling more effective feature mixing within the transformer encoder. TransformEEG is evaluated on Parkinson's disease classification against seven other consolidated EEG deep learning models, showing better results in terms of balanced accuracy's median and interquartile range. To provide reliable performance estimates, the evaluation is conducted on four publicly available datasets comprising 290 subjects (140 PD patients, 150 healthy controls), using a 10-outer, 10-inner Nested-Leave-N-Subjects-Out cross-validation scheme. Compared to previous literature, this study emphasizes model generalizability in two key ways. First, it presents a model designed specifically to address the high variability commonly observed in EEG deep learning. Second, it analyzes how gradually incorporating data augmentation and classification threshold correction affects result variability, providing a comprehensive assessment of the generalizability of the investigated architectures.

Paper structure: The outline of this paper is as follows. Section 2 describes in detail the experimental setting. Section 3 presents the comparative analysis between TransformEEG and seven other EEG deep learning models. Section 4 critically discusses the results, highlighting potential limitations and future directions. Finally, a conclusion is drawn in Section 5.

2. Methods

This section outlines key methodological aspects of the study. First, it provides a concise description of the selected datasets and their preprocessing steps. Next, it introduces the proposed TransformEEG architecture. Finally, it details additional training information, including data partitioning, data augmentation, training hyperparameters, and model evaluation strategies. Together with the openly available source code repository and supplementary data, this section provides all the features listed in [22] to enhance the reproducibility of this study.

2.1. Dataset selection

The analysis was conducted on four open-source datasets, briefly described in the following subsections and summarized in Table 1. All datasets are publicly available on OpenNeuro[1] [23], an established platform for sharing neuroimaging data in BIDS format [24]. They were selected to provide a sufficiently large number of EEG recordings while

[1] [Online] Available: https://openneuro.org/.

Table 1
Dataset description.

Dataset ID	Original reference	# Chan	f_s [Hz]	# Subj	# Samples[1]
ds004148	FCZ	64	500	60	2880
ds002778	CMS/DRL	41	512	31	456
ds003490	CPZ	64	500	50	2408
ds004584	PZ	63	500	149	1775
Total				290	7519

[1] Assuming EEG windows of 16s and 25 % overlap.

maintaining a balanced distribution of Parkinson's disease and healthy control subjects (140 and 150, respectively). A large number of subjects is particularly important for training deep learning models, as further discussed in Section 3.5.

2.1.1. Dataset 1: ds004148—EEG test-retest

This dataset [25] comprises resting-state (both eyes-open and eyes-closed) and cognitive state recordings from 60 healthy subjects, with an average age of 20.0 ± 1.9 years. All subjects participated in three recording sessions, during which both resting-state and cognitive tasks were performed. To prevent unbalancing the sample distribution across different datasets, only the resting-state recordings from session one were considered, as done in [14]. The selected resting-state recordings have a fixed duration of exactly 300 s.

2.1.2. Dataset 2: ds002778—UC San Diego

This dataset [26] includes resting-state, eyes-open EEG recordings from 15 individuals with Parkinson's disease and 16 age-matched healthy controls. The average age of the two groups is 63.3 ± 8.2 years for the Parkinson's patients and 63.5 ± 9.7 years for the healthy controls. Healthy subjects participated in only one session, while Parkinson's individuals underwent two sessions: the first, recorded after they discontinued medication for at least 12 hours before the session (*ses-off*); the second, recorded while they were under medication (*ses-on*). For the subsequent analysis, only the off-medication recordings were included. The selected recordings have an average duration of 195.7 ± 18.8 s.

2.1.3. Dataset 3: ds003490—EEG 3-stim

This dataset [27] comprises resting-state EEG recordings (both eyes-open and eyes-closed) and auditory oddball tasks from 25 individuals with Parkinson's disease and 25 age-matched healthy controls. The average age of the two groups is 69.7±8.7 years for the Parkinson's patients and 69.3±9.6 years for the healthy controls. Healthy controls participated in a single recording session, while Parkinson's individuals underwent acquisitions both off-medication (at least 15 hours) and on-medication. For the subsequent analysis, only the off-medication recordings were included. The selected records have an average duration of 595.9 ± 74.0 s.

2.1.4. Dataset 4: ds004584—EEG PD

This dataset [28] comprises resting-state, eyes-open EEG recordings from 100 Parkinson's patients and 49 age-matched healthy controls. The average age of the two groups is respectively 68.5 ± 8.1 years for the Parkinson's patients and 70.9 ± 7.6 years for the healthy controls. All subjects underwent a single recording session, with an average duration of 144.4 ± 46.0 s.

2.2. Data preprocessing

All the selected recordings were preprocessed and harmonized using an automatic pipeline implemented in *BIDSAlign*[2] [29] (v1.0.0). This pipeline is based on the findings from a recent analysis on the role of preprocessing in EEG deep learning applications [14], which showed that adding more intensive steps, such as noisy channel removal and window correction via artifact subspace reconstruction (ASR) [16], does not improve model performance. The final standardization, downsampling, and window extraction steps were performed online within the Python environment during data loading for model training. The preprocessing steps are detailed below in their execution order.

1. *Non-EEG channels removal*: Other biosignals (e.g., ECG, EOG) included as extra channels were removed.
2. *Time segments removal*: The first and last 8 s of each recording were removed to exclude potential divergences in the signal. This step limits the removal of brain activity while enhancing the quality of the subsequent independent component rejection step.
3. *DC component removal*: The direct current (DC) voltage was subtracted from each EEG channel. This operation is equivalent to removing the mean from each EEG channel.
4. *Resampling*: EEG recordings were resampled to 250 Hz to reduce the computational burden of the subsequent preprocessing operations.
5. *Filtering*: EEG signals were filtered with a passband Hamming windowed sinc FIR filter with passband edges of 1 Hz and 45 Hz.
6. *Automatic independent components rejection*: Independent Components (ICs) were extracted using the *runica* algorithm [30], with no limit on the number of components. These components were automatically rejected using IClabel [15] by applying the following rejection thresholds: [90 %, 100 %] confidence for components considered noisy (e.g., muscle, eye, heart, line noise, channel noise) and [0 %, 10 %] confidence for components representing brain activity, following the approach in [14].
7. *Re-referencing*: EEG recordings were re-referenced to the common average.
8. *Channel selection*: A subset of 32 EEG channels, common to all four selected datasets, was extracted and used for subsequent analysis.
9. *Downsampling*: EEG signals were further downsampled to 125 Hz to improve computation and reduce the GPU's memory occupation during model training. As explained in [14], direct resampling from 250 Hz to 125 Hz was avoided to preserve the quality of the automatic independent component rejection process.
10. *Standardization*: The z-score operator was applied along the EEG channel dimension. This step transforms each EEG channel into a signal with mean $\mu = 0$ and standard deviation $\sigma = 1$.
11. *Windows extraction*: EEG data were partitioned into 16-second windows with 25 % overlap to increase the number of samples. This step generates a dataset of 7519 samples, 4947 from healthy controls and 2572 from Parkinson's subjects.

Preprocessed data from all the four datasets were combined to construct a binary classification task that aims to distinguish between Parkinson's and healthy subjects.

2.3. Model architecture

TransformEEG is a hybrid convolutional-transformer architecture composed of three modules, each stacking a series of neural blocks. The first module is the depthwise convolutional tokenizer, which creates EEG tokens describing local portions of the input window with channel-specific features. The second module is the transformer encoder, which recombines the generated tokens using the self-attention mechanism. The third module is the classification MLP, which produces class predictions based on the output of the transformer encoder. Fig. 1 illustrates the structure of TransformEEG. Table 11 in Section A.7 of the Supplementary data includes a detailed description of the input and output dimensions and the number of parameters for each layer. A PyTorch [31] implementation of the model is available in the open-source code repository.[3] A description of each module is provided below.

[2] https://github.com/MedMaxLab/BIDSAlign.

[3] https://github.com/MedMaxLab/transformeeg.

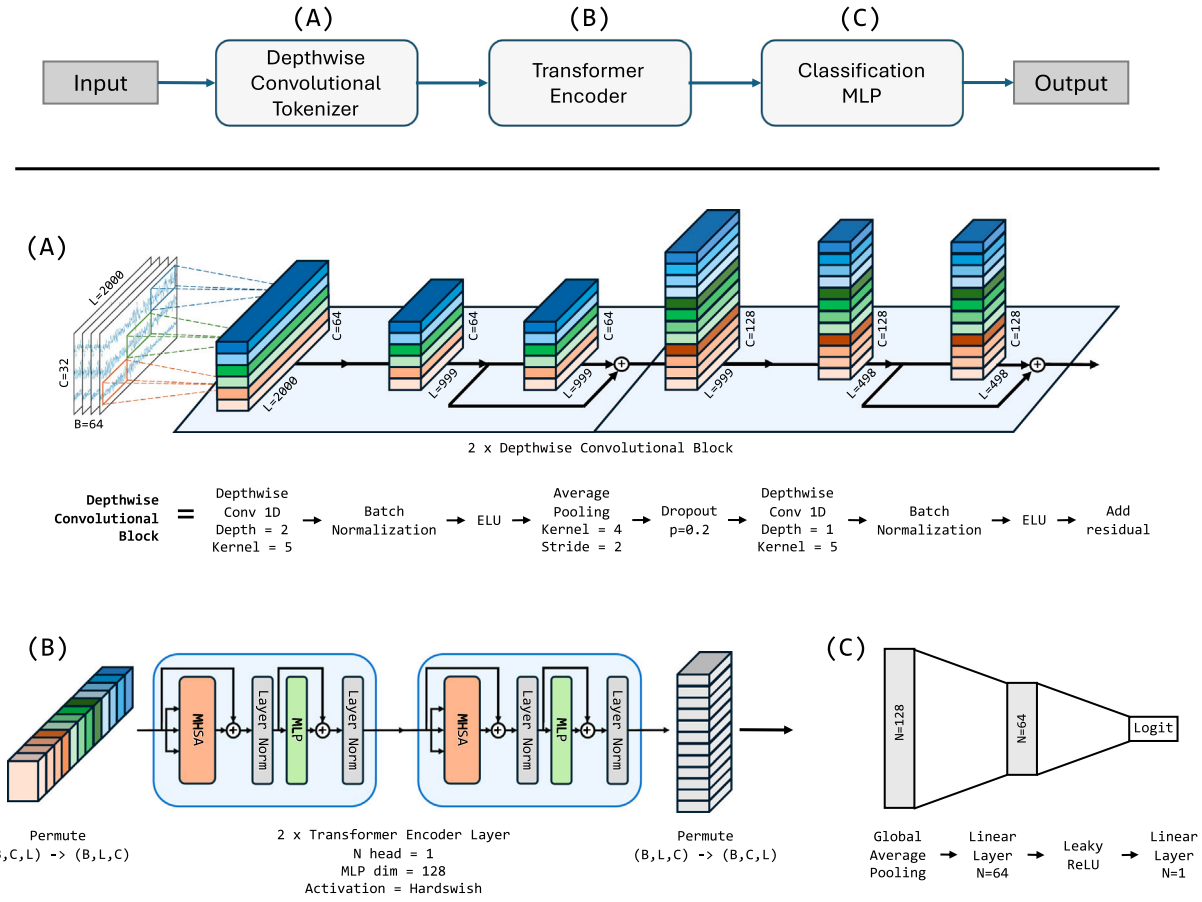

Fig. 1. Schematic representation of the TransformEEG architecture. TransformEEG consists of three modules: a depthwise convolutional tokenizer (A), a transformer encoder (B), and a classification MLP (C). The depthwise convolutional tokenizer creates EEG tokens describing local time portions of the input window with channel-specific features. The transformer encoder recombines the tokens with the self-attention mechanism. The classification MLP outputs class predictions.

2.3.1. *Depthwise convolutional tokenizer*

The depthwise convolutional tokenizer is a module designed to transform an input EEG window into a sequence of tokens. This module consists of two stacked depthwise convolutional blocks, which extract features based on local temporal patterns within individual EEG channels. The depthwise convolutional block performs the following operations:

1. Depthwise 1D convolution with a depth multiplier of 2, which doubles the feature dimension.
2. Batch normalization, followed by ELU activation.
3. Average pooling with stride 2, which halves the sequence length.
4. Dropout ($p = 0.2$), which is applied during training for regularization.
5. Depthwise 1D convolution with a depth multiplier of 1.
6. Batch normalization, followed by ELU activation.
7. Residual connection addition (from the output after dropout), to improve gradient stability.

Depthwise convolutions ensure that, first, each feature is associated with a unique input EEG channel (even after multiple blocks are applied sequentially) and, second, that the number of parameters remains low. Maintaining a small parameter count helps prevent overparameterization and reduces overfitting tendencies, which can be particularly detrimental in EEG-based pathology classification tasks, as discussed in [32,33]. Average pooling halves the sequence length, thereby reducing memory usage and computational load. As a result, the combination of depthwise convolutions and pooling produces tokens that represent local portions (consecutive time steps) of the input EEG window, effectively capturing channel-specific features.

Denoting the following variables:

1. B as the batch size,
2. C as the number of input EEG channels,
3. L as the window length,
4. D as the depth multiplier of the depthwise convolutional layer,
5. S and K as the stride and kernel size of the average pooling layer, respectively.

The combination of two depthwise convolutional blocks produces a new multidimensional array with the following dimensions:

$$(B, C, L) \rightarrow \left(B, C \times D, \left\lfloor \frac{L + (S-K)(1+S)}{S^2} \right\rfloor\right) \tag{1}$$

Therefore, with input dimensions of (64, 32, 2000), the output dimensions are (64, 128, 498).

The TransformEEG's tokenizer module diverges from typical EEGNet-like convolutional encoders used in attention-based architectures such as EEGConformer [18] or ATCNet [19]. EEGNet-like encoders treat the input window as a single-channel pseudo image, processed with 2D convolutional layers that operate along either the temporal or spatial dimensions. Consequently, spatial convolutions extract features that are linear combinations of the original EEG channels. Additionally, since no padding is used, this process conveniently removes the EEG channel dimension and prepares the input for the transformer encoder (see

Supplementary data, Section A.7). However, as a result, attention layers are forced to perform linear projections on features that may not optimally represent the original channels. This redundancy can negatively impact the effectiveness of feature mixing within the transformer encoder, as such mixing heavily depends on the quality of channel recombination performed by the spatial layers.

In contrast, each module of TransformEEG has a clear and complementary role. The convolutional tokenizer processes EEG segments as multi-channel time-series data. Depthwise convolutions apply a separate filter to each input EEG channel independently, aiming to extract channel-specific features based on local temporal patterns. No channel recombination occurs at this stage, ensuring that the extracted features better reflect the properties of each individual channel. This design potentially enhances the generalizability of the model because it allows for improved feature mixing across channels within the transformer encoder. In summary, the complementary roles of convolutional and attention layers enhance the network's ability to capture both local and global patterns within the EEG window, leading to more consistent performance, as discussed in Section 3.

2.3.2. Transformer module

The sequence of tokens generated by the depthwise convolutional tokenizer is given to the transformer module (Fig. 1, Panel B), which consists of a series of transformer encoder layers. Each transformer encoder layer processes the input EEG tokens using a self-attention mechanism and applies a subsequent non-linear transformation with a multi-layer perceptron (one hidden layer with size equal to the embedding dimension, 128). Layer normalization is applied after each skip connection, following the approach described in [34]. Permutation operations are included for compatibility with PyTorch library objects. In contrast to traditional implementations, no positional embeddings are added to the sequence of tokens, nor is a special class token concatenated to the output of the convolutional tokenizer [35]. An ablation analysis, presented in Section A.4.1 of the Supplementary data, demonstrates that including one or both of these elements does not improve the performance or generalizability of TransformEEG. The number of heads in the self-attention layers is set to one. As discussed in Section A.4.2 of the Supplementary data, increasing the number of heads does not improve the model's performance. While using more attention heads allows TransformEEG to capture multiple types of relationships within the input data simultaneously, it also increases the chances that the model will overfit the training set, especially if the number of subjects (samples) is low.

2.3.3. Classification MLP

The output of the transformer encoder is fed into the final classification module to generate predictions. Since there is no class token, global average pooling is initially performed to produce an embedding vector. This embedding vector is then fed into a multi-layer perceptron (MLP) comprising one hidden layer. The hidden layer halves the embedding dimension and applies a Leaky ReLU activation function (negative slope of 0.01) as non-linearity. The MLP's output is subsequently passed through a sigmoid function to produce a probability indicating whether the sample originates from an EEG of a patient with Parkinson's disease. Sections 3.1 and 3.2 present the results using a default probability threshold of 0.5 to classify samples as positive or negative, while Section 3.3 presents the results when correction procedures are applied to adjust this threshold.

2.4. Implementation details

Models were trained using *SelfEEG*[4] [36] (v0.2.0) and figures were generated with *Seaborn* [37]. Experiments were conducted on an NVIDIA A30 GPU device with CUDA 12.2. Further implementation details are provided in the Supplementary data and in the openly available source code.

2.4.1. Data partition and model evaluation

After dividing the data into 16-second windows with 25 % overlap, the datasets were partitioned and evaluated using the Nested-Leave-N-Subjects-Out (N-LNSO) cross-validation (CV) method described in [14]. This method involves concatenating two Leave-N-Subjects-Out CV procedures to generate a set of unique train-validation-test splits. For each split, models are trained on the training set, monitored on the validation set, and evaluated on the test set. This study uses ten outer folds and ten inner folds, for a total of 100 splits per N-LNSO procedure. Splits were stratified by both class and dataset. Specifically, each of the 100 test sets included subjects from all datasets and both classes. This stratification provides a more reliable estimate of performance, as the model must generalize across datasets and classes to achieve high accuracy. Additional details are provided in Section A.1 of the Supplementary data.

Seven established deep learning models were considered for the evaluation. The list is composed of xEEGNet [33], EEGNet [21], ShallowNet [38], ATCNet [19], EEGConformer [18], DeepConvNet [38], and EEGResNet [39]. These models encompass different architectural types, such as convolutional (e.g., EEGNet, ShallowNet) and attention-based models (e.g., EEGConformer, ATCNet). The number of learnable parameters spans from few hundreds (xEEGNet, 245 parameters) to more than a million (EEGResNet, 1,337,665 parameters). This diversity enriches the analysis and enables a comprehensive comparison with state-of-the-art models.

Models were evaluated using balanced accuracy, defined as the macro-average of recall across classes [40]. In binary classification, balanced accuracy has a random chance level of 50 % and gives equal weight to each class regardless of its distribution. Results based on additional metrics such as F1-Score and Cohen's kappa are reported in Section A.2 of the Supplementary data. The results from all splits are aggregated to compare TransformEEG with the other selected EEG deep learning models. Comparison metrics include the median value, interquartile range (IQR), and the [1st–99th] percentile range.

2.4.2. Data augmentation

To enable a fair comparison of the considered architectures, data augmentation effects were assessed by selecting the optimal configuration for each model. A set of 10 different data augmentations was selected for the comparison presented in Section 3.2, based on dedicated analyses performed on EEG signals [41–44]. Each augmentation was investigated individually and in combination with another, resulting in a total of 100 possible data augmentation configurations. For each of these configurations, a N-LNSO cross-validation was performed on the architectures listed in Section 2.4.1. The median and interquartile range of balanced accuracy were compared to identify the most effective augmentation combination for each model.

Determining the optimal data augmentation is a challenging process, as both the median (central measure) and the interquartile range (variability) are crucial in the comparison. In addition, these two values evolve on different scales, increasing the challenges in weighting their improvement over the reference baseline. In order to identify a consistent and objective measure, which takes into account the relative improvements of both the baseline median and interquartile range, this work proposes the Augmentation Relative Improvement Score (ARIS). ARIS is defined as follows

$$ARIS = \begin{cases} 0, & \text{if } M_b > M \text{ or } \mathrm{IQR}_b < \mathrm{IQR}, \\ \dfrac{M_b - M}{M_b} \times \dfrac{\mathrm{IQR} - \mathrm{IQR}_b}{\mathrm{IQR}_b}, & \text{otherwise.} \end{cases} \quad (2)$$

where:

1. M_b is the median baseline without data augmentation.

[4] https://github.com/MedMaxLab/selfEEG.

Table 2
Data augmentation hyperparameters: hyperparameter values are randomly selected at each call to increase sample variability.

Data Augmentation	Hyperparameter Name	Hyperparameter Value
Sign Flip	None	None
Time Reverse	None	None
Band Noise	Bandwidth	band ∈ {delta, theta, alpha, beta, gamma low}
	σ	$\sigma \in [0.8, 0.9]$
Signal Drift	Slope	$m/(125 \times 16)$ $m \in \{\pm 0.5, \pm 0.4, \pm 0.3\}$
SNR Scaling	SNR	SNR ∈ [8, 10]
Channel Dropout	Channels to drop	$n \in [4, 16], n \in \mathbb{Z}$
Masking	Masked portions	$k \in \{3, 4, 5\}$
	Masking ratio	$p \in [0.2, 0.35]$
Signal Warp	Number of segments	$n \in \{6, 7, 8\}$
	Stretch strength	$k_{st} \in [1.25, 1.50]$
	Squeeze strength	$k_{sq} = 1$
Phase Randomizer	perturbation strength	$s = 0.9$
Phase Swap	None	None

2. M is the current median value with data augmentation.
3. IQR_b is the baseline interquartile range.
4. IQR is the current interquartile range.

This score ensures that data augmentations that do not improve both the median and interquartile range of the balanced accuracy are discarded. Additionally, it ensures improvements are properly weighted by the scale of the baseline value.

A description of each data augmentation, along with the set of parameters used (detailed in Table 2), is provided below. Their Python implementation is available within the *selfEEG* source code, which is openly accessible on GitHub. An analytical formulation is also provided to clarify their description. The input EEG window is denoted as a matrix $\text{X} \in \mathbb{R}^{C \times N}$, where C is the number of channels and N is the number of time steps. The value of N is calculated as $N = f_s \times L$, with f_s being the sampling frequency in Hertz (125 Hz) and L the sequence length in seconds (16 s). The augmented version of X is denoted as $\tilde{\text{X}} \in \mathbb{R}^{C \times N}$. Additionally, $x_{c,t}$ is used to indicate the value of channel c at the time step t of an EEG sample X.

- **Time Reverse**: This transformation flips the signal horizontally by applying the following operation:

$$\tilde{x}_{c,t} = x_{c,L-t} \tag{3}$$

- **Sign Flip**: The signal is flipped vertically. This data augmentation simulates an inversion of polarity in the electrodes and it is described by the transformation

$$\tilde{\text{X}} = -\text{X} \tag{4}$$

- **Band Noise**: This augmentation adds random noise filtered within specific EEG bandwidths. Given the impulse response $h(t)$ of a band-pass filter that preserves a particular EEG bandwidth, the augmented EEG sample can be defined as:

$$\tilde{x}_{c,t} = x_{c,t} + (\varepsilon * h)(t) \tag{5}$$

  where $\varepsilon(t)$ is Gaussian noise with variance σ^2, an augmentation hyperparameter, and $*$ denotes the convolution operation.
- **Signal drift**: The signal is drifted with either a positive or negative slope by applying the transformation:

$$\tilde{x}_{c,t} = x_{c,t} + mt \tag{6}$$

  where m is an augmentation hyperparameter that determines the slope of the drift.
- **SNR Scaling**: This augmentation adds random noise to an EEG sample, scaled to reach a specified signal-to-noise ratio (SNR). Given a target SNR, the augmented EEG sample can be defined as:

$$\tilde{x}_{c,t} = x_{c,t} + k\varepsilon_t \tag{7}$$

  where
  - $\varepsilon_t \sim \mathcal{N}(0, 1)$ is Gaussian noise with unitary variance
  - k is a scaling factor calculated as

$$k = 10^{(-\frac{\text{SNR}}{20})} \sqrt{\frac{1}{NC} \sum_c \sum_t x_{c,t}^2} \tag{8}$$

- **Channel dropout**: This augmentation sets a predefined number of EEG channels to zero. Specifically, given a dropout percentage p and a bijective function $\sigma : S \to S$ that defines a random permutation of the elements of a vector, the augmented EEG sample is computed as

$$\tilde{\text{X}} = \text{X} \odot (\mathbf{s}_C \mathbf{1}_N^T) \tag{9}$$

  where

$$\mathbf{s}_C = \sigma(\mathbf{0}_{\lfloor Cp \rfloor} \oplus \mathbf{1}_{C - \lfloor Cp \rfloor}) \tag{10}$$

  In this context:
  - $\odot$ is the Hadamard product, representing the element-wise product of two matrices.
  - $\mathbf{s}_C$ is a random column vector composed of zeros and ones, indicating which channels are set to zero.
  - The permutation σ shuffles the elements of this vector.
  - $\oplus$ is the operator describing the concatenation of two vectors.
- **Signal masking**: This augmentation sets portions of the signals to zero. Specifically, given a masking percentage p and a predefined number of masking blocks k, the augmented signal can be defined as:

$$\tilde{\text{X}} = \text{X} \odot (\mathbf{1}_C \mathbf{m}^T) \tag{11}$$

  where:
  - $\mathbf{1}_C$ is a column vector of ones with length equal to the number of channels C,
  - $\mathbf{m}$ is a column mask vector of length equal to the signal length, constructed as a concatenation of blocks of ones and zeros, i.e.,

$$\mathbf{m} = \mathbf{1}_{b_1} \oplus \mathbf{0}_{b_2} \oplus \mathbf{1}_{b_3} \oplus \mathbf{0}_{b_4} \oplus \cdots \oplus \mathbf{0}_{b_{2k}} \oplus \mathbf{1}_{b_{2k+1}} \tag{12}$$

  with each b_i representing the length of the corresponding block. The total length of $\mathbf{m}$ is:

$$N = \sum_{i=1}^{2k+1} b_i$$

  and the total number of masked samples, corresponding to the sum of the zero blocks, is

$$Np = \sum_{i=2,4,6,\ldots,2k} b_i$$

  The length of each masked portion b_i is randomly generated, while ensuring that the total masked proportion equals the hyperparameter p.
- **Signal warp**: This augmentation randomly stretch or squeeze portions of the EEG signal along the time axis, proceeding as follows:

1. The EEG signal is divided into multiple chunks (segments).
2. Up to half of these segments are randomly chosen to be stretched, while the remaining segments are squeezed.

3. Based on the random selection of the segments to squeeze or stretch, a non-uniform time grid is constructed. The values of this grid are defined according to the stretch and squeeze strengths hyperparameters.
4. The EEG signal is interpolated onto this new grid using the Piecewise Cubic Hermite Interpolating Polynomial (PCHIP) [45].
5. the new signal is treated as uniformly sampled between 0 and the sequence length L, and a final PCHIP interpolation is performed to resample the signal back onto the original time grid.

Therefore, the augmented EEG sample $\tilde{\mathrm{X}}$ is computed as:

$$\tilde{\mathrm{X}} = pchip(pchip(\mathrm{X}, \mathbf{t}_{old}, \mathbf{t}_{nu}), \mathbf{t}_{\mathrm{u}}, \mathbf{t}_{old}) \tag{13}$$

where:
- $pchip(\mathrm{X}, \mathbf{t}_1, \mathbf{t}_2)$ is the function performing the interpolation of a signal from a time grid $\mathbf{t}_1$ to a time gird $\mathbf{t}_2$.
- $\mathbf{t}_{\mathrm{old}}$ is the original time grid.
- $\mathbf{t}_{\mathrm{nu}}$ is the non-uniform, warped time grid constructed at step 3.
- $\mathbf{t}_{\mathrm{u}}$ is the uniformly sampled time grid used for the final interpolation at step 5.

- **Phase Randomizer**: This augmentation randomly perturbs the phase of the EEG signal. The augmented sample is computed by applying the transformation:

$$\tilde{\mathrm{X}} = \mathrm{Re}(\mathcal{F}^{-1}[\mathcal{F}[\mathrm{X}] \odot (\mathbf{1}_c e^{\circ j s \boldsymbol{\phi}^T})]) \tag{14}$$

where:
- $\mathcal{F}[\mathrm{X}]$ and $\mathcal{F}^{-1}[\mathrm{X}]$ denote the discrete Fourier transform (DFT) and the inverse discrete Fourier transform (IDFT), respectively, applied to each channel of the EEG sample X.
- $e^{\circ X} = (e^{x_{ij}})$ is the Hadamard exponential, which computes the element-wise exponential of a matrix X.
- $\boldsymbol{\phi} \sim \mathcal{U}(0, 2\pi)^N$ is a vector of length N, where each element is independently drawn from a uniform distribution over $[0, 2\pi)$, used to randomize the phase of each EEG channel.
- s is a scalar between 0 and 1 defining the strength of the phase perturbation.

- **Phase Swap**: Presented in [44], this data augmentation consists in merging the amplitude and phase components of biosignals from different sources to help the model learn their coupling. Specifically, the amplitude and phase of two randomly selected EEG samples are extracted using the discrete Fourier transform. New samples are then generated by applying the inverse discrete Fourier transform, combining the amplitude from one sample with the phase from the other. This process is described by the following transformation:

$$\tilde{\mathrm{X}}^i = \mathrm{Re}\left(\mathcal{F}^{-1}\left[|\mathcal{F}[\mathrm{X}^i]| \odot e^{\circ j \arg(\mathcal{F}[\mathrm{X}^j])}\right]\right) \tag{15}$$

where:
- $|\mathcal{F}[\mathrm{X}^i]|$ is the amplitude of the EEG sample X^i.
- $\arg(\mathcal{F}[\mathrm{X}^j])$ is the phase of the EEG sample X^j.

Whenever possible, the same data augmentation is applied to all channels of an EEG sample and broadcasted across the batch dimension to improve computational efficiency. Additionally, data augmentation was performed during 75 % of the training iterations. In other words, when a batch is created, there is a 75 % probability that it will be augmented with the selected data augmentations; otherwise, the identity function is applied. This additional randomness helps balance the number of original and augmented samples provided to the model.

Table 3 reports the top three data augmentation strategies for each model, except for TransformEEG, whose results are presented in detail in Section 3.2. To select the optimal data augmentation, the median and IQR of balanced accuracy variations were computed using a 10-Outer, 5-Inner N-LNSO cross-validation. The number of inner folds was reduced for this specific analysis to halve the total number of required training sessions and account for computational limitations.

Table 3
Top three data augmentations for each model. The median and interquartile range (IQR) of balanced accuracy are reported as differences from a baseline obtained using 10-Outer, 5-Inner N-LNSO cross-validation, chosen due to computational limitations. The best data augmentations for TransformEEG are detailed in Section 3.2.

Model	Top 3 data augmentations	Δ Bal. Acc.		ARIS
		Median	IQR	
xEEGNet	Baseline	73.51	10.98	–
	Signal drift + phase swap	+2.57	−4.12	$1.31 \cdot 10^{-2}$
	Masking + phase swap	+2.80	−3.69	$1.28 \cdot 10^{-2}$
	Phase swap + sign flip	+1.58	−5.32	$1.04 \cdot 10^{-2}$
EEGNet	Baseline	74.41	6.61	–
	Chan. drop. + band noise	+1.28	−0.56	$1.46 \cdot 10^{-3}$
	Band noise	+0.58	−0.76	$9.00 \cdot 10^{-4}$
	Band noise + time reverse	+0.32	−0.73	$4.70 \cdot 10^{-4}$
ShallowNet	Baseline	77.03	11.37	–
	Chan. drop. + masking	+3.37	−3.35	$1.29 \cdot 10^{-2}$
	Chan. drop. + SNR scale	+2.62	−3.61	$1.08 \cdot 10^{-2}$
	Chan. drop.	+1.45	−4.38	$7.26 \cdot 10^{-3}$
ATCNet	Baseline	69.40	10.76	–
	Phase swap + masking	+5.78	−4.91	$3.81 \cdot 10^{-2}$
	Time reverse + phase swap	+5.64	−4.36	$3.30 \cdot 10^{-2}$
	SNR scaling + phase swap	+5.24	−4.62	$3.24 \cdot 10^{-2}$
EEGConformer	Baseline	77.39	7.58	–
	Chan. drop. + SNR scale	+1.88	−1.98	$6.32 \cdot 10^{-3}$
	Signal drift + band noise	+0.68	−2.04	$2.37 \cdot 10^{-3}$
	Chan. drop. + signal drift	+1.13	−1.15	$2.21 \cdot 10^{-3}$
DeepConvNet	Baseline	70.81	16.11	–
	Phase swap + band noise	+4.88	−8.26	$3.53 \cdot 10^{-2}$
	Chan. drop. + band noise	+4.67	−7.99	$3.27 \cdot 10^{-2}$
	Chan. drop. + phase swap	+4.18	−8.70	$3.19 \cdot 10^{-2}$
EEGResNet	Baseline	72.94	7.81	–
	Rand. phase + signal drift	+1.24	−1.55	$3.37 \cdot 10^{-3}$
	Phase swap + SNR scale	+0.98	−1.37	$2.33 \cdot 10^{-3}$
	Time reverse	+0.99	−0.86	$1.48 \cdot 10^{-3}$

2.4.3. Training hyperparameters

A fixed random seed value of 42 was used to minimize randomness in the code and enhance reproducibility, as recommended in [22]. An analysis of how the random seed affects the accuracy estimation of N-LNSO cross-validation is provided in Section A.3 of the Supplementary data. This analysis confirms that the study's conclusions are not influenced by the choice of the seed.

Model weights were initialized using Pytorch's default initialization settings and were trained with Adam optimizer ($\beta_1 = 0.75$, $\beta_2 = 0.999$, no weight decay) [46]. The batch size was set to 64, and the initial learning rate was $2.5 \cdot 10^{-4}$. An exponential scheduler with $\gamma = 0.99$ was used to reduce the learning rate after each epoch. The learning rate at the epoch i is computed as:

$$\mathrm{lr}_i = \mathrm{lr}_0 \times \gamma^i \tag{16}$$

Binary cross entropy was used as loss function. The maximum number of epochs was set to 300 to ensure convergence across all the selected models. To prevent overfitting, early stopping with a patience of 20 epochs was implemented as an additional regularization method. Early stopping can be integrated into N-LNSO cross-validation procedures, as each partition includes a unique triplet of training, validation, and test sets.

3. Results

This section presents the results of the comparative analysis of TransformEEG. It is structured as follows:

1. Section 3.1 presents the baseline results for TransformEEG and the seven models used for the comparison. These results are for a pipeline without data augmentation and threshold correction.

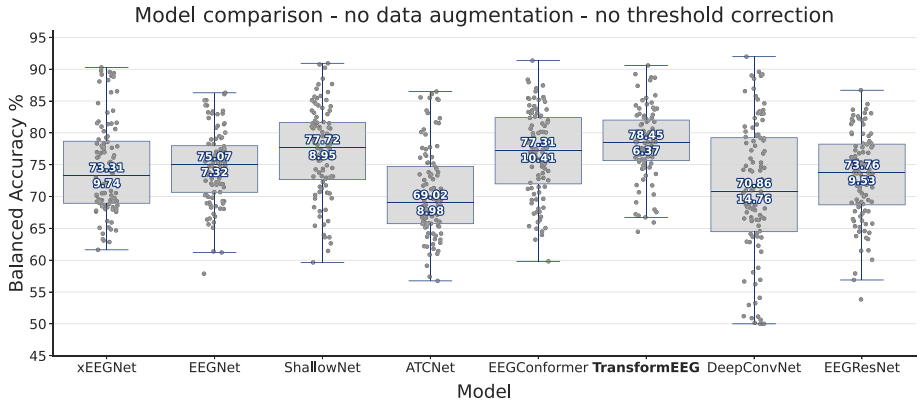

Fig. 2. Performance comparison of the selected models using a 10-outer, 10-inner N-LNSO cross-validation scheme. Models are organized in ascending order based on the number of learnable parameters from left to right. All models were trained without data augmentation or threshold correction to serve as a baseline reference. TransformEEG ranks first in median balanced accuracy and interquartile range. It also achieved the highest minimum accuracy. These results demonstrate the superior generalizability of the proposed TransformEEG architecture for the investigated task.

2. Sections 3.2 and 3.3 evaluate TransformEEG and the seven models used for the comparison within a richer training pipeline that includes data augmentation and threshold correction.
3. Section 3.4 explores how performance changes when window-level predictions are aggregated to create a single prediction for an entire subject's EEG.
4. Section 3.5 examines performance variations when the number of subjects is reduced, assessing the importance of data harmonization.
5. Section 3.6 presents an evaluation of TransformEEG and three machine learning models trained with hand-crafted features, including subject age.

During the presentation of the results, no statistical test outcomes will be reported. As explained in [17], the Nested-Leave-N-Subjects-Out (N-LNSO) cross-validation procedure generates more reliable performance estimates. However, it does not align with the key assumptions of statistical tests designed to identify significant differences in variance between models (e.g., Levene's test, Fligner-Killeen's test). Specifically, sample independence is violated because different N-LNSO splits share overlapping training sets. This violation increases the risk of Type I errors and can lead to misinterpretation of the results.

3.1. Baseline comparison

Fig. 2 presents the results of TransformEEG compared to the other seven selected models. No data augmentation or threshold correction was included at this stage to provide a baseline reference for subsequent analysis. TransformEEG ranks first in terms of median balanced accuracy (78.45 %) and IQR (6.37 %). It also achieved the highest minimum accuracy (64.46 %). ShallowNet and EEGConformer achieve comparable median accuracies (77.72 % and 77.31 %, respectively), but they exhibit higher variability in the results. Additionally, TransformEEG demonstrates a high median number of effective training epochs, defined as the number of epochs required to reach the best validation accuracy during training. This metric indicates how well the model's reduction in training loss is accompanied by a corresponding reduction in validation loss, reflecting better generalization to unseen data. A very low number of effective training epochs suggests that the model quickly reaches its optimal validation performance, which may be indicative of overfitting tendencies and lower generalization capabilities. On the contrary, a higher number of epochs can imply better generalization, as the model continues to improve over more training iterations. With a median of 21 effective training epochs, TransformEEG ranks among the best models, second only to the minimalist xEEGNet, which has a median of 37 epochs. This result demonstrates that, despite having 210,561 learnable parameters, TransformEEG does not rapidly overfit the training set as some comparable transformer architectures do, such as EEGConformer, which has a median of only two effective training epochs.

3.2. Comparison including data augmentation

This subsection examines how performance varies when an optimally selected data augmentation is incorporated into the training pipeline. It highlights how TransformEEG maintains leading performance within a fair comparison and, more generally, how this step can enhance performance and reduce variability across the models considered.

Fig. 3, Panels A and B, show the results of TransformEEG when different data augmentation compositions are applied during training, following the procedure described in Section 2.4.2. A summary of the results from the same analysis performed on the other seven models is presented in Table 3. Focusing on TransformEEG, several compositions are identified as a good candidate. In particular, using "time reverse" in combination with either "masking" or "phase swap" ensures a good balance between the median and IQR of the balanced accuracy. The combination of time reverse with phase swap results in a decreased IQR (from 6.4 % to 5.6 %) at the cost of a slight decrease of the median value (−1.3 %). However, this composition does not alter the power spectral density (PSD) of the input EEG window, which can be advantageous in scenarios where spectral information are combined with temporal patterns to enrich the quality of the representations (see Section 4). Masking with time reverse allows for a decrease of the IQR (from 6.4 % to 6.0 %) and a slight increase in the median value (+0.6 %), achieving the highest ARIS. Therefore, it was selected and tested against the optimal combination of other models.

Fig. 3, Panel C, shows the results of TransformEEG compared to the seven other selected deep learning models when optimal data augmentation is applied during training. TransformEEG ranks first in terms of median balanced accuracy (79.21 %) and interquartile range (5.97 %). As in the previous subsection, ShallowNet and EEGConformer achieve comparable median accuracies (78.85 % and 77.62 %, respectively), but they exhibit higher variability in their results. These results improve upon the baseline values reported in Section 3.1 and confirm that data augmentation can be effectively integrated into EEG deep learning pipelines to enhance the model's generalizability. Additionally, if the single outlier is discarded from the TransformEEG's N-LNSO set of training instances, the highest minimum balanced accuracy would rise from 64.5 % to 71.2 %. In fact, TransformEEG is the only model to have nearly all test accuracies (99 out of 100) above 70 %, achieving

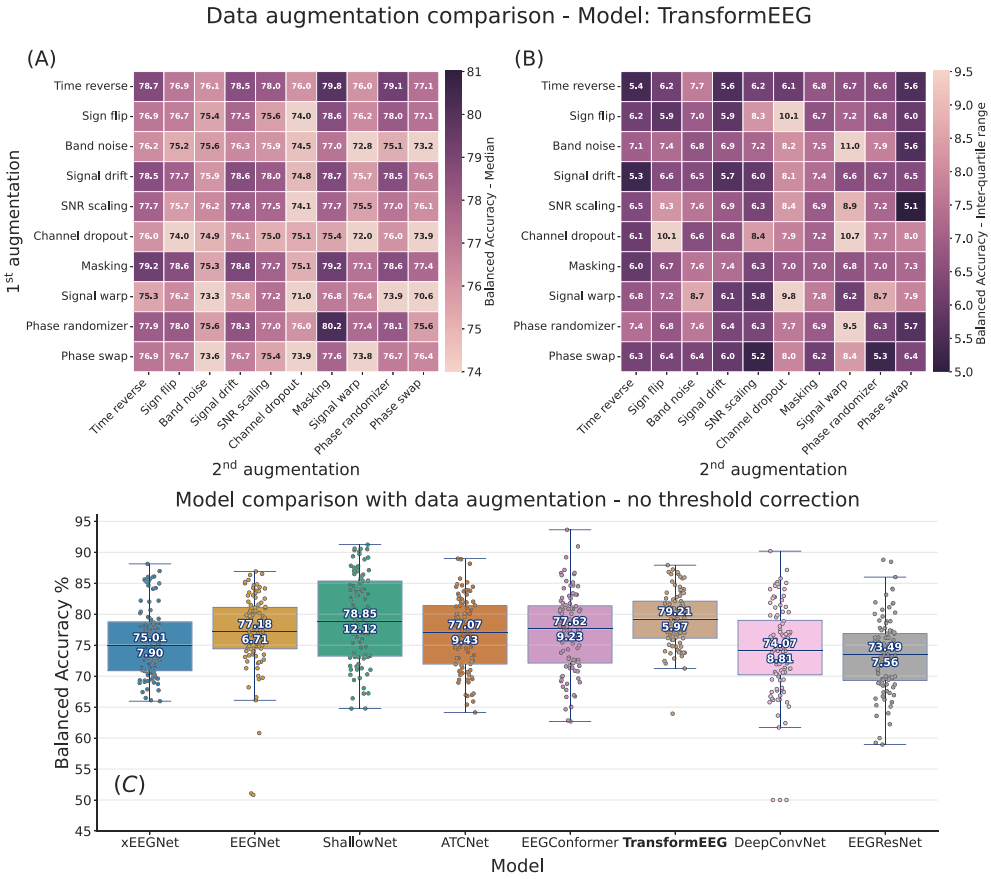

Fig. 3. Effect of data augmentation on the performance of TransformEEG and the other selected EEG-based deep learning models. Panels A and B display the median and interquartile range of balanced accuracy, respectively, when different data augmentation compositions are incorporated during the training of TransformEEG. For each augmentation combination, an N-LNSO cross-validation was performed. Panel C shows the results of a 10-outer 10-inner N-LNSO performance comparison between the selected models when the optimal data augmentation composition for each model is applied during training. Models are organized in ascending order based on the number of learnable parameters from left to right. TransformEEG ranks first in median balanced accuracy and interquartile range. These results improves baseline values and underscores the potential benefit of integrating properly tuned data during training.

the lowest [1st–99th] percentile range (16.05 %). Low variability revealed that TransformEEG can achieve competitive performance with more consistent results.

Considering the entire set of results, all the models achieved an increase in the median balanced accuracy. The interquartile range also decreases in most of the models, with ShallowNet being the only exception (from the baseline 8.95 % to 12.12 %). Despite the increase in the IQR, ShallowNet's [1st–99th] percentile range improved from 29.90 % to 25.81 %. These improvements over baseline values underscore the potential of integrating properly tuned data augmentations during the training of EEG deep learning models.

3.3. Comparison including threshold correction

Table 4 shows the results of TransformEEG against the other selected models when the threshold correction method is incorporated to the training pipeline. This approach consists of adjusting the default classification threshold of 0.5 by searching for an optimal value that maximizes the balanced accuracy of the validation set. This adjustment serves as a minor correction in case the trained model generates too aggressive or too conservative predictions, particularly when the class ratio is slightly unbalanced. Using the validation data avoids introducing data leakage, as test data cannot be used to inflate the results.

Even in this scenario, TransformEEG ranks first in terms of median and IQR of the balanced accuracy. It is the sole model to achieve a median balanced accuracy higher than 80 % while also reducing the IQR from the baseline value of 6.37 % to 5.74 %. ShallowNet maintains a comparable median accuracy (79.97 %) but with a higher variability of results. This denotes how EEG-specific deep learning models trained in combination with carefully selected data augmentations and label assignment methods can effectively address generalizability issues.

3.4. Comparison including subject-level predictions

This section explores how performance changes when using window aggregation to create a single prediction for an entire subject's EEG. This method is especially important in pathology classification, where it's more clinically relevant to know if an entire EEG recording indicates a disease, rather than just a small segment.

The analysis shows that TransformEEG maintains its strong performance, not only achieving leading results but also showing the lowest variability. This is illustrated in the boxplot in Fig. 4, which shows the results after predictions from all EEG windows are aggregated into a single, subject-level prediction for each recording. Predictions were produced using the following procedure:

- A prediction is made for each individual EEG window using a standard classification threshold of 0.5.
- A minimal ratio of positive windows needed to classify the entire recording as positive is calculated. This ratio is determined using the validation data, ensuring that data leakage is not introduced.
- Based on this ratio, a class label is assigned to each EEG in the test set, and the confusion matrix is computed.

Table 4
Balanced accuracy of the selected models trained with different training modalities and varying number of datasets. Acronyms in the "Training Method" column are: B (baseline), DA (data augmentation), T (threshold correction). IQR indicates interquartile range.

Models	# Param	Training method	Balanced accuracy					
			2 Datasets			4 Datasets		
			Median	IQR	[1st–99th]	Median	IQR	[1st–99th]
xEEGNet [33]	245	B	67.99	18.02	47.00	73.31	9.74	26.90
		B + DA	**77.28**	24.83	51.01	75.01	7.90	20.86
		B + DA + T	**77.35**	22.69	48.88	78.03	9.84	20.39
EEGNet [21]	2609	B	70.03	20.96	44.76	75.07	7.32	24.90
		B + DA	70.58	18.01	48.98	77.18	6.71	35.44
		B + DA + T	67.42	20.20	48.73	78.44	6.88	25.10
ShallowNet [38]	57,441	B	**78.73**	18.18	**41.73**	77.72	8.95	29.90
		B + DA	71.49	18.48	48.06	78.85	12.12	25.81
		B + DA + T	72.29	16.97	49.07	79.91	11.00	25.44
ATCNet [19]	146,629	B	72.73	15.97	44.83	69.02	8.98	28.71
		B + DA	73.02	17.56	46.95	77.07	9.43	23.46
		B + DA + T	71.89	17.32	49.14	76.85	9.66	23.71
EEGConformer [18]	191,153	B	75.98	15.83	45.14	77.31	10.41	25.18
		B + DA	75.46	16.67	49.56	77.62	9.23	28.13
		B + DA + T	75.77	**14.74**	50.00	77.72	9.59	27.71
TransformEEG	210,561	B	72.09	15.93	44.83	**78.45**	**6.37**	**23.33**
		B + DA	69.92	**14.39**	44.62	**79.21**	**5.97**	**16.05**
		B + DA + T	70.62	15.07	**43.42**	**80.10**	**5.74**	**18.21**
DeepConvNet [38]	287,901	B*	50.09	0.64	37.17	70.86	14.76	39.60
		B + DA	67.97	29.47	**40.09**	74.07	8.81	37.16
		B + DA + T	74.14	32.49	46.62	75.64	9.35	36.04
EEGResNet [39]	1,337,665	B	73.12	**13.65**	47.71	73.76	9.53	27.68
		B + DA	74.24	16.86	44.34	73.49	7.56	29.21
		B + DA + T	74.96	17.80	48.29	75.10	6.92	25.10

* Baseline results of DeepConvNet trained with two datasets were ignored because the model collapsed.

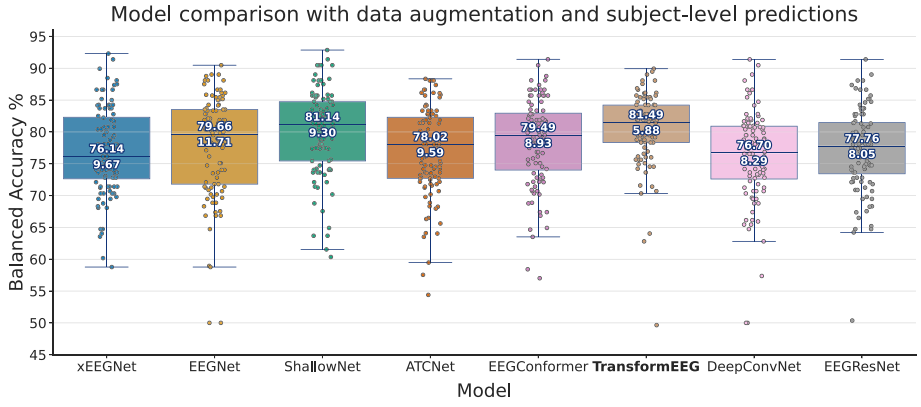

Fig. 4. Performance comparison of the selected models using a 10-outer, 10-inner N-LNSO cross-validation scheme. Models are arranged from left to right in order of increasing learnable parameters. All models were trained with their optimal data augmentation and predictions of windows of the same EEG recording were aggregated to provide subject-level predictions.

TransformEEG ranks first with a median balanced accuracy of 81.49 % and an IQR of 5.88 %. This performance improves on the results from the previous sections and maintains a significant gap in variability (IQR) compared to other models. While all models show improvement, ShallowNet is the only model to achieve a comparable median accuracy (81.14 %), although with a much higher IQR (9.30 %).

Despite the overall improvement, the boxplot reveals a few outliers. These may be due to a suboptimal choice for the minimum ratio of positive windows, highlighting the need for more advanced techniques that can robustly aggregate window-level predictions.

It is important to note a key limitation of this aggregation method: the number of samples in the confusion matrix is reduced. Since each subject can contribute more than 3 % to the total accuracy, each prediction error carries more weight. This makes it difficult to definitively state whether the performance improvements are solely due to the benefits of the aggregation procedure or if they are influenced by the reduced sample size.

3.5. Model scalability

This subsection examines performance variations when the amount of training data is reduced. It highlights generalizability limitations common to all architectures, including smaller models, which are favored in this type of analysis due to their lower number of parameters.

Previous Sections 3.1–3.3 demonstrate the efficacy of TransformEEG, which outperforms other models in terms of the interquartile range and the [1st–99th] percentile range of the balanced accuracy. These findings are consistent across different metrics, as confirmed in Section A.2 of the Supplementary data. However, training deep learning models such as TransformEEG often heavily depends on the amount of training data available. Table 4 illustrates how performance varies when the number of subjects decreases from 290 to 81, using data from the ds002778 and ds003490 datasets, as in [14]. Focusing on the baseline results (Table 4, "B" rows), performance variability across all investigated models increases substantially, even in lighter models such as xEEGNet (from

Table 5
Balanced accuracy comparison between TransformEEG and three machine learning models.

Models	Balanced accuracy		
	Median	IQR	[1st–99th]
Without age			
Logistic regression	72.76	9.37	28.00
SVM	73.43	9.54	33.73
Random Forest	75.39	6.89	19.45
With age			
Logistic regression	72.87	9.54	27.82
SVM	73.00	8.31	32.49
Random Forest	77.89	6.40	18.47
TransformEEG	**79.21**	**5.97**	**16.05**

9.74 % to 18.02 %). TransformEEG's interquartile range rises similarly, despite remaining one of the lowest among all the investigated models. Its median balanced accuracy also drops by 6.01 %, ranking below other lighter models such as ShallowNet. When training with only two datasets, adding data augmentation or threshold correction does not produce the same benefits observed when using data from all four datasets. This result emphasizes the importance of standardizing and aggregating EEG recordings from multiple centers to facilitate more effective learning of disease-specific EEG features and to ensure consistent results across different splits.

3.6. Comparison with machine learning models

This subsection presents a comparison between TransformEEG and three machine learning (ML) models: Logistic Regression, Support Vector Machine (SVM), and Random Forest (RF). It shows that while ML models can achieve good performance, they still underperform compared to TransformEEG in terms of both median accuracy and interquartile range.

Training a machine learning model requires a set of hand-crafted features. Based on recent literature [47], a set of 24 intra-channel and 12 inter-channel features, capturing information from both time and frequency domains, was selected. Intra-channel features were extracted for each of the 32 EEG channels, resulting in a feature vector of length 780. Feature extraction was performed on 16-second windows obtained after the segmentation step described in Section 2.2, ensuring that dataset length was preserved and comparisons remained fair and reliable.

For each ML model, N-LNSO splits were used to generate the same ensemble of test metrics as for TransformEEG. Model hyperparameters were optimized using the validation set of each N-LNSO split to prevent data leakage. The best-performing model was subsequently evaluated on the corresponding test set. The complete list of selected features and optimized hyperparameters for each model is provided in Section A.8 of the Supplementary data.

Table 5 shows that TransformEEG remains the best-performing model, even when ML models are trained with age included as an additional predictor to facilitate the identification of young controls from the ds004148 dataset. These results suggest that TransformEEG can automatically extract features that generalize better to unseen subjects compared to the provided hand-crafted features.

4. Discussion

EEG-based deep learning models often exhibit high variability, which can only be assessed through the use of appropriate model evaluation techniques [17]. This aspect is particularly relevant in pathology detection tasks, where the direct association between patient ID and health status (class label) can strongly influence training dynamic, serving as a shortcut for minimizing the training loss [48]. Nested approaches, such as the N-LNSO, offer more reliable performance estimates but also reveal strong variability related to changes in data splits. This variability complicates the comparison of different EEG deep learning models, as median accuracy values become less informative due to the wide range of accuracies observed across different splits.

TransformEEG was specifically designed to advance current generalizability challenges in EEG pathology detection, focusing on Parkinson's disease. It features a depthwise convolutional tokenizer that specializes in generating tokens describing local time segments of the EEG window with channel-specific features. This tokenizer differs from EEGNet-like convolutional blocks and enables more effective feature mixing operations within the transformer's self-attention layers, as presented in Section 2.3.

Baseline results in Section 3.1 show that TransformEEG ranks first in terms of the median (78.45 %) and IQR (6.37 %) balanced accuracy. It also achieved the highest minimum accuracy (64.5 %). These results improve further if additional regularization techniques, such as data augmentation, are applied during training. After applying an optimal data augmentation composition for each model and adjusting the classification threshold, TransformEEG achieves a median balanced accuracy of 80.10 % and an IQR of 5.74 %, remaining the best among the models investigated (see Section 3.3).

The strength of TransformEEG lies in the clear and complementary roles assigned to each of its modules. For example, the depthwise convolutional tokenizer does not perform feature mixing between EEG channels, unlike other EEG transformer models [18]. Instead, it specializes in extracting representations based on local temporal patterns within each individual EEG channel. This design allows the transformer encoder, where most of the model's parameters reside, to be the sole module responsible for feature mixing across channels. The combination of convolutional and attention layers enhances the network's ability to capture both local and global patterns within the EEG window, leading to more consistent performance across different N-LNSO splits. This consistency highlights the superior capability of the model to generalize to diverse data and partitions.

To enable attention layers to capture long-range temporal patterns without overfitting the training set, it is essential to use a sufficient number of EEG recordings from different subjects. Results in Section 3.5 investigate how model performance changes when the number of subjects is reduced from 290 to 81, replicating the experimental setting presented in [14]. When only 81 subjects are used, performance variability across all investigated models increases drastically. TransformEEG was not excluded by this trend. The interquartile range increase of 9.56 % and the median balanced accuracy drop of 6.01 %. This result suggests that a larger number of subjects is necessary to enable the effective training of EEG deep learning architectures. It also emphasizes the need for creating novel large-scale, multi-center datasets.

As discussed in recent works, EEG deep learning classifiers can easily exploit the unique association between labels and participant IDs as a shortcut to minimize the training loss [17,32]. This association induces severe overfitting and prevents the model from learning disease-specific representations. Inter-subject heterogeneity is therefore a primary source of variability, and the most effective way to address this challenge is by aggregating data from different sources, as done in this study. Harmonization tools like BIDSAlign [29] can preprocess and standardize EEG recordings from multiple centers, enabling the training of more robust EEG deep learning models and facilitating fair benchmarking. Open platforms like OpenNeuro provide access to a wide range of EEG datasets. Although these datasets may differ in experimental paradigms (e.g., resting-state vs. task-based) or population characteristics, they offer an opportunity to enhance model generalizability, especially when label-free paradigms, such as self-supervised learning, are used [49]. Self-supervised learning is a recent approach designed to improve model generalizability by learning meaningful representations from large, unlabeled datasets. It has been widely adopted to boost the performance of EEG deep learning models, often yielding promising results. Future research could explore how aggregating heterogeneous,

unlabeled data with self-supervised methods impacts model generalizability, building on the findings of this study. This approach would also enable the exploration of alternative transformer architectures, which are known to require large datasets for optimal performance. Ultimately, these strategies have the potential to advance the field and improve the reliability of EEG-based deep learning decision support systems [50].

TransformEEG demonstrates promising results in Parkinson's disease detection and lays the foundation for addressing generalizability issues in EEG-based deep learning pathology classification tasks [32]. However, model generalizability depends not only on the architecture design, but also on how data are preprocessed, partitioned, and used to train and evaluate the model. TransformEEG is a model based on temporal analysis. It takes an EEG window in the time domain as input and outputs the probability that the window belongs to one of the investigated diseases. While this approach aligns with an optimal engineering solution for addressing the challenges of small dataset sizes and GPU memory limitations, it does not necessarily represent the best approach for disease detection. In pathology classification tasks, it is of greater clinical interest to determine whether the entire EEG originates from an individual with the target disease, rather than focusing solely on a single time window. Therefore, voting or aggregation procedures should be considered and carefully integrated into the evaluation process. Not all EEG segments are equally informative of a neurological disease, especially when short window lengths are used. Artifacts or involuntary movements can obscure disease-specific brain dynamics and alter the segment's power spectral density. Aggregation procedures can help address this issue by smoothing disease detection predictions across all segments from the same EEG recording. An analysis of aggregation procedures is presented in Section 3.4, demonstrating that TransformEEG's performance improves, achieving a median balanced accuracy of 81.49 %. Importantly, this improvement does not negatively impact the IQR, which remains the lowest (5.88 %) among all the selected models. However, this analysis is based on a single approach, and future research should explore more advanced aggregation methods that build on the findings of this study.

Contrary to other models such as xEEGNet, TransformEEG does not process the EEG window in the time domain to specifically extract features that are linearly proportional to the mean power of EEG bands across channels. While this design choice enhances the performance of the model, it would be of great clinical interest to assess whether combining both temporal and spectral information could further improve model generalizability. Researchers have already explored approaches that integrate temporal and spectral information to boost model performance, typically by extracting the spectrogram of an EEG window [13]. This is also supported in [51], where it is stated that the integration of spectral information, such as the power spectral density (PSD), can aid in the assessment of certain neurodegenerative disorders. If resting-state data are used, as in this study, a potential strategy can involve incorporating the PSD of the EEG segment in a separate branch to generate an additional set of features, or tokens if the model includes transformer blocks. Temporal and spectral information can then be combined with mid- or late-fusion strategies. See Section A.6.1 of the Supplementary data for a preliminary analysis on the integration of the PSD within the TransformEEG model.

Beyond the integration of temporal and spectral information, future work should also investigate methods that combine spatial, spectral, and temporal features in a unified framework. A promising direction is highlighted in recent work on fine-grained spatial-frequency-time representations for EEG decoding tasks [52], which demonstrates how jointly modeling these three dimensions can enhance classification performance, particularly in motor imagery BCI applications. Similar principles could be applied to capture disease-specific patterns across spatially distributed brain regions. Incorporating such approaches within models like TransformEEG may improve their ability to generalize by leveraging richer and more structured representations of EEG activity. This could involve adapting tokenization mechanisms to explicitly encode spatial location and frequency-band information alongside temporal dynamics.

Results in Section 3 demonstrate that TransformEEG can learn features that improve accuracy on unseen subjects. This suggests that the learned representations might capture more effectively disease-specific characteristics rather than biometric properties of the EEG signal. However, this study focuses on Parkinson's disease detection using a binary classification task. Parkinson's disease is a complex and heterogeneous disorder characterized by a variety of symptoms that can lead to different disease subtypes, often influenced by the severity of cognitive decline. Differentiating these subtypes is of great clinical importance, as it is essential for developing personalized therapies. The availability of new, large, dedicated datasets will facilitate their integration with existing data and support the investigation of deep learning approaches to automatically identify PD subcategories within a multi-class classification framework. This aggregation can address current challenges, such as overlapping clinical symptoms and involved brain regions [53], and enhance the baseline performances reported in recent studies on similar classification problems [14,33]. In addition, future research on different neurodegenerative diseases could explore how multimodal approaches might address generalizability issues and improve model reliability. Combining EEG signals with other neuroimaging (e.g., MRI, fMRI) or clinical data (e.g., MoCA, MMSE) can capture complementary information, enhancing feature quality at both the unimodal and multimodal levels.

TransformEEG represents the first step toward more generalizable deep learning-based EEG analysis. However, it is necessary to further validate the results with additional datasets, particularly those focused on the early stages of Parkinson's disease. Another important limitation concerns the mixing of younger and older control subjects from different datasets. Although this strategy increases the number of control samples and helps reduce inter-subject variability, it may introduce age-related confounds that the model could exploit. Future work should investigate the extent to which subject-specific features are encoded, how strongly they correlate with covariates, and whether they influence model predictions. A preliminary discussion of this topic is provided in Section A.9 of the Supplementary data, with an additional analysis on the CAUEEG [54] dataset provided in Section A.9.4.

Following this direction, future studies should also focus on designing rigorous XAI experiments to provide deeper insights into the features learned by EEG deep learning models, including the proposed TransformEEG. Such analyses should consider the predictions of all input EEG windows, using established techniques such as GradCAM-like methods [55] across all models trained during the cross-validation scheme. Additionally, XAI-specific modifications to the model architecture should be introduced to map intermediate outputs to scalp topographies and temporally related oscillations, as demonstrated in the xEEGNet model [33].

Addressing this limitation, along with the future directions outlined in this section, will improve the reliability of EEG-based deep learning models for pathology detection. These improvements will also facilitate the integration of such systems into real-world clinical settings, ultimately supporting clinicians and patients more effectively.

5. Conclusion

This work introduces TransformEEG, a novel hybrid convolutional-transformer model designed to address generalizability issues in Parkinson's disease detection from EEG data. TransformEEG features a depthwise convolutional tokenizer that specializes in generating tokens representing temporally local segments of the EEG window with channel-specific features. This module enables more effective feature mixing within the transformer encoder, which is included to capture long-range temporal patterns. The model was evaluated using harmonized EEG recordings from four publicly available datasets, comprising

290 subjects (140 Parkinson's Disease individuals and 150 healthy controls), through a Nested-Leave-N-Subjects-Out cross-validation scheme. When trained with a sufficient amount of data, TransformEEG achieved promising results, ranking first among seven other consolidated EEG deep learning models in terms of median balanced accuracy (80.10 %), interquartile range (5.74 %). These findings demonstrate that designing deep learning models tailored to EEG data and evaluating them on large, multi-center datasets can help address generalizability challenges, despite the path to solutions that generalize to clinical settings remains an open challenge for the global scientific community.

CRediT authorship contribution statement

Federico Del Pup: Conceptualization, Methodology, Software, Visualization, Writing – original draft. **Riccardo Brun:** Methodology, Software, Writing – review & editing. **Filippo Iotti:** Methodology, Software, Writing – review & editing. **Edoardo Paccagnella:** Software (EEGConformer), Writing – review & editing. **Mattia Pezzato:** Software (Phase Swap), Writing – review & editing. **Sabrina Bertozzo:** Writing – review & editing. **Andrea Zanola:** Methodology, Writing – review & editing. **Louis Fabrice Tshimanga:** Methodology, Writing – review & editing. **Henning Müller:** Writing – review and editing. **Manfredo Atzori:** Supervision, Funding acquisition, Project administration, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This document is the result of the research project funded in part by the European Unions Horizon Europe research and innovation programme under Grant agreement no 101137074—HEREDITARY, in part by the STARS@UNIPD funding program of the University of Padua, Italy, project: MEDMAX.

Appendix A. Supplementary data

Supplementary data to this article can be found online at doi:https://doi.org/10.1016/j.neucom.2025.132075.

Data availability

The code used to produce both results and figures is openly available at https://github.com/MedMaxLab/transformeeg. All data that support the findings of this study are openly available within the OpenNeuro platform.

References

[1] T. Pringsheim, N. Jette, A. Frolkis, T.D.L. Steeves, The prevalence of Parkinson's disease: a systematic review and meta-analysis, Mov. Disord. 29 (13) (2014) 1583–1590, https://doi.org/10.1002/mds.25945
[2] W. Dauer, S. Przedborski, Parkinson's disease: mechanisms and models, Neuron 39 (6) (2003) 889–909, https://doi.org/10.1016/S0896-6273(03)00568-3
[3] S. Sveinbjornsdottir, The clinical symptoms of parkinson's disease, J. Neurochem. 139 (S1) (2016) 318–324, https://doi.org/10.1111/jnc.13691
[4] S.-Y. Lim, A.E. Lang, The nonmotor symptoms of Parkinson's disease—an overview, Mov. Disord. 25 (S1) (2010) S123–S130, https://doi.org/10.1002/mds.22786
[5] M.J. Armstrong, M.S. Okun, Diagnosis and treatment of Parkinson disease: a review, Jama 323 (6) (2020) 548–560, https://doi.org/10.1001/jama.2019.22360
[6] L. Shirahige, M. Berenguer-Rocha, S. Mendonça, S. Rocha, M.C. Rodrigues, K. Monte-Silva, Quantitative electroencephalography characteristics for Parkinson's disease: a systematic review, J. Parkinson's Dis. 10 (2) (2020) 455–470, https://doi.org/10.3233/JPD-191840
[7] V.J. Geraedts, L.I. Boon, J. Marinus, A.A. Gouw, J.J. van Hilten, C.J. Stam, M.R. Tannemaat, M.F. Contarino, Clinical correlates of quantitative EEG in Parkinson disease: a systematic review, Neurology 91 (19) (2018) 871–883, https://doi.org/10.1212/WNL.0000000000006473
[8] R. Soikkeli, J. Partanen, H. Soininen, A. Pääkkönen, P. Riekkinen, Slowing of EEG in Parkinson's disease, Electroencephalogr. Clin. Neurophysiol. 79 (3) (1991) 159–165, https://doi.org/10.1016/0013-4694(91)90134-P
[9] A.M. Maitin, J.P. Romero Muñoz, A.J. Garcıa-Tejedor, Survey of machine learning techniques in the analysis of EEG signals for Parkinson's disease: a systematic review, Appl. Sci. 12 (14) (2022) 6967, https://doi.org/10.3390/app12146967
[10] H.W. Loh, W. Hong, C.P. Ooi, S. Chakraborty, P.D. Barua, R.C. Deo, J. Soar, E.E. Palmer, U.R. Acharya, Application of deep learning models for automated identification of Parkinson's disease: a review (2011–2021), Sensors 21 (21) (2021) 7034, https://doi.org/10.3390/s21217034
[11] S.K. Khare, V. Bajaj, U.R. Acharya, PDCNNet: an automatic framework for the detection of Parkinson's disease using EEG signals, IEEE Sens. J. 21 (15) (2021) 17017–17024, https://doi.org/10.1109/JSEN.2021.3080135
[12] E. Balaji, D. Brindha, V.K. Elumalai, R. Vikrama, Automatic and non-invasive Parkinson's disease diagnosis and severity rating using LSTM network, Appl. Soft Comput. 108 (2021) 107463, https://doi.org/10.1016/j.asoc.2021.107463
[13] A. Miltiadous, E. Gionanidis, K.D. Tzimourta, N. Giannakeas, A.T. Tzallas, DICE-net: a novel convolution-transformer architecture for Alzheimer detection in EEG signals, IEEE Access 11 (2023) 71840–71858, https://doi.org/10.1109/ACCESS.2023.3294618
[14] F. Del Pup, A. Zanola, L. Fabrice Tshimanga, A. Bertoldo, M. Atzori, The more, the better? Evaluating the role of EEG preprocessing for deep learning applications, IEEE Trans. Neural Syst. Rehabil. Eng. 33 (2025) 1061–1070, https://doi.org/10.1109/TNSRE.2025.3547616
[15] L. Pion-Tonachini, K. Kreutz-Delgado, S. Makeig, ICLabel: an automated electroencephalographic independent component classifier, dataset, and website, NeuroImage 198 (2019) 181–197, https://doi.org/10.1016/j.neuroimage.2019.05.026
[16] T.R. Mullen, C.A.E. Kothe, Y.M. Chi, A. Ojeda, T. Kerth, S. Makeig, T.-P. Jung, G. Cauwenberghs, Real-time neuroimaging and cognitive monitoring using wearable dry EEG, IEEE Trans. Biomed. Eng. 62 (11) (2015) 2553–2567, https://doi.org/10.1109/TBME.2015.2481482
[17] F. Del Pup, A. Zanola, L.F. Tshimanga, A. Bertoldo, L. Finos, M. Atzori, 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. 196 (2025) 110608, https://doi.org/10.1016/j.compbiomed.2025.110608
[18] Y. Song, Q. Zheng, B. Liu, X. Gao, EEG conformer: convolutional transformer for EEG decoding and visualization, IEEE Trans. Neural Syst. Rehabil. Eng. 31 (2023) 710–719, https://doi.org/10.1109/TNSRE.2022.3230250
[19] H. Altaheri, G. Muhammad, M. Alsulaiman, Physics-informed attention temporal convolutional network for EEG-based motor imagery classification, IEEE Trans. Ind. Inform. 19 (2) (2023) 2249–2258, https://doi.org/10.1109/TII.2022.3197419
[20] N. Delfan, M. Shahsavari, S. Hussain, R. Damaševičius, U.R. Acharya, A hybrid deep spatiotemporal attention-based model for parkinson's disease diagnosis using resting state EEG signals, Int. J. Imaging Syst. Technol. 34 (4) (2024) e23120, https://doi.org/10.1002/ima.23120
[21] V.J. Lawhern, A.J. Solon, N.R. Waytowich, S.M. Gordon, C.P. Hung, B.J. Lance, EEGNet: a compact convolutional neural network for EEG-based brain–computer interfaces, J. Neural Eng. 15 (5) (2018) 056013, https://doi.org/10.1088/1741-2552/aace8c
[22] F. Del Pup, M. Atzori, Toward improving reproducibility in neuroimaging deep learning studies, Front. Neurosc. 18 (2024) https://doi.org/10.3389/fnins.2024.1509358
[23] C.J. Markiewicz, K.J. Gorgolewski, F. Feingold, R. Blair, Y.O. Halchenko, E. Miller, N. Hardcastle, J. Wexler, O. Esteban, M. Goncavles, A. Jwa, R. Poldrack, The Openneuro resource for sharing of neuroscience data, eLife 10 (2021) e71774, https://doi.org/10.7554/eLife.71774
[24] C.R. Pernet, S. Appelhoff, K.J. Gorgolewski, G. Flandin, C. Phillips, A. Delorme, R. Oostenveld, EEG-BIDS, an extension to the brain imaging data structure for electroencephalography, Sci. Data 6 (1) (2019) 103, https://doi.org/10.1038/s41597-019-0104-8
[25] Y. Wang, W. Duan, D. Dong, L. Ding, X. Lei, A test-retest resting and cognitive state EEG dataset (2022) https://doi.org/10.18112/openneuro.ds004148.v1.0.1
[26] A.P. Rockhill, N. Jackson, J. George, A. Aron, N.C. Swann, UC SAN Diego resting state EEG data from patients with parkinson's disease (2021) https://doi.org/10.18112/openneuro.ds002778.v1.0.5
[27] J.F. Cavanagh, EEG: 3-stim auditory oddball and rest in parkinson's (2021) https://doi.org/10.18112/openneuro.ds003490.v1.1.0
[28] A. Singh, R. Cole, A. Espinoza, J. Cavanagh, N. Narayanan, Rest eyes open (2023) https://doi.org/10.18112/openneuro.ds004584.v1.0.0
[29] A. Zanola, F. Del Pup, C. Porcaro, M. Atzori, BIDSAlign: a library for automatic merging and preprocessing of multiple EEG repositories, J. Neural Eng. 21 (4) (2024) 046050, https://doi.org/10.1088/1741-2552/ad6a8c
[30] A.J. Bell, T.J. Sejnowski, An information-maximization approach to blind separation and blind deconvolution, Neural Comput. 7 (6) (1995) 1129–1159, https://doi.org/10.1162/neco.1995.7.6.1129
[31] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, et al., Pytorch: an imperative style, high-performance deep learning library, Adv. Neural Inf. Process. Syst. 32 (2019) https://doi.org/10.48550/arXiv.1912.01703
[32] G. Brookshire, J. Kasper, N.M. Blauch, Y.C. Wu, R. Glatt, D.A. Merrill, S. Gerrol, K.J. Yoder, C. Quirk, C. Lucero, Data leakage in deep learning studies of translational EEG, Front. Neurosci. 18 (2024) https://doi.org/10.3389/fnins.2024.1373515
[33] A. Zanola, L. Fabrice Tshimanga, F. Del Pup, M. Baiesi, M. Atzori, xEEGNett: towards explainable AI in EEG dementia classification, J. Neural Eng. 22 (4) (2025) 046042, https://doi.org/10.1088/1741-2552/adf6e6
[34] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L.U. Kaiser, I. Polosukhin, Attention is all you need, in: Advances in Neural Information Processing Systems, vol. 30, Curran Associates, Inc., 2017, pp. 5998–6008, https://doi.org/10.48550/arXiv.1706.03762

[35] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: pre-training of deep bidirectional transformers for language understanding, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186, https://doi.org/10.18653/v1/N19-1423
[36] F. Del Pup, A. Zanola, L.F. Tshimanga, P.E. Mazzon, M. Atzori, SelfEEG: a Python library for self-supervised learning in electroencephalography, J. Open Source Softw. 9 (95) (2024) 6224, https://doi.org/10.21105/joss.06224
[37] M.L. Waskom, Seaborn: statistical data visualization, J. Open Source Softw. 6 (60) (2021) 3021, https://doi.org/10.21105/joss.03021
[38] R.T. Schirrmeister, J.T. Springenberg, L.D.J. Fiederer, M. Glasstetter, K. Eggensperger, M. Tangermann, F. Hutter, W. Burgard, T. Ball, Deep learning with convolutional neural networks for EEG decoding and visualization, Hum. Brain Mapp. 38 (11) (2017) 5391–5420, https://doi.org/10.1002/hbm.23730
[39] K.H. Cheah, H. Nisar, V.V. Yap, C.-Y. Lee, G.R. Sinha, Optimizing residual networks and VGG for classification of EEG signals: identifying ideal channels for emotion recognition, J. Healthc. Eng. 2021 (1) (2021) 5599615, https://doi.org/10.1155/2021/5599615
[40] M. Grandini, E. Bagli, G. Visani, Metrics for multi-class classification: an overview, arXiv Preprint, 2020, https://doi.org/10.48550/arXiv.2008.05756
[41] E. Lashgari, D. Liang, U. Maoz, Data augmentation for deep-learning-based electroencephalography, J. Neurosci. Methods 346 (2020) 108885, https://doi.org/10.1016/j.jneumeth.2020.108885
[42] C. Rommel, J. Paillard, T. Moreau, A. Gramfort, Data augmentation for learning predictive models on EEG: a systematic comparison, J. Neural Eng. 19 (6) (2022) 066020, https://doi.org/10.1088/1741-2552/aca220
[43] A. Le Guennec, S. Malinowski, R. Tavenard, Data augmentation for time series classification using convolutional neural networks, in: ECML/PKDD Workshop on Advanced Analytics and Learning on Temporal Data, Riva Del Garda, Italy, 2016, pp. 8, https://shs.hal.science/halshs-01357973.
[44] A. Lemkhenter, P. Favaro, Boosting generalization in bio-signal classification by learning the phase-amplitude coupling, in: Pattern Recognition, Springer International Publishing, Cham, 2021, pp. 72–85, https://doi.org/10.1007/978-3-030-71278-5_6
[45] F.N. Fritsch, J. Butland, A method for constructing local monotone piecewise cubic interpolants, SIAM J. Sci. Stat. Comput. 5 (2) (1984) 300–304, https://doi.org/10.1137/0905021
[46] D. Kingma, J. Ba, Adam: a method for stochastic optimization, in: 3rd International Conference on Learning Representations (ICLR), San Diega, CA, USA, 2015, pp. 13, https://doi.org/10.48550/arXiv.1412.6980
[47] A.M. Maitin, J.P. Romero Muñoz, Á.J. García-Tejedor, Survey of machine learning techniques in the analysis of EEG signals for Parkinson's disease: a systematic review, Appl. Sci. 12 (14) (2022) 6967, https://doi.org/10.3390/app12146967
[48] H.-T. Lee, H.-R. Cheon, S.-H. Lee, M. Shim, H.-J. Hwang, Risk of data leakage in estimating the diagnostic performance of a deep-learning-based computer-aided system for psychiatric disorders, Sci. Rep. 13 (1) (2023) 16633, https://doi.org/10.1038/s41598-023-43542-8
[49] M.H. Rafiei, L.V. Gauthier, H. Adeli, D. Takabi, Self-supervised learning for electroencephalography, IEEE Trans. Neural Netw. Learn. Syst. 35 (2) (2024) 1457–1471, https://doi.org/10.1109/TNNLS.2022.3190448
[50] S. Hussain, Challenges and future research directions in automated detection of mental illness using machine learning, Acad. Bull. Ment. Health. 2 (2024) 56–60, https://doi.org/10.25259/ABMH_19_2024
[51] C. Babiloni, R. Lizio, N. Marzano, P. Capotosto, A. Soricelli, A.I. Triggiani, S. Cordone, L. Gesualdo, C. Del Percio, Brain neural synchronization and functional coupling in Alzheimer's disease as revealed by resting state EEG rhythms, Int. J. Psychophysiol. 103 (2016) 88–102, Research on Brain Oscillations and Connectivity in A New Take-Off State. https://doi.org/10.1016/j.ijpsycho.2015.02.008
[52] G. Liu, R. Zhang, L. Tian, W. Zhou, Fine-grained spatial-frequency-time framework for motor imagery brain–computer interface, IEEE J. Biomed. Health Inform. 29 (6) (2025) 4121–4133, https://doi.org/10.1109/JBHI.2025.3536212
[53] R. Nardone, L. Sebastianelli, V. Versace, L. Saltuari, P. Lochner, V. Frey, S. Golaszewski, F. Brigo, E. Trinka, Y. Höller, Usefulness of EEG techniques in distinguishing frontotemporal dementia from Alzheimer's disease and other dementias, Dis. Markers 2018 (1) (2018) 6581490, https://doi.org/10.1155/2018/6581490
[54] M.J. Kim, Y.C. Youn, J. Paik, Deep learning-based EEG analysis to classify normal, mild cognitive impairment, and dementia: algorithms and dataset, NeuroImage 272 (2023) 120054, https://doi.org/10.1016/j.neuroimage.2023.120054
[55] R.R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra, Grad-CAM: visual explanations from deep networks VIA gradient-based localization, in: 2017 IEEE International Conference on Computer Vision (ICCV), 2017, pp. 618–626, https://doi.org/10.1109/ICCV.2017.74

Author biography

Federico Del Pup received a B.Sc. and an M.Sc. in Bioengineering from the University of Padua, Italy, in 2019 and 2022. He is currently pursuing the Ph.D. degree in Bioengineering at the Department of Information Engineering of the University of Padua. In 2025, Federico was a visiting researcher at the Institute of Information Systems of the University of Applied Sciences Western Switzerland (HES-SO Valais). His research interests include biomedical signal processing and analysis using deep learning methods, with a particular focus on non-invasive neuroimaging data such as EEG.

Riccardo Brun received his B.Sc. degree in biomedical engineering from the University of Padua, Italy, in 2023. He is currently pursuing an M.Sc. degree in bioengineering for neuroscience at the Department of Information Engineering of the University of Padua. He is conducting his thesis in collaboration with the Reykjavík University, Iceland. His work focuses on applying deep learning and machine learning techniques to biomedical signals, with a growing interest in human–machine interaction and robotics.

Filippo Iotti received his B.Sc. degree in biomedical engineering from the University of Padua, Italy, in 2023. He is currently pursuing an M.Sc. degree in bioengineering for neuroscience at the Department of Information Engineering of the University of Padua. He is conducting his thesis in collaboration with the KTH Royal Institute of Technology in Stockholm, Sweden. His research interests include deep learning application for medical imaging data processing and analysis.

Edoardo Paccagnella received a B.Sc. in Bioengineering from the University of Padua, Italy, in 2023. He is currently pursuing an M.Sc. degree in bioengineering for neuroscience at the Department of Information Engineering of the University of Padua. He is conducting his thesis in collaboration with the KTH Royal Institute of Technology in Stockholm, Sweden. His work focuses on the neuroimaging data analysis through deep learning methods.

Mattia Pezzato received a B.Sc. degree in Biomedical Engineering from the University of Padua, Italy, in 2023. He is currently pursuing an M.Sc. degree in bioengineering for neuroscience at the Department of Information Engineering of the University of Padua. He is conducting his thesis research at the Venetian Institute of Molecular Medicine, focusing on brain plasticity and reorganization across health and disease conditions using multimodal neuroimaging data (EEG and MRI). His research interests include neurorehabilitation and neuromodulation strategies.

Sabrina Bertozzo received a B.Sc. in biomedical engineering from the University of Padua, Italy, in 2023. She is currently pursuing an M.Sc. degree in bioengineering for neuroscience at the Department of Information Engineering of the University of Padua. Her research interests include neural technologies for biomedical applications and neuroimaging data analysis using deep learning methods.

Andrea Zanola received a B.Sc. in physics and an M.Sc. in physics of data from the University of Padua, Italy, in 2020 and 2022. He is currently pursuing the Ph.D. degree in Neuroscience at the Padova Neuroscience center. In 2024, Andrea was a visiting researcher at the Venetian Institute of Molecular Medicine. In 2025, he was a visiting researcher at the Institute of Information Systems of the University of Applied Sciences Western Switzerland (HES-SO Valais). His research interests focus on multimodal deep learning, including the integration of imaging, tabular, and EEG data, as well as clustering applied to neurological diseases.

Louis Fabrice Tshimanga received a B.Sc. in biomedical engineering from the Polytechnic University of Milan, Italy, in 2018 and an M.Sc. in data science from the University of Milano-Bicocca, Italy, in 2021. He is currently pursuing the Ph.D. degree in Neuroscience at the Padova Neuroscience Center, Italy. In 2025, Louis Fabrice was visiting researcher at the Institute of Information Systems of the University of Applied Sciences Western Switzerland (HES-SO Valais). His research interests focus on machine learning application to multimodal data in neuroscience, particularly to MRI, EEG and psychometric data.

Henning Müller studied medical informatics at the University of Heidelberg, Germany, then worked at Daimler-Benz research in Portland, OR, USA. He received his Ph.D. degree at the University of Geneva, Switzerland with a research stay at Monash University, Melbourne, Australia. Since 2007, he has been a full professor at the HES-SO Valais. Since 2014, he is professor at the medical faculty of the University of Geneva. Henning was the coordinator of the ExaMode EU project, the coordinator of the Khresmoi EU project, and the scientific coordinator of the VISCERAL EU project. He is also the initiator of the ImageCLEF benchmark.

Manfredo Atzori received a M.Sc. in Physics and a Ph.D. in Bioengineering in 2006 and 2009 from the University of Padua, Italy. He is Assistant Professor at the Department of Neuroscience of the University of Padova, Italy, and a research scientist at the Institute of Information Systems of the University of Applied Sciences Western Switzerland (HES-SO Valais). He was the Scientific Coordinator of the Horizon 2020 project ExaMode, targeting multimodal weakly-supervised knowledge discovery in digital pathology. He has also been the coordinator of the Hasler Foundation financed ProHand project, targeting the development of 3D printed robotic prosthetic hands controlled via machine learning approaches.