

# MEG-GPT: A transformer-based foundation model for magnetoencephalography data

Rukuang Huang<sup>1,2</sup>, SungJun Cho<sup>1,3</sup>, Chetan Gohil<sup>1,2,\*</sup>, Oiwi Parker Jones<sup>1,4</sup>, and Mark Woolrich<sup>1,2</sup>

<sup>1</sup>Oxford Centre for Integrative Neuroimaging (OxCIN), University of Oxford, United Kingdom

<sup>2</sup>Department of Psychiatry, University of Oxford, United Kingdom

<sup>3</sup>Nuffield Department of Clinical Neurosciences, University of Oxford, United Kingdom

<sup>4</sup>Department of Engineering Science, University of Oxford, United Kingdom

\*Corresponding author: chetan.gohil@psych.ox.ac.uk

## Abstract

Modelling the complex spatiotemporal patterns of large-scale brain dynamics is crucial for neuroscience, but traditional methods fail to capture the rich structure in modalities such as magnetoencephalography (MEG). Recent advances in deep learning have enabled significant progress in other domains, such as language and vision, by using *foundation models* at scale. Here, we introduce **MEG-GPT**, a transformer-based foundation model that uses time-attention and next time-point prediction. To facilitate this, we also introduce a novel data-driven *tokeniser* for continuous MEG data, which preserves the high temporal resolution of continuous MEG signals without lossy transformations. We trained **MEG-GPT** on tokenised brain region time-courses extracted from a large-scale MEG dataset ( $N=612$ , eyes-closed rest, Cam-CAN data), and show that the learnt model can generate data with realistic spatio-spectral properties, including transient events and population variability. Critically, it performs well in downstream *decoding* tasks, improving downstream supervised prediction task, showing improved zero-shot generalisation across sessions (improving accuracy from 0.56 to 0.59) and subjects (improving accuracy from 0.45 to 0.49) compared to a PCA baseline methods. Furthermore, we show the model can be efficiently *fine-tuned* on a smaller labelled dataset to boost performance in cross-subject decoding scenarios. This work establishes a powerful foundation model for electrophysiological data, paving the way for applications in computational neuroscience and neural decoding.

**Keywords:** Electrophysiology; MEG; GPT; Transformer; Foundation Model; Tokenisation

## Highlights

- **MEG-GPT**, a transformer-based foundation model that learns complex spatiotemporal dynamics from large-scale resting-state MEG data.

- A novel data-driven *tokenizer* converts continuous MEG signals into discrete tokens without loss of temporal or spectral resolution.
- MEG-GPT generates realistic synthetic MEG signals that replicate key spatio-spectral properties, including transient events, inter-subject variability and subject-specific fingerprints.
- Boosts *zero-shot decoding accuracy* across subjects (from 0.45 to 0.49) and sessions (from 0.56 to 0.59) compared to traditional PCA-derived baselines.
- The pre-trained model *efficiently fine-tunes* on small, labelled datasets to enhance cross-subject decoding.

## 1 Introduction

A central aim in neuroscience is to understand the rich, dynamic patterns observed in large-scale brain activity. Electrophysiological imaging techniques provide a direct measure of neural activity with millisecond temporal resolution, making them particularly useful for studying fast brain dynamics. Among these techniques, magnetoencephalography (MEG) offers a non-invasive measure of neural activity that is localised with good spatial accuracy and excellent temporal resolution (Proudfoot, Woolrich, Nobre, & Turner, 2014).

Traditional analyses of source-localised MEG data often involve averaging over space or time, leading to a loss of spatiotemporal information (Alexander et al., 2013; Brookes et al., 2010; König et al., 2015; Tzovara et al., 2012). For instance, power spectral density (PSD) analyses average over time and sometimes space (Luck, 2014), while static functional connectivity analyses average over time (Gohil, Kohl, Huang, et al., 2024). Furthermore, these analyses are often applied to small datasets in isolation. The recent availability of large-scale MEG datasets, like Cam-CAN (Shafto et al., 2014; Taylor et al., 2017), offers a new opportunity to move beyond these limitations. Modern deep learning approaches can be trained on large-scale data to extract rich spatiotemporal structure and patterns of population variability, which can then be leveraged to benefit inferences on smaller, specialized datasets (Géron, 2022; Murphy, 2012).

*Foundation models* (Bommasani et al., 2021) are a class of models trained on large amounts of data at scale, often using *self-supervised learning* (Géron, 2022; Murphy, 2012). This contrasts with *supervised learning*, which aims to learn a mapping from the data to *labels*<sup>1</sup>. In self-supervised learning, the objective is to model the statistical dependencies within the data using a label derived from the data itself. In time series data, and in the approach we take here, this is often achieved by predicting the next time step; note that this can also be thought of as fitting an autoregressive generative model to the data. Foundation models can extract ‘general features’ from the training data that can be applied to new datasets. Often, a pre-trained foundation model is *fine tuned*<sup>2</sup> on an independent dataset, which may be labelled, for a particular *downstream* application, such as a prediction task (Bommasani et al., 2021). Adopting this approach has achieved remarkable success in modelling language and vision (Brown et al., 2020; Dosovitskiy et al., 2020; Nie, Nguyen, Sinthong, & Kalagnanam, 2022; Radford et al., 2021; Ramesh et al., 2021; Touvron et al., 2023; Zhai, Kolesnikov, Houlsby, & Beyer, 2022).

---

<sup>1</sup>Auxiliary quantities that we would like to predict.

<sup>2</sup>Trained for a short period. This can be with the same or a different objective (loss function).

This approach is potentially valuable in the analysis of MEG data, where unlabelled resting-state data is abundant and labelled data, such as that for cognitive tasks or clinical populations, is limited. Improving supervised learning performance on MEG data is critical for a variety of neuroscience studies and applications. This includes: predicting a cognitive state based on data, referred to as *decoding* (Thomas, Ré, & Poldrack, 2022); disease classification and biomarker discovery (Craik, He, & Contreras-Vidal, 2019); patient stratification (Bosl, Tager-Flusberg, & Nelson, 2018; Cassani, Estarellas, San-Martin, Fraga, & Falk, 2018); and brain-computer interfaces (Dash, Ferrari, Heitzman, & Wang, 2019; Lotte et al., 2018; Tang, LeBel, Jain, & Huth, 2023). Foundation models offer a powerful framework for such applications, as they can be adapted to specific tasks with minimal labelled data while leveraging prior knowledge from large-scale unlabelled data (Bommasani et al., 2021).

While existing foundation models for electrophysiological data often rely on time-frequency transformations, such as wavelet methods (Wang et al., 2023; Yuan et al., 2024), that compromise data resolution, our approach circumvents this entirely. Inspired by the Generative Pre-Trained Transformer (GPT) family of models (Brown et al., 2020; Radford, Narasimhan, Salimans, Sutskever, et al., 2018; Radford et al., 2019), we introduce **MEG-GPT**, a foundation model that learns directly from tokenised MEG data. **MEG-GPT** is a nonlinear autoregressive (AR) model (Marple Jr, 2019) based on a transformer architecture<sup>3</sup> (Géron, 2022; Vaswani et al., 2017). This model predicts the next token from a sequence of previous tokens. This is made possible by another contribution: a bespoke, data-driven tokeniser for continuous electrophysiological data that operates with no loss in temporal or spectral resolution.

Foundation models are typically evaluated in terms of their performance on downstream tasks, such as classification or prediction (Bommasani et al., 2021; Zhai et al., 2022). However, the generative nature of GPT models means they are able to produce new synthetic data. This offers a complementary approach for evaluating model performance. We can assess whether the model can generate important features of interest that are present in the training data in the new synthetic data. In language and vision models, this is done by qualitatively examining the generated text (Brown et al., 2020; Touvron et al., 2023) or images (Ramesh et al., 2021). Here, we evaluate **MEG-GPT** quantitatively by assessing the model’s ability to generate realistic features, such as the transient spatio-spectral structure and inter-subject variability in neural data.

In the following, we first validate the performance of our tokeniser. Then, we train **MEG-GPT** on the tokenised parcel time courses extracted from the eyes closed resting-state data in Cam-CAN (612 subjects,  $\sim 2.8$  billion time points). Next, we demonstrate that the trained **MEG-GPT** can generate new data with realistic spectral properties, transient dynamics, and inter-subject variability. Finally, we showcase its practical utility in a downstream decoding task (Grootswagers, Wardle, & Carlson, 2017; King & Dehaene, 2014), where it achieves superior zero-shot generalisation compared to traditional approaches.

---

<sup>3</sup>More specifically, **MEG-GPT** is based on the decoder block of a transformer architecture.

## 2 Methods

### 2.1 Tokeniser

GPT models typically learn statistical dependencies between *tokens* of data. These are discrete ‘building blocks’ of the data. In large language models (Tunstall, Von Werra, & Wolf, 2022), these tokens correspond to words (or individual characters). Here, we have continuous MEG data. Hence, before we can feed the data into the GPT model, we need a “tokeniser” that can map each time-series of continuous MEG data to a time series of discrete tokens; this process is then repeated separately for all parcels of brain regions. In this paper, we work with source reconstructed and parcellated MEG data (see Section 2.5.2 and Section 2.5.3), i.e. we have time series data for each parcel of brain region, and each parcel can be thought of as virtual electrodes. Our proposed tokeniser accomplishes this using an autoencoder framework, which learns a discrete set of short, reusable temporal patterns – the ‘tokens’ – directly from the data.

Training the foundation model on tokenised, rather than continuous, MEG data means we can benefit from using the cross entropy as the loss function (see Section 2.2.4). Cross entropy has been shown to have better convergence properties compared to the mean-square-error (MSE) loss, which is used for continuous data (Géron, 2022).

The simplest type of tokenisation would quantise the data into bins, e.g. after a  $\mu$ -law transformation (Csaky, van Es, Jones, & Woolrich, 2024; ITU-T, 1988; Van Den Oord et al., 2016). However, this does not account for the temporal-spectral properties of the data. We therefore took a data-driven approach.

One option is the vector-quantised variational autoencoder (VQ-VAE) (Van Den Oord, Vinyals, et al., 2017), which learns to represent the data as a time series of discrete vectors with the data at each time point **compressed** into one of a set of vectors. The set of vectors is learnt from the data and is referred to as a *codebook* (or *dictionary*).

In contrast to this, we wanted a tokeniser that is not aiming to compress the data, i.e. we want it to be a near-lossless transform. This means that it does no regularisation, and has no loss of temporal or spatial resolution. Instead, we want all the modelling of rich spatio-temporal structure to be done downstream by the more sophisticated transformer-based foundation model.

Our tokeniser, illustrated in Figure 1, is described below. It is inspired by the VQ-VAE, but the codebook is embedded in the decoder. Note that, the tokeniser does not need a straight-through estimator (which is used in a VQ-VAE) to calculate the gradient, and that it is a standard autoencoder, i.e. no stochastic sampling is performed.

#### 2.1.1 Encoder

The *encoder* learns a mapping from the continuous MEG data to sequence of categorical *token labels*, which are unique indices for each token. The same tokeniser is applied to each parcel independently.

Let  $\mathbf{x} = [x_1, \dots, x_T] \in \mathbb{R}^T$  be the continuous MEG data for a single parcel, where  $T$  is the number of time points. The goal of the encoder is to learn a sequence of categorical token labels  $\mathbf{z} = [z_1, \dots, z_T] \in \mathbb{R}^T$ . The token label at each time point can take one of  $K$  values (a pre-specified hyperparameter). The encoder learns a set of *logits* at each time point  $\boldsymbol{\alpha}_t = [\alpha_1, \dots, \alpha_K] \in \mathbb{R}^K$ , which reflects the underlying probability of each token:

$$\boldsymbol{\alpha}_t = \text{Encoder}(\mathbf{x}). \tag{1}$$

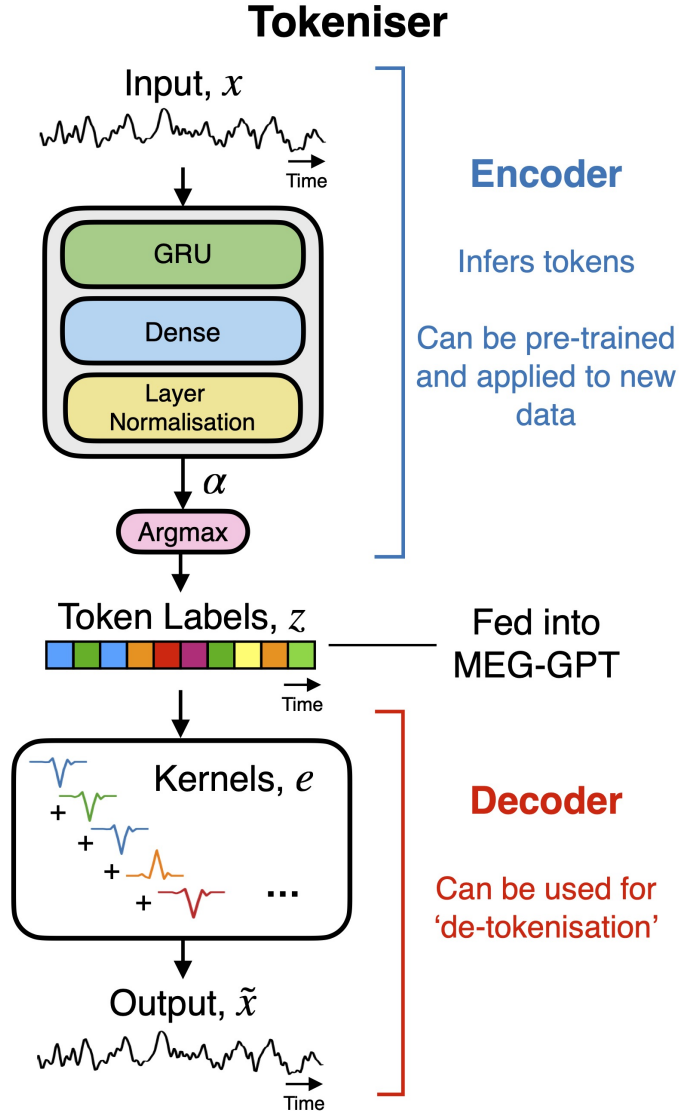


Figure 1: **Illustration of the tokeniser.** The input to the tokeniser is a single sequence of continuous MEG data and its encoder’s output is a single sequence of token labels. The tokeniser is based on an autoencoder framework. The encoder maps the continuous data  $\boldsymbol{x}$  onto logits  $\boldsymbol{\alpha}$  that are used to calculate token labels  $\boldsymbol{z}$ . The decoder reconstructs the data  $\tilde{\boldsymbol{x}}$  from token labels  $\boldsymbol{z}$  using a weighted sum of token kernels  $\boldsymbol{e}$ . The token labels  $\boldsymbol{z}$  are then fed into MEG-GPT.

In our tokeniser, the encoder is a single layer of the GRU (Gated Recurrent Unit) (Cho et al., 2014) layer and a Dense layer, followed by a Layer Normalisation (Ba, Kiros, & Hinton, 2016). The token label is calculated from the logits using

$$z_t = \operatorname{argmax}(\boldsymbol{\alpha}_t). \quad (2)$$

### 2.1.2 Decoder

The continuous MEG data is built by combining the tokens (i.e. building blocks). The decoder reconstructs the continuous data based on the token labels inferred by the encoder. To model the temporal characteristics of the data, we decided to use a dictionary of tokens

based on 1D convolution kernels. These kernels are learnt from the data alongside the token labels.

Let  $\mathbf{e} = [\mathbf{e}_1, \dots, \mathbf{e}_K] \in \mathbb{R}^{K \times d_{\text{token}}}$  be the set of *token kernels*, where each kernel is a vector of dimensionality  $d_{\text{token}}$  (a pre-specified hyperparameter). We reconstruct the data at each time point as a weighted sum of token kernels with the one-hot vectors  $\zeta_t = \text{OneHot}(\alpha_t)$ <sup>4</sup>:

$$\tilde{x}_t = \sum_{k=1}^K \left( w_k \left( \sum_{\tau=-d_{\text{token}}/2}^{d_{\text{token}}/2} e_{k,\tau} \zeta_{k,t+\tau} \right) + b_k \right) \quad (3)$$

where  $\mathbf{w} = [w_1, \dots, w_K] \in \mathbb{R}^K$  are learnable weights in the decoder.

### 2.1.3 Training

The tokeniser is trained using *stochastic gradient descent* (Géron, 2022), where the learnable parameters of the model are updated iteratively to minimise a *loss function*. In this work, the parameter updates were calculated using an Adam optimiser (Kingma, 2014).

**Loss function:** The loss function is the MSE between the reconstructed data  $\tilde{\mathbf{x}}$  and the input data  $\mathbf{x}$ .

**Annealing:** An important component of our tokeniser is the *argmax* operation that maps the logits to a categorical token label (Equation (2)). A key challenge is that the *argmax* operation is non-differentiable, which prevents end-to-end training.<sup>5</sup> To overcome this, we use an annealing technique similar to the Gumbel-Softmax relaxation introduced in (Jang, Gu, & Poole, 2016). In our work, we replace the *argmax* operation with a weighted sum of *argmax* and *softmax*<sup>6</sup> during training:

$$z_t = (1 - \kappa) \cdot \text{argmax}(\boldsymbol{\alpha}_t) + \kappa \cdot \text{softmax}(\boldsymbol{\alpha}_t), \quad (4)$$

where  $\kappa$  starts from 1 at the beginning of the training process and gradually decreases to 0. During back-propagation, the gradient is effectively taken only through the *softmax* component, while the *argmax* component is treated as constant. This allows end-to-end training to occur by smoothly transitioning from a *softmax* to an *argmax*-like output during training. After training (during inference),  $\kappa$  is set to zero.

### 2.1.4 Token re-factorisation

In practice, after we trained the tokeniser, we found that not all tokens were used to reconstruct the data. Because these tokens do not appear in the data, we do not need to include them when training the foundation model. We performed *token re-factorisation* to remove these tokens. This involves:

1. Relabelling the  $K$  tokens in descending order (i.e. from 1 to  $K$ ) in terms of their rate of occurrence in the tokenised training data.
2. Identifying the tokens that do not appear in the tokenised training data

<sup>4</sup>A one-hot vector is a binary vector in which exactly one element is equal to 1 and all others are 0.

<sup>5</sup>The *argmax* operation is not differentiable, which impedes the backpropagation algorithm, and prevents us from calculating the gradient for model parameters that precede this operation.

<sup>6</sup>The *softmax* operation can be performed incorporating a *temperature* parameter, which we set to a small value (of 0.1) in this paper to encourage categorical-like outputs.

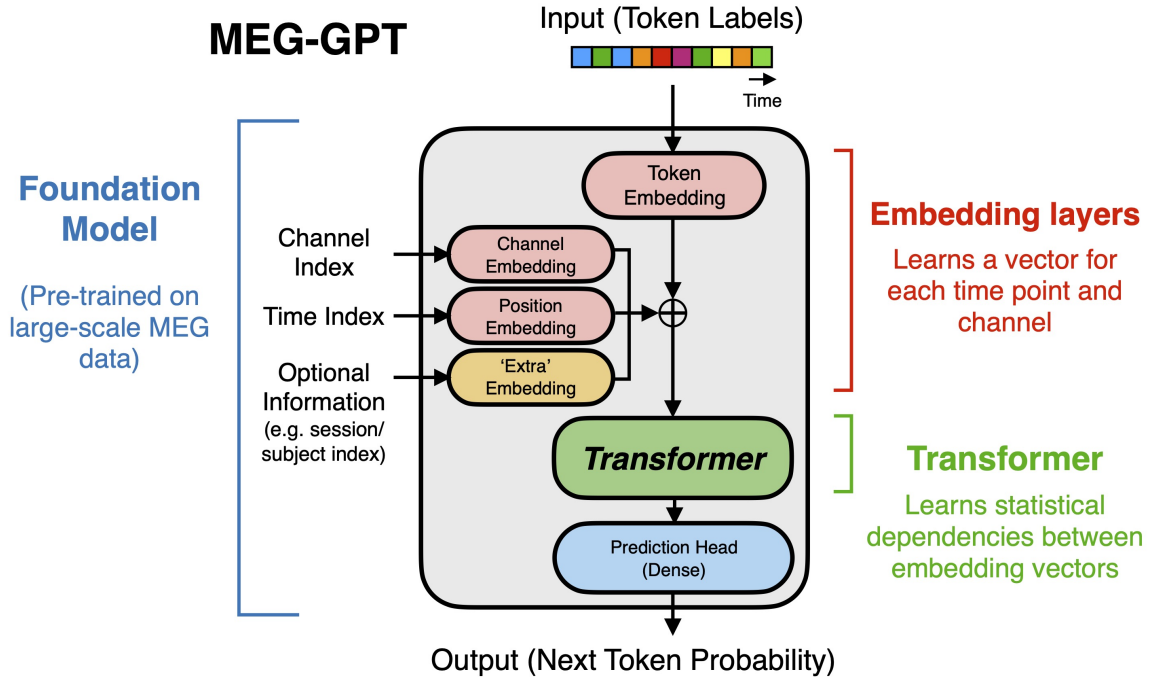


Figure 2: **MEG-GPT foundation model**. MEG-GPT is a nonlinear autoregressive model that predicts the token at the next timepoint from a preceding sequence of token labels. However, unlike classical autoregressive models, the autoregressive weighting can change as a function of the data. A key part of the foundation model is learning an *embedding space* for the tokens, which generates vector for each token  $\mathbf{v}_z$ , parcel  $\mathbf{v}_c$ , temporal position  $\mathbf{v}_p$ , any additional information  $\mathbf{v}_s$  (e.g. subject ID) and combines (adds) these to provide a single embedding vector  $\mathbf{v}$  at each time point. This combined embedding vector is fed into the Transformer Decoder, and the output of the Transformer Decoder is used to predict the next token via the Prediction Head.

3. Assigning the tokens with zero occurrence a label of 0.

Following token re-factorisation, we end up with  $K^*$  tokens, where tokens with labels from 1 to  $K^* - 1$  correspond to tokens that appeared in the training data of the tokeniser.

## 2.2 The foundation model: MEG-GPT

MEG-GPT is a transformer-decoder-based foundational model that can be pre-trained on a large-scale datasets. The input to MEG-GPT is a sequence of token labels and the output is the probability for each token to be next. The architecture for MEG-GPT is shown in Figure 2. We describe each part in detail below.

### 2.2.1 Input embedding

The model first transforms its discrete token label inputs into a rich, continuous vector space (the *token embedding space*) through learnt embeddings. This process assigns a unique vector to each token label. This can be thought of as a compressed summary of characteristics of each token, e.g. tokens that have similar characteristics will get grouped together in the embedding space.

We also learn embedding vectors for the other inputs provided to MEG-GPT, i.e. the parcel index, time index, and any other auxiliary information like a session, participant or task index. Each embedding vector is denoted by:

- $\mathbf{v}_z \in \mathbb{R}^{K^* \times d_z}$  for the token embedding.
- $\mathbf{v}_c \in \mathbb{R}^{C \times d_c}$  for the parcel embedding, where  $C$  is the number of parcels.
- $\mathbf{v}_p \in \mathbb{R}^{L \times d_p}$  for position (time index) embedding, where  $L$  is the sequence length.
- $\mathbf{v}_s \in \mathbb{R}^{N \times d_s}$  for the additional information embedding, e.g. the participant ID, where  $N$  is the number of unique identifiers for each piece of additional information.

$d_*$  denotes the length of each embedding vector. All of these embeddings are summed to obtain the overall input embedding

$$\mathbf{v} = \text{Sum}(\mathbf{v}_z, \mathbf{v}_c, \mathbf{v}_p, \mathbf{v}_s) \in \mathbb{R}^{L \times C \times d}, \quad (5)$$

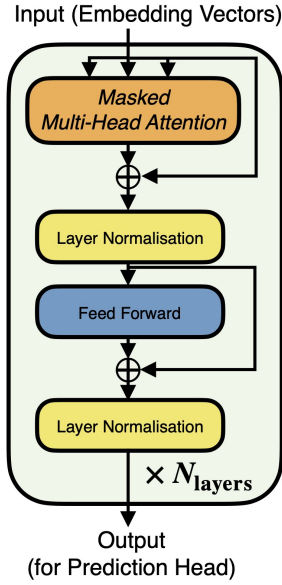
which is passed to the Transformer Decoder block. A Dense layer is used to map the individual embedding vector lengths ( $d_z, d_c, d_p, d_s$ ) to  $d$  if there is a mismatch in dimensionality. Summation of embeddings is chosen here over concatenation as summation provides a more compact representation and is commonly used in transformer-based architectures. All embeddings are learnt when the MEG-GPT is trained. This combined vector,  $\mathbf{v}$ , effectively encodes the ‘**what**’ (token), ‘**where**’ (parcel), ‘**when**’ (time) and ‘**who**’ (participant) of the signal, providing a comprehensive input to the Transformer Decoder block.

### 2.2.2 Transformer and Prediction Head

The input to the Transformer Decoder is  $\mathbf{v} \in \mathbb{R}^{L \times C \times d}$ , which is a time ( $L$ ) by parcels ( $C$ ) by embedding length ( $d$ ) tensor. The Transformer Decoder models the statistical dependencies between these embedding vectors. It can be understood as a powerful nonlinear autoregressive (AR) model, where the AR coefficients are not fixed but instead adapt dynamically to the input data via its multi-head attention mechanism.

The Transformer Decoder is based on the architecture in (Vaswani et al., 2017). Figure 3A shows the architecture of the MEG-GPT Transformer Decoder. In each layer, we use residual connections around the Masked Multi-Head Attention block and a Feed Forward layer, followed by Layer Normalisation. The output of the Transformer Decoder is passed to the Prediction Head – a single Dense layer – which generates the final logits used to predict the probability of the next token.

## A) Transformer



## B) Masked Multi-Head Attention

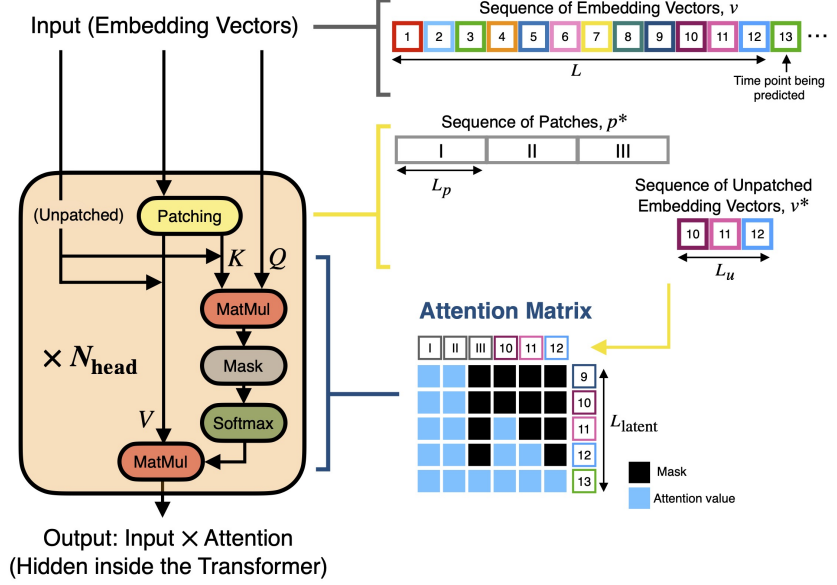


Figure 3: **Transformer Decoder and the masked multi-head attention used in MEG-GPT.** A) Architecture of the Transformer Decoder in Figure 2. B) Masked multi-head (self-)attention used in the Transformer Decoder. Shows an example of predicting the token at the next time step with a receptive field of  $L = 12$ . The input embeddings  $v$  are divided into 3 patches of patch size  $L_p = 4$ , resulting in *patched inputs*  $p^*$ . The last  $L_u = 3$  unpatched embeddings are retained as the *unpatched inputs*  $v^*$ , allowing for finer temporal resolution near the prediction point. The attention matrix used in the self-attention mechanism has a latent sequence length of  $L_{\text{latent}} = 5$  corresponding to the output tokens. Each output attends to both patched inputs  $p^*$  and unpatched inputs  $v^*$ . A temporal mask (indicated by black squares) is applied to prevent information leakage from future time points during training.

### 2.2.3 Masked multi-head attention

A key innovation in the Transformer Decoder (Vaswani et al., 2017) is the use of *self-attention* layers. Here, the Transformer Decoder can learn to *attend* to different parts of the input data’s history in different ways that depend on the input data. How far back in time the Transformer Decoder can attend depends on the *receptive field*. A key challenge for transformers is that the computational cost of the self-attention mechanism scales quadratically with sequence length, making it prohibitive for long time-series. To extend MEG-GPT’s receptive field without incurring this significant computational cost, we incorporate several recent modifications into the standard self-attention architecture, as illustrated in Figure 3B. These include patching, the use of unpatched sequences, and the Perceiver AR architecture.

**Patching.** Patching is a technique to divide the input embeddings  $v \in \mathbb{R}^{L \times C \times d}$  into non-overlapping patches of length  $L_p$ , called the patch size. The resulting patched input embeddings becomes  $p \in \mathbb{R}^{P \times L_p \times C \times d}$ . As discussed in (Nie et al., 2022), aggregating time steps within patches can help extract local semantic information, which is typically not available in the single time level, and reduce computational cost. A linear Dense layer (shared across all patches) is used to collapse the  $L_p$  dimension and extract information

at the patch level, resulting in patched input  $\mathbf{p}^* \in \mathbb{R}^{P \times C \times d}$ .

**Unpatched sequences.** Although patching can extract local information, it does mean there is a loss of temporal resolution in the time attention. This could be particularly detrimental to the model’s ability to predict tokens over the short time scales, e.g. over time scales that correspond to the temporal window width of the token kernels ( $d_{\text{token}}$ ). Hence, in addition to the patched inputs, unpatched inputs  $\mathbf{v}^* = \mathbf{v}[T - L_u + 1 : T] \in \mathbb{R}^{L_u \times C \times d}$  are also used to predict the future time point.

**Perceiver AR.** Our Transformer Decoder also makes use of the Perceiver AR architecture (Hawthorne et al., 2022) to encode input sequences to latent representations of shorter sequence length  $L_{\text{latent}}$ , and keep the autoregressive nature of the model by applying the correct masking. In this paper, we only shrink the sequence length of the latent representations in the first layer of the Transformer Decoder, so that all subsequent layers work with inputs of length  $L_{\text{latent}}$ , reducing computational cost and memory requirement.

## 2.2.4 Loss function

MEG-GPT is trained using stochastic gradient descent using the Adam optimiser (Kingma, 2014). The loss function used in training is the cross entropy (Murphy, 2012) between the predicted token probabilities and the real token labels. The loss for a sequence of token labels is computed as

$$\mathcal{L} = - \sum_{t=L-L_{\text{loss}}}^L \sum_{k=1}^{K^*} \{\text{OneHot}(\mathbf{z})\}_{tk} \log(\hat{p}_{tk}), \quad (6)$$

where  $\hat{p}_{tk}$  is the predicted probability of token  $k$  at time  $t$ . Notice we only calculate the loss based on the last  $L_{\text{loss}}$  time points in the sequence. This is because the first few predictions made by MEG-GPT in the sequence have a relatively short input to base predictions on. Only including the predictions at the end of the sequence improves the stability of the computed loss.

## 2.2.5 Generating new data

Once trained, MEG-GPT can be used as a generative model to synthesise new MEG data by following a three-step process:

1. **Prompt Initialisation.** A prompt sequence of tokens  $\mathbf{z}^* \in \mathbb{R}^{L \times C}$  is created by sampling tokens from a categorical distribution weighted by their rate of occurrence in the training data. Extra inputs used during training (such as parcel labels, session/participant indices) should also be provided.
2. **Autoregressive Generation.** The model then predicts subsequent tokens one at a time. At each step, the next token is drawn from the model’s output probability distribution using *nucleus sampling* (top-p = 0.99) (Holtzman, Buys, Du, Forbes, & Choi, 2019). This restricts sampling to the smallest possible set of tokens whose cumulative probability exceeds 0.99.
3. **Signal Reconstruction.** The complete sequence of generated token labels is then passed through the pre-trained tokeniser’s decoder. This converts the discrete sequence back into a continuous, synthetic MEG time series, reversing the tokenisation process.

## 2.3 Fine tuning the foundation model

After the foundation model has been trained, it can then be *fine-tuned* on another dataset to adapt to aspects that may not be present in the original training data. For example, in MEG data this may be potential variations in recording devices, sensor configurations, preprocessing pipelines, population, experimental design, etc. Fine tuning allows the model to adapt to the specific statistical and physiological characteristics of the new dataset, improving performance on downstream tasks.

Fine-tuning process is computationally efficient. It typically involves training the model on the new, often smaller, dataset for only a few epochs using a small learning rate. This prevents the model from catastrophically forgetting the general features learnt during pre-training while gently adjusting weights to fit the new data.

A critical aspect of fine-tuning is determining which model components to update. Embeddings specific to the original training set, such as **subject embeddings**, must be discarded and retrained on the new dataset. In contrast, embeddings that are transferable, such as position (time) and parcel (space) embedding, can be held fixed. Exact choices for the datasets studied in this work are given in Section 2.5.4.

## 2.4 The foundation model as a feature extractor

To evaluate MEG-GPT’s practical utility, we use it to extract features for a downstream decoding task. We compare the performance of a classifier when trained on three distinct feature sets derived from the same underlying MEG data epochs:

- **Baseline Features:** Parcel MEG time courses are time-locked to the task onset and epoched. The epochs (with dimensions  $C \times L$ ) are flattened and used as features.
- **Zero shot Features:** The prediction head of the MEG-GPT foundation model is discarded and the outputs (with dimensions  $C \times L \times d$ ) from the decoder are collapsed over the time dimension by taking the average. The resulting outputs (with dimensions  $C \times d$ ) are flattened and used as features.
- **Fine-tuned Features:** MEG-GPT is fine-tuned on the training set of the task dataset (see Section 2.5.4) and features are extracted in the same way as the zero-shot case.

## 2.5 Datasets

Two publicly available MEG datasets were used: one for training MEG-GPT (Cam-CAN) and another to illustrate a downstream decoding task (Wakeman-Henson). Both datasets were collected using an Elekta Neuromag Vectorview scanner at a sampling frequency of 1 kHz and were processed in the same way.

**Cam-CAN** (Shafto et al., 2014; Taylor et al., 2017). We used resting-state (eyes closed;  $\sim 8.5$  minutes) recordings from 612 healthy participants (310 males, 302 females, aged 18-88 years) from this dataset.

**Wakeman-Henson** (Wakeman & Henson, 2015). This dataset contains 19 healthy participants (11 males, 8 females, aged 23-37 years), which were scanned 6 times each. In each recording session, each participant performed a visual perception task. They were presented with 3 types of visual stimuli, including an image of a famous, unfamiliar or scrambled face. Each recording session was around 7.5 minutes and contain  $\sim 200$  trials which are evenly split across the three stimulus types. To ensure participants focus on the

image, they were also asked to press one of the keys depending on whether they regarded each of the images as symmetric.

### 2.5.1 Preprocessing

Both public MEG datasets were band-pass filtered between 0.03 and 330 Hz and MaxFiltered (Taulu & Simola, 2006). They were then further preprocessed using the `osl-ephys` toolbox, which is based on MNE (van Es, Gohil, Quinn, & Woolrich, 2025):

1. Band-pass filtered between 0.5 and 125 Hz.
2. Notch filtered at 50 and 100 Hz to remove power line artefacts.
3. Downsampled to 250 Hz.
4. Automated bad segment and channel detection using the generalised extreme Studentised deviate procedure (Rosner, 1983).
5. FastICA (Hyvarinen, 1999) with 64 components to detect artefacts. Components with high correlation (threshold of 0.9) with the electrooculogram/electrocardiogram (EOG/ECG) channels were marked as noise and removed. Between 0 and 3 EOG components were rejected in each recording (mean 0.99, standard deviation 0.79) and between 0 and 5 ECG components were rejected (mean 2.25, standard deviation 0.84).

### 2.5.2 Coregistration and source reconstruction

Coregistration and source reconstruction were also performed using the `osl-ephys` toolbox: coregistration was carried out using the `osl-ephys` tool RHINO, making use of a structural magnetic resonance imaging image and digitised headshape points (acquired with a Polhemus pen) for each subject; then, the data were source reconstructed onto an 8 mm isotropic grid using a volumetric linearly constrained minimum variance (unit noise gain) beamformer (Van Veen & Buckley, 1988; Van Veen, Van Drongelen, Yuchtman, & Suzuki, 1997).

### 2.5.3 Parcellation, leakage correction and sign flipping

Parcellation, leakage correction and sign flipping was also performed with the `osl-ephys` toolbox. An anatomical parcellation was used to estimate the activity as 52 regions of interest using the first principal component across voxels associated with each parcel. Details of the parcellation can be found in (Kohl, Woolrich, Nobre, & Quinn, 2023).

A common problem in the estimation of source activity using electrophysiological data, is ‘spatial leakage’ in the activity between neighbouring regions, which can lead to ‘ghost interactions’ (Colclough, Brookes, Smith, & Woolrich, 2015). The symmetric multivariate leakage reduction algorithm proposed by (Colclough et al., 2015) was used to reduce spatial leakage and ghost interactions, by removing all zero-lag correlation between parcel time courses.

The principal component analysis (PCA) step in performing the parcellation means the sign of each parcel time course is arbitrary. This poses a challenge for group-level analysis. We used the method proposed in (Vidaurre et al., 2016) to align the sign of each parcel time course across sessions/subjects. This algorithm uses a greedy search based on

randomly flipping the sign of each parcel time course to maximises the agreement between different sessions/subjects.

Finally, we temporally standardise each sign-flipped parcel time course (i.e. subtract the temporal mean and divide by the temporal standard deviation). All subsequent analysis is done on these standardised parcel (i.e. brain region) time courses.

#### 2.5.4 Training, validation and test sets

We divide the Cam-CAN and the Wakeman-Henson datasets into training, validation and testing sets, which differ depending on the study:

- **Tokenizer:** The training set of the tokenizer includes the parcel time courses of the first 50 of 612 subjects in the Cam-CAN dataset. The testing set of the tokenizer includes the rest of the subjects in the Cam-CAN dataset and the entire Wakeman-Henson dataset. The hyperparameters for training the tokenizer are shown in Table A.1.
- **MEG-GPT:** For training MEG-GPT, we employ a nine-to-one train-validation split for each subject in the Cam-CAN dataset, i.e. for each subject in the Cam-CAN dataset, 90% of their data is used for training and 10% of the data is used for evaluating the validation loss and accuracy. The hyperparameters for training MEG-GPT are available in Table B.1.
- **Fine-tuning and task decoder:** During fine-tuning MEG-GPT and training of task decoder on the Wakeman-Henson dataset, the first 5 sessions of the first 18 subjects are used as the training set. The testing set includes the sixth session of the first 18 subjects (used for testing within subject generalisability) and all sessions of subject 19 (used for testing out of subject generalisability). When we fine-tuned the MEG-GPT model, we froze the token, parcel and position embeddings, and only trained the Transformer Decoder and Prediction Head. The subject embeddings learnt from the Cam-CAN dataset were discarded at this stage. The hyperparameters for fine-tuning on the Wakeman-Henson dataset are available in Table B.2.

## 3 Results

### 3.1 The tokenizer reconstructs MEG data with high accuracy and generalises to unseen data

First, we studied the performance of the novel data-adaptive tokenizer after training it on a subset of the Cam-CAN dataset (50 subjects), resulting in  $K^* = 61$  tokens after token re-factorisation. Details regarding the hyperparameters of the tokenizer are given in Appendix A, along with the training curve (loss vs number of training epochs, Figure A.1) and distribution of token occurrences and token shapes (which shows fundamental building blocks of MEG data learnt by the model) of the final model (Figure A.2).

Qualitatively, the tokenizer provides a high-fidelity reconstruction of the original MEG signal, closely matching its waveform across the training set, a held-out test set, and a separate, held-out dataset (Figure 4). To quantify this, we looked at the percentage of variance explained (PVE) of the reconstructed data using the tokenizer vs the original MEG parcel time courses (defined in Appendix A). We calculated the PVE on three

datasets: the training set of the tokeniser in Cam-CAN (Cam-CAN train), the testing set in Cam-CAN (Cam-CAN test) and the Wakeman-Henson dataset. As shown in Figure 4B, with only  $K^* = 61$  tokens, the tokeniser achieved more than 97% PVE on most of the sessions and also generalised to unseen data, with only a slight drop of PVE in Cam-CAN test compared with Cam-CAN train. This high reconstruction accuracy was maintained on the Wakeman-Henson dataset, confirming that the tokeniser generalises well. Notably, reconstruction performance was highest on the Wakeman-Henson dataset.

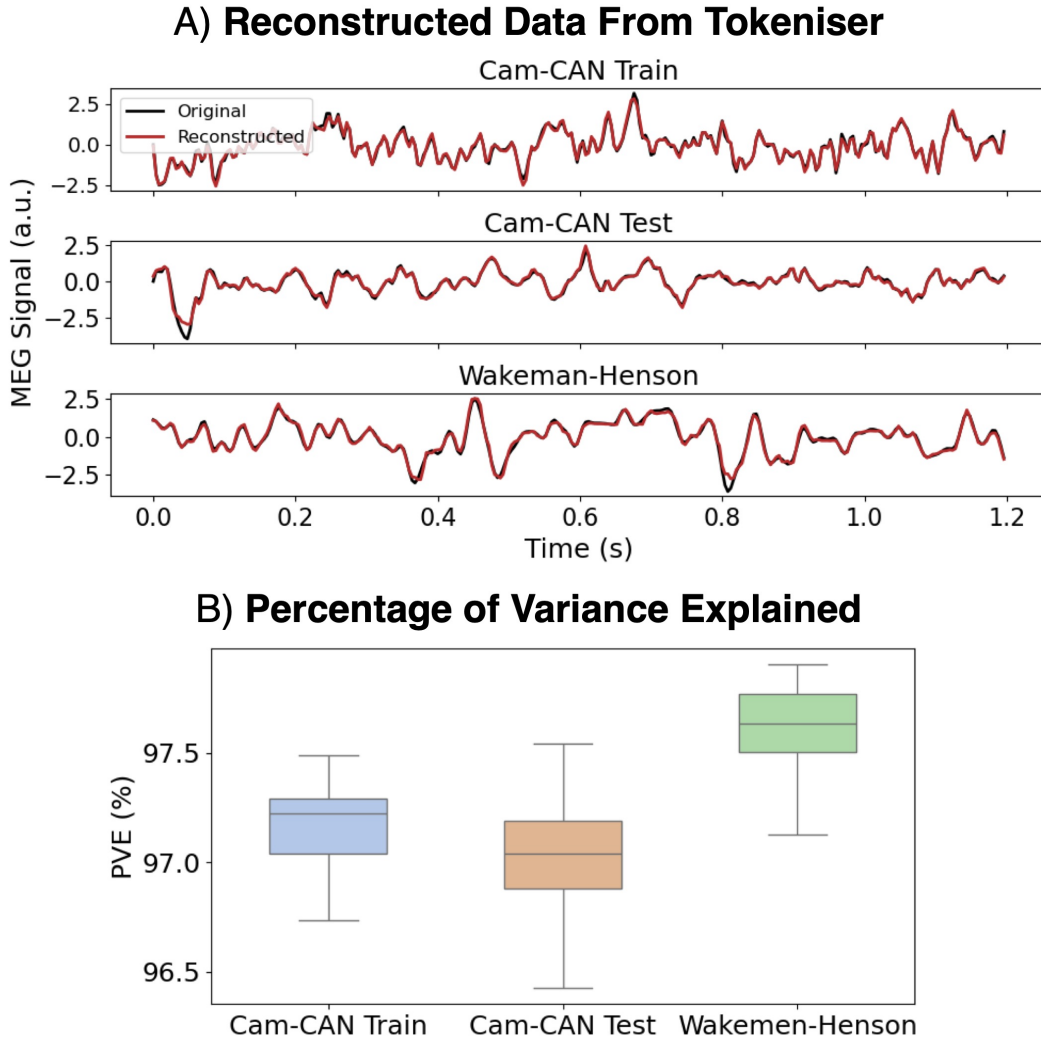


Figure 4: **The data-adaptive tokeniser reconstructs parcellated MEG data with high accuracy and generalises well to unseen data.** A) Original signals (black) and tokeniser reconstructions (where the original signals are tokenised and then de-tokenised) (red) are shown for each session from the Cam-CAN training set (top row), Cam-CAN testing set (middle row), and the Wakeman-Henson dataset (bottom row). Only the first 1.2s of each session are displayed. B) Percentage of variance explained (PVE) of the reconstructed data across different datasets.

### 3.2 MEG-GPT captures spatial and spectral characteristics of real data

We tokenised all 612 recording sessions in the Cam-CAN dataset using the trained tokenizer from Section 3.1 and trained the MEG-GPT foundation model on the Cam-CAN dataset (see Section 2.5.4 for details regarding the training and validation split). The subject ID (index) was also passed as an extra input to MEG-GPT. More details on choices of hyperparameter and training curves are available in Appendix B.

After training MEG-GPT on the full Cam-CAN dataset, we evaluated its ability to capture the spatio-spectral features of real brain activity. We generated 60 seconds of synthetic data for each subject (see Section 2.2.5) and calculated the power spectral density (PSD) for every brain parcel using Welch’s method (2s window, 50% overlap) (Welch, 2003). For a direct comparison, we generated data using a linear autoregressive (AR) model (see Appendix D) with the same receptive field as MEG-GPT to serve as a baseline.

Figure 5A shows the group-average PSD for each parcel for the generated data from both MEG-GPT and the linear AR model. Qualitatively, we see that MEG-GPT outperforms the linear AR model in capturing key features seen in the real data’s PSD, e.g. the  $1/f$  component and the size of the alpha peak. Next, in Figure 5B, we integrated the PSD over five different frequency bands and plotted the spatial map of power (variance). Again, MEG-GPT is superior at capturing the characteristics of the real data, e.g. the frontal power in  $\delta$  and  $\theta$ , compared to the linear AR model.

### 3.3 MEG-GPT captures subject-specific fingerprints

As mentioned in Section 3.2, subject ID (index) were included as an extra input to MEG-GPT during training. We next investigated whether MEG-GPT learnt to generate data with subject-specific characteristics.

As an initial test, we examined if the model could reproduce the well-known effects of ageing on neural oscillations (Gohil, Kohl, Pitt, et al., 2024; Quinn et al., 2024). Both the real data and data generated by MEG-GPT successfully replicated the decrease in  $\alpha$ -peak frequency and increase in  $\beta$  power observed in older subjects (Figure 6).

Next, we performed a more stringent test to determine if the model could generate unique individual “fingerprints”. We extracted four different features from the real data and the MEG-GPT generated data (see Appendix C.1). For each of the features, a nearest neighbour classifier (where correlation was used as the a measure of similarity) was used for classifying subject labels. The results show that TDE features yielded the highest top-1 accuracy (Figure 6B). However, this did not persist for the top-5 accuracy where combining spatial and spectral information yielded the highest top-5 classification accuracy, confirming that MEG-GPT learns multifaceted fingerprints expressed in both domains.

Finally, to assess whether MEG-GPT has learnt the relationships between subjects, we measured the consistency score (defined in Appendix C.3). This measures the strength of agreement between the subject-pairwise distance within real data subjects and generated data subjects. All four features yielded significantly high consistency scores (under permutation test, Appendix C.3) (last column of Figure 6B).

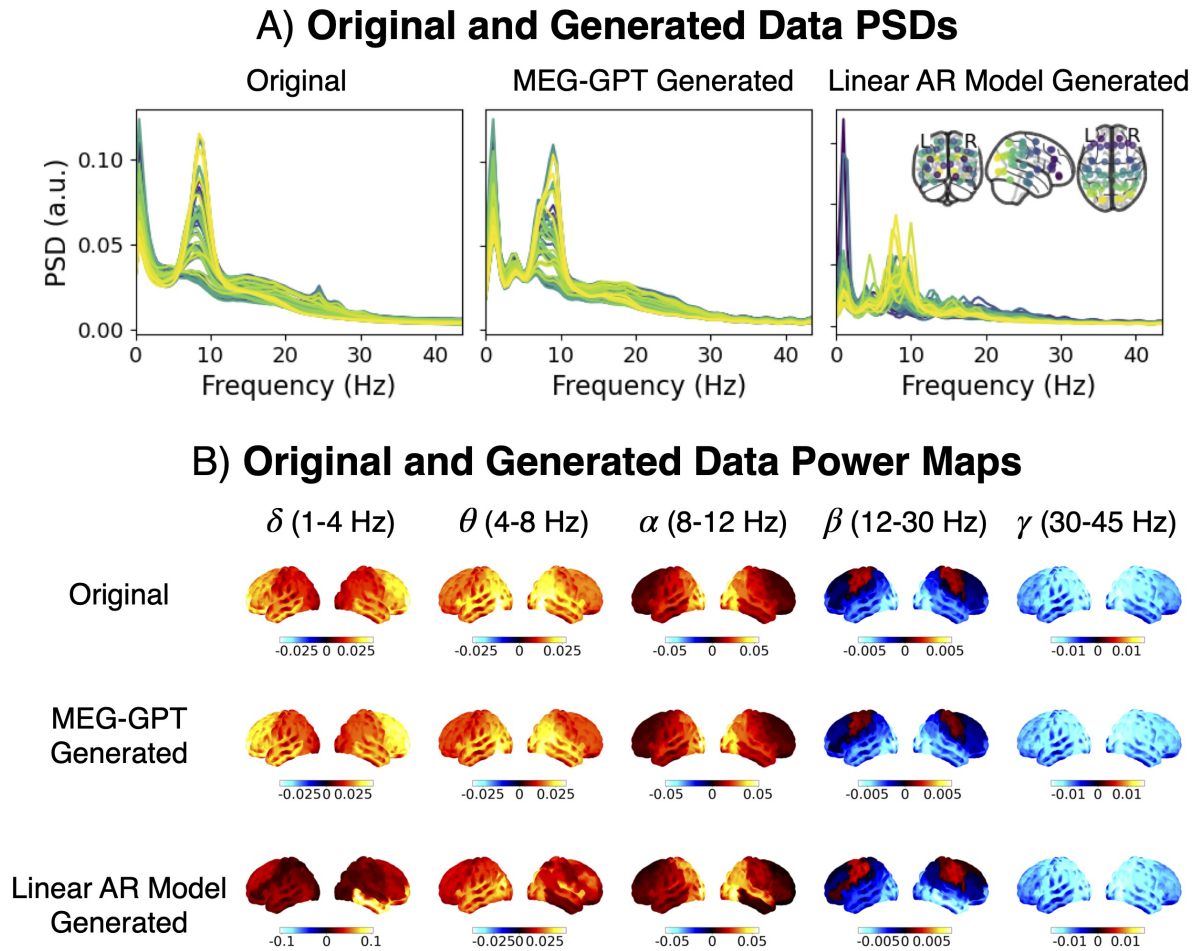


Figure 5: **MEG-GPT captures spatial and spectral characteristics of real data.** Each plot is calculated for the real data, MEG-GPT generated data, and linear AR model generated data. A) Group-average PSD for each parcel. The glass brain plot in the top right indicates the location of each parcel. B) Narrow-band power maps relative to the average across frequency bands.

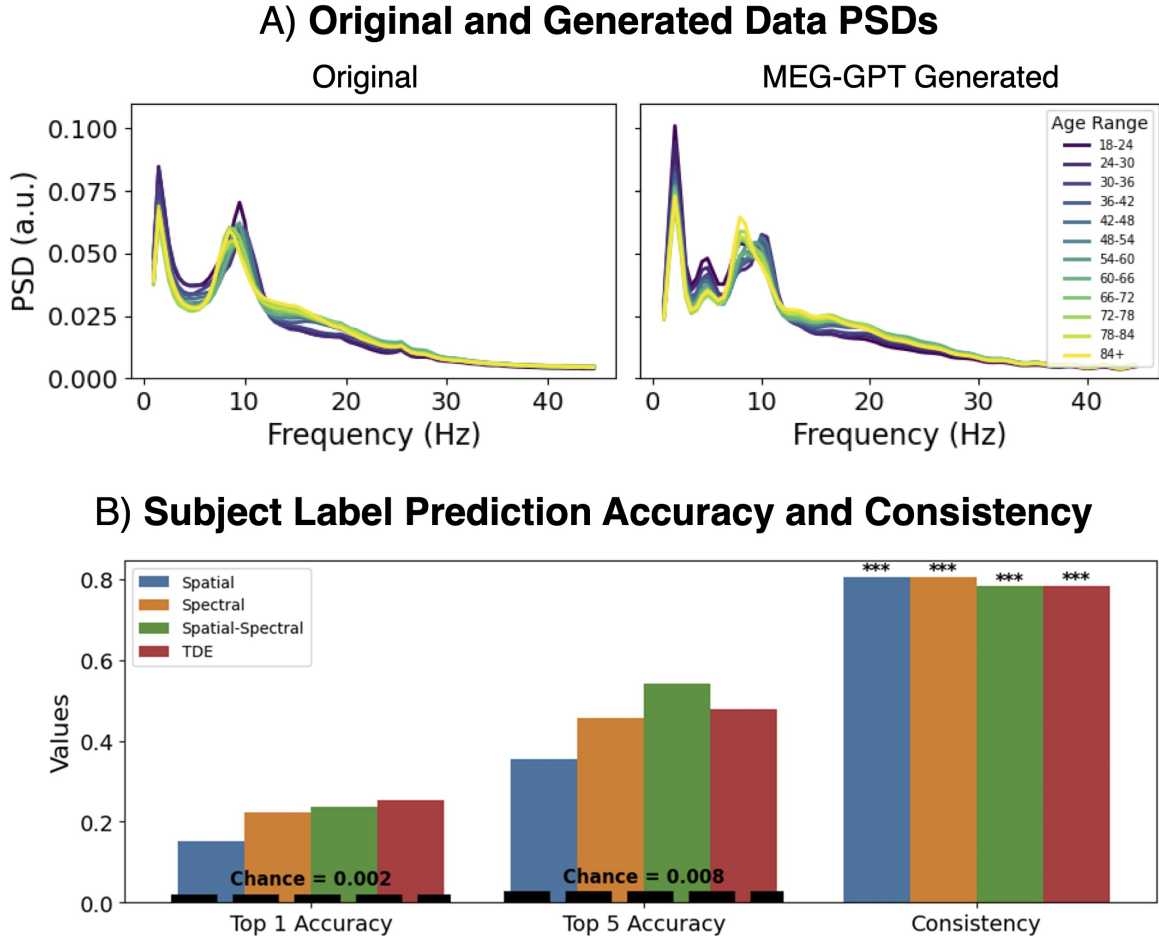


Figure 6: **MEG-GPT captures subject-specific fingerprints.** A) PSD across different age groups for real data (left) and MEG-GPT generated data (right). B) Top-1 accuracy, top-5 accuracy of predicting subject labels, and consistency score for 4 different features are shown. Chance level of top 1 (red dotted line) and top 5 (blue dotted line) are also illustrated. The asterisks (\*\*\*) indicate a  $p$ -value  $< 0.001$ .

### 3.4 MEG-GPT captures bursting dynamics in MEG data

We next investigated the ability of MEG-GPT to capture transient spectral bursting, an important characteristic of brain activity that is only apparent when temporal or trial averaging is not carried out (Quinn et al., 2019). Initially, we focussed on a parcel in the motor cortex, an area known to exhibit bursting in the  $\beta$  band (Bonaiuto et al., 2021). We qualitatively examined the spectrograms and found that MEG-GPT generates data with transient bursting in the  $\beta$  and  $\alpha$  bands in a similar manner to the real data, whereas the linear AR model overly produces continuous bursting, particularly in the  $\alpha$  band (Figure 7).

To quantitatively characterise these bursting dynamics, we employed a Time-Delay Embedded Hidden Markov Model (TDE-HMM) (see Appendix E.1). The combination of TDE and HMM has been shown to be a reliable way to capture state-specific oscillatory burstings in single channel MEG data (Gohil, Huang, et al., 2024; Quinn et al., 2019), without the need to pre-specify frequency bands, amplitude thresholds, or durations of bursts. We first established a ground truth by fitting a 3-state HMM to the real

data, which identified states 2 and 3 corresponding to activity in the  $\delta/\theta$  and  $\alpha/\beta$  bands respectively (Figure 7D, left panel). This is discussed further in Appendix E.1.

We then applied the same HMM inference procedure independently to both the MEG-GPT and linear AR model generated data <sup>7</sup>. Remarkably, we found that the MEG-GPT generated data uncovered a set of three states whose spectral profiles were nearly identical to those found in the real data (Figure 7D, middle panel). In contrast, the HMM completely fails on the linear AR generated data (Figure 7D, right panel).

Finally, we compared the temporal statistics of the HMM states across the datasets (see Appendix E.2). The summary statistics for MEG-GPT's states – including the burst count, mean interval, and mean lifetime – were significantly more similar to the real data statistics than those from the AR model (Figure 7C). This entire analysis was successfully repeated on a parcel in the visual cortex. We found that the findings were reproduced albeit with different bursting behaviour to the motor cortex, confirming the model's ability to capture region-specific bursting dynamics (Appendix E.3)

---

<sup>7</sup>The order of states output by the HMM is arbitrary and so we used the Hungarian algorithm to match the order from different runs using the state covariance matrices (Appendix Figure E.1A).

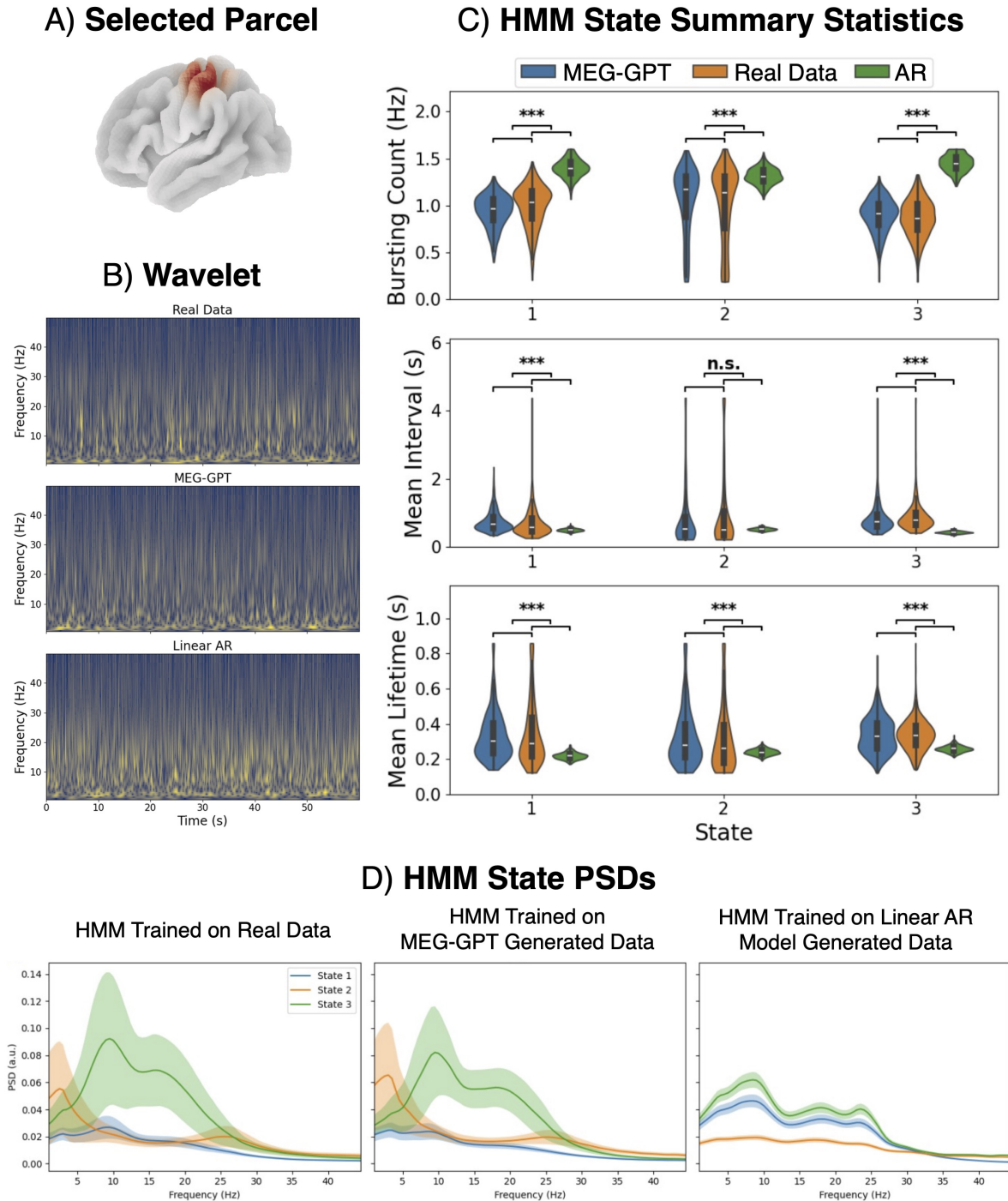


Figure 7: **MEG-GPT captures region-specific bursting dynamics in MEG data.** A) Location of the selected motor parcel. B) Wavelet transform of the first 60s from the first subject: real data (top), MEG-GPT generated data (middle), and linear AR model generated data (bottom). C) Summary statistics for each HMM state, including bursting count (top), mean interval (middle), and mean lifetime (bottom), are plotted for each of the HMM states. Results for MEG-GPT are shown in blue, real data in orange, and linear AR model in green. Asterisks mark statistics and states where MEG-GPT results more closely matched to real data compared to the AR model. The asterisks (\*\*\*) indicate a  $p$ -value  $< 0.001$ , and “n.s.” indicates a non-significant result. D) State-specific PSD profiles with the solid line representing the group average and the shaded area indicating one standard deviation across subjects. The analysis was successfully reproduced using a visual cortex parcel, confirming the model’s ability to capture region-specific bursting dynamics (Appendix E.3).

### 3.5 MEG-GPT extracts features that enhance decoding performance

To demonstrate MEG-GPT’s practical utility, we evaluated its features on a downstream visual decoding task from the Wakeman-Henson dataset. We trained a single, group-level multinomial logistic regression classifier to predict four distinct task labels (i.e. famous faces, unfamiliar faces, scrambled images, and button press). The classifier’s performance was then assessed on its ability to generalise to unseen data in two challenging scenarios: new sessions from subjects seen during training (“Within Subject”) and data from an entirely new, held-out participant (“New Subject”) (subject 19 in this figure). Four feature sets were compared: **baseline** (i.e. raw epoched time courses), **baseline+PCA** (i.e. PCA applied to the raw epoched time courses<sup>8</sup>), **zero-shot** (i.e. epoched and time-averaged output from MEG-GPT’s Transformer Decoder) and **fine-tuned** (i.e. as zero-shot, but after fine-tuning the training data from Wakeman-Henson).

As shown in Figure 8, features extracted from MEG-GPT provided a substantial boost in decoding accuracy compared to the baseline. In the **zero-shot** setting, simply using features from a pre-trained MEG-GPT improved “Within Subject” accuracy from 0.54 (baseline) and 0.56 (baseline+PCA) to 0.59 and, more dramatically, “New Subject” accuracy from 0.41 (baseline) and 0.45 (baseline+PCA) to 0.49. **Fine-tuning** the model on the task data provided an additional, targeted benefit for the most difficult scenario, further increasing “New Subject” accuracy to 0.51. These results demonstrate that MEG-GPT learns generalisable representations of brain activity that can significantly enhance the performance of simple linear decoders. A breakdown of performances over different labels is also demonstrated by the confusion matrices in Figure F.2 and it shows the enhancement in performance is universal across all labels. In Figure 8, the “New Subject” results were computed using only participant 19 as the held-out subject. We repeated the analysis for baseline and zero-shot features with all other subjects being held out in turn, and the conclusion holds (Appendix F.2).

---

<sup>8</sup>The number of PCA components was selected based on nested cross validation. 60% explained variance led to the best test prediction accuracy.

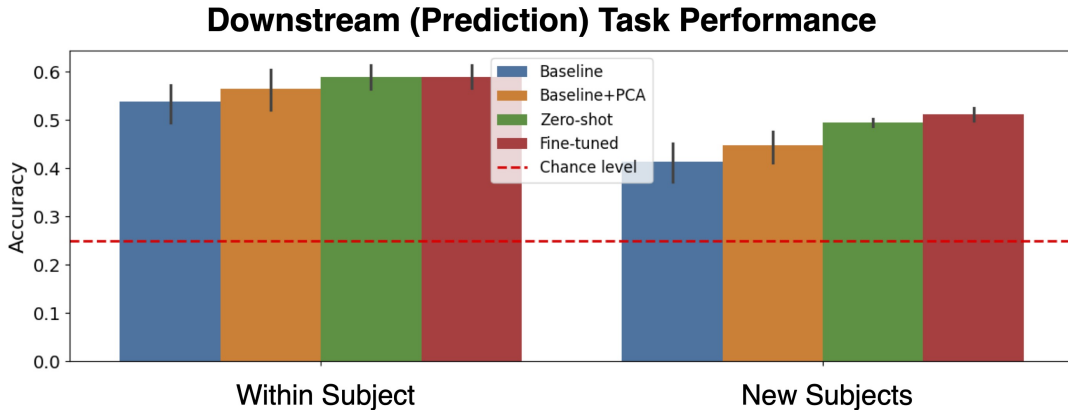


Figure 8: **MEG-GPT extracts features that enhance decoding accuracy.** Group-level multinomial logistic regression was used to predict task labels in a visual task MEG dataset (i.e. famous faces, unfamiliar faces, scrambled images, and button press). Four different feature sets are compared: baseline (raw epoched parcel time courses), baseline+PCA (PCA applied to the raw epoched parcel time courses), zero-shot (time-averaged output from MEG-GPT’s Transformer decoder) and fine-tuned (same as zero shot but after fine-tuning on training data from Wakeman-Henson). Generalisation to unseen data was assessed “Within Subject” (i.e. held-out sessions from subjects already seen) [left], and to “New Subjects” (i.e. held-out participants) [right]. Within subject accuracies and new subject accuracies of each of the four approaches are plotted. The error bars are 95% confidence intervals over sessions and the chance level is indicated by the red dotted line at 0.25.

## 4 Discussion

In this work, we have demonstrated the feasibility and power of applying a self-supervised, transformer-based foundation model to continuous MEG signals. Our results show that MEG-GPT not only learns to generate realistic MEG data with complex, non-stationary dynamics (Figures 5-7) but also provides highly generalisable features that significantly improve decoding performance on unseen data and subjects (Figure 8). This work establishes a promising new framework for large-scale modelling in neuroscience, though several key areas for future development remain.

### 4.1 Tokeniser

MEG-GPT is designed to receive tokenised data so that it can leverage the benefits of the cross entropy loss function. The proposed novel tokeniser is a data-driven approach that does not aim to compress the data – this means that it does no regularisation and has no loss of temporal or spatial resolution. The philosophy behind this is that we consider the downstream transformer-based foundation model the best place to do any modelling of the rich spatio-temporal structure, rather than risk doing a relatively poor job during tokenisation. We showed that the tokeniser reconstructs MEG data with high fidelity and generalises well to unseen subjects and datasets (Section 3.1).

One important aspect of the tokeniser is the use of 1D convolution kernels in the decoder. The reconstruction of the continuous data based on these kernels affects both future and past time points, i.e. a token at time  $t$  can invoke activity at time points

before  $t$ . This may potentially leak information temporally when paired with a causal AR foundation model. A natural extension of the tokeniser is to implement a causal decoder.

## 4.2 Modelling transient spatio-spectral structure

MEG-GPT captures the spectral and spatial characteristics of MEG data, as shown in the PSD of its generated signals (Section 3.2). However, the current receptive field of 80 samples (320 ms at 250 Hz) limits the model’s ability to capture slow, low-frequency fluctuations. Extending the context window is an important future direction for improving the model.

At present, spatial information is encoded via learnable parcel embeddings, which capture anatomical organisation (Appendix B.3), but the model lacks explicit inter-parcel dependencies. This results in generated signals that are incapable of expressing functional connectivity, either static or dynamic, that is known to be present in MEG/EEG (Gohil, Huang, et al., 2024; Gohil, Kohl, Huang, et al., 2024). Incorporating cross-parcel attention or a similar interaction mechanism is an important extension for the model. Furthermore, rather than relying solely on the model to infer spatial structure implicitly, incorporating it explicitly could provide additional information about co-activation patterns across brain regions and guide the model towards more interpretable representations.

In Section 3.4, we demonstrated that MEG-GPT does an excellent job of capturing single region transient bursting dynamics, such as beta bursts in sensorimotor cortex. These features are not at all well represented by linear AR models. More broadly, the ability to inspect and validate the generated data itself (rather than evaluating the model solely through downstream performance) offers an important assessment of the fidelity of the foundation model. This approach parallels developments in generative vision and language models, where qualitative examination of generated samples has become central to understanding representational capacity. Here, we do this more quantitatively assessing the bursting dynamics of neural data. We showed that the activity generated by MEG-GPT reproduces hallmark neurophysiological patterns such as transient beta bursts, which are increasingly recognised as critical markers of brain function in health and disease (Khanna et al., 2025; Lundqvist, Miller, Nordmark, Liljefors, & Herman, 2024; Quinn et al., 2019; Torrecillos, He, Kühn, & Tan, 2023).

## 4.3 Learning individual differences

Functional neuroimaging data shows a high level of variability across a population. There is increasing awareness that there is a wealth of information present in how subjects vary across population, and that large-scale datasets and more sophisticated modelling approaches are needed to unlock this potential (Seghier & Price, 2018). MEG-GPT represents a step forward in this direction.

In MEG-GPT, individual variability is captured using subject/session embedding vectors. These have been successfully used in computational neuroscience to characterise individual variability (Chehab, Defossez, Loiseau, Gramfort, & King, 2021; Csaky, Van Es, Parker Jones, & Woolrich, 2023; Défossez, Caucheteux, Rapin, Kabeli, & King, 2023; Huang, Gohil, & Woolrich, 2024; Jayalath, Landau, Shillingford, Woolrich, & Jones, 2024). Section 3.3 highlighted MEG-GPT’s ability to learn subject-specific information (fingerprints) in the data through the use of subject embeddings. This aspect of the model allows MEG-GPT to improve individual predictions by leveraging information from simi-

lar subjects. Further work is needed to see the impact foundation models like **MEG-GPT** can have on scientific and clinical studies, when the models are fine-tuned on bespoke studies (He et al., 2022).

In addition to learning differences between individuals, **MEG-GPT** can be trained on the data from the same subject, but different preprocessing, source localisation, parcellation pipelines, to learn the effect of these different data processing choices have on the data.

#### 4.4 Fine tuning and feature generalisability

**MEG-GPT** demonstrates strong zero-shot performance: features extracted from the trained model on the Cam-CAN dataset generalise to new subjects and sessions in the Wakeman-Henson dataset without further training. The extracted features improve decoding performance in both within and across-subject prediction tasks (Section 3.5). This makes the model attractive for researchers with limited computational resources or those working with small datasets. We also showed that **MEG-GPT** can be efficiently fine tuned on new datasets. Pre-training on Cam-CAN took  $\sim 400$  GPU hours, whereas fine tuning on the Wakeman-Henson dataset required only 3 GPU hours. This adaptability provided by fine tuning is critical for deployment across different scanner types, sampling rates, and study populations, and supports domain adaptation for diverse real-world use cases.

#### 4.5 Scaling to larger multi-modal datasets

Both **MEG-GPT** and its tokeniser were trained exclusively on MEG data. We did this as we believe source localisation provides a powerful denoising step and source reconstructing EEG is often difficult due to needing an accurate head model and is only possible for medium/high density recordings. Many EEG foundation models are based on time-frequency features and we believe developing a foundation model for the raw time series would be a valuable contribution. Our approach gives us the ability to generate MEG data directly, which is particularly attractive for validation and downstream use cases. However, a defining feature of foundation models is their scalability across diverse data modalities. MEG datasets are often limited in size and availability compared to EEG. Hence future research should explore cross-modal training on both MEG and EEG. This raises challenges such as handling differences in sampling rate, channel layout, and number of sensors (which could be circumvented by data preprocessing, source reconstruction and parcellation), but offers a compelling opportunity to investigate whether **MEG-GPT** can capture the shared and modality-specific structure of brain activity across recording techniques.

#### 4.6 Limitations and future directions

In this work, when using **MEG-GPT** derived features on downstream decoding task, we simply averaged the features over time to reduce dimensionality for decoding tasks. While this yields strong performance, it potentially discards temporal dynamics encoded in the token sequence. This may disadvantage **MEG-GPT** relative to the baseline approach in some settings. Developing classification architectures that retain temporally resolved representations, e.g., via convolutional, recurrent or attention-based decoders, could further improve task performance.

Our decoding experiments (Figure 8) used a linear classifier trained on extracted features. Prior work (Yuan et al., 2024) suggests that end-to-end fine-tuning of both the

decoder and a more flexible classification head yields superior results. Future work should explore fine tuning the entire model to directly optimise supervised objectives.

A limitation of our current evaluation is the use of a relatively simple baseline based on raw epoched time-courses in the downstream decoding task (Figure 8). While this provides a clear reference point, it does not reflect the range of feature extraction approaches commonly used in MEG decoding. For examples, methods such as PCA, or other low-dimensional representations that capture spatiotemporal structure. Comparing **MEG-GPT** representations against such feature-based baselines would provide a more comprehensive assessment of its advantages. We chose a simple baseline here to isolate the benefit of learned representations from the foundation model, but a systematic comparison with classical feature extraction pipelines is an important direction for future work.

Another important direction for evaluation is a systematic study of the contribution of different model components, for example by training variants with and without specific embeddings. Unfortunately, due to limited computational resources, a comprehensive ablation study on the full MEG dataset is currently prohibitive. However, this is substantially more feasible in simulation. In Appendix G, we demonstrate the effect of removing parcel and subject embeddings using MEG-like simulated data. These experiments confirm the importance of these embeddings within the model.

## 5 Conclusion

In this work, we introduced **MEG-GPT**, a foundation model trained on large-scale resting-state MEG data, and a novel, data-driven tokeniser that is used to provide **MEG-GPT** with tokenised data inputs. Our results demonstrate that this self-supervised approach is highly effective: the model generates realistic data capturing complex neural dynamics, and its features significantly improve zero-shot generalisation in downstream decoding tasks. This confirms the feasibility of applying large-scale generative models to electrophysiological signals, paving the way for versatile, powerful, and general-purpose tools in neural decoding and computational neuroscience.

## Ethics statement

The study that collected the CamCAN dataset was conducted in compliance with the Helsinki Declaration, and had been approved by the local ethics committee, Cambridgeshire 2 Research Ethics Committee. Written informed consent was given by participants. See (Shafto et al., 2014) for details regarding protocols. The Wakeman-Henson dataset ((Wakeman & Henson, 2015)) was approved by Cambridge University Psychological Ethics Committee. Written informed consent was obtained from participants.

## Data and code availability statement

Data used are publicly available. For the Wakeman-Henson dataset, we refer the readers to the original paper (Wakeman & Henson, 2015). For the Cam-CAN dataset, we refer the readers to the original paper (Taylor et al., 2017).

Source code and scripts for reproducing results in the paper are available on GitHub: <https://github.com/OHBA-analysis/osl-foundation>. Example code and tutorials for training a foundation model and applying it to new data are also provided.

## Credit authorship contribution statement

RH: Conceptualisation, Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing - original draft, Writing - review and editing, Visualisation. SC: Conceptualisation, Methodology, Software, Writing - review and editing, CG: Conceptualisation, Data curation, Software, Visualisation, Writing - original draft, Writing - review and editing. OPJ: Conceptualisation, Methodology, Writing - review and editing. MW: Conceptualisation, Methodology, Data curation, Software, Writing - review and editing, Supervision.

## Acknowledgements

Research is supported by the Wellcome Trust (106183/Z/14/Z, 215573/Z/19/Z), the New Therapeutics in Alzheimer’s Diseases (NTAD) study supported by UK MRC, the Dementia Platform UK (RG94383/RG89702) and supported by the NIHR Oxford Health Biomedical Research Centre (NIHR203316). The views expressed are those of the author(s) and not necessarily those of the NIHR or the Department of Health and Social Care. The Wellcome Centre for Integrative Neuroimaging is supported by core funding from the Wellcome Trust (203139/Z/16/Z and 203139/A/16/Z). RH is supported by the Wellcome Trust (215573/Z/19/Z). SC is supported by the Medical Sciences Graduate School Studentship, funded by the Medical Research Council (MR/W006731/1), the Hertford Claire Clifford Lusardi Scholarship, and the Nuffield Department of Clinical Neurosciences. CG is supported by the Wellcome Trust (215573/Z/19/Z). OPJ is supported by the MRC (MR/X00757X/1), Royal Society (RG/R1/241267), NSF (2314493), NFRF (NFRFT-2022-00241), and SSHRC (895-2023-1022). MW is supported by the Wellcome Trust (106183/Z/14/Z, 215573/Z/19/Z).

## Declaration of competing interest

No competing interests.

## References

- Alexander, D. M., Jurica, P., Trengove, C., Nikolaev, A. R., Gepshtein, S., Zvyagintsev, M., Mathiak, K., Schulze-Bonhage, A., Ruescher, J., Ball, T., et al. (2013). Traveling waves and trial averaging: the nature of single-trial and averaged brain responses in large-scale cortical signals. *Neuroimage*, *73*, 95–112.
- Ba, J. L., Kiros, J. R., & Hinton, G. E. (2016). Layer normalization. *arXiv preprint arXiv:1607.06450*.
- Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., et al. (2021). On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*.
- Bonaiuto, J. J., Little, S., Neymotin, S. A., Jones, S. R., Barnes, G. R., & Bestmann, S. (2021). Laminar dynamics of high amplitude beta bursts in human motor cortex. *NeuroImage*, *242*, 118479.
- Bosl, W. J., Tager-Flusberg, H., & Nelson, C. A. (2018). EEG analytics for early detection of autism spectrum disorder: a data-driven approach. *Scientific reports*, *8*(1), 6828.

- Brookes, M. J., Zumer, J. M., Stevenson, C. M., Hale, J. R., Barnes, G. R., Vrba, J., & Morris, P. G. (2010). Investigating spatial specificity and data averaging in MEG. *Neuroimage*, *49*(1), 525–538.
- Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. (2020). Language models are few-shot learners. *Advances in neural information processing systems*, *33*, 1877–1901.
- Cassani, R., Estarellas, M., San-Martin, R., Fraga, F. J., & Falk, T. H. (2018). Systematic review on resting-state EEG for Alzheimer’s disease diagnosis and progression assessment. *Disease markers*, *2018*(1), 5174815.
- Chehab, O., Defossez, A., Loiseau, J.-C., Gramfort, A., & King, J.-R. (2021). Deep recurrent encoder: A scalable end-to-end network to model brain signals. *arXiv preprint arXiv:2103.02339*.
- Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.
- Colclough, G. L., Brookes, M. J., Smith, S. M., & Woolrich, M. W. (2015). A symmetric multivariate leakage correction for MEG connectomes. *Neuroimage*, *117*, 439–448.
- Craik, A., He, Y., & Contreras-Vidal, J. L. (2019). Deep learning for electroencephalogram (EEG) classification tasks: a review. *Journal of neural engineering*, *16*(3), 031001.
- Csaky, R., Van Es, M. W., Parker Jones, O., & Woolrich, M. (2023). Group-level brain decoding with deep learning. *Human Brain Mapping*, *44*(17), 6105–6119.
- Csaky, R., van Es, M. W., Jones, O. P., & Woolrich, M. (2024). Foundational gpt model for meg. *arXiv preprint arXiv:2404.09256*.
- Dash, D., Ferrari, P., Heitzman, D., & Wang, J. (2019). Decoding speech from single trial MEG signals using convolutional neural networks and transfer learning. In *2019 41st annual international conference of the IEEE engineering in medicine and biology society (embc)* (pp. 5531–5535). IEEE.
- Défossez, A., Caucheteux, C., Rapin, J., Kabeli, O., & King, J.-R. (2023). Decoding speech perception from non-invasive brain recordings. *Nature Machine Intelligence*, *5*(10), 1097–1107.
- Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.
- Géron, A. (2022). *Hands-on machine learning with scikit-learn, keras, and tensorflow*. " O’Reilly Media, Inc."
- Gohil, C., Huang, R., Roberts, E., van Es, M. W., Quinn, A. J., Vidaurre, D., & Woolrich, M. W. (2024). osl-dynamics, a toolbox for modeling fast dynamic brain activity. *Elife*, *12*, RP91949.
- Gohil, C., Kohl, O., Huang, R., van Es, M. W., Parker Jones, O., Hunt, L. T., Quinn, A. J., & Woolrich, M. W. (2024). Dynamic network analysis of electrophysiological task data. *Imaging Neuroscience*, *2*, 1–19.
- Gohil, C., Kohl, O., Pitt, J., van Es, M. W., Quinn, A. J., Vidaurre, D., Turner, M. R., Nobre, A. C., & Woolrich, M. W. (2024). Effects of age on resting-state cortical networks. *bioRxiv*, 2024–09.
- Grootswagers, T., Wardle, S. G., & Carlson, T. A. (2017). Decoding dynamic brain patterns from evoked responses: A tutorial on multivariate pattern analysis applied to time series neuroimaging data. *Journal of cognitive neuroscience*, *29*(4), 677–697.

- Hawthorne, C., Jaegle, A., Cangea, C., Borgeaud, S., Nash, C., Malinowski, M., Dieleman, S., Vinyals, O., Botvinick, M., Simon, I., et al. (2022). General-purpose, long-context autoregressive modeling with perceiver AR. In *International conference on machine learning* (pp. 8535–8558). PMLR.
- He, T., An, L., Chen, P., Chen, J., Feng, J., Bzdok, D., Holmes, A. J., Eickhoff, S. B., & Yeo, B. T. (2022). Meta-matching as a simple framework to translate phenotypic predictive models from big to small data. *Nature neuroscience*, *25*(6), 795–804.
- Holtzman, A., Buys, J., Du, L., Forbes, M., & Choi, Y. (2019). The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751*.
- Huang, R., Gohil, C., & Woolrich, M. (2024). Modelling variability in functional brain networks using embeddings. *bioRxiv*, 2024–01.
- Hyvarinen, A. (1999). Fast and robust fixed-point algorithms for independent component analysis. *IEEE transactions on Neural Networks*, *10*(3), 626–634.
- ITU-T. (1988). Pulse code modulation (PCM) of voice frequencies.
- Jang, E., Gu, S., & Poole, B. (2016). Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*.
- Jayalath, D., Landau, G., Shillingford, B., Woolrich, M., & Jones, O. P. (2024). The Brain’s Bitter Lesson: Scaling Speech Decoding With Self-Supervised Learning. *arXiv preprint arXiv:2406.04328*.
- Khanna, P., Farrokhi, B., Choi, H., Griffin, S., Heimbuch, I., Novik, L., Thiesen, K., Morrison, J., Morecraft, R. J., & Ganguly, K. (2025). Separable global and local beta burst dynamics in motor cortex of primates. *bioRxiv*, 2025–05.
- King, J.-R., & Dehaene, S. (2014). Characterizing the dynamics of mental representations: the temporal generalization method. *Trends in cognitive sciences*, *18*(4), 203–210.
- Kingma, D. P. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- Kohl, O., Woolrich, M., Nobre, A. C., & Quinn, A. (2023). Glasser52: A parcellation for MEG-Analysis. *Zenodo*. Retrieved from <https://zenodo.org/records/11099418>
- König, R., Matysiak, A., Kordecki, W., Sielużycki, C., Zacharias, N., & Heil, P. (2015). Averaging auditory evoked magnetoencephalographic and electroencephalographic responses: a critical discussion. *European Journal of Neuroscience*, *41*(5), 631–640.
- Lotte, F., Bougrain, L., Cichocki, A., Clerc, M., Congedo, M., Rakotomamonjy, A., & Yger, F. (2018). A review of classification algorithms for EEG-based brain–computer interfaces: a 10 year update. *Journal of neural engineering*, *15*(3), 031005.
- Luck, S. J. (2014). *An introduction to the event-related potential technique*. MIT press.
- Lundqvist, M., Miller, E. K., Nordmark, J., Liljefors, J., & Herman, P. (2024). Beta: bursts of cognition. *Trends in Cognitive Sciences*, *28*(7), 662–676.
- Marple Jr, S. L. (2019). *Digital spectral analysis*. Courier Dover Publications.
- Murphy, K. P. (2012). *Machine learning: a probabilistic perspective*. MIT press.
- Nie, Y., Nguyen, N. H., Sinthong, P., & Kalagnanam, J. (2022). A time series is worth 64 words: Long-term forecasting with transformers. *arXiv preprint arXiv:2211.14730*.
- Proudfoot, M., Woolrich, M. W., Nobre, A. C., & Turner, M. R. (2014). Magnetoencephalography. *Practical neurology*, *14*(5), 336–343.
- Quinn, A. J., Atkinson, L. Z., Gohil, C., Kohl, O., Pitt, J., Zich, C., Nobre, A. C., & Woolrich, M. W. (2024). The GLM-spectrum: A multilevel framework for spectrum analysis with covariate and confound modelling. *Imaging Neuroscience*, *2*, 1–26.
- Quinn, A. J., van Ede, F., Brookes, M. J., Heideman, S. G., Nowak, M., Seedat, Z. A., Vidaurre, D., Zich, C., Nobre, A. C., & Woolrich, M. W. (2019). Unpacking tran-

- sient event dynamics in electrophysiological power spectra. *Brain topography*, 32(6), 1020–1034.
- Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. (2021). Learning transferable visual models from natural language supervision. In *International conference on machine learning* (pp. 8748–8763). Pmlr.
- Radford, A., Narasimhan, K., Salimans, T., Sutskever, I., et al. (2018). Improving language understanding by generative pre-training.
- Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al. (2019). Language models are unsupervised multitask learners. *OpenAI blog*, 1(8), 9.
- Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., & Sutskever, I. (2021). Zero-shot text-to-image generation. In *International conference on machine learning* (pp. 8821–8831). Pmlr.
- Rosner, B. (1983). Percentage points for a generalized ESD many-outlier procedure. *Technometrics*, 25(2), 165–172.
- Seghier, M. L., & Price, C. J. (2018). Interpreting and utilising intersubject variability in brain function. *Trends in cognitive sciences*, 22(6), 517–530.
- Shafto, M. A., Tyler, L. K., Dixon, M., Taylor, J. R., Rowe, J. B., Cusack, R., Calder, A. J., Marslen-Wilson, W. D., Duncan, J., Dalgleish, T., et al. (2014). The Cambridge Centre for Ageing and Neuroscience (Cam-CAN) study protocol: a cross-sectional, lifespan, multidisciplinary examination of healthy cognitive ageing. *BMC neurology*, 14, 1–25.
- Tang, J., LeBel, A., Jain, S., & Huth, A. G. (2023). Semantic reconstruction of continuous language from non-invasive brain recordings. *Nature Neuroscience*, 26(5), 858–866.
- Taulu, S., & Simola, J. (2006). Spatiotemporal signal space separation method for rejecting nearby interference in MEG measurements. *Physics in Medicine & Biology*, 51(7), 1759.
- Taylor, J. R., Williams, N., Cusack, R., Auer, T., Shafto, M. A., Dixon, M., Tyler, L. K., Henson, R. N., et al. (2017). The Cambridge Centre for Ageing and Neuroscience (Cam-CAN) data repository: Structural and functional MRI, MEG, and cognitive data from a cross-sectional adult lifespan sample. *neuroimage*, 144, 262–269.
- Thomas, A. W., Ré, C., & Poldrack, R. A. (2022). Interpreting mental state decoding with deep learning models. *Trends in Cognitive Sciences*, 26(11), 972–986.
- Torrecillos, F., He, S., Kühn, A. A., & Tan, H. (2023). Average power and burst analysis revealed complementary information on drug-related changes of motor performance in Parkinson’s disease. *npj Parkinson’s Disease*, 9(1), 93.
- Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al. (2023). Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.
- Tunstall, L., Von Werra, L., & Wolf, T. (2022). *Natural language processing with transformers*. " O’Reilly Media, Inc."
- Tzovara, A., Murray, M. M., Plomp, G., Herzog, M. H., Michel, C. M., & De Lucia, M. (2012). Decoding stimulus-related information from single-trial EEG responses based on voltage topographies. *Pattern Recognition*, 45(6), 2109–2122.
- Van Den Oord, A., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A., Kavukcuoglu, K., et al. (2016). Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*, 12.

- Van Den Oord, A., Vinyals, O. et al. (2017). Neural discrete representation learning. *Advances in neural information processing systems*, 30.
- Van Veen, B. D., & Buckley, K. M. (1988). Beamforming: A versatile approach to spatial filtering. *IEEE assp magazine*, 5(2), 4–24.
- Van Veen, B. D., Van Drongelen, W., Yuchtman, M., & Suzuki, A. (1997). Localization of brain electrical activity via linearly constrained minimum variance spatial filtering. *IEEE Transactions on biomedical engineering*, 44(9), 867–880.
- van Es, M. W., Gohil, C., Quinn, A. J., & Woolrich, M. W. (2025). osl-ephys: a Python toolbox for the analysis of electrophysiology data. *Frontiers in Neuroscience*, 19, 1522675.
- Waswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.
- Vidaurre, D., Quinn, A. J., Baker, A. P., Dupret, D., Tejero-Cantero, A., & Woolrich, M. W. (2016). Spectrally resolved fast transient brain states in electrophysiological data. *Neuroimage*, 126, 81–95.
- Wakeman, D. G., & Henson, R. N. (2015). A multi-subject, multi-modal human neuroimaging dataset. *Scientific data*, 2(1), 1–10.
- Wang, C., Subramaniam, V., Yaari, A. U., Kreiman, G., Katz, B., Cases, I., & Barbu, A. (2023). BrainBERT: Self-supervised representation learning for intracranial recordings. *arXiv preprint arXiv:2302.14367*.
- Welch, P. (2003). The use of fast Fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms. *IEEE Transactions on audio and electroacoustics*, 15(2), 70–73.
- Yuan, Z., Shen, F., Li, M., Yu, Y., Tan, C., & Yang, Y. (2024). BrainWave: A Brain Signal Foundation Model for Clinical Applications. *arXiv preprint arXiv:2402.10251*.
- Zhai, X., Kolesnikov, A., Houlsby, N., & Beyer, L. (2022). Scaling vision transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 12104–12113).